Cognitively Inspired Natural Language Processing

Ning Shi, ning.shi@ualberta.ca

Supervised by Prof. Grzegorz Kondrak

Amii AI Seminar

2022/12/23

Ning Shi is a 1st-year Ph.D. student working with Prof. Grzegorz Kondrak at the University of Alberta, associated with Alberta Machine Intelligence Institute (Amii).

Education:

- Georgia Institute of Technology
- Syracuse University
- New York University
- Donghua University

Experience:

- Beijing Academy of Artificial Intelligence (BAAI)
- Alibaba Group
- Learnable

Outline

- Part1 - RoChBert: Towards Robust BERT Fine-tuning for Chinese
- Part2 - Revisit Systematic Generalization via Meaningful Learning
- Part3 - Text Editing as Imitation Game

Knowledge Fusion

RoChBert: Towards Robust BERT Fine-tuning for Chinese

Zihan Zhang, Jinfeng Li, Ning Shi, Bo Yuan, Xiangyu Liu, Rong Zhang, Hui Xue, Donghong Sun, and Chao Zhang

Findings of EMNLP 2022

Introduction

Knowledge

- Conscience
- Sum of our memories
- All the knowledge
- Natural language processing (NLP)

semantics, syntax, imagination, association, etc.

e.g., What a beautiful day.

Introduction

Adversarial texts

- Discrete
- Small perturbation significant change
- Small change significant perturbation

Typoglycemia

- More than what we can see

Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae.

Davis, Matt (2012)

Introduction

Adversarial texts in Chinese

- Chinese characters or 漢字 (hànzì)
- Pronunciation (homophones)

  English: I'll ***bury*** the ***berry***.

  Chinese: gambling - 博彩 (bó cǎi) v.s. 菠菜 (bō cài)
- Glyph (homoglyph)

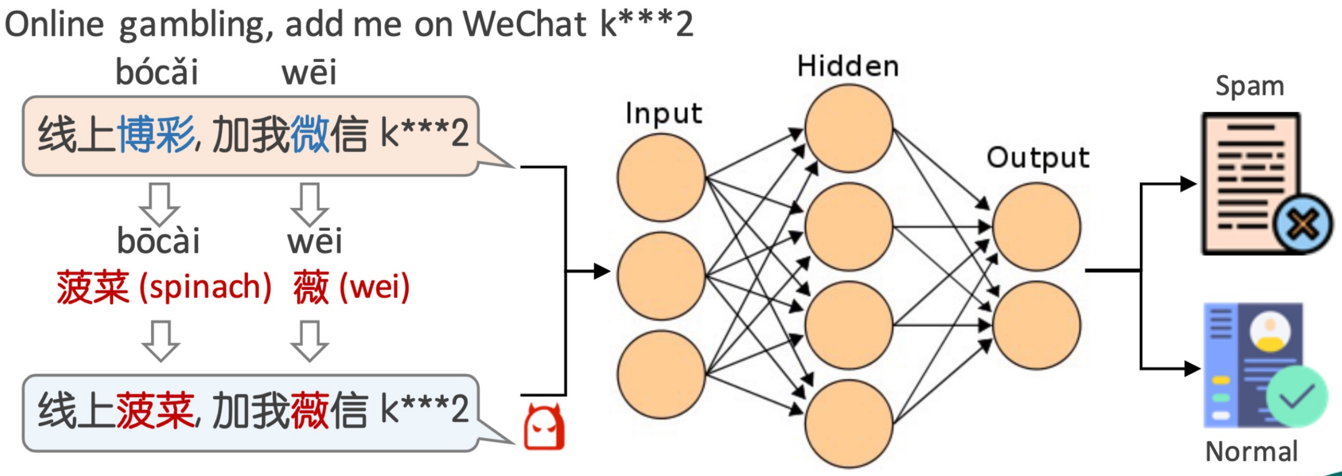
  English: internationa***l***bank.c***0***m - i, l v.s., I, L; o, O v.s., 0

  Chinese: WeChat - 微信 (wēi xìn) v.s. 薇信 (wēi xìn)

Method

Knowledge fusion for model robustness

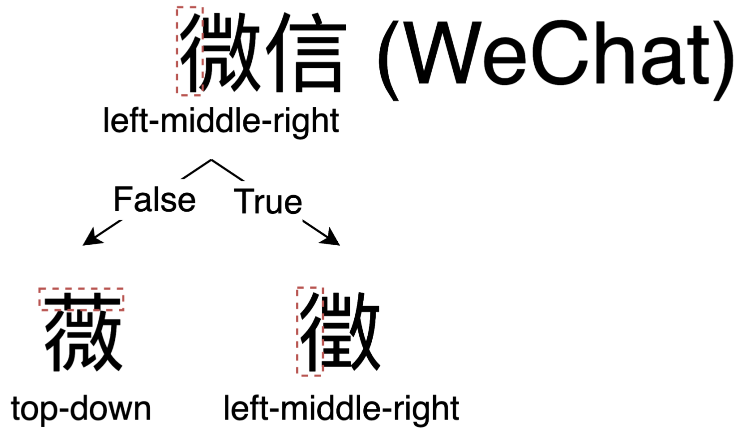
- Adversarial graph
- Multimodal fusion
- Data augmentation by curriculum learning



Adversarial Graph

Knowledge representation

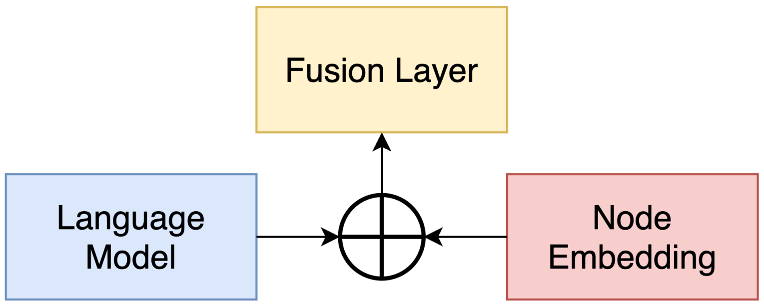
- Adversarial graph involving stroke code
    - StoneSkipping (Jiang et al., 2019)
    - AdvGraph (Li et al., 2021)
- Node -> Chinese character
- Edge -> phonetic or glyph relationship
- 3,000 -> 7,707 nodes
- 109,706 edges



Multimodal Fusion

Knowledge fusion

- Knowledge as a second modal
- Graph embedding (e.g., node2vec)
- Word embedding (e.g., BERT)
- Concatenation and fusion (e.g., self-attention)



Data Augmentation

Curriculum learning

- Adversarial texts
- Not only adversarial examples

  samples mislead the target model
- But also intermediate texts

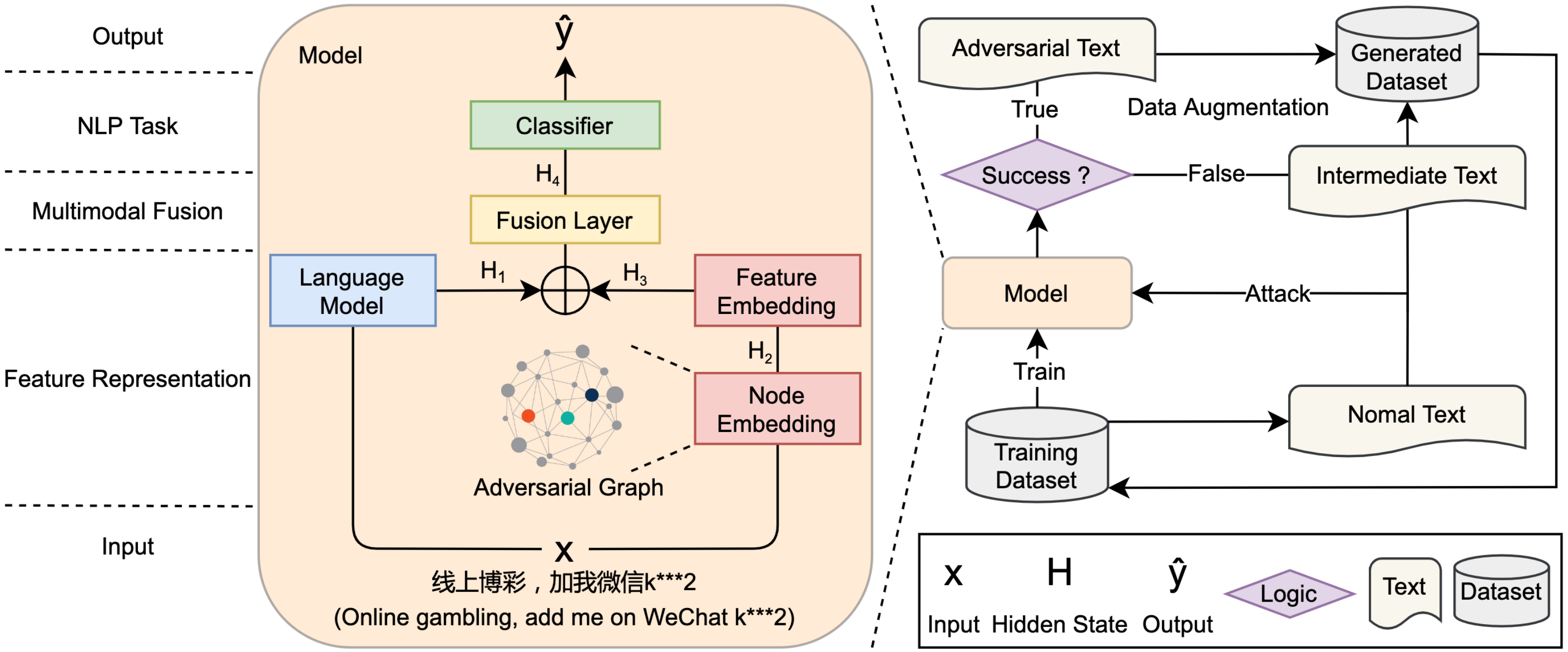
  samples lead to a confidence decline

Algorithm 1 : The detail of data augmentation.

