Coupling Retrieval and Meta-learning for Context-dependent Semantic Parsing

Daya Guo[1], Duyu Tang[2], Nan Duan[2], Ming Zhou[2] and Jian Yin[1]

[1]Sun Yat-sen University [2]Microsoft Research

Task definition

- **Context-Dependent Semantic Parsing**
  - Input: a natural language with context
  - Output: a structural logical form

Task definition

- **Context-Dependent Semantic Parsing**
  - Input: a natural language with context
  - Output: a structural logical form

Task I: Conversational Question Answering over KB
Conversational history (Context): Q1: Where was the president of the United States born? A1: New York City **Current Question:** Q2: Where did he graduate from? **Output :** $find(find(United\ States, isPresidentof), graduateFrom)$ $\downarrow$ Executed on KB A2: University of Pennsylvanian

Task definition

- **Context-Dependent Semantic Parsing**
  - Input: a natural language with context
  - Output: a structural logical form

Task I: Conversational Question Answering over KB
Conversational history (Context): Q1: Where was the president of the United States born? A1: New York City **Current Question:** Q2: Where did he graduate from? **Output:** $find(find(United\ States, isPresidentof), graduateFrom)$ $\downarrow$ Executed on KB A2: University of Pennsylvanian

Task II: Code Generation
Class Environment: **Variables: [Type, Name]** double[] vecElements double[] weights **Methods: [Return Type, Name]** void add(double) **Natural Language Query:** Increment this vector in this place **Source Code:** public void inc() { this.add(1);}

Related works

- Standard methods
  - Sequence-to-Sequence approaches: translate a sentence to a logical form [Dong and Lapata, 2016; Jia and Liang, 2016; Ling et al., 2016; Xiao et al., 2016]
  - Sequence-to-Action approaches: generate an action sequence that can construct logical forms [Yin and Neubig, 2017; Krish-namurthy et al., 2017; Iyyer et al., 2017; Chen et al., 2018]

Related works

- Standard methods
  - Sequence-to-Sequence approaches: translate a sentence to a logical form [Dong and Lapata, 2016; Jia and Liang, 2016; Ling et al., 2016; Xiao et al., 2016]
  - Sequence-to-Action approaches: generate an action sequence that can construct logical forms [Yin and Neubig, 2017; Krish-namurthy et al., 2017; Iyyer et al., 2017; Chen et al., 2018]

- Retrieve-and-Edit based methods
  - Increase the probability of actions that can derive the retrieved subtrees [Hayati et al., 2018]
  - Treat each example with retrieved datapoints as a new task [Huang et al., 2018a]
  - Encoder-decoder based retrieval model [Hashimoto et al., 2018]

Motivation

- Programmers usually leverage similar examples as a guidance
- The pattern of a structural output may come from multiple examples

Current Question:

Which rivers flow through India but does not originate in China?

Logical Form:

$$\boxed{find(India, river\_flow\_through)}$$

$$-\ \boxed{find(China, river\_\text{originate})}$$

Retrieved Question Q1:

Which rivers flow through India?

Logical Form:

$$\boxed{find(India, river\_flow\_through)}$$

Retrieved Question Q2:

Which rivers originates in China?

Logical Form:

$$\boxed{find(China, river\_\text{originate})}$$

Motivation

- Programmers usually leverage similar examples as a guidance
- The pattern of a structural output may come from multiple examples

Current Question:

Which rivers flow through India but does not originate in China?

Logical Form:

$$find(India, river\_flow\_through) - find(China, river\_\text{originate})$$

Retrieved Question Q1:

Which rivers flow through India?

Logical Form:

$$find(India, river\_flow\_through)$$

Retrieved Question Q2:

Which rivers originates in China?

Logical Form:

$$find(China, river\_\text{originate})$$

- How to automatically retrieve similar examples?
- How to use them as the supporting evidence to improve semantic parsing?

