RetroNLU: Retrieval Augmented Task Oriented Semantic Parsing

https://retronlu.github.io/

Vivek Gupta[1], Akshat Shrivastava[2], Adithya Sagar[2],
Armen Aghajanyan[2], Denis Savenkov[3]

[1]University of Utah; [2]Facebook (Meta) Conversational AI[2]

[1]on academic job market
[1]Bloomberg Ph.D. Fellow
[1]work done as an intern

Takeaway

1. In this work, we explore RetroNLU: retrieval based modeling approach for task-oriented semantic parsing problem.
2. RetroNLU makes explicit use of memory of retrieve examples of semantic parses that the model learn to adapt for other similar input utterance.
3. We analyse the robustness and sensitivity of RetroNLU in several dimensions as follows:
   a. Data Efficiency
   b. Limited Supervision
   c. Noise Robustness
   d. Utterance Complexity
   e. Knowledge Efficiency

Task Oriented Semantic Parsing

utterance : “please add 20 minutes on the lasagna timer”

↓

semparse (coupled) : [in:add_time_timer please add [sl:date_time 20 minutes] on the [sl:timer_name lasagna] [sl:method_timer timer]]

Task Oriented Semantic Parsing

Utterance : “please add 20 minutes on the lasagna timer”

semparse (coupled) : [in:add_time_timer please add [sl:date_time 20 minutes] on the [sl:timer_name lasagna] [sl:method_timer timer]]

semparse (decoupled) : [in:add_time_timer ~~please add~~ [sl:date_time 20 minutes] ~~on the~~ [sl:timer_name lasagna] [sl:method_timer timer]]

Task Oriented Semantic Parsing

Utterance : "please add 20 minutes on the lasagna timer"

↓

semparse (coupled) : [in:add_time_timer please add [sl:date_time 20 minutes] on the [sl:timer_name lasagna] [sl:method_timer timer]]

↓

semparse (decoupled) : [in:add_time_timer [sl:date_time 20 minutes] [sl:timer_name lasagna] [sl:method_timer timer]]

Structured Prediction

Utterance: Driving directions to the Eagles game

Semantic Parse: [IN:GET_DIRECTIONS Driving directions to [SL:DESTINATION [IN:GET_EVENT the [SL:NAME_EVENT Eagles] [SL:CAT_EVENT game]]]]

Tree Representation:

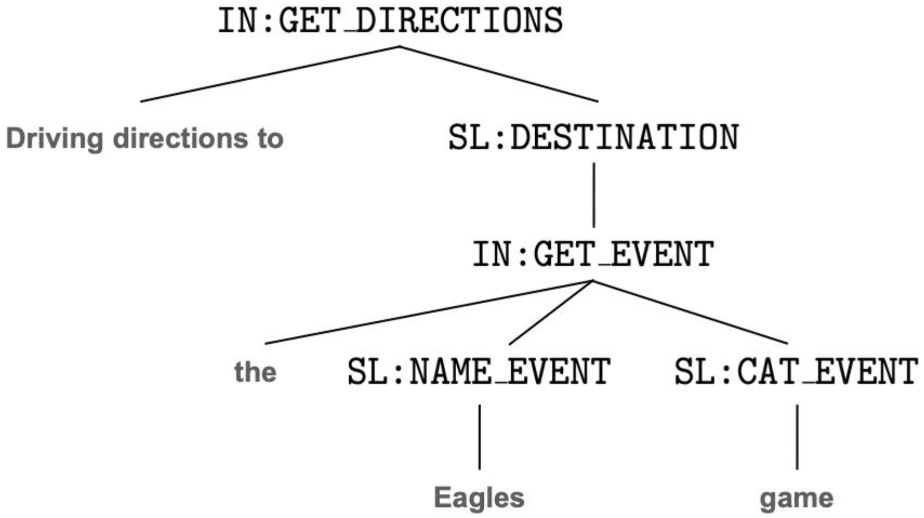


Figure 1: An compositional query from TOP dataset.

RETRIEVAL AUGMENTATION

NN Index	
$utterance_1$	$semprase_1$
$utterance_2$	$semparse_2$
….	….
….	….
$utterance_n$	$semparse_n$

Initial Problem

→ $utterance_p$ → $semparse_p$

NN index is build using pre-train BART Model

Retrieval Augmentation

initial: utterance$_p$ → **nn-context**: semparse$_2$

NN Index	
utterance$_1$	semprase$_1$
utterance$_2$	semparse$_2$
….	….
….	….
utterance$_n$	semparse$_n$

Initial Problem

→ utterance$_p$ → semparse$_p$

Nearest Neighbour

NN index is build using pre-train BART Model

RETRIEVAL AUGMENTATION

initial: utterance_p → **nn-context**: semparse_2 → **augment**: semparse_2 | utterance_p

NN Index	
utterance_1	semprase_1
utterance_2	semparse_2
….	….
….	….
utterance_n	semparse_n

Initial Problem

→ utterance_p → semparse_p

After Retrieval Augmentation

→ semparse_2 | utterance_p → semparse_p

NN index is build using pre-train BART Model

EXAMPLE : RETRIEVAL AUGMENTATION

initial utterance : “please add 20 minutes on the lasagna timer”

expected semparse(decoupled): [in:add_time_timer [sl:date_time 20 minutes] [sl:timer_name lasagna] [sl:method_timer timer]]