Input: Training dataset D, the target classifier $\mathcal{F}$ with mapping f.
Output: New training dataset D_{ag}.

for $x \in D$ **do**
    $tmp \leftarrow \{\}$
    $\hat{y} = \mathcal{F}_f(x)$
    if $\hat{y}$ is not the ground-truth label **then**
        continue
    end if
    $x^* = x, \hat{y}^* = \hat{y}$
    while $\hat{y}^* == \hat{y}$ **do**
        $x^* = x^* + \Delta x$ ▷ According to attack algorithms
        $\hat{y}^* = \mathcal{F}_f(x^*)$
        $tmp \leftarrow tmp \bigcup \{x^*\}$
        if all the words in x are modified **then**
            break
        end if
    end while
    if $\hat{y}^* \neq \hat{y}$ and $\|x^* - x\|_p < \epsilon_{max}$ **then**
        $D_{ag} \leftarrow D_{ag} \bigcup tmp$
    end if
    if $size(D_{ag}) > size(D)$ **then**
        $D_{ag} \leftarrow D_{ag} \bigcup D$
        break
    end if
end for
return D_{ag};

Method



Output
NLP Task
Multimodal Fusion
Feature Representation
Input
Model
ŷ
Classifier
H4
Fusion Layer
Language Model
H1
H3
Feature Embedding
H2
Node Embedding
Adversarial Graph
x
线上博彩，加我微信k***2
(Online gambling, add me on WeChat k***2)
Adversarial Text
Generated Dataset
True
Data Augmentation
Success ?
False
Intermediate Text
Model
Attack
Train
Training Dataset
Nomal Text
x
Input
H
Hidden State
ŷ
Output
Logic
Text
Dataset

Experiments

Data

Sentiment analysis, text classification, and natural language inference

- ChnSentiCorp
  github.com/pengming617/bert_classification/tree/master/data
- DMSC
  https://www.kaggle.com/utmhikari/doubanmovieshortcomments
- THUCNews (Sun et al., 2016)
- OCNLI
  https://github.com/cluebenchmark/OCNLI

Experiments

Baselines

- ChineseBERT (Sun et al., 2021)
- Chinese spelling corrector (SC)

  https://github.com/shibing624/pycorrector

Evaluation

- Accuracy
- Modification rate (MR)
- Unlimited attack success rate (UASR)
- Limited attack success rate (LASR)

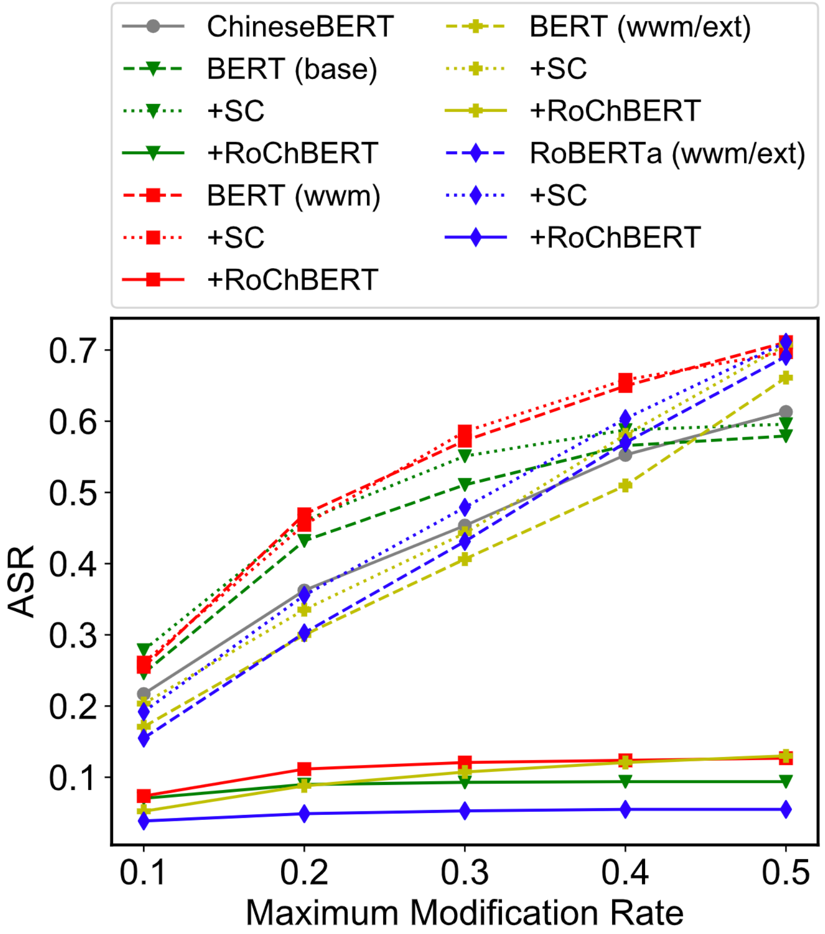
Experiments

Model	Chnsenti.	DMSC	THUC.	OCNLI
ChineseBERT	95.25	92.95	97.87	**73.20**
$BERT_{base}$	95.33	93.02	98.07	71.57
+SC	94.42	92.85	98.07	70.57
+`RoChBert` (PWWS)	95.58	93.05	97.87	67.76
+`RoChBert` (TextBugger)	95.83	92.75	98.00	67.34
+`RoChBert` (Random)	95.92	92.70	**98.13**	70.25
$BERT_{wwm}$	94.58	92.51	97.87	70.33
+SC	93.58	92.45	97.93	69.08
+`RoChBert` (PWWS)	94.92	92.70	97.93	67.23
+`RoChBert` (TextBugger)	95.75	93.30	97.80	67.71
+`RoChBert` (Random)	95.25	91.45	98.00	70.09
$BERT_{wwm/ext}$	**96.00**	93.29	97.73	71.16
+SC	95.00	93.20	97.80	70.68
+`RoChBert` (PWWS)	95.58	**94.00**	97.80	68.31
+`RoChBert` (TextBugger)	95.42	93.30	97.87	68.65
+`RoChBert` (Random)	95.83	93.60	97.73	71.40
$RoBERTa_{wwm/ext}$	95.58	92.89	98.00	71.29
+SC	94.50	92.95	97.87	70.88
+`RoChBert` (PWWS)	95.58	93.10	**98.13**	68.31
+`RoChBert` (TextBugger)	95.83	93.56	97.93	69.09
+`RoChBert` (Random)	95.50	93.45	98.07	72.45

Experiments

Model	PWWS			TextBugger			Random			PWWS			TextBugger			Random		
	UASR	LASR	MR	UASR	LASR	MR	UASR	LASR	MR	UASR	LASR	MR	UASR	LASR	MR	UASR	LASR	MR
	ChnSetiCorp									THUCNews								
ChineseBERT	79.73	40.97	27.22	93.25	42.67	**23.64**	54.91	**3.38**	**51.23**	71.55	23.44	44.82	69.80	36.23	23.63	78.40	1.13	64.68
BERT$_{base}$	83.62	67.66	12.96	97.45	69.26	16.12	52.77	8.19	42.85	81.31	25.64	44.60	58.43	43.21	14.87	80.59	2.56	61.42
+SC	82.75	64.86	13.74	96.49	71.25	14.85	54.42	7.56	43.01	80.51	29.18	40.87	59.90	46.02	13.26	79.08	2.55	61.96
+RoChBert	65.18	31.63	29.49	64.45	34.92	20.35	39.49	10.98	37.48	66.35	5.11	63.71	9.66	9.10	8.05	51.17	0.81	**66.57**
BERT$_{wwm}$	87.53	56.56	19.86	98.28	64.73	16.89	48.27	6.45	44.91	73.77	36.27	35.17	78.28	46.93	21.07	74.90	2.67	58.81
+SC	84.62	57.42	18.75	97.63	68.60	15.95	50.54	5.59	45.62	76.66	30.19	38.37	72.26	45.45	17.96	73.49	2.97	58.68
+RoChBert	64.42	22.94	**41.94**	62.24	35.23	19.6	42.31	6.36	44.88	76.60	5.19	**64.00**	13.09	11.15	11.94	**45.77**	0.72	64.67
BERT$_{wwm/ext}$	72.04	42.93	22.21	92.93	53.27	20.16	56.96	5.80	40.44	79.63	21.29	48.05	86.39	29.99	**31.85**	78.81	1.74	60.79
+SC	75.00	50.53	18.19	90.93	57.59	17.47	57.07	5.91	41.04	82.72	21.98	46.59	79.04	33.54	26.08	79.24	2.76	60.04
+RoChBert	**62.75**	24.44	37.08	65.74	31.70	23.17	**38.45**	6.45	44.26	66.80	7.89	60.09	13.10	8.80	16.31	51.07	**0.61**	66.04
RoBERTa$_{wwm/ext}$	76.46	44.11	23.25	99.78	54.19	21.49	57.58	5.83	44.14	72.58	17.02	50.12	81.24	30.27	28.46	79.51	1.22	62.27
+SC	80.47	52.23	19.62	98.20	59.66	18.26	55.94	6.90	44.00	81.51	19.41	48.52	77.22	35.55	24.33	79.57	1.23	62.43
+RoChBert	65.85	**22.69**	39.14	**54.18**	**28.57**	20.49	38.59	8.41	36.60	**59.92**	**4.68**	63.54	**5.49**	**4.88**	7.70	59.04	1.83	60.33
	DMSC									OCNLI								
ChineseBERT	78.76	60.35	16.64	92.20	59.37	18.47	53.30	7.04	48.75	62.57	46.22	17.32	73.78	35.27	25.16	38.92	8.38	40.92
BERT$_{base}$	76.70	61.06	15.74	78.75	60.19	13.79	56.31	7.66	46.69	58.68	42.29	17.39	65.84	35.95	22.96	40.08	10.06	38.59
+SC	83.24	63.24	17.87	82.49	63.24	13.54	57.51	6.38	47.25	56.50	42.38	15.81	65.73	36.08	21.87	38.88	8.81	38.82
+RoChBert	68.67	46.70	23.66	**36.36**	**29.22**	12.43	44.94	10.13	39.21	**43.58**	**29.73**	17.90	**48.99**	24.21	24.53	**36.50**	**5.65**	**47.04**
BERT$_{wwm}$	95.66	76.66	13.53	99.67	76.33	13.92	52.88	6.30	43.41	56.06	40.42	17.09	64.23	32.54	23.97	36.34	8.03	42.91
+SC	94.46	74.38	13.66	98.91	77.96	13.07	55.70	7.38	43.59	55.33	40.68	16.28	63.30	33.43	22.69	36.56	8.68	41.33
+RoChBert	64.94	50.00	14.00	44.85	33.47	15.94	46.78	11.91	40.22	50.00	30.18	20.79	50.36	20.32	27.54	39.21	7.05	46.48
BERT$_{wwm/ext}$	85.52	60.30	17.63	99.79	69.31	16.35	57.51	**5.90**	**49.93**	62.22	45.56	16.46	69.31	37.08	23.15	40.69	8.61	41.58
+SC	88.41	63.95	16.98	99.36	72.85	15.36	59.12	6.76	48.85	61.79	47.70	15.26	69.32	37.10	21.83	43.10	10.74	40.65
+RoChBert	75.40	46.31	22.37	40.36	29.98	14.65	**37.38**	6.87	47.99	51.75	32.75	21.27	54.69	**20.23**	**28.24**	39.52	7.45	45.33
RoBERTa$_{wwm/ext}$	69.70	50.76	20.04	83.12	53.90	18.69	52.71	7.79	39.31	66.21	46.02	17.65	80.08	37.09	26.35	40.66	7.01	43.34
+SC	75.54	55.19	18.54	85.82	57.47	17.57	54.87	7.68	40.70	65.10	47.31	17.48	79.03	40.14	24.01	41.66	8.83	40.91
+RoChBert	**55.15**	**28.06**	**34.80**	59.25	35.27	**19.52**	44.86	7.92	43.98	60.14	35.11	**22.33**	51.43	21.57	27.31	36.86	7.70	42.60

Experiments



Observation

- Robustness in adaptive settings (figure)
- Ablation study (table)

Model	Acc.	UASR	LASR	MR
$\text{BERT}_{\text{base}}$	93.02	78.75	60.19	13.79
+graph	93.05	71.53	50.32	16.40
+aug.	93.85	68.01	48.01	17.56
+graph+aug.	94.15	73.16	19.36	40.69
+RoChBert	92.75	36.36	29.22	12.43