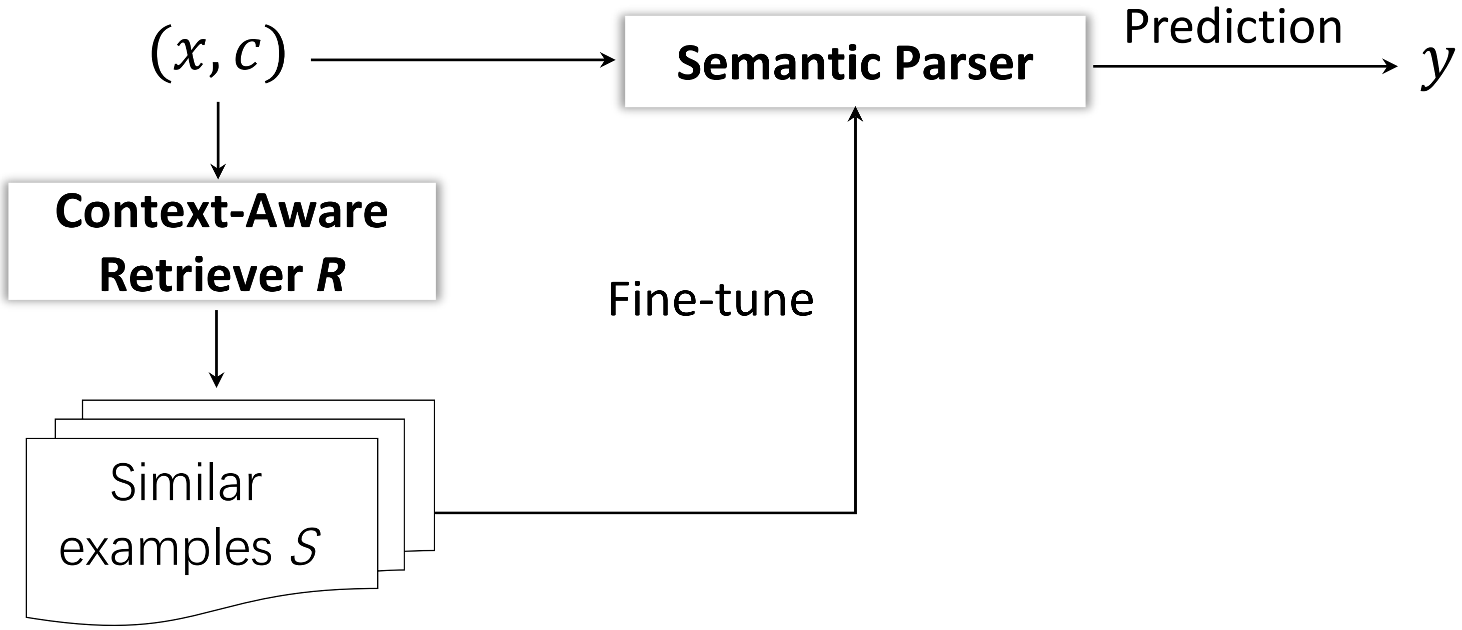
Overview

- Standard prediction

Overview

- Standard prediction

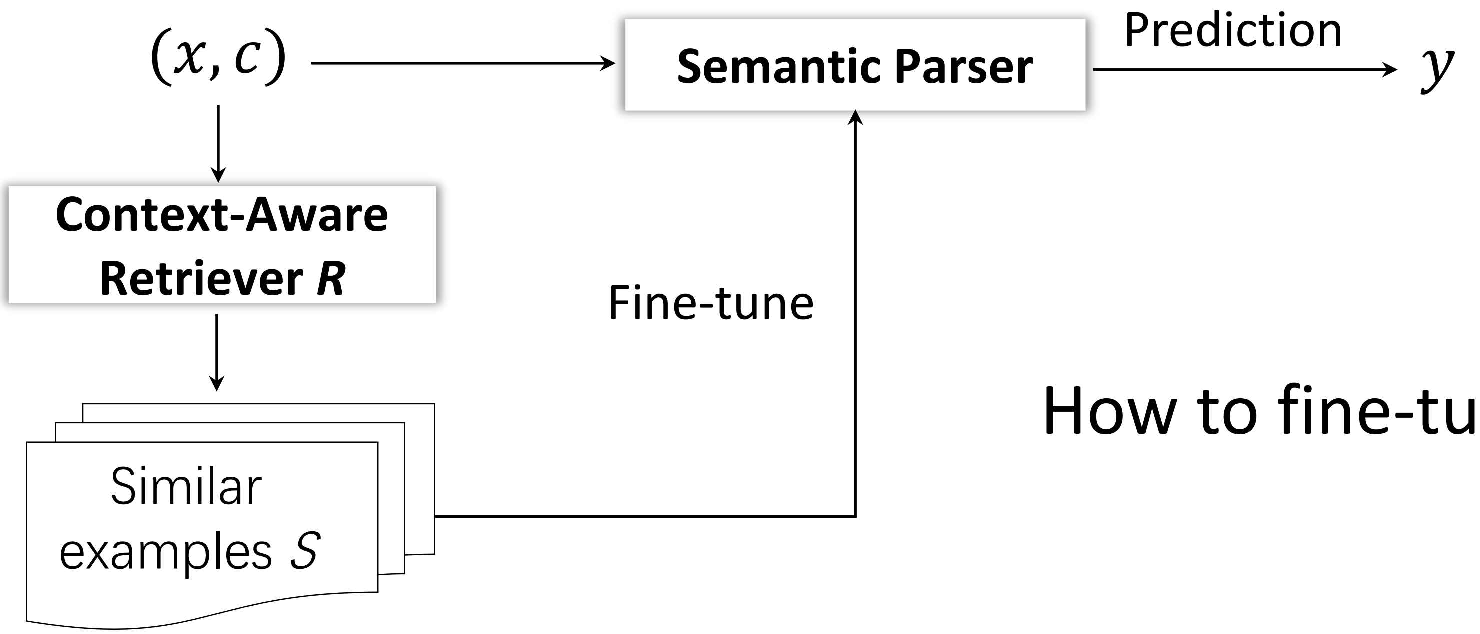
- Fine-tune semantic parser before prediction



Overview

- Standard prediction

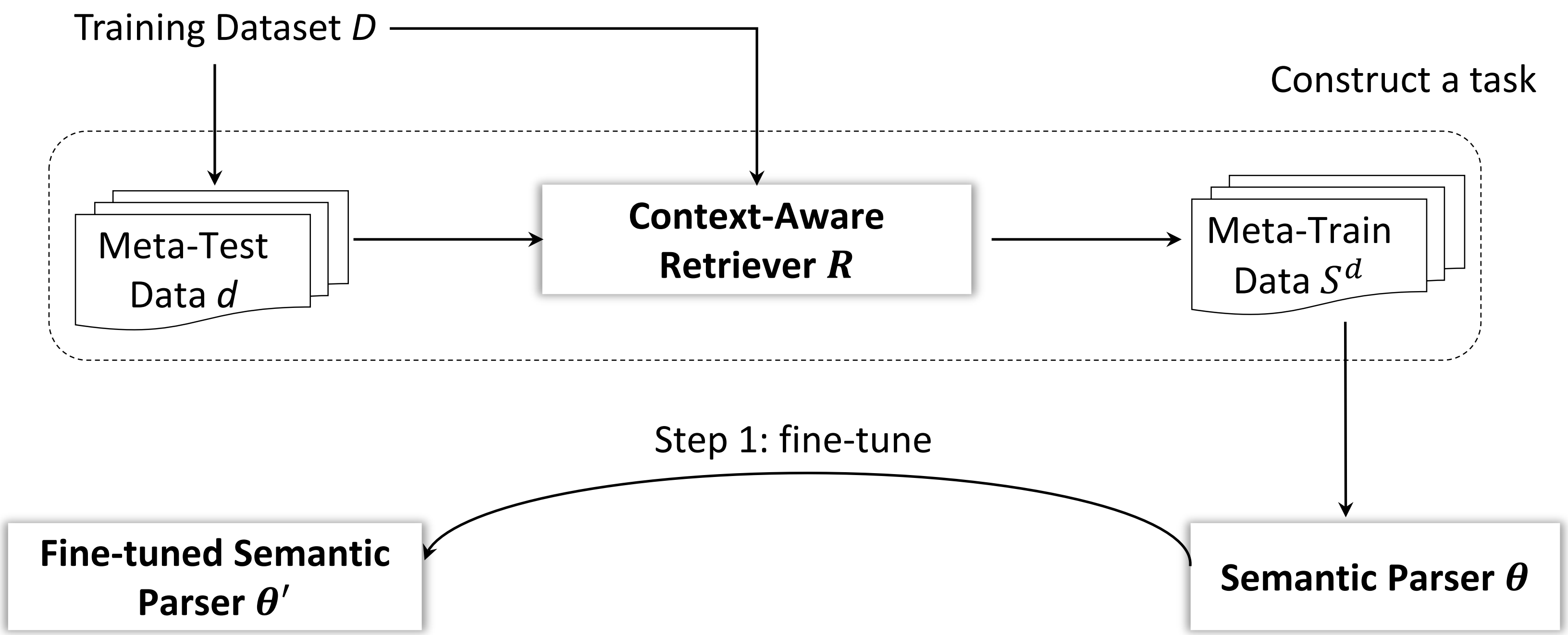
- Fine-tune semantic parser before prediction



How to fine-tune the semantic parser?