Experiments

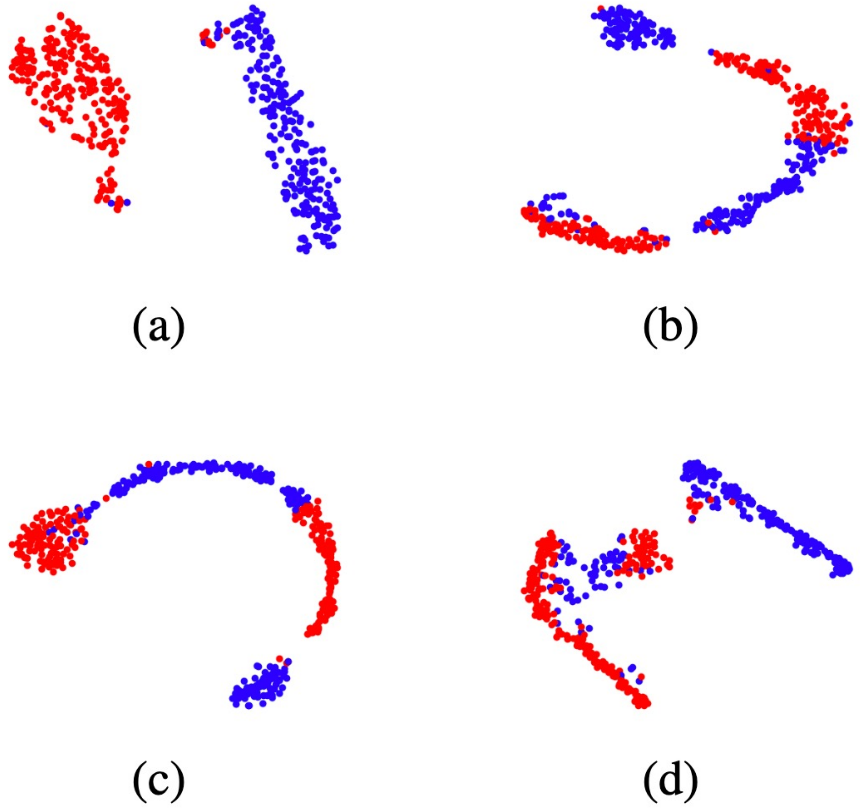
Representation Visualization

(a) Benign texts $\text{BERT}_{\text{base}}$

(b) Adversarial texts $\text{BERT}_{\text{base}}$

(c) Adversarial texts ChineseBERT

(d) Adversarial texts RoChBert



TLDR

Conclusion

- RoChBert - a plug-in for the robustness of Chinese language model
- Incorporating human knowledge (e.g., adversarial graph)
- Knowledge representation -> knowledge fusion
- Knowledge can be helpful in many NLP tasks

e.g., Incorporating External POS Tagger for Punctuation Restoration (Shi et al., 2021)

Q&A

arXiv

Meaningful Learning

Revisit Systematic Generalization via Meaningful Learning

Ning Shi, Boxin Wang, Wei Wang, Xiangyu Liu, and Zhouhan Lin

the Fifth BlackboxNLP at EMNLP 2022

Introduction

Systematic Generalization

- Talent of human
- How about neural networks?
- Pessimistic view
- Optimistic results

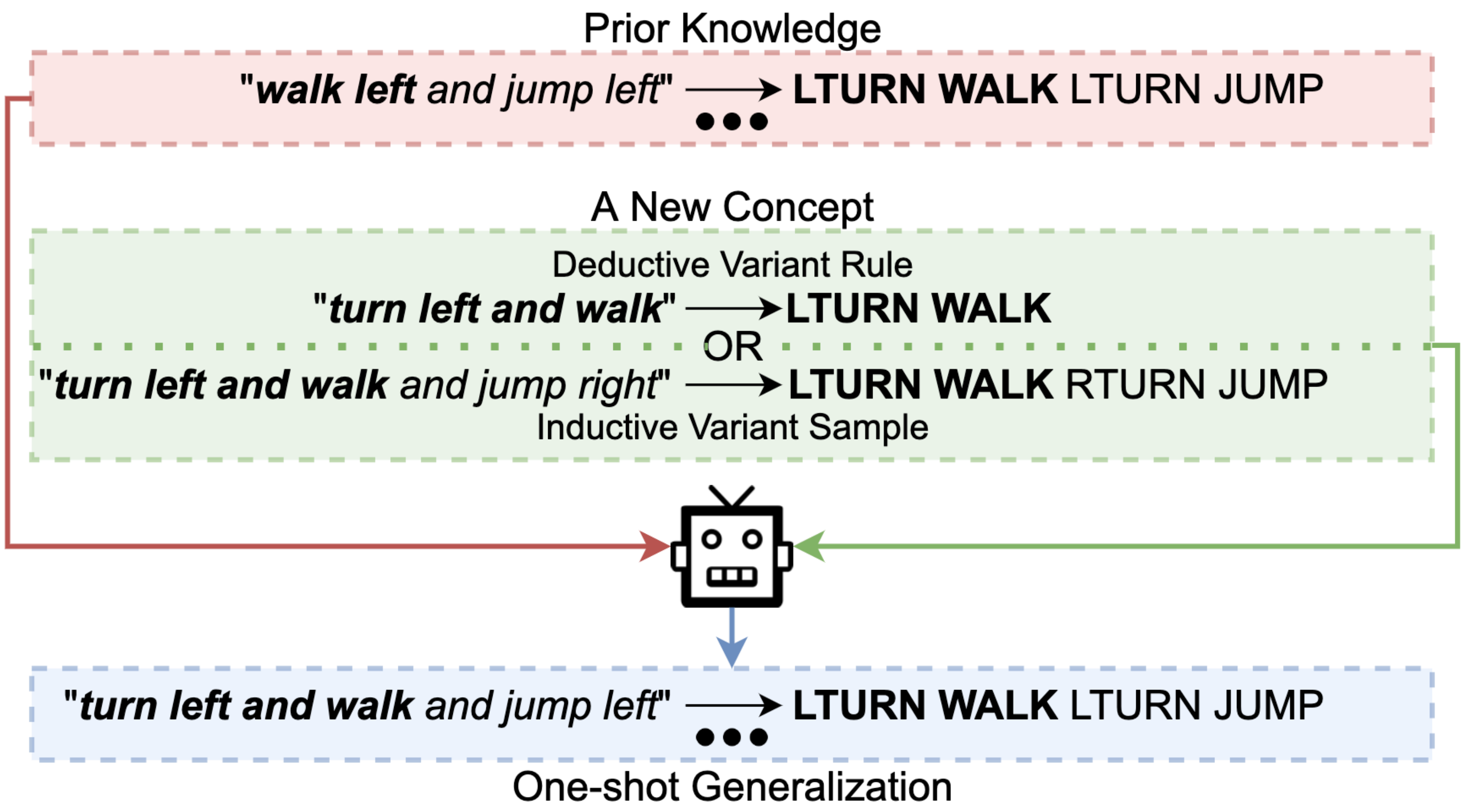
walk twice -> WALK WALK

jump -> JUMP

↓

jump twice -> ?

Introduction



amii

Introduction

Question by Lake and Baroni (2018) on page 8:

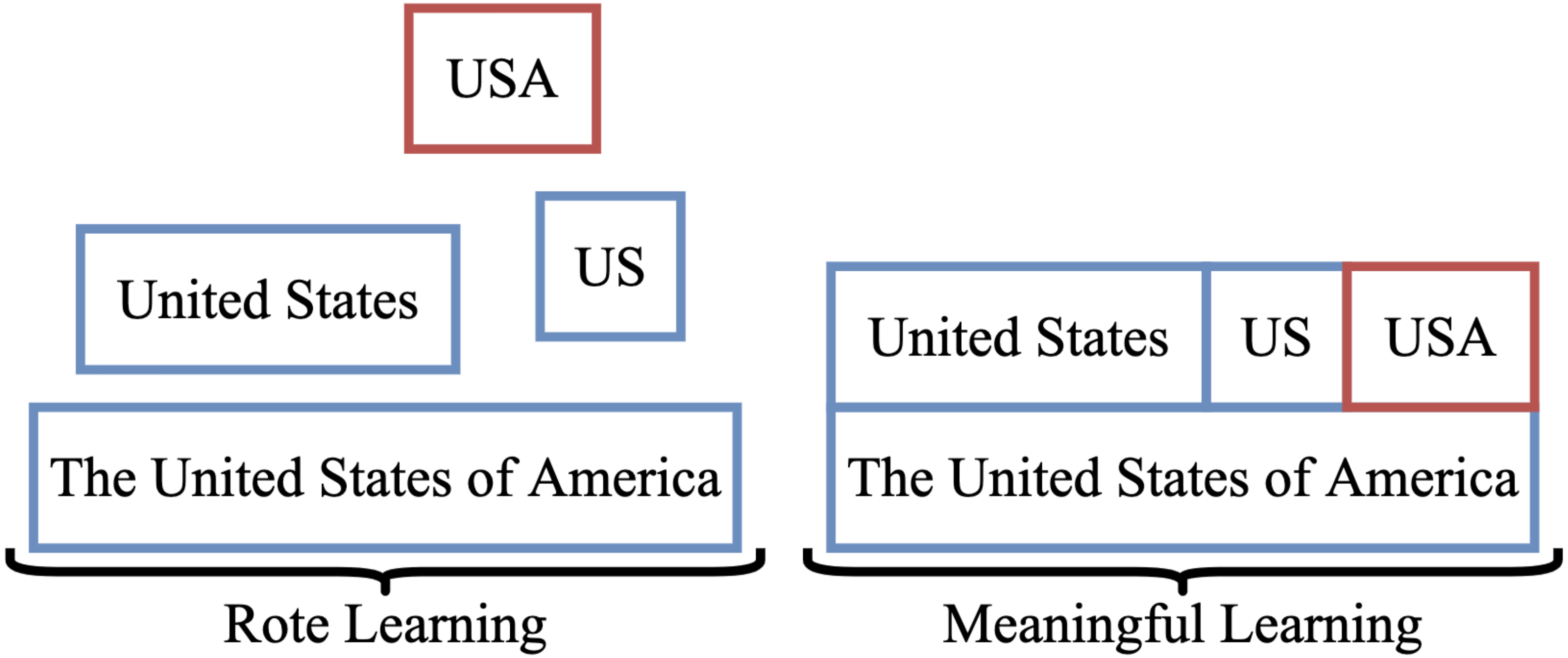
What are, precisely, the generalization mechanisms that subtend the networks' success in these experiments?

Meaningful Learning

In educational psychology, meaningful learning refers to learning new concepts by relating them to old ones (Ausubel, 1963).

On the contrary, rote learning stands for learning new concepts without the consideration of relationships.

Meaningful Learning



Inductive Learning

Inductive learning is a bottom-up approach from the more specific to the more general.

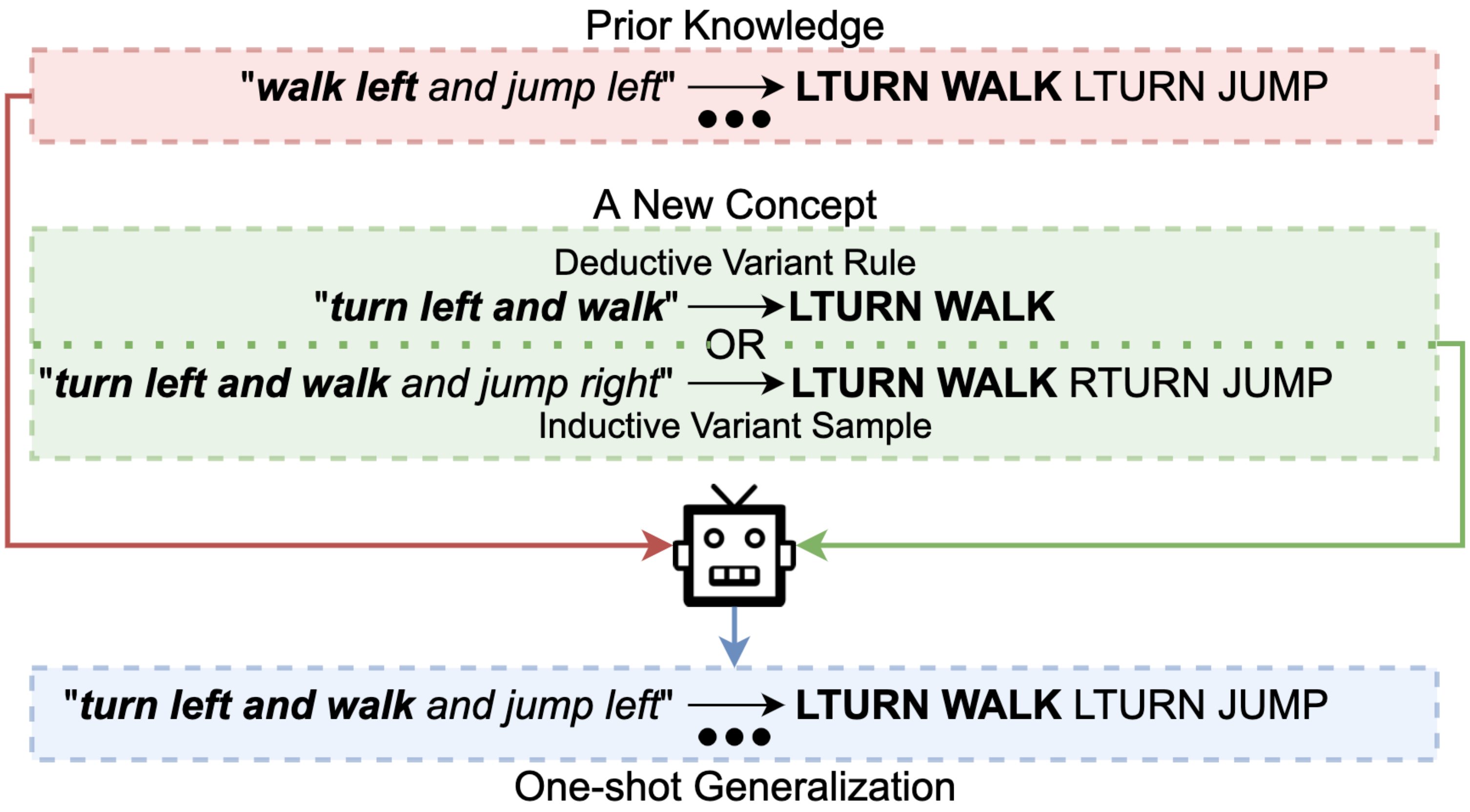
In grammar teaching, inductive learning is a rule discovery approach starting with the presentation of specific examples from which a general rule can be inferred.

Deductive Learning

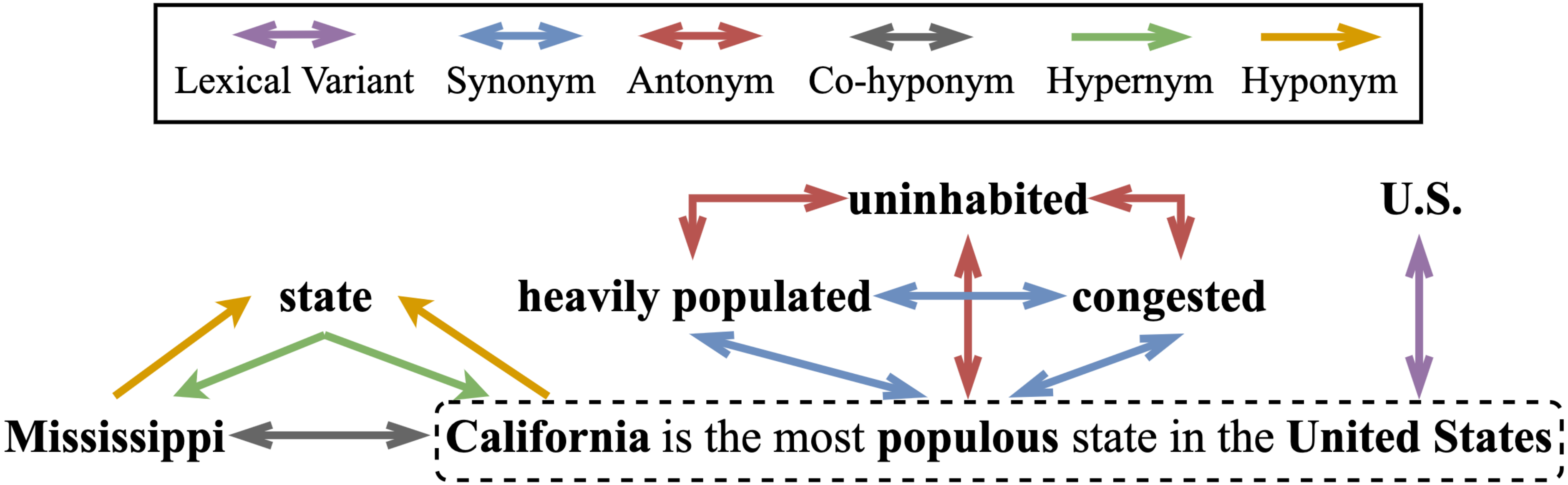
Deductive Learning, the opposite of inductive learning, is a top-down approach from the more general to the more specific.

As a rule-driven approach, teaching in a deductive manner often begins with presenting a general rule followed by specific examples in practice where the rule is applied.

Meaningful Learning



Semantic Links



Experimental Setup

Data

- SCAN (Lake and Baroni, 2018)
- GEO from Geography - https://github.com/jkkummerfeld/text2sql-data
- ADV from Advising - https://github.com/jkkummerfeld/text2sql-data

Seq2seq models

- RNN - bi-directional recurrent networks with LSTM units
- CNN - convolutional seq2seq structure (Gehring et al., 2017)
- TFM – Transformer (Vaswani et al., 2017)

amii

Experimental Setup

Data	Sequence	
SCAN	Source	*jump twice*
	Target	JUMP JUMP
GEO	Source	*how many people in new york city*
	Target	SELECT CITY alias0 . POPULATION FROM CITY AS CITY alias0 WHERE CITY alias0 . CITY_NAME = CITY_NAME ;
ADV	Source	*Which department includes a history of american film ?*
	Target	SELECT DISTINCT COURSE alias0 . DEPARTMENT FROM COURSE AS COURSE alias0 WHERE COURSE alias0 . NAME LIKE TOPIC ;
Geography	Source	*how many people live in new york*
	Target	SELECT STATE alias0 . POPULATION FROM STATE AS STATE alias0 WHERE STATE alias0 . STATE_NAME = " new york " ;
Advising	Source	*I would like to see A History of American Film courses of 2 credits .*
	Target	SELECT DISTINCT COURSE alias0 . DEPARTMENT , COURSE alias0 . NAME , COURSE alias0 . NUMBER FROM COURSE AS COURSE alias0 WHERE (COURSE alias0 . DESCRIPTION LIKE "% A History of American Film %" OR COURSE alias0 . NAME LIKE "% A History of American Film %") AND COURSE alias0 . CREDITS = 2 ;

Experimental Setup

Evaluation

- Token accuracy (Token Acc.)
- Sequence accuracy (Seq. Acc.)

Data	SCAN					GEO					ADV					Geography		Advising	
	Exp. 1			Exp. 2		Exp. 1			Exp. 2		Exp. 1			Exp. 2		Bas.	Aug.	Bas.	Aug.
	Sta.	Dif.	Cha.	Sta.	Dif.	Sta.	Dif.	Cha.	Sta.	Dif.	Sta.	Dif.	Cha.	Sta.	Dif.				
Train Size	20946	20942	20928	20950	20946	724	720	711	728	724	6038	6034	5969	6040	6036	598	701	3814	5660
Test Size	308240	308240	308240	308240	308240	21350	21350	21350	21350	21350	107614	107614	107614	107614	107614	279	279	573	573
Time RNN			21					5					19			4	5	27	35
Time CNN			17					1.2					11			1	1.2	12	19
Time TFM			7					0.5					5			0.4	0.5	6	8

Experiment#1

1. We augment the original training set with variants samples and rules as more as possible.

2. We decrease the number of augmented samples for each variant until the one-shot learning setting.

3. We train the same model on these various datasets to format a gradual transition from baselines to the one-shot learning.

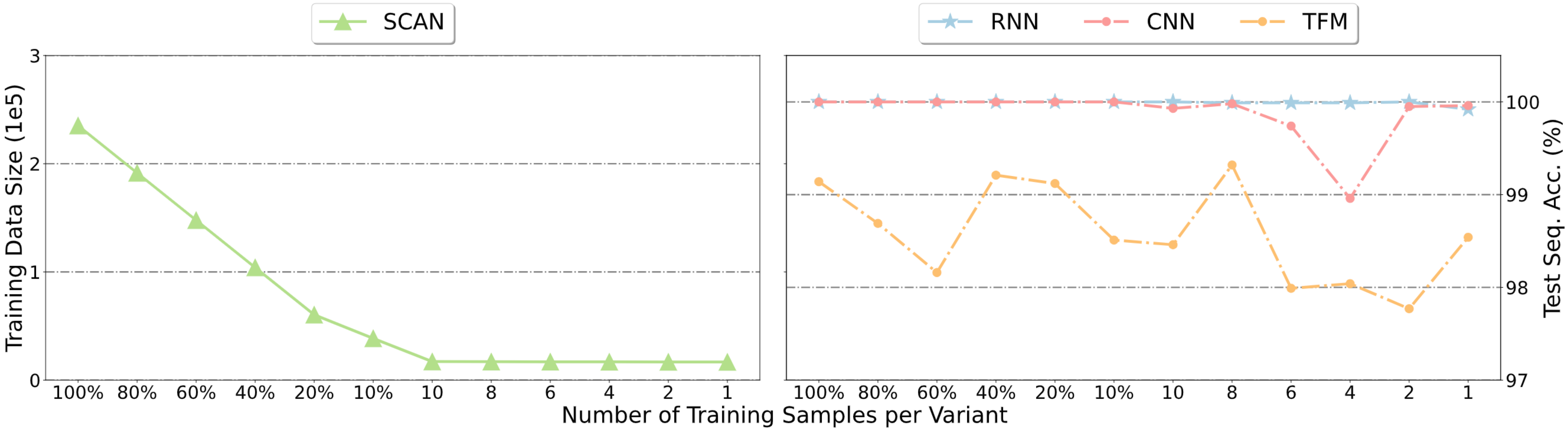
Experiment#1

Data	Primitive	Semantic Links	Variant	Concept Rule	
				Primitive Rule	Variant Rule
SCAN	*jump* *look* *run* *walk*	Lexical Variant	*jump_0* *look_0* *run_0* *walk_0*	*jump* → JUMP *look* → LOOK *run* → RUN *walk* → WALK	*jump_0* → JUMP *look_0* → LOOK *run_0* → RUN *walk_0* → WALK
GEO	*new york city* *mississippi rivier* *dc* *dover*	Co-hyponym	*houston city* *red rivier* *kansas* *salem*	*new york city* → CITY_NAME *mississippi rivier* → RIVER_NAME *dc* → STATE_NAME *dover* → CAPITAL_NAME	*houston city* → CITY_NAME *red rivier* → RIVER_NAME *kansas* → STATE_NAME *salem* → CAPITAL_NAME
ADV	*a history of american film* *aaron magid* *aaptis* *100*	Co-hyponym	*advanced ai techniques* *cargo* *survmeth* *171*	*a history of american film* → TOPIC *aaron magid* → INSTRUCTOR *aaptis* → DEPARTMENT *100* → NUMBER	*advanced ai techniques* → TOPIC *cargo* → INSTRUCTOR *survmeth* → DEPARTMENT *171* → NUMBER

Experiment#1

Data	Primitive	Variant	#Variants	Prompt
SCAN	*jump*	*jump_0*	10	*[concept] twice*
GEO	*new york city*	*houston city*	39	*how many people in [concept]*
	mississippi rivier	*red rivier*	9	*how long is [concept]*
	dc	*kansas*	49	*where is [concept]*
	dover	*salem*	8	*what states capital is [concept]*
ADV	*a history of american film*	*advanced ai techniques*	5/424	*who teaches [concept] ?*
	aaron magid	*cargo*	5/492	*does [concept] give upper-level courses ?*
	aaptis	*survmeth*	5/1720	*name core courses for [concept] .*
	100	*171*	5/1895	*can undergrads take [concept] ?*

Experiment#1



Experiment#2

- Standard: models are trained on prior knowledge and one variant sample per variant (i.e., the same one-shot setting).
- Difficult: We remove from the prior knowledge primitive samples sharing the same context with their variant samples.