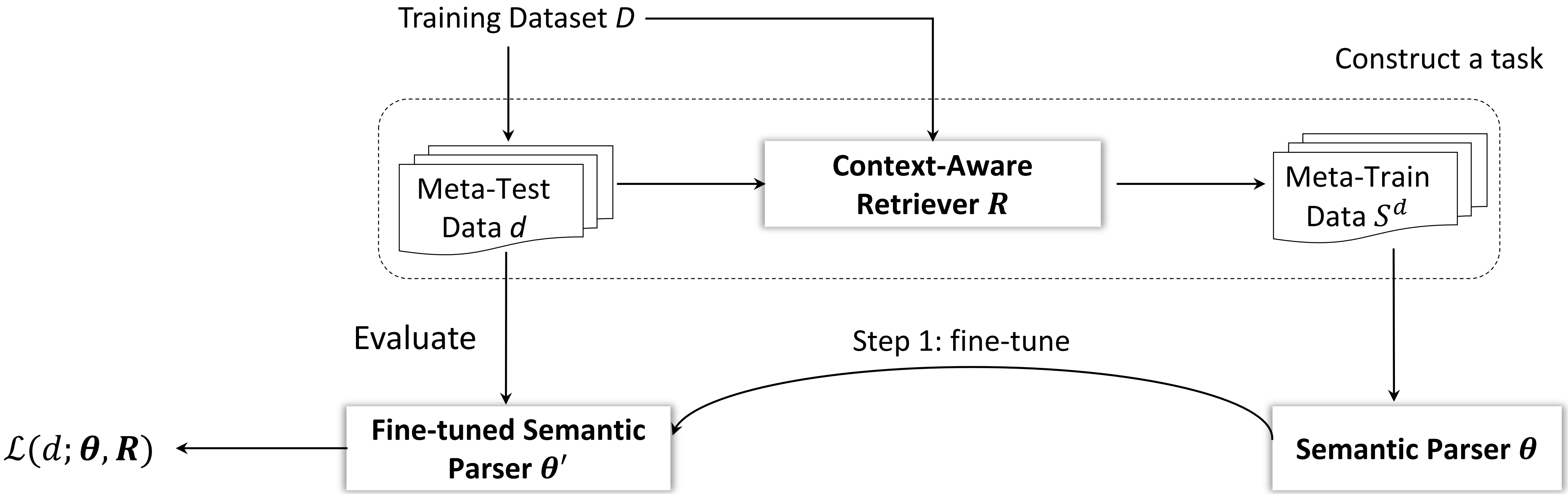
Framework

- Fine-tune semantic parser



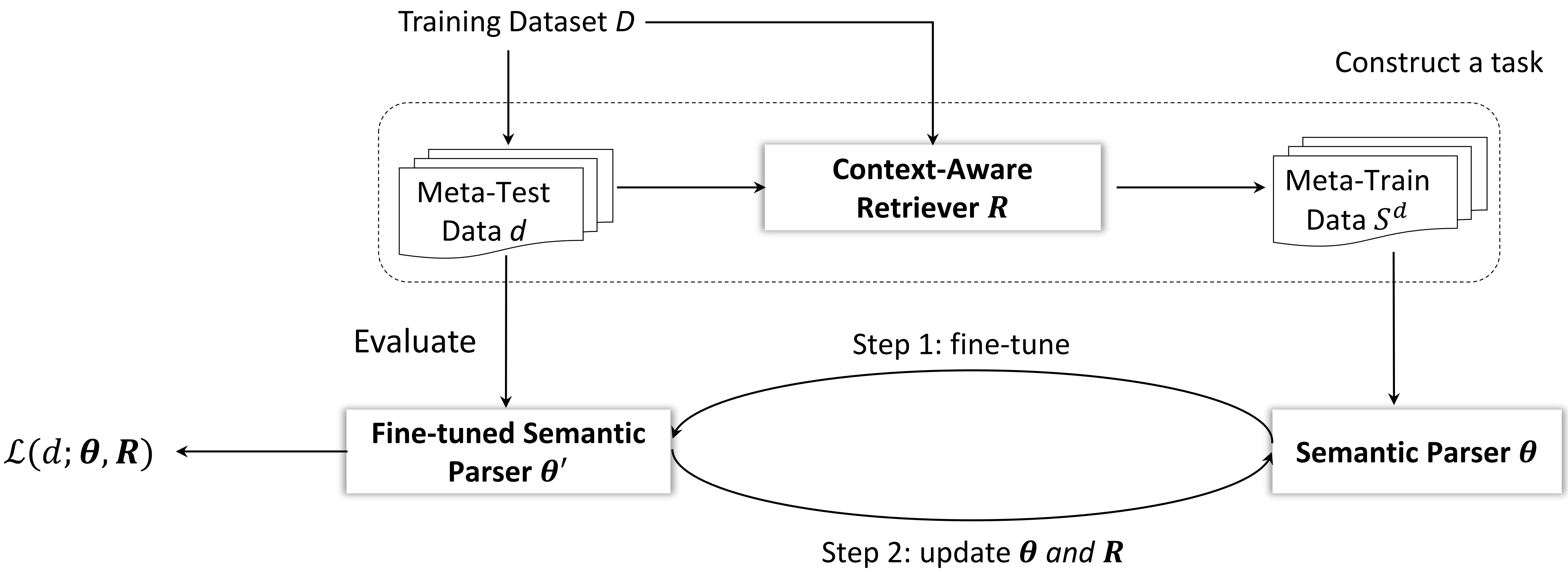
Framework

- Calculate the loss function $\mathcal{L}(d; \boldsymbol{\theta}, \boldsymbol{R})$