  (e.g., we remove "jump twice" due to "jump_0 twice")
- Challenging: We also exclude from the prior knowledge primitive samples of the same length as their variant samples.

  (e.g., we remove "jump twice", "jump right", "jump left")

amii

Experiment#2

Data	Model	Token Acc.%			Seq. Acc.%		
		Standard	Difficult	Challenging	Standard	Difficult	Challenging
SCAN	RNN	99.99 ± 0.03	99.89 ± 0.19	99.96 ± 0.02	99.95 ± 0.08	99.85 ± 0.08	99.80 ± 0.31
	CNN	99.96 ± 0.08	99.76 ± 0.54	98.89 ± 2.44	99.85 ± 0.34	99.52 ± 1.07	97.57 ± 5.24
	TFM	98.91 ± 0.78	98.90 ± 1.10	98.76 ± 0.85	97.35 ± 1.62	96.86 ± 2.64	96.38 ± 2.81
GEO	RNN	75.71 ± 8.42	75.69 ± 6.12	73.46 ± 3.05	44.95 ± 14.69	43.27 ± 13.47	36.77 ± 5.60
	CNN	87.99 ± 2.67	79.51 ± 6.03	77.40 ± 2.48	69.46 ± 5.78	51.20 ± 8.64	48.58 ± 3.40
	TFM	75.37 ± 7.84	75.11 ± 4.88	68.41 ± 4.76	45.93 ± 12.42	44.59 ± 9.76	36.93 ± 7.47
ADV	RNN	58.61 ± 6.18	59.74 ± 5.67	58.11 ± 5.82	36.18 ± 5.75	35.69 ± 6.05	35.45 ± 6.69
	CNN	57.83 ± 7.55	54.05 ± 5.74	53.66 ± 2.57	45.08 ± 9.32	42.14 ± 6.90	41.37 ± 4.04
	TFM	53.43 ± 2.80	51.51 ± 4.50	49.17 ± 2.58	42.59 ± 3.65	41.28 ± 4.35	38.88 ± 2.68

Experiment#3

- Standard: models are trained on the prior knowledge, primitive rules, and variant rules.
- Difficult: We remove primitive rules from the training set. Consequently, semantic links are weakened and depend on variant rules only.

amii

Experiment#3

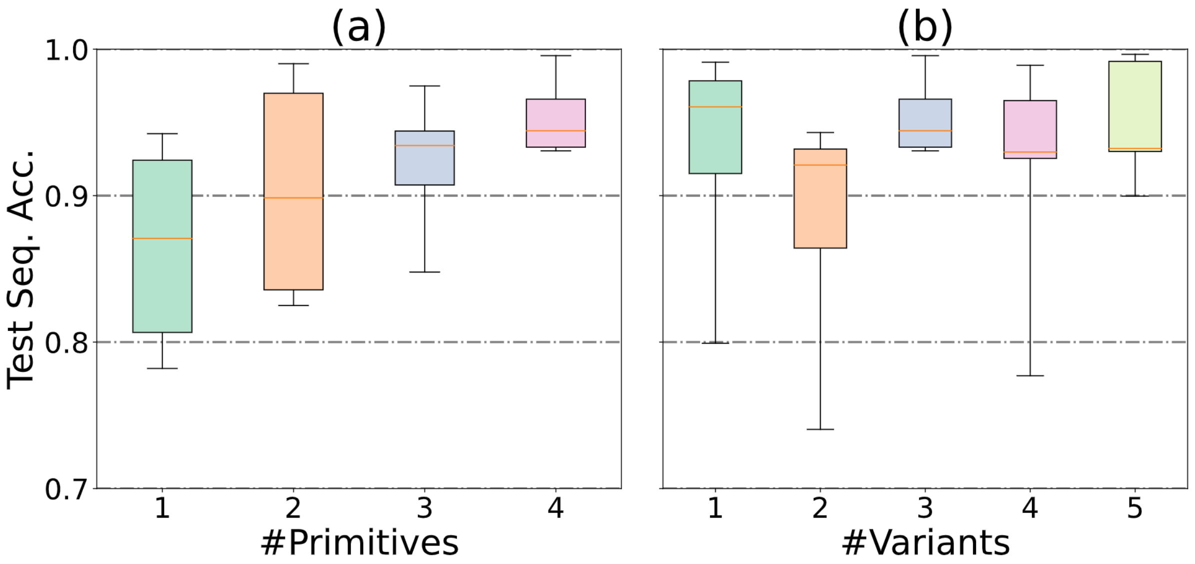
Concept Rule	
Primitive Rule	**Variant Rule**
jump → JUMP	*jump_0* → JUMP
look → LOOK	*look_0* → LOOK
run → RUN	*run_0* → RUN
walk → WALK	*walk_0* → WALK
new york city → CITY_NAME	*houston city* → CITY_NAME
mississippi rivier → RIVER_NAME	*red rivier* → RIVER_NAME
dc → STATE_NAME	*kansas* → STATE_NAME
dover → CAPITAL_NAME	*salem* → CAPITAL_NAME
a history of american film → TOPIC	*advanced ai techniques* → TOPIC
aaron magid → INSTRUCTOR	*cargo* → INSTRUCTOR
aaptis → DEPARTMENT	*survmeth* → DEPARTMENT
100 → NUMBER	*171* → NUMBER

Experiment#3

Data	Model	Token Acc.%		Seq. Acc.%	
		Standard	Difficult	Standard	Difficult
SCAN	RNN	99.48 ± 0.71	98.70 ± 0.92	98.27 ± 2.38	95.39 ± 2.72
	CNN	99.99 ± 0.01	98.59 ± 3.10	99.96 ± 0.03	96.66 ± 7.27
	TFM	96.90 ± 1.78	96.68 ± 2.21	91.94 ± 4.04	91.26 ± 5.80
GEO	RNN	54.44 ± 7.15	39.71 ± 18.38	13.61 ± 7.08	7.76 ± 5.34
	CNN	41.86 ± 3.38	41.07 ± 7.48	4.85 ± 4.66	4.04 ± 2.18
	TFM	67.02 ± 6.91	65.97 ± 5.17	36.38 ± 10.08	31.57 ± 7.42
ADV	RNN	36.50 ± 7.66	36.42 ± 7.39	12.84 ± 4.31	12.66 ± 5.19
	CNN	43.51 ± 11.31	35.34 ± 14.68	32.33 ± 12.93	23.58 ± 16.04
	TFM	56.82 ± 3.79	53.33 ± 3.85	47.43 ± 3.71	43.24 ± 5.14

Experiments

- Experiments over RNN on SCAN with vary- ing #primitives (a) and #variants (b).



Proof of Concept

Model	IWSLT'14				IWSLT'15			
	En-De		De-En		En-Fr		Fr-En	
	BLEU	SacreBLEU	BLEU	SacreBLEU	BLEU	SacreBLEU	BLEU	SacreBLEU
Baselines								
LSTM (Luong et al., 2015)	24.98	24.88	30.18	32.62	38.06	42.93	37.34	39.36
Transformer (Vaswani et al., 2017)	28.95	28.85	35.24	37.60	41.82	46.41	40.45	42.61
Dynamic Conv. (Wu et al., 2019)	27.39	27.28	33.33	35.54	40.41	45.32	39.61	41.42
+Vocabulary Augmentation								
LSTM (Luong et al., 2015)	$25.35\uparrow_{0.37}$	$25.38\uparrow_{0.50}$	$30.99\uparrow_{0.81}$	$33.63\uparrow_{1.01}$	$38.32\uparrow_{0.26}$	$43.30\uparrow_{0.37}$	$37.77\uparrow_{0.43}$	$39.83\uparrow_{0.47}$
Transformer (Vaswani et al., 2017)	$29.40\uparrow_{0.45}$	$29.29\uparrow_{0.44}$	$35.72\uparrow_{0.48}$	$38.07\uparrow_{0.47}$	$42.19\uparrow_{0.37}$	$46.68\uparrow_{0.27}$	$41.04\uparrow_{0.59}$	$43.15\uparrow_{0.54}$
Dynamic Conv. (Wu et al., 2019)	$27.60\uparrow_{0.21}$	$27.50\uparrow_{0.22}$	$33.62\uparrow_{0.29}$	$36.00\uparrow_{0.46}$	$40.87\uparrow_{0.46}$	$45.95\uparrow_{0.63}$	$39.95\uparrow_{0.34}$	$41.86\uparrow_{0.44}$

Proof of Concept

Model	Geography				Advising			
	Train		Test		Train		Test	
	Token Acc.%	Seq. Acc.%	Token Acc.%	Seq. Acc.%	Token Acc.%	Seq. Acc.%	Token Acc.%	Seq. Acc.%
Baselines								
RNN	89.05	17.39	69.81	9.68	92.22	3.64	60.41	6.11
CNN	98.45	70.74	78.44	55.91	99.74	81.62	81.74	51.13
TFM	99.45	84.95	80.24	49.82	99.68	76.90	78.51	29.67
+Entity Augmentation								
RNN	87.47	29.96	72.39$\uparrow_{2.58}$	15.05$\uparrow_{5.37}$	88.82	30.97	71.17$\uparrow_{10.76}$	16.06$\uparrow_{9.95}$
CNN	97.54	76.03	80.32$\uparrow_{1.88}$	60.93$\uparrow_{5.02}$	99.65	87.01	84.50$\uparrow_{2.76}$	56.02$\uparrow_{4.89}$
TFM	99.30	85.73	81.09$\uparrow_{0.85}$	54.84$\uparrow_{5.02}$	99.57	86.94	84.26$\uparrow_{5.75}$	35.08$\uparrow_{5.41}$

TLDR

Conclusion

- We revisit systematic generalization from a meaningful learning perspective by either inductive or deductive semantic linking.
- Modern seq2seq models can generalize to new concepts and compositions after semantic linking, which empirically answers the question by Lake and Baroni (2018).
- Both semantic linking and prior knowledge play a key role, in line with meaningful learning theory.
- Meaningful learning already benefits models in solving realistic problems

Q&A

arXiv

GitHub

Imitation Learning

Text Editing as Imitation Game

Ning Shi, Bin Tang, Bo Yuan, Longtao Huang, Yewen Pu, Jie Fu, and Zhouhan Lin

Findings of EMNLP 2022

Introduction

Text Editing

- Text simplification (e.g., dyslexia friendly)
- Grammatical error correction (e.g., Grammarly)
- Post processing (e.g., MT)
- Punctuation restoration (e.g., ASR)
- To name a few

Source Text (x)

1 1 2

↓

Target Text (y)

1 + 1 = 2

amii

Introduction

From End to End (End2end)

- Simplicity
- Good results
- Not much effort

But

- Copy mechanism
- Translate overlap

Source Text (x)

1 1 2 <pad>

↓

Target Text (y)

<s> 1 + 1 = 2 </s>

amil

Introduction

Sequence Tagging

(Token-level Action Generation)

- Tag <keep> for overlap

But

- Action bounded by token

Source Text (x)

1 1 2

↓

Target Tag (y′)

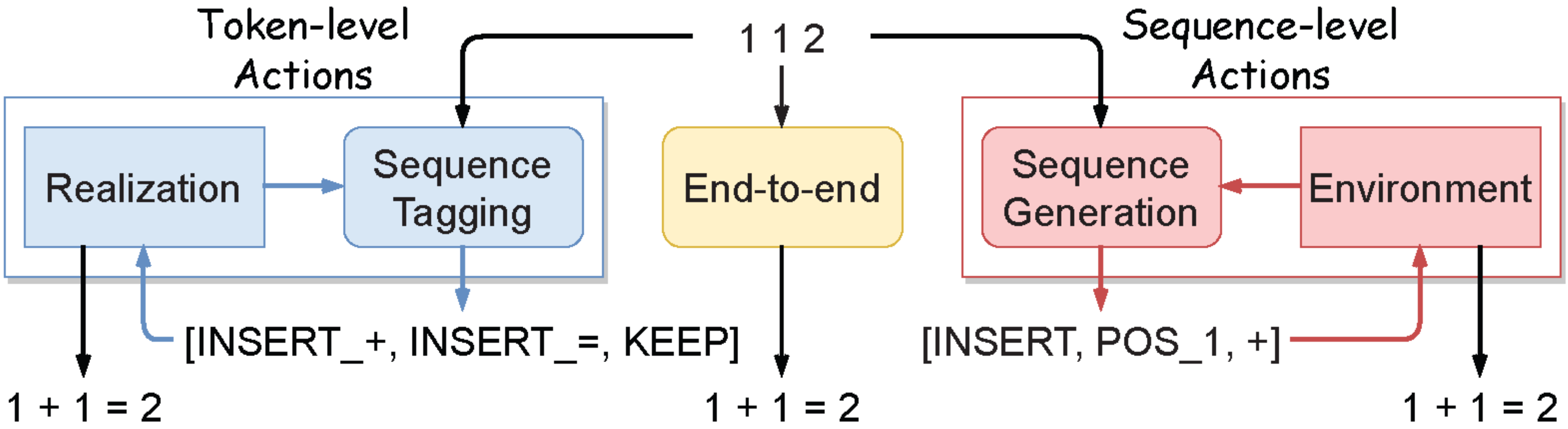
<insert_+> <insert_=> <keep>

Imitation Game

Imitation Learning (IL) & Recurrent Inference (Sequence-level Action Generation)

- Dynamic encoder context matrix
- Complex task decomposed into easier sub-tasks
- Highest degrees of action flexibility at sequence-level

Imitation Game



Three approaches – sequence tagging (left), end-to-end (middle), sequence generation (right).

Imitation Game

Markov Decision Process (MDP) Definition

- State S – a set of text sequences

Source text x as initial state s_1 (e.g., 1 1 2)

Target text y as target state s_T (e.g., 1 + 1 = 2)

Every edited texts as intermediate states s_t (e.g., 1 + 1 2)

Thus, the path $X \mapsto Y$ can be a set of sequential states $s_{<T}$

Imitation Game

Markov Decision Process (MDP) Definition

- State S – a set of text sequences
- Action A – a set of action sequences

Edit metric $\boldsymbol{E}$ (e.g., Levenshtein distance)

As long as $X \mapsto Y$ given $A_{\boldsymbol{E}}$

Examples: [INSERT, POS_3, =]

INSERT -> operation token

POS_3 -> position token

Imitation Game

Markov Decision Process (MDP) Definition

- State S – a set of text sequences
- Action A – a set of action sequences
- Transition matrix P – the probability that a_t leads s_t to s_{t+1}

Due to the nature of text editing, we know it is always 1, meaning always happen.

Imitation Game

Markov Decision Process (MDP) Definition

- State S – a set of text sequences
- Action A – a set of action sequences
- Transition matrix P – the probability that a_t leads s_t to s_{t+1}
- Environment $\mathcal{E}$ - to update state by $s_{t+1} = \mathcal{E}(s_t, a_t)$

The game environment is episodic and allows control of the editing process.

Imitation Game

Markov Decision Process (MDP) Definition

- State S – a set of text sequences
- Action A – a set of action sequences
- Transition matrix P – the probability that a_t leads s_t to s_{t+1}
- Environment $\mathcal{E}$ - to update state by $s_{t+1} = \mathcal{E}(s_t, a_t)$
- Reward function R – to calculate a reward for each action

In this work, we focus on behavior cloning (BC), so the reward function can be omitted for now.

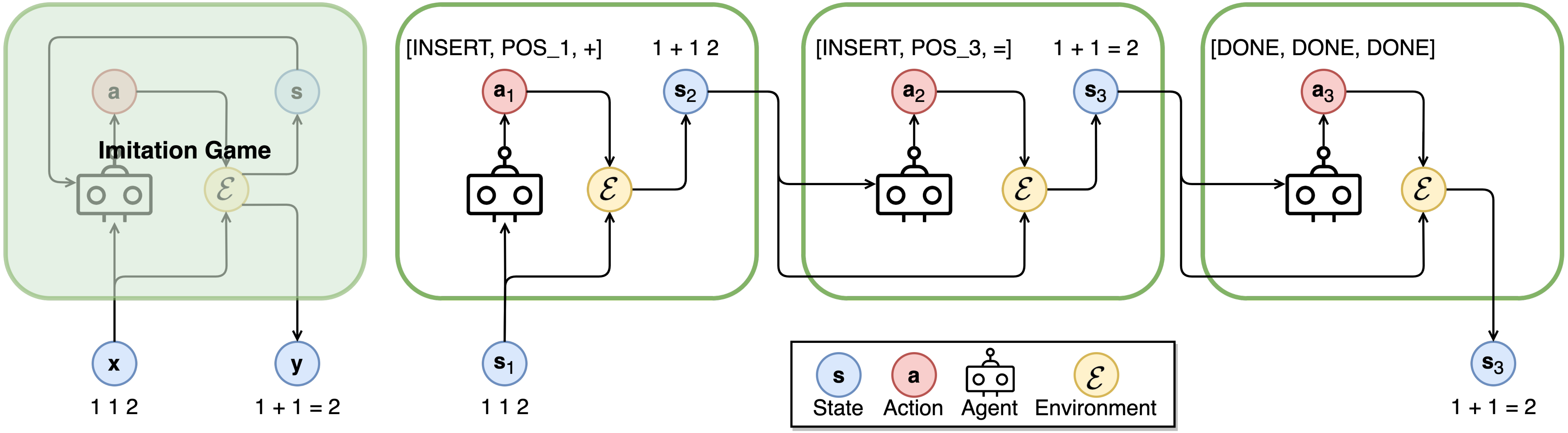
Imitation Game

Markov Decision Process (MDP) Definition

- State S – a set of text sequences
- Action A – a set of action sequences
- Transition matrix P – the probability that a_t leads s_t to s_{t+1}
- Environment $\mathcal{E}$ - to update state by $s_{t+1} = \mathcal{E}(s_t, a_t)$
- Reward function R – to calculate a reward for each action

The formulation turns out to be a simplified $M_{BC} = (S, A, \mathcal{E})$

Imitation Game



An example of the imitation game to complete "1 1 2" as "1 + 1 = 2".

Imitation Game

Trajectory Generation (TG)

How to convert conventional sequence-to-sequence data into state-to-action demonstrations?

Dynamic programming (DP) to back trace the minimum edit distance given the edit metric.

Algorithm 1 Trajectory Generation (TG)

Input: Initial state $\mathbf{x}$, goal state $\mathbf{y}$, environment $\mathcal{E}$, and edit metric $\mathbf{E}$.

Output: Trajectories τ.

$\tau \leftarrow \emptyset$

$\mathbf{s} \leftarrow \mathbf{x}$

$ops \leftarrow \mathrm{DP}(\mathbf{x}, \mathbf{y}, E)$

for $op \in ops$ **do**

  $\mathbf{a} \leftarrow \mathrm{Action}(op)$ ▷ Translate operation to action

  $\tau \leftarrow \tau \cup [(\mathbf{s}, \mathbf{a})]$

  $\mathbf{s} \leftarrow \mathcal{E}(\mathbf{s}, \mathbf{a})$

end for

$\tau \leftarrow \tau \cup [(\mathbf{s}, \mathbf{a}_T)]$ ▷ Append goal state and output action

return τ

Imitation Game

Trajectory Augmentation (TA)

IL suffers from distribution shift and error accumulation.

TA to expand the expert demonstrations and actively expose shifted states utilizing the divide-and-conquer technique.

Algorithm 2 Trajectory Augmentation (TA)

Input: States $\mathbf{S}$, state $\mathbf{s}_t$, expert states $\mathbf{S}^*$, actions $\mathbf{A}$, and environment $\mathcal{E}$.
Output: Augmented states $\mathbf{S}$.

if $|\mathbf{A}| > 1$ **then**
    $\mathbf{a}_t \leftarrow \mathbf{A}.\mathrm{pop}(0)$
    $\mathbf{s}_{t+1} \leftarrow \mathcal{E}(\mathbf{s}_t, \mathbf{a}_t)$
    $\mathbf{S} \leftarrow \mathbf{S} \cup \mathrm{TA}(\mathbf{S}, \mathbf{s}_{t+1}, \mathbf{S}^*, \mathbf{A}, \mathcal{E})$ ▷ Execute action
    $\mathbf{A} \leftarrow \mathrm{Update}(\mathbf{A}, \mathbf{s}_t, \mathbf{s}_{t+1})$
    $\mathbf{S} \leftarrow \mathbf{S} \cup \mathrm{TA}(\mathbf{S}, \mathbf{s}_t, \mathbf{S}^*, \mathbf{A}, \mathcal{E})$ ▷ Skip action
else if $\mathbf{s}_t \notin \mathbf{S}^*$ **then**
    $\mathbf{S} \leftarrow \mathbf{S} \cup [\mathbf{s}_t]$ ▷ Merge shifted state
end if
return $\mathbf{S}$

Imitation Game

Trajectory Augmentation (TA)

Advantages:

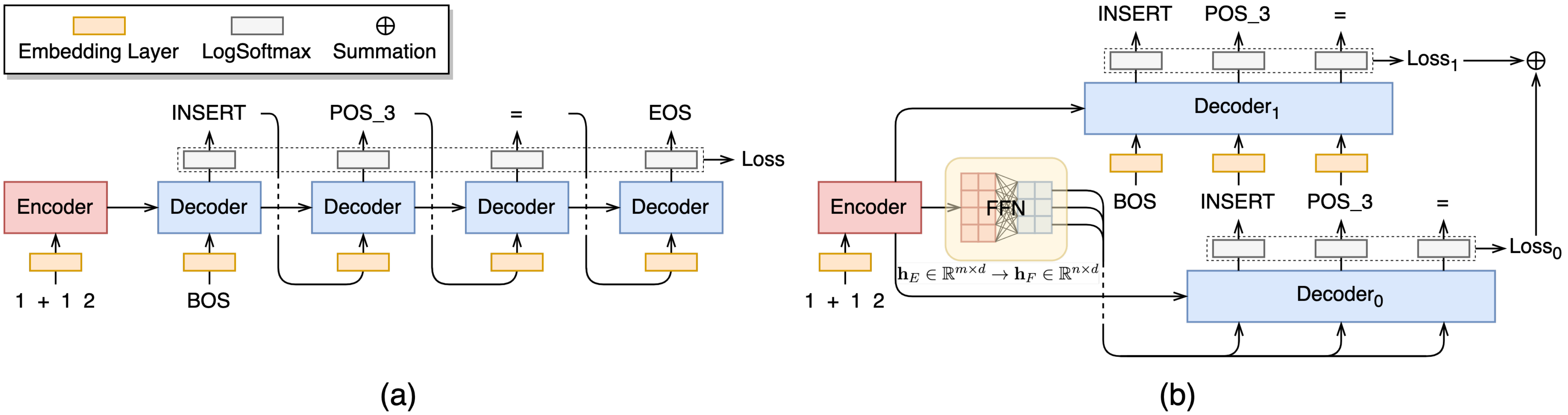
- To preserve the i.i.d. assumption
- No dependency on the task
- No domain knowledge
- No labeling work
- No further evaluation

Algorithm 2 Trajectory Augmentation (TA)

Input: States $\mathbf{S}$, state $\mathbf{s}_t$, expert states $\mathbf{S}^*$, actions $\mathbf{A}$, and environment $\mathcal{E}$.
Output: Augmented states $\mathbf{S}$.

if $|\mathbf{A}| > 1$ **then**
    $\mathbf{a}_t \leftarrow \mathbf{A}.\text{pop}(0)$
    $\mathbf{s}_{t+1} \leftarrow \mathcal{E}(\mathbf{s}_t, \mathbf{a}_t)$
    $\mathbf{S} \leftarrow \mathbf{S} \cup \text{TA}(\mathbf{S}, \mathbf{s}_{t+1}, \mathbf{S}^*, \mathbf{A}, \mathcal{E})$ ▷ Execute action
    $\mathbf{A} \leftarrow \text{Update}(\mathbf{A}, \mathbf{s}_t, \mathbf{s}_{t+1})$
    $\mathbf{S} \leftarrow \mathbf{S} \cup \text{TA}(\mathbf{S}, \mathbf{s}_t, \mathbf{S}^*, \mathbf{A}, \mathcal{E})$ ▷ Skip action
else if $\mathbf{s}_t \notin \mathbf{S}^*$ **then**
    $\mathbf{S} \leftarrow \mathbf{S} \cup [\mathbf{s}_t]$ ▷ Merge shifted state
end if
return $\mathbf{S}$

Non-Autoregressive Decoding



The conventional autoregressive decoder (a) compared with the proposed non-autoregressive D2 (b) in which the linear layer aligns the sequence length dimension for the subsequent parallel decoding.