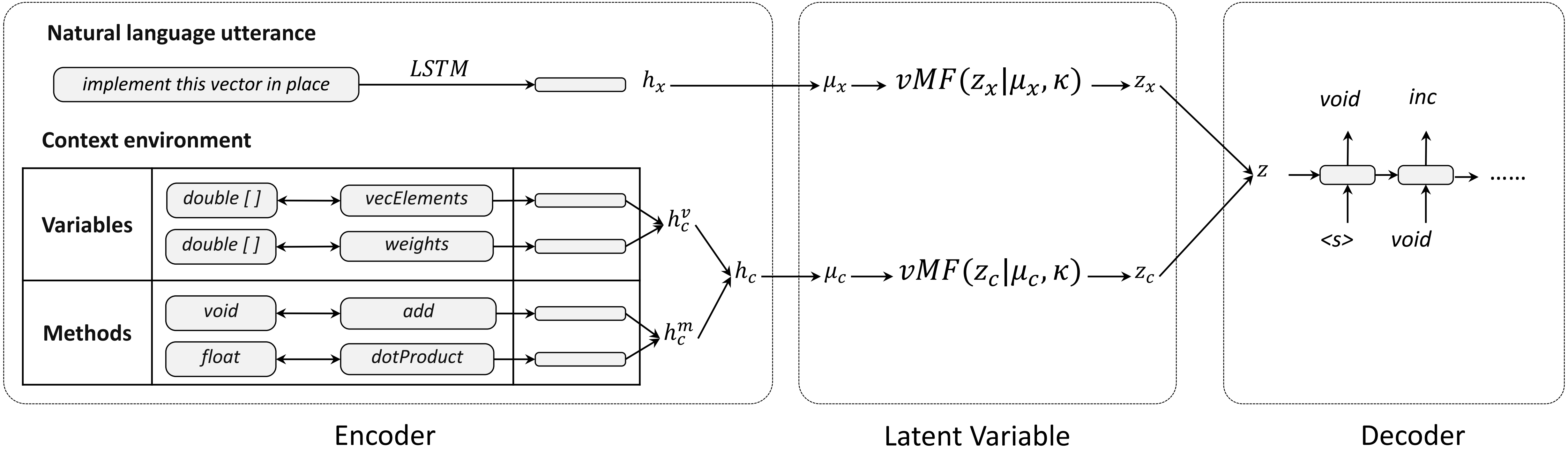


Framework

- Update semantic parser and retriever by meta-learning

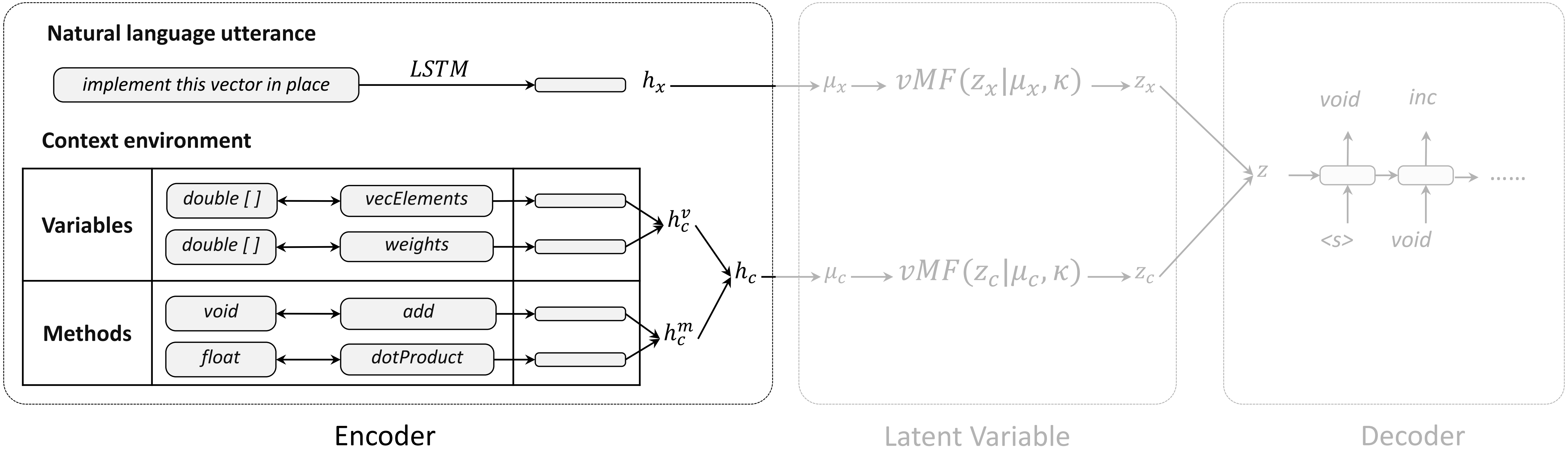


Context-Aware Retrieval Model



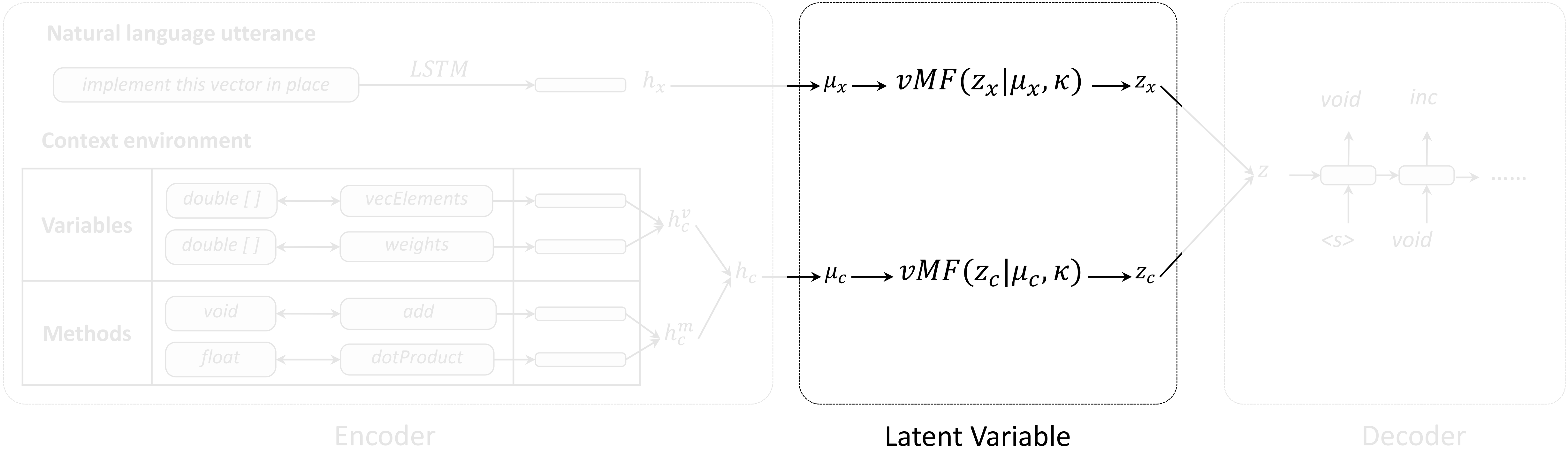
- The retrieval model is based on variational autoencoder framework
- Train the model using semantic parsing supervised dataset

Context-Aware Retrieval Model



- Encode a natural language with the context environment into hidden states

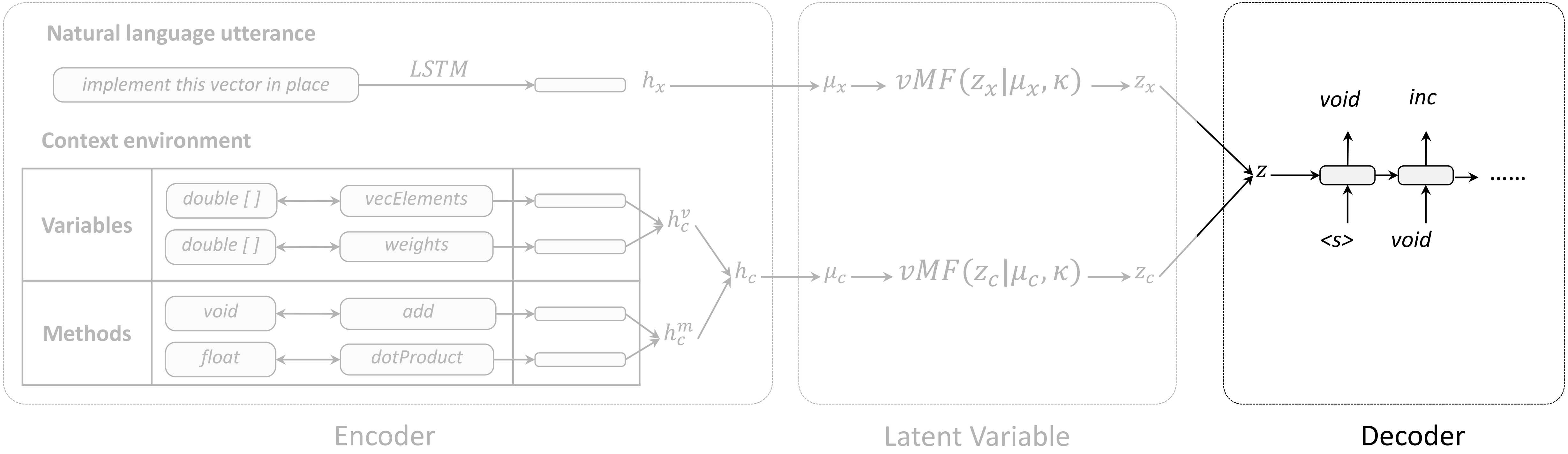
Context-Aware Retrieval Model



- Our choice of encoder noise: the von-Mises Fisher (vMF) distribution

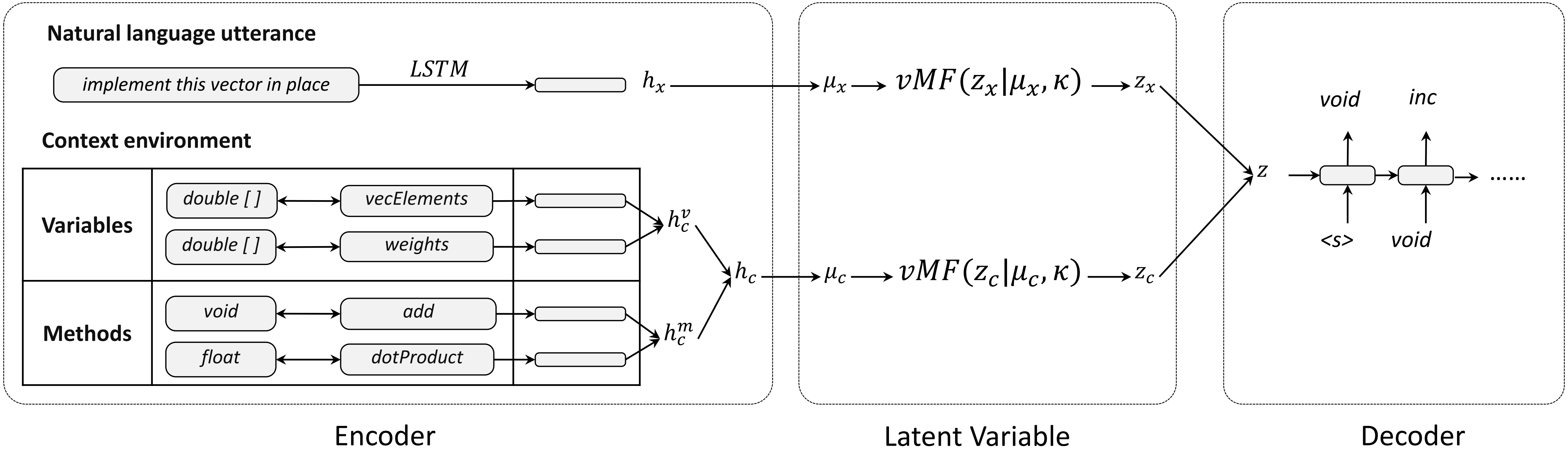
$$p(z_x|x) = vMF(z_x; \mu_x, k) = Z_k^{-1} e^{(\kappa \mu_k^T z_x)}$$