Arithmetic Equation (AE)

AOR ($N = 10$, $L = 5$, $D = 10$K)			**AES** ($N = 100$, $L = 5$, $D = 10$K)			**AEC** ($N = 10$, $L = 5$, $D = 10$K)		
Train/Valid/Test	**Train TA**	**Traj. Len.**	**Train/Valid/Test**	**Train TA**	**Traj. Len.**	**Train/Valid/Test**	**Train TA**	**Traj. Len.**
7,000/1,500/1,500	145,176	6	7,000/1,500/1,500	65,948	6	7,000/1,500/1,500	19,764	4

Table 1: Data statistics of AE benchmarks.

Term	**AOR** ($N = 10$, $L = 5$, $D = 10$K)	**AES** ($N = 100$, $L = 5$, $D = 10$K)	**AEC** ($N = 10$, $L = 5$, $D = 10$K)
Source $\mathbf{x}$	3 6 2 9 3	65 + (25 - 20) - (64 + 32) + (83 - 24) = (- 25 + 58)	- 2 * + 4 10 + 8 / 8 = 8
Target $\mathbf{y}$	- 3 - 6 / 2 + 9 = 3	65 + 5 - 96 + 59 = 33	- 2 + 10 * 8 / 8 = 8
State $\mathbf{s}_t^*$	- 3 - 6 / 2 9 3	65 + 5 - (64 + 32) + (83 - 24) = (- 25 + 58)	- 2 + **4** 10 + 8 / 8 = 8
Action $\mathbf{a}_t^*$	[POS_6, +]	[POS_4, POS_8, 96]	[DELETE, POS_3, POS_3]
Next State $\mathbf{s}_{t+1}^*$	- 3 - 6 / 2 **+** 9 3	65 + 5 - **96** + (83 - 24) = (- 25 + 58)	- 2 + 10 + 8 / 8 = 8
Shifted State $\mathbf{s}_t'$	- 3 - 6 / 2 9 **=** 3	65 + 5 - (64 + 32) + **59** = (- 25 + 58)	- 2 + 4 10 ***** 8 / 8 = 8

Table 2: Examples from AE with specific N for integer size, L for the number of integers, and D for data size.

AE benchmarks: Arithmetic Operators Restoration (AOR), Arithmetic Equation Simplification (AES), and Arithmetic Equation Correction (AEC)

Models

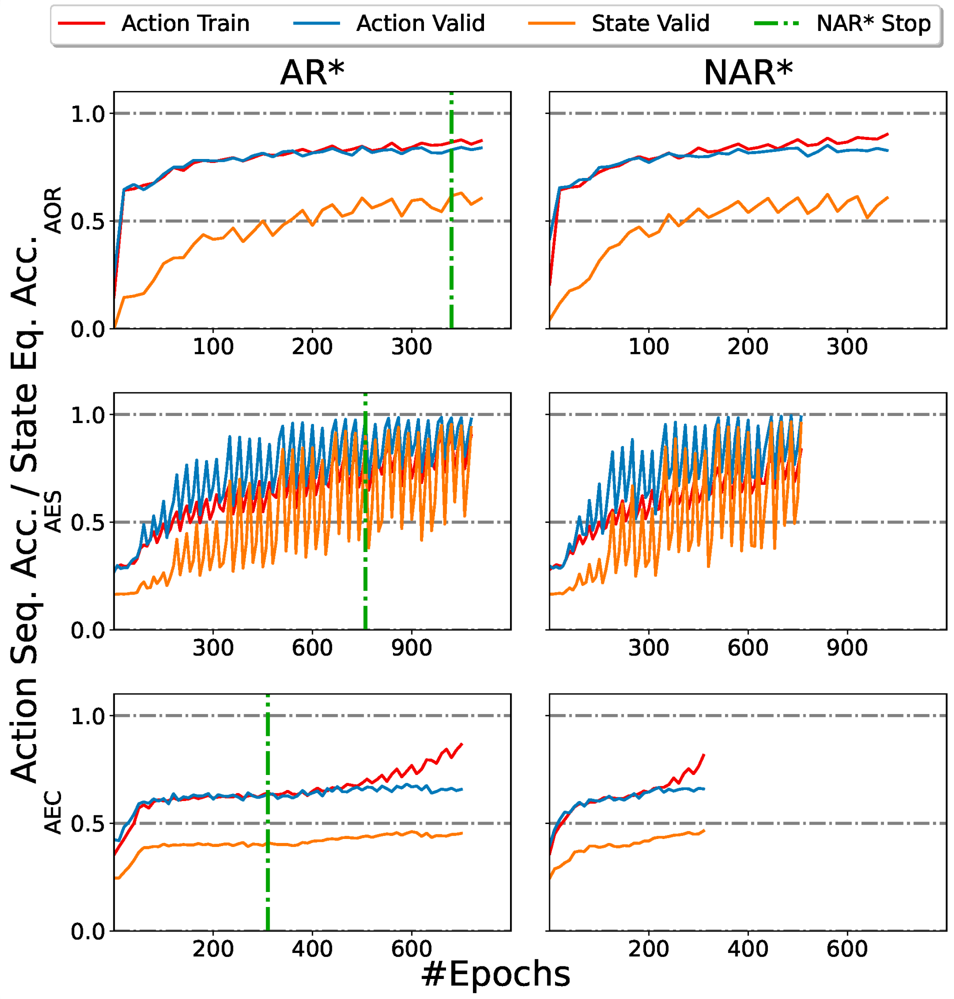
- **End2end** – translate x to y from end to end
- **Tagging** – token level action
- **Recurrence** – recurrent inference via autoregressive LSTM
- **Recurrence*** - rerun the source code of Recurrence that only has access to the fixed training set
- **AR** – our reproduction of Recurrence* in our pipeline
- **AR*** - increase the encoder layers in AR from 1 to 4
- **NAR** – replace autoregressive decoder of AR* with a linear layer to enable non-autoregressive decoding
- **NAR*** - our method with D2 non-autoregressive decoder
- **+TA** – enable trajectory augmentation