Context-Aware Retrieval Model



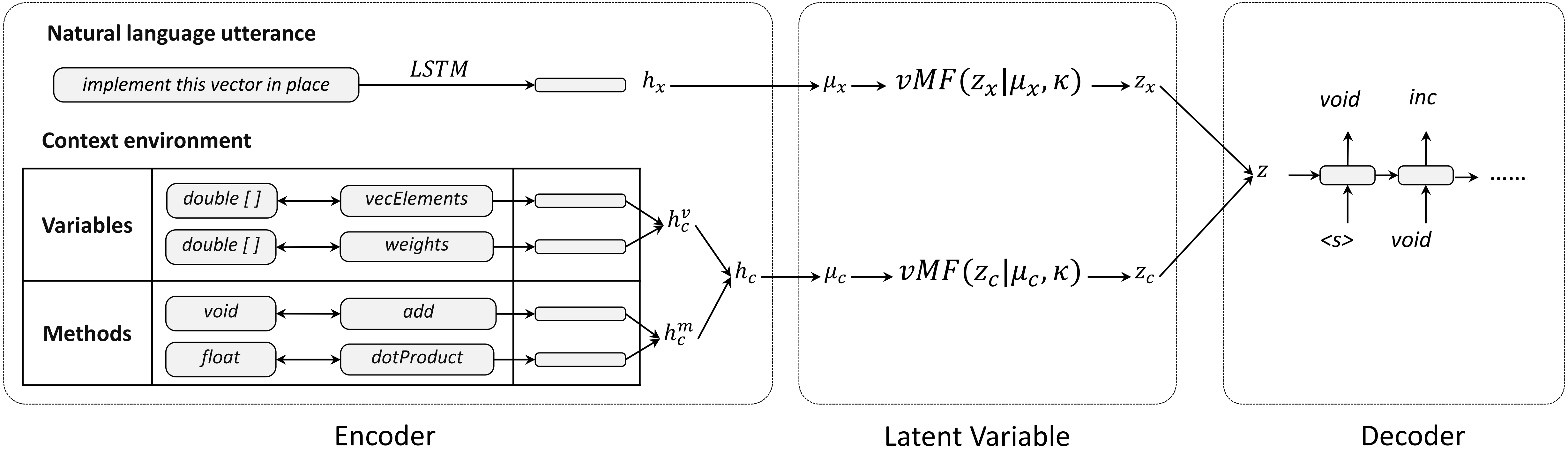
- The decoder predicts logical form from the latent variable z

Context-Aware Retrieval Model



How to retrieve similar examples?

Context-Aware Retrieval Model



- KL-divergence between latent variables z as distance

$$distance = KL(p(z|x,c)||p(z|x',c')) = C_k(\|\mu_x - \mu_{x'}\|_2^2 + \|\mu_c - \mu_{c'}\|_2^2)$$

Sequence-to-Action Model

- Definition of grammar
  - A set of actions that can construct logical forms

Sequence-to-Action Model

- Definition of grammar
  - A set of actions that can construct logical forms

Grammar I: Conversational Question Answering over KB

Action	Operation	Description
A1-A3	S—>Set | Num | Bool	S is start symbol
A4	Set—>Find(R, E)	Set of entities with a r edge to e
A5	Num—>Count(Set)	Total number of set
A6	Bool—>(∈, E, Set)	Whether E∈Set
A7	Set—>Set ∪ Set	Union of Sets
A8	Set—>Set ∩ Set	Intersection of Sets
A9	Set—>Set —Set	Difference of Sets
A10	Set->larger(set, r, num)	Entity from set linking to more than num entities with relation r
A11	Set->less(set, r, num)	Entity from set linking to less than num entities with relation r
A12	Set->equal(set, r, num)	Entity from set linking to num entities with relation r
A13	Set->arxmax(set, r, num)	Entity from set linking to most entities with relation r
A14	Set->argmin(set, r, num)	Entity from set linking to least entities with relation r
A15	Set->{e}	
A16-A18	e | r | num -> constant	instantiation for entity e, predicate r or number num
A19-A21	Set | Num | Bool ->action(i-1)	Replicate previous operation sequence

Sequence-to-Action Model

- Definition of grammar
  - A set of actions that can construct logical forms

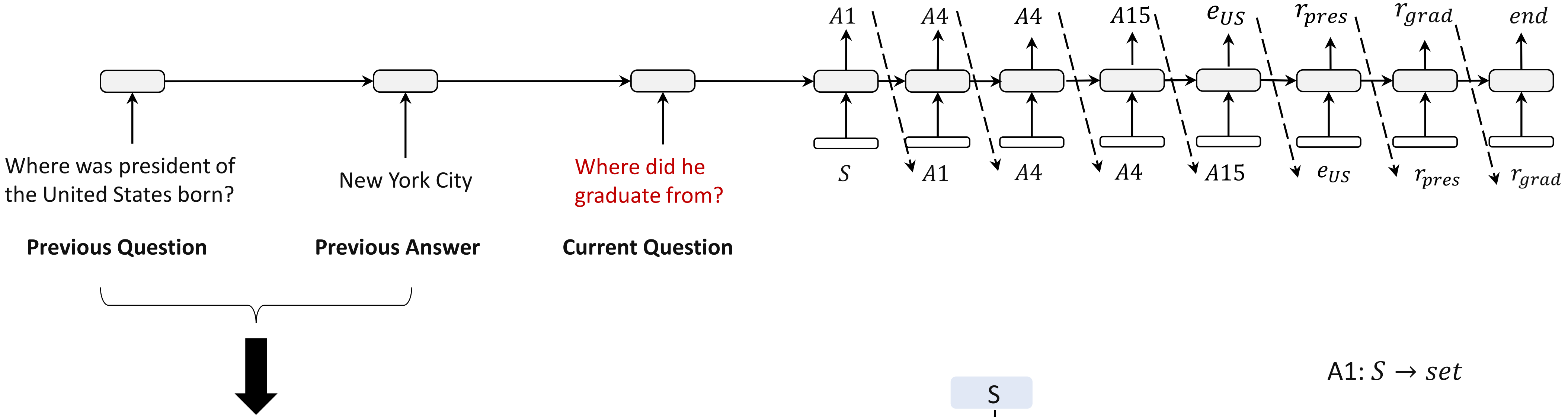
Grammar I: Conversational Question Answering over KB

Action	Operation	Description
A1-A3	S—>Set | Num | Bool	S is start symbol
A4	Set—>Find(R, E)	Set of entities with a r edge to e
A5	Num—>Count(Set)	Total number of set
A6	Bool—>(∈, E, Set)	Whether E∈Set
A7	Set—>Set ∪ Set	Union of Sets
A8	Set—>Set ∩ Set	Intersection of Sets
A9	Set—>Set —Set	Difference of Sets
A10	Set->larger(set, r, num)	Entity from set linking to more than num entities with relation r
A11	Set->less(set, r, num)	Entity from set linking to less than num entities with relation r
A12	Set->equal(set, r, num)	Entity from set linking to num entities with relation r
A13	Set->arxmax(set, r, num)	Entity from set linking to most entities with relation r
A14	Set->argmin(set, r, num)	Entity from set linking to least entities with relation r
A15	Set->{e}	
A16-A18	e | r | num -> constant	instantiation for entity e, predicate r or number num
A19-A21	Set | Num | Bool ->action(i-1)	Replicate previous operation sequence

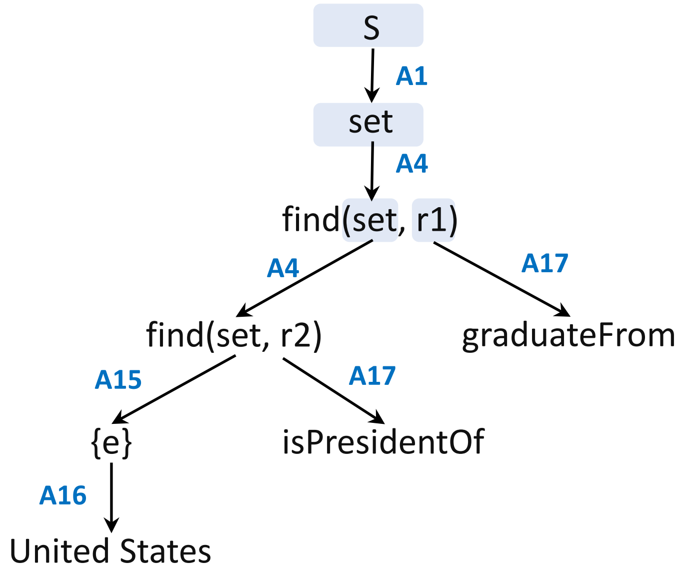
Grammar II: Code Generation

Action	Production rule	Description
A1	MemberDeclaration→MethodDeclaration	Declaration of member
A2	TypeTypeOrVoid→void	Type of function
A3	IdentifierNT→ClassMethod	Identifier comes from member methods
A4	IdentifierNT→ClassVariable	Identifier comes from member variables
A5	ClassMethod→Constant	Instantiation for the member method or variables
A6	ClassVariable→Constant	Instantiation for the member variable
....	.....	.....