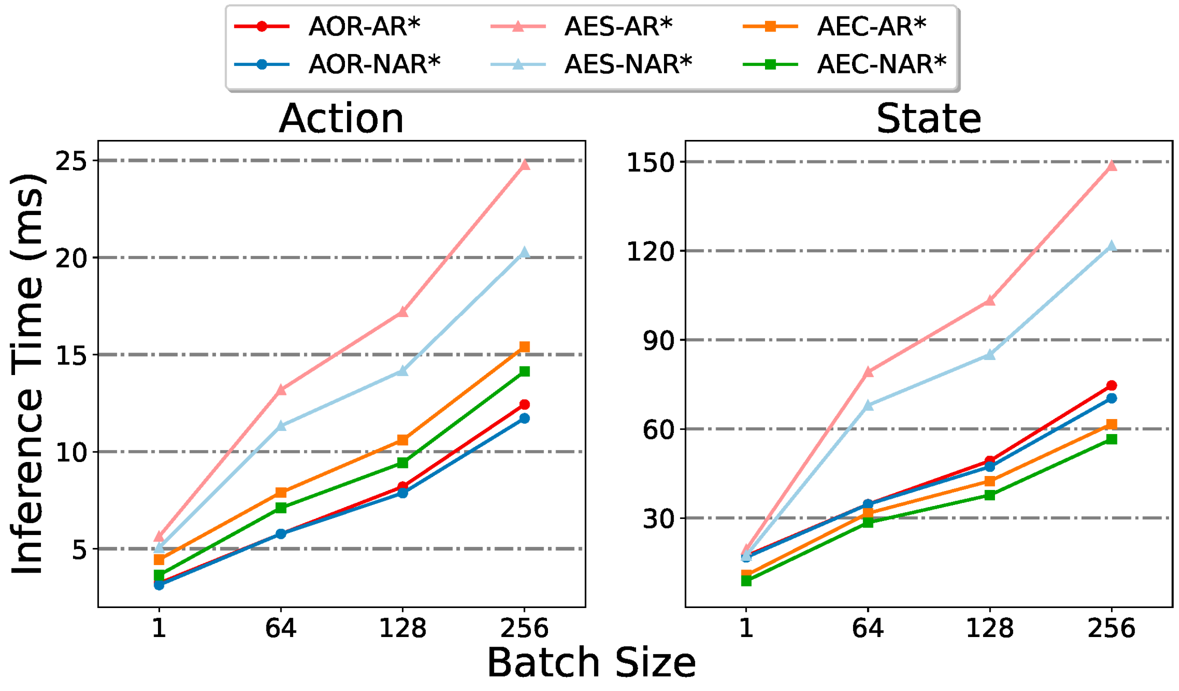
Experimental Results

Method	**AOR** ($N = 10$, $L = 5$, $D = 10$K)			**AES** ($N = 100$, $L = 5$, $D = 10$K)		**AEC** ($N = 10$, $L = 5$, $D = 10$K)		
	Tok. Acc. %	**Seq. Acc. %**	**Eq. Acc. %**	**Tok. Acc. %**	**Eq. Acc. %**	**Tok. Acc. %**	**Seq. Acc. %**	**Eq. Acc. %**
End2end	–	–	29.33	84.60	25.20	88.08	57.27	57.73
Tagging	–	–	51.40	87.00	36.67	84.46	46.93	47.33
Recurrence	–	–	58.53	98.63	87.73	83.64	57.47	58.27
Recurrence*	60.30 ± 1.30	27.31 ± 1.33	56.73 ± 1.33	79.82 ± 0.37	22.28 ± 0.52	82.32 ± 0.56	41.72 ± 0.74	42.13 ± 0.75
AR	61.85 ± 0.51	28.83 ± 1.14	59.09 ± 0.95	88.12 ± 2.37	37.05 ± 6.57	$\mathbf{82.61 \pm 0.53}$	45.81 ± 0.36	46.31 ± 0.31
AR*	62.51 ± 0.62	$\mathbf{30.85 \pm 0.41}$	61.35 ± 0.33	99.27 ± 0.32	93.57 ± 2.91	82.29 ± 0.39	45.99 ± 0.49	46.35 ± 0.52
NAR	59.72 ± 0.70	24.16 ± 1.16	51.64 ± 1.97	83.87 ± 1.60	29.49 ± 2.51	80.28 ± 0.76	44.91 ± 1.71	45.40 ± 1.78
NAR*	$\mathbf{62.81 \pm 0.89}$	30.13 ± 1.31	$\mathbf{61.45 \pm 1.61}$	$\mathbf{99.51 \pm 0.13}$	$\mathbf{95.67 \pm 0.93}$	81.82 ± 0.68	$\mathbf{45.97 \pm 1.07}$	$\mathbf{46.43 \pm 1.10}$
AR +TA	62.35 ± 0.61	32.28 ± 0.67	63.56 ± 1.06	88.05 ± 1.20	38.39 ± 3.45	$\mathbf{83.94 \pm 0.42}*$	49.36 ± 1.23	49.83 ± 1.21
AR* +TA	62.58 ± 0.63	33.01 ± 1.31	65.73 ± 1.38	99.44 ± 0.27	95.24 ± 2.38	83.39 ± 0.74	48.95 ± 0.65	49.47 ± 0.73
NAR +TA	61.30 ± 0.86	32.04 ± 1.99	63.75 ± 2.08	90.38 ± 2.21	47.91 ± 8.18	81.36 ± 0.40	48.01 ± 1.07	48.47 ± 1.15
NAR* +TA	$\mathbf{63.48 \pm 0.38}^*$	$\mathbf{34.23 \pm 0.92}^*$	$\mathbf{67.13 \pm 0.99}^*$	$\mathbf{99.58 \pm 0.15}^*$	$\mathbf{96.44 \pm 1.29}^*$	82.70 ± 0.42	$\mathbf{49.64 \pm 0.59}^*$	$\mathbf{50.15 \pm 0.55}^*$

Table 3: Evaluation results on AOR, AES, and AEC with specific N, L, and D. The token and sequence accuracy for AOR were not reported, thus we leave these positions blank here. With or without TA, our proposed NAR* achieves the best performance in terms of equation accuracy across the board.

Experimental Results





Analysis

Action Design

Due to the liberty of sequence generation, the same operation can be represented as different action sequences by, for example, a simple swap of action tokens.

Our NAR* stays nearly consistent across three designs.

Design	Action Sequence	Method	Tok. Acc. %	Eq. Acc. %
#1	[Pos.$_L$, Pos.$_R$, Tok.]	AR*	99.27 ± 0.32	93.57 ± 2.91
		NAR*	$\mathbf{99.51 \pm 0.13}$	$\mathbf{95.67 \pm 0.93}$
		AR* +TA	99.44 ± 0.27	95.24 ± 2.38
		NAR* +TA	$\mathbf{99.58 \pm 0.15}^*$	$\mathbf{96.44 \pm 1.29}^*$
#2	[Pos.$_L$, Tok., Pos.$_R$]	AR*	99.08 ± 0.93	92.35 ± 7.21
		NAR*	$\mathbf{99.50 \pm 0.27}$	$\mathbf{95.55 \pm 2.28}$
		AR* +TA	99.52 ± 0.29	95.68 ± 2.49
		NAR* +TA	$\mathbf{99.54 \pm 0.20}^*$	$\mathbf{95.97 \pm 1.64}^*$
#3	[Tok., Pos.$_L$, Pos.$_R$]	AR*	98.06 ± 0.79	83.79 ± 6.25
		NAR*	$\mathbf{99.53 \pm 0.14}$	$\mathbf{95.99 \pm 0.81}$
		AR* +TA	98.43 ± 0.49	87.29 ± 3.70
		NAR* +TA	$\mathbf{99.61 \pm 0.06}^*$	$\mathbf{96.55 \pm 0.46}^*$

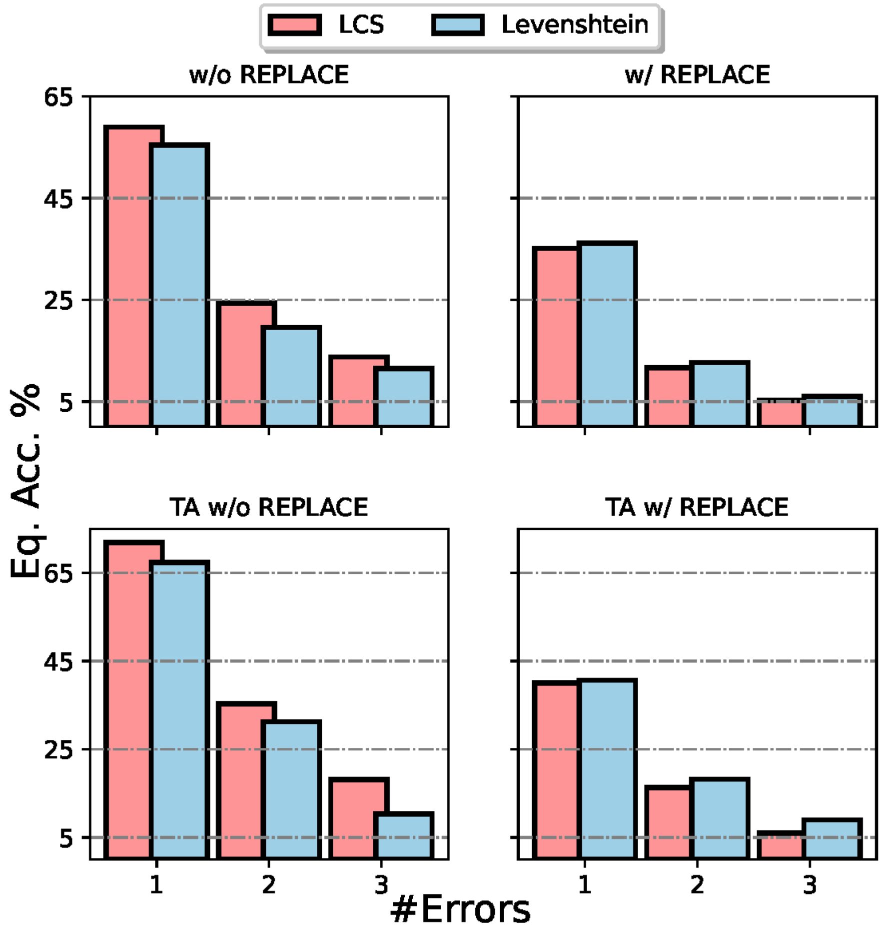
Table 4: Evaluation of AR* and NAR* in AES across three action designs that vary from each other by token order. They directs to the same operation with Pos.$_L$/Pos.$_R$/Tok. denoting left parenthesis/right parenthesis/target token.

Analysis

Trajectory Optimization

A better edit metric $\boldsymbol{E}$ often means a smaller action vocabulary space, shorter trajectory length, and, therefore, an easier IL.

An appropriate edit metric $\boldsymbol{E}$ depends on the specific task.



Analysis

Dual Decoders

As an ablation study, we freeze the encoder of NAR* and vary its decoder to reveal the contributions of each component in D2.

- **Linear** – replace the decoder with a linear layer
- **Decoder$_0$** – remove the second decoder from D2
- **Shared D2** – share the parameters between two decoders in D2
- **D2 (NAR*)** – our method with D2 non-autoregressive decoder
- **+TA** – enable trajectory augmentation

Analysis

Dual Decoders

As an ablation study, we freeze the encoder of NAR* and vary its decoder to reveal the contributions of each component in D2.

Decoder	**AOR** ($N = 10$, $L = 5$, $D = 10$K)			**AES** ($N = 100$, $L = 5$, $D = 10$K)		**AEC** ($N = 10$, $L = 5$, $D = 10$K)		
	Tok. Acc. %	**Seq. Acc. %**	**Eq. Acc. %**	**Tok. Acc. %**	**Eq. Acc. %**	**Tok. Acc. %**	**Seq. Acc. %**	**Eq. Acc. %**
Linear	61.84 ± 0.94	28.55 ± 1.57	57.72 ± 1.55	99.41 ± 0.26	95.01 ± 2.01	81.35 ± 0.92	42.47 ± 1.85	42.81 ± 1.87
Decoder_0	61.78 ± 0.83	28.20 ± 1.57	58.36 ± 1.58	99.24 ± 0.23	93.49 ± 2.03	80.84 ± 0.66	43.97 ± 1.82	44.32 ± 1.82
Shared D2	61.74 ± 0.71	28.68 ± 0.94	58.05 ± 1.01	99.28 ± 0.24	93.85 ± 2.14	81.38 ± 1.04	43.64 ± 2.03	44.09 ± 2.02
D2 (NAR*)	$\mathbf{62.81 \pm 0.89}$	$\mathbf{30.13 \pm 1.31}$	$\mathbf{61.45 \pm 1.61}$	$\mathbf{99.51 \pm 0.13}$	$\mathbf{95.67 \pm 0.93}$	$\mathbf{81.82 \pm 0.68}$	$\mathbf{45.97 \pm 1.07}$	$\mathbf{46.43 \pm 1.10}$
Linear +TA	61.41 ± 0.28	31.75 ± 0.93	63.15 ± 0.96	99.42 ± 0.17	95.08 ± 1.47	81.54 ± 0.66	46.79 ± 2.26	47.33 ± 2.30
Decoder_0 +TA	62.50 ± 1.24	32.48 ± 1.87	64.47 ± 1.88	99.47 ± 0.13	95.33 ± 1.13	82.02 ± 0.40	46.80 ± 2.04	47.32 ± 1.91
Shared D2 +TA	61.64 ± 0.87	31.21 ± 0.34	62.77 ± 0.85	99.53 ± 0.12	95.91 ± 1.25	81.80 ± 0.47	47.23 ± 1.07	47.61 ± 1.14
D2 (NAR*) +TA	$\mathbf{63.48 \pm 0.38}^*$	$\mathbf{34.23 \pm 0.92}^*$	$\mathbf{67.13 \pm 0.99}^*$	$\mathbf{99.58 \pm 0.15}^*$	$\mathbf{96.44 \pm 1.29}^*$	$\mathbf{82.70 \pm 0.42}^*$	$\mathbf{49.64 \pm 0.59}^*$	$\mathbf{50.15 \pm 0.55}^*$

Table 6: Evaluation of agents equipped with same encoders but different decoders on AE benchmarks.

Conclusion

Contributions:

- Frame text editing into an imitation game

This allows the *highest* degree of flexibility to design actions at the sequence-level, which are arguably more *controllable*, *interpretable*, and *similar* to human behavior.

Conclusion

Contributions:

- Frame text editing into an imitation game
- We involve TG to translate standard datasets

Free to translate the conventional input-output data to state-action demonstrations for a friendly IL.

amii

Conclusion

Contributions:

- Frame text editing into an imitation game
- We involve TG to translate standard datasets
- We introduce D2 as a novel non-autoregressive decoder

To boost the learning in terms of *accuracy*, *efficiency*, and *robustness*

Conclusion

Contributions:

- Frame text editing into an imitation game
- We involve TG to translate standard datasets
- We introduce D2 as a novel non-autoregressive decoder
- We propose TA technique

To mitigate the distribution shift problem IL often suffers

Conclusion

Contributions:

- Frame text editing into an imitation game
- We involve TG to translate standard datasets
- We introduce D2 as a novel non-autoregressive decoder
- We propose TA technique

Future work:

- Reward function, action design, trajectory optimization

amii

Conclusion

Limitations

- Efficiency issue due to multiple calls of encoder (e.g., a heavy pretrained language model)
- Application in more realistic editing tasks (e.g., text simplification)

TLDR

Turning tasks into games that agents feel more comfortable with sheds light on future studies in the direction of reinforcement learning in the application of text editing.

amii

Thanks

arXiv

GitHub