Sequence-to-Action Model



Dialog Memory	
Entity	{United States, tag=utterance} {New York City, tag=answer}
Predicate	{isPresidentOf} {placeOfBirth}
Action Subsequence	$set \rightarrow A4\ A15\ e_{US}\ r_{pres}$ $set \rightarrow A4\ A15$ $set \rightarrow A4\ A4\ A15\ e_{US}\ r_{pres}\ r_{bth}$ $set \rightarrow A4\ A4\ A15$



A1: $S \rightarrow set$

A4: $set \rightarrow find(set, r1)$

A4: $set \rightarrow find(set, r2)$

A15: $set \rightarrow \{e\}$

A16: $e \rightarrow United\ States$

A17: $r2 \rightarrow isPresidentOf$

A17: $r1 \rightarrow graduateFrom$

Sequence-to-Action Model

Class Environment:

Variables: [Type, Name]

double[] vecElements

double[] weights

Methods: [Return Type, Name]

void add(double)

NL Query: Increment this vector in this place

Source Code:

public void inc() { this.add(1);}

Action Sequence:

MemberDeclaration→MethodDeclaration

MethodDeclaration→TypeTypeOrVoid IdentifierNT FormalParameters MethodBody

TypeTypeOrVoid→void

....

Primary→IdentifierNT

IdentifierNT→1

Experiment

- Dataset
  - Conversational Question answering
    - CSQA dataset [Saha et al., 2018]
    - Created based on Wikidata with 12.8M entities, including 200k dialogs
    - Contain conversational history and current questions with the answer.
  - Code Generation
    - CONCODE dataset [Iyer et al., 2018]
    - Built from about 33,000 public Java projects on Github
    - Contain natural description and codes together with class environment information

Experiment

Methods	Dev		Test	
	Exact	BLEU	Exact	BLEU
Retrieval ONLY				
TFIDF	1.25	17.78	1.50	19.73
Context-independent Retrieval	0.85	19.63	0.80	21.98
Context-dependent Retrieval	1.30	21.21	1.00	24.94
Parsing-based methods without retrieved examples				
Seq2Seq	2.90	21.00	3.20	23.51
Seq2Prod (Yin and Neubig, 2017)	5.55	21.00	6.65	21.29
Iyer et al. (2018)	7.05	21.28	8.60	22.11
Seq2Action	**7.75**	**22.47**	**9.15**	**23.34**
Parsing-based methods with retrieved examples				
Seq2Action+Edit vector (Context-independent Retrieval)	6.6	21.27	7.90	22.51
Seq2Action+Edit vector (Context-aware Retrieval)	7.75	20.69	9.20	22.68
Seq2Action+Retrieve-and-edit (Context-independent Retrieval)	5.55	21.27	7.05	22.74
Seq2Action+Retrieve-and-edit (Context-aware Retrieval)	7.55	22.20	9.30	23.95
Seq2Action+MAML (Context-independent Retrieval)	9.15	21.48	9.85	23.22
Seq2Action+MAML (Context-aware Retrieval, w/o finetune)	8.30	21.27	10.30	24.12
Seq2Action+MAML (Context-aware Retrieval)	**8.45**	**21.32**	**10.50**	**24.40**

Table 1: Performance of different approaches on the CONCODE dataset.

Experiment

Methods	Dev		Test	
	Exact	BLEU	Exact	BLEU
Retrieval ONLY				
TFIDF	1.25	17.78	1.50	19.73
Context-independent Retrieval	0.85	19.63	0.80	21.98
Context-dependent Retrieval	1.30	21.21	1.00	24.94
Parsing-based methods without retrieved examples				
Seq2Seq	2.90	21.00	3.20	23.51
Seq2Prod (Yin and Neubig, 2017)	5.55	21.00	6.65	21.29
Iyer et al. (2018)	7.05	21.28	8.60	22.11
Seq2Action	**7.75**	**22.47**	**9.15**	**23.34**
Parsing-based methods with retrieved examples				
Seq2Action+Edit vector (Context-independent Retrieval)	6.6	21.27	7.90	22.51
Seq2Action+Edit vector (Context-aware Retrieval)	7.75	20.69	9.20	22.68
Seq2Action+Retrieve-and-edit (Context-independent Retrieval)	5.55	21.27	7.05	22.74
Seq2Action+Retrieve-and-edit (Context-aware Retrieval)	7.55	22.20	9.30	23.95
Seq2Action+MAML (Context-independent Retrieval)	9.15	21.48	9.85	23.22
Seq2Action+MAML (Context-aware Retrieval, w/o finetune)	8.30	21.27	10.30	24.12
Seq2Action+MAML (Context-aware Retrieval)	**8.45**	**21.32**	**10.50**	**24.40**

Table 1: Performance of different approaches on the CONCODE dataset.

Experiment

Methods	Dev		Test	
	Exact	BLEU	Exact	BLEU
Retrieval ONLY				
TFIDF	1.25	17.78	1.50	19.73
Context-independent Retrieval	0.85	19.63	0.80	21.98
Context-dependent Retrieval	1.30	21.21	1.00	24.94
Parsing-based methods without retrieved examples				
Seq2Seq	2.90	21.00	3.20	23.51
Seq2Prod (Yin and Neubig, 2017)	5.55	21.00	6.65	21.29
Iyer et al. (2018)	7.05	21.28	8.60	22.11
Seq2Action	**7.75**	**22.47**	**9.15**	**23.34**
Parsing-based methods with retrieved examples				
Seq2Action+Edit vector (Context-independent Retrieval)	6.6	21.27	7.90	22.51
Seq2Action+Edit vector (Context-aware Retrieval)	7.75	20.69	9.20	22.68
Seq2Action+Retrieve-and-edit (Context-independent Retrieval)	5.55	21.27	7.05	22.74
Seq2Action+Retrieve-and-edit (Context-aware Retrieval)	7.55	22.20	9.30	23.95
Seq2Action+MAML (Context-independent Retrieval)	9.15	21.48	9.85	23.22
Seq2Action+MAML (Context-aware Retrieval, w/o finetune)	8.30	21.27	10.30	24.12
Seq2Action+MAML (Context-aware Retrieval)	**8.45**	**21.32**	**10.50**	**24.40**

Table 1: Performance of different approaches on the CONCODE dataset.

Experiment

Methods	Dev		Test	
	Exact	BLEU	Exact	BLEU
Retrieval ONLY				
TFIDF	1.25	17.78	1.50	19.73
Context-independent Retrieval	0.85	19.63	0.80	21.98
Context-dependent Retrieval	1.30	21.21	1.00	24.94
Parsing-based methods without retrieved examples				
Seq2Seq	2.90	21.00	3.20	23.51
Seq2Prod (Yin and Neubig, 2017)	5.55	21.00	6.65	21.29
Iyer et al. (2018)	7.05	21.28	8.60	22.11
Seq2Action	**7.75**	**22.47**	**9.15**	**23.34**
Parsing-based methods with retrieved examples				
Seq2Action+Edit vector (Context-independent Retrieval)	6.6	21.27	7.90	22.51
Seq2Action+Edit vector (Context-aware Retrieval)	7.75	20.69	9.20	22.68
Seq2Action+Retrieve-and-edit (Context-independent Retrieval)	5.55	21.27	7.05	22.74
Seq2Action+Retrieve-and-edit (Context-aware Retrieval)	7.55	22.20	9.30	23.95
Seq2Action+MAML (Context-independent Retrieval)	9.15	21.48	9.85	23.22
Seq2Action+MAML (Context-aware Retrieval, w/o finetune)	8.30	21.27	10.30	24.12
Seq2Action+MAML (Context-aware Retrieval)	**8.45**	**21.32**	**10.50**	**24.40**

Retrieve-and-Edit

Retrieve-and-MAML

Table 1: Performance of different approaches on the CONCODE dataset.

Experiment

Methods	HRED +KVmem	D2A	S2A	S2A +EditVec	S2A +RAndE	S2A +MAML
Question Type	F1					
Simple Question (Direct)	13.64%	91.41%	92.01%	91.95%	92.08%	**92.66%**
Simple Question (Co-referenced)	7.26%	69.83%	71.40%	72.94%	**73.19%**	71.18%
Simple Question (Ellipsis)	9.95%	81.98%	81.75%	83.31%	**84.61%**	82.21%
Logical Reasoning (All)	8.33%	43.62%	42.00%	43.85%	41.83%	**44.34%**
Quantitative Reasoning (All)	0.96%	50.25%	45.37%	46.93%	42.64%	**50.30%**
Comparative Reasoning (All)	2.96%	44.20%	41.51%	43.96%	44.46%	**48.13%**
Clarification	16.35%	18.31%	18.9%	18.42%	18.70%	**19.12%**
Question Type	Accuracy					
Verification (Boolean)	21.04%	45.05%	51.17%	47.81%	**55.00%**	50.16%
Quantitative Reasoning (Count)	12.13%	40.94%	46.01%	44.67%	43.07%	**46.43%**
Comparative Reasoning (Count)	5.67%	17.78%	16.52%	17.52%	16.43%	**18.91%**

Table 2: Performance of different approaches on the CSQA dataset.

Analysis

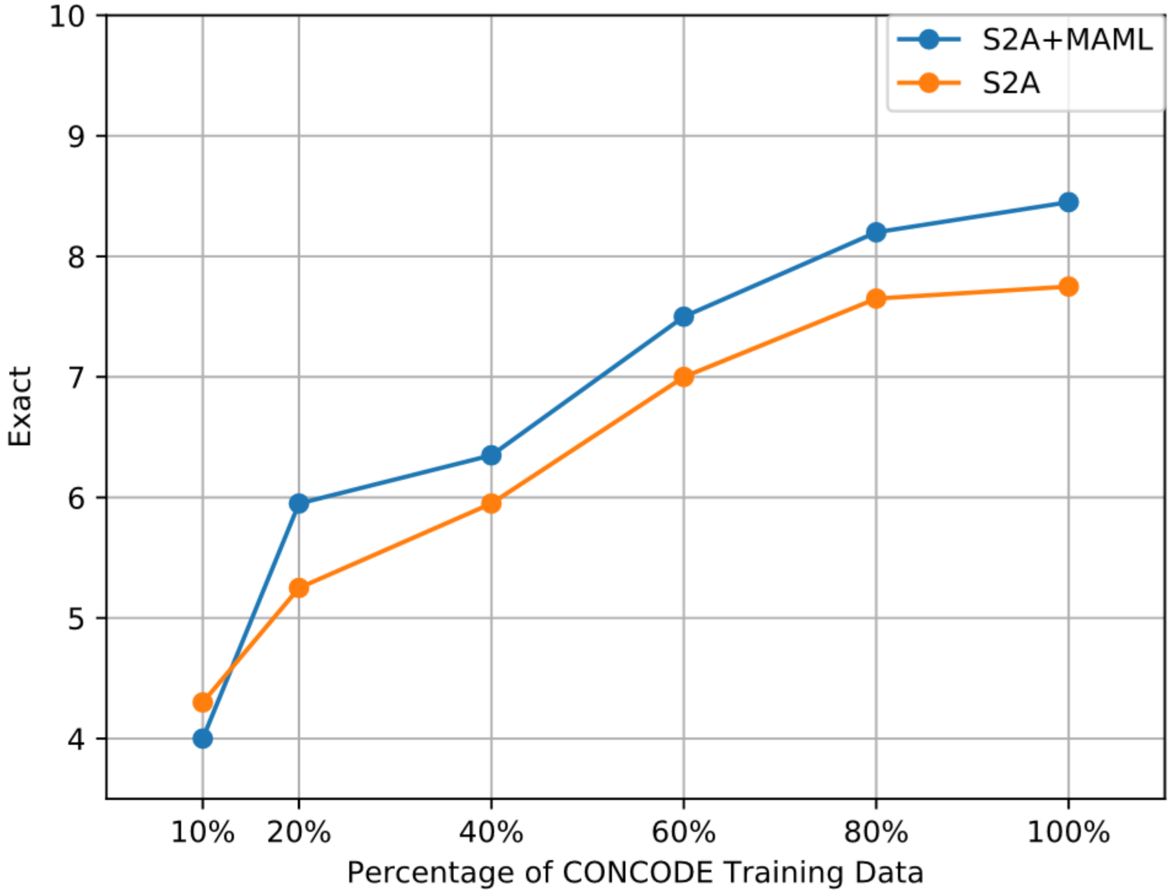


Figure 4: Comparison between S2A and S2A+MAML with different portions of supervised data.

Analysis

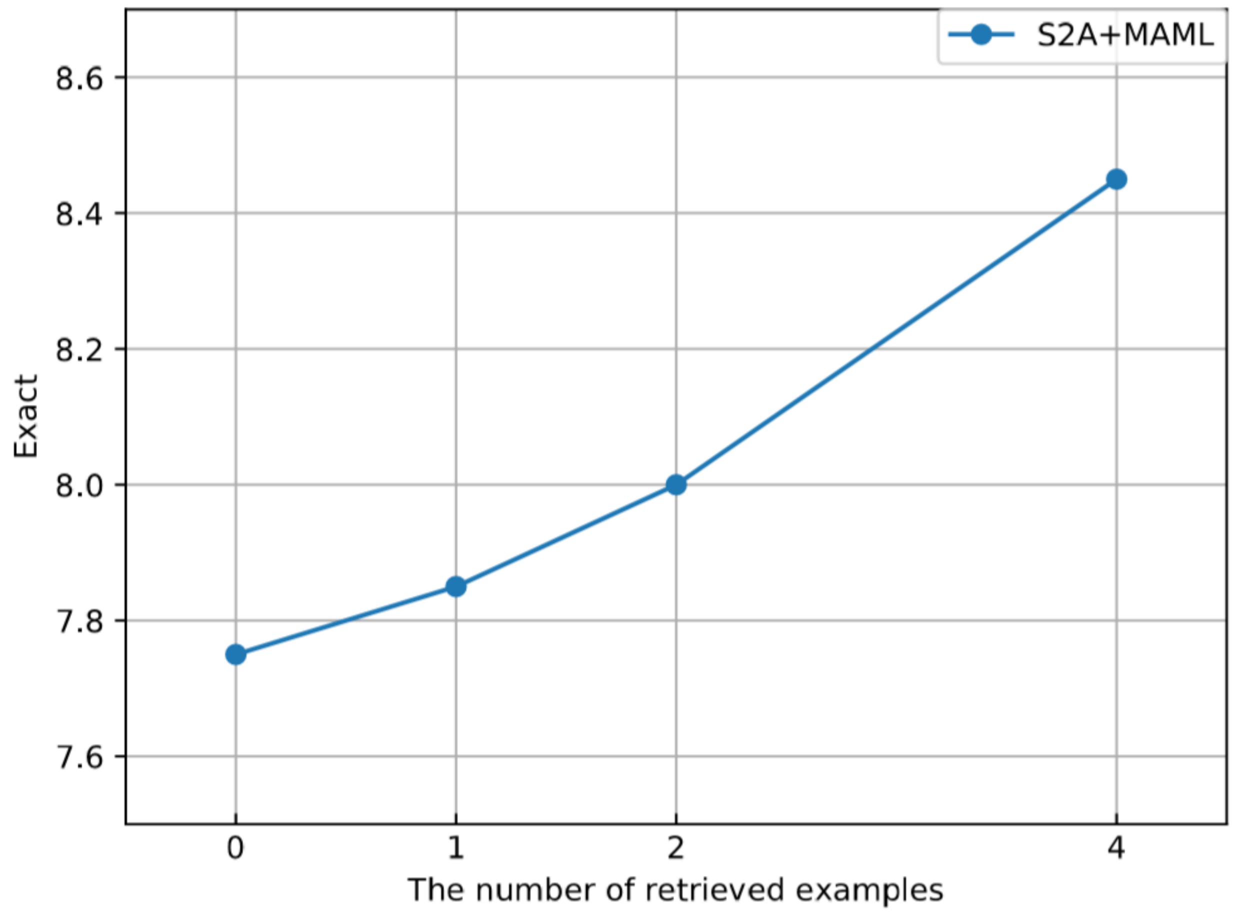


Figure 4: S2A+MAML with different number of retrieved examples on the CONCODE devset.

Analysis

Input	Context-Aware Retriever	Context-Independent Retriever

Class environment:
HashMap<lalr_item, lalr_item> _all;
NL:
Does the set contain a particular item
Code:
boolean function(lalr_item arg0){
return _all.constainsKey(arg0); }
	Class environment:
	Map<Point, RailwayNode> _nodeMap;
	NL:
	Check if a node at a specific position exits.
	Code:
	boolean function(Point arg0){
	return _nodeMap.constainsKey(arg0);}
		Class environment:
		Node root;
		Node get(Node x, String key, int d);
		NL: Does the set contain the given key
		Code:
		boolean function(String arg0){
		Node loc0==get(root,arg0,0);
		if (loc0==null) return false;
		return loc0.isString; }

Q1: who is the dad of jorgen ottesen brahe? **A1**: otte brahe **Q2**: who is the spouse of that one?	**Q1**: whose child are gio batta bellotti? **A1**: matteo bellotti, paola cresipi guzzo **Q2**: which person is married to that one?	**Q1**: which abstract beings have marge simpson as an offspring? **A1**: clancy bouvier, jacqueline bouvier **Q2**: who is the spouse of that one?

Analysis

Input	Context-Aware Retriever	Context-Independent Retriever

Class environment:
HashMap<lalr_item, lalr_item> _all;
NL:
Does the set contain a particular item
Code:
boolean function(lalr_item arg0){
return _all.constainsKey(arg0); }
	Class environment:
	Map<Point, RailwayNode> _nodeMap;
	NL:
	Check if a node at a specific position exits.
	Code:
	boolean function(Point arg0){
	return _nodeMap.constainsKey(arg0);}
		Class environment:
		Node root;
		Node get(Node x, String key, int d);
		NL: Does the set contain the given key
		Code:
		boolean function(String arg0){
		Node loc0==get(root,arg0,0);
		if (loc0==null) return false;
		return loc0.isString; }

Q1: who is the dad of jorgen ottesen brahe? **A1**: otte brahe **Q2**: who is the spouse of that one?	**Q1**: whose child are gio batta bellotti? **A1**: matteo bellotti, paola cresipi guzzo **Q2**: which person is married to that one?	**Q1**: which abstract beings have marge simpson as an offspring? **A1**: clancy bouvier, jacqueline bouvier **Q2**: who is the spouse of that one?

Analysis

Input	Context-Aware Retriever	Context-Independent Retriever

Class environment:
HashMap<lalr_item, lalr_item> _all;
NL:
Does the set contain a particular item
Code:
boolean function(lalr_item arg0){
return _all.constainsKey(arg0); }
	Class environment:
	Map<Point, RailwayNode> _nodeMap;
	NL:
	Check if a node at a specific position exits.
	Code:
	boolean function(Point arg0){
	return _nodeMap.constainsKey(arg0);}
		Class environment:
		Node root;
		Node get(Node x, String key, int d);
		NL: Does the set contain the given key
		Code:
		boolean function(String arg0){
		Node loc0==get(root,arg0,0);
		if (loc0==null) return false;
		return loc0.isString; }

Q1: who is the dad of jorgen ottesen brahe? **A1**: otte brahe **Q2**: who is the spouse of that one?	**Q1**: whose child are gio batta bellotti? **A1**: matteo bellotti, paola cresipi guzzo **Q2**: which person is married to that one?	**Q1**: which abstract beings have marge simpson as an offspring? **A1**: clancy bouvier, jacqueline bouvier **Q2**: who is the spouse of that one?

Analysis

Input	Context-Aware Retriever	Context-Independent Retriever

Class environment:
HashMap<lalr_item, lalr_item> _all;
NL:
Does the set contain a particular item
Code:
boolean function(lalr_item arg0){
return _all.constainsKey(arg0); }
	Class environment:
	Map<Point, RailwayNode> _nodeMap;
	NL:
	Check if a node at a specific position exits.
	Code:
	boolean function(Point arg0){
	return _nodeMap.constainsKey(arg0);}
		Class environment:
		Node root;
		Node get(Node x, String key, int d);
		NL: Does the set contain the given key
		Code:
		boolean function(String arg0){
		Node loc0==get(root,arg0,0);
		if (loc0==null) return false;
		return loc0.isString; }

Q1: who is the dad of jorgen ottesen brahe? **A1**: otte brahe **Q2**: who is the spouse of that one?	**Q1**: whose child are gio batta bellotti? **A1**: matteo bellotti, paola cresipi guzzo **Q2**: which person is married to that one?	**Q1**: which abstract beings have marge simpson as an offspring? **A1**: clancy bouvier, jacqueline bouvier **Q2**: who is the spouse of that one?

Conclusion

- We present an approach to couple retrieval and meta-learning to improve context-dependent semantic parsing
- In conversational question answering and code generation tasks, our approach achieves state-of-the-art performances on CSQA and CONCODE datasets.

Thanks

Learning Method

- Basic idea to learn models by jointly learning

$$p(y|x,c) = \sum_{S} p_{edit}(y|x,c,S)p_{ret}(S|x,c)$$

- Assuming that an oracle semantic parser p^*_{edit} provides the true conditional distribution

$$\log p(y|x,c) \geq \underbrace{E_{z\sim p_{encoder}(z|x,\,c)} log p_{decoder}(y|z)}_{\text{Reconstruction term}} - \underbrace{E_{\{(x',c')\}\sim p_{ret}} KL(p_{encoder}(z|x,c)||p_{encoder}(z|x',c'))}_{\text{KL-divergence}}$$

- We can maximize the worst-case lower bound

$$distance = KL(p_{encoder}(z|x,c)\big|\big|p_{encoder}(z|x',c')\big) = C_k(\|\mu_x - \mu_{x'}\|_2^2 + \|\mu_c - \mu_{c'}\|_2^2) \leq 8C_k$$

$$\downarrow$$

$$\log p(y|x,c) \geq E_{z\sim p_{encoder}(z|x,\,c)} log p_{decoder}(y|z) - distance \geq \underbrace{E_{z\sim p_{encoder}(z|x,\,c)} log p_{decoder}(y|z)}_{\text{Reconstruction term}} - 8C_k$$