EXAMPLE : RETRIEVAL AUGMENTATION

initial utterance : “please add 20 minutes on the lasagna timer”
expected semparse(decoupled): [in:add_time_timer [sl:date_time 20 minutes] [sl:timer_name lasagna] [sl:method_timer timer]]

nn utterance : “add ten minutes to the oven timer”
nn semparse (coupled): [in:add_time_timer add [sl:date_time ten minutes] to the [sl:timer_name oven] [sl:method_timer timer]]

EXAMPLE : RETRIEVAL AUGMENTATION

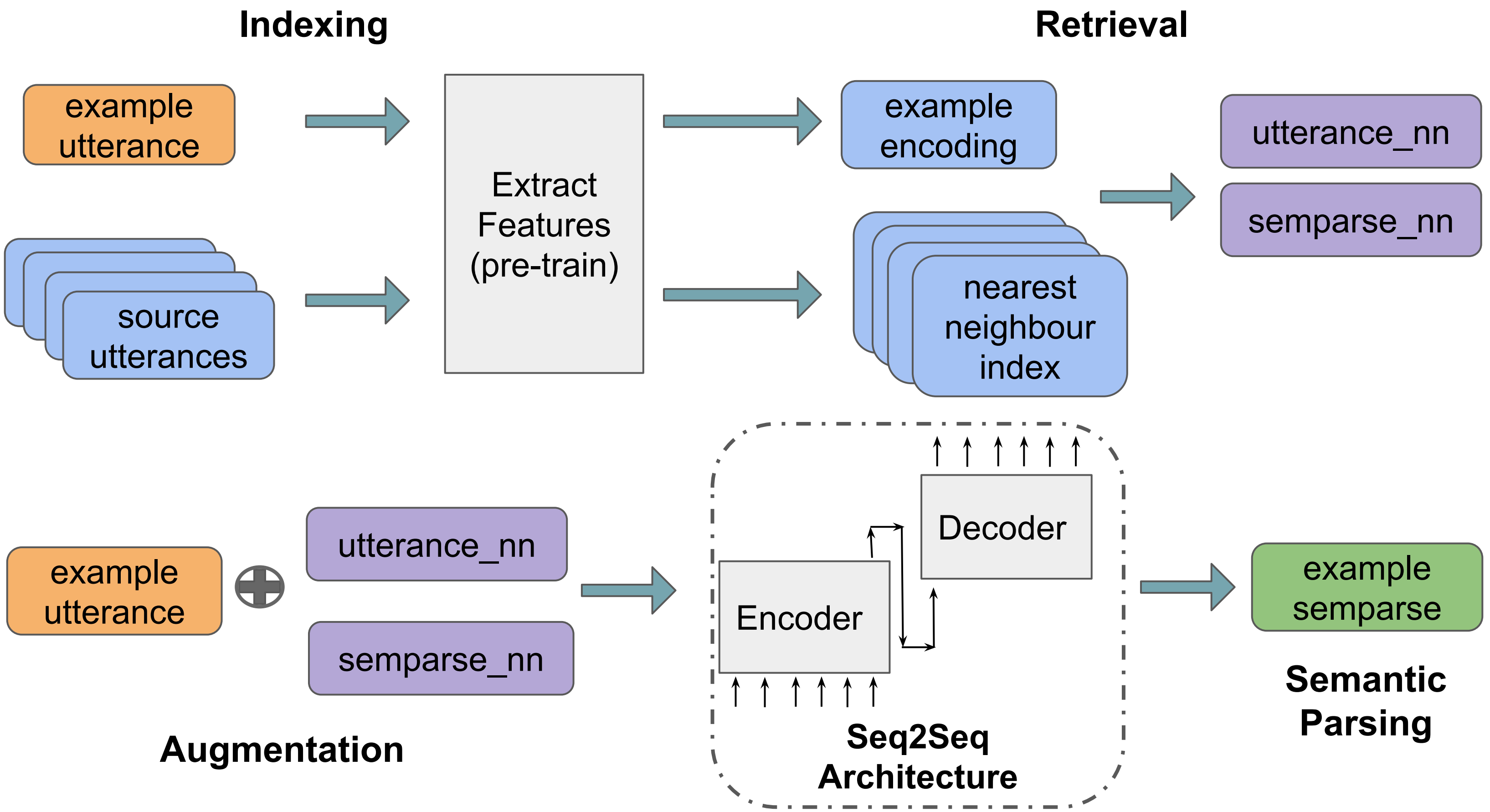
initial utterance : "please add 20 minutes on the lasagna timer"
expected semparse(decoupled): [in:add_time_timer [sl:date_time 20 minutes] [sl:timer_name lasagna] [sl:method_timer timer]]

nn utterance : "add ten minutes to the oven timer"
nn semparse (coupled): [in:add_time_timer add [sl:date_time ten minutes] to the [sl:timer_name oven] [sl:method_timer timer]]

final utterance :[in:add_time_timer add [sl:date_time ten minutes] to the [sl:timer_name oven] [sl:method_timer timer]] | please add 20 minutes on the lasagna timer

expected semparse (decoupled): [in:add_time_timer [sl:date_time 20 minutes] [sl:timer_name lasagna] [sl:method_timer timer]]

RetroNlu



Indexing
Retrieval
example utterance
source utterances
Extract Features (pre-train)
example encoding
nearest neighbour index
utterance_nn
semparse_nn
example utterance
utterance_nn
semparse_nn
Augmentation
Encoder
Decoder
Seq2Seq Architecture
example semparse
Semantic Parsing

TOP-v2 DATASET

Source Domains	
Alarm	20,431
Event	9,171
Music	10,019
Navigation	11,564
Timer	23,055

High Resource Setting

Train 70%, Validation 10%, Test 20%

Target Domains	
weather	23055
reminder	17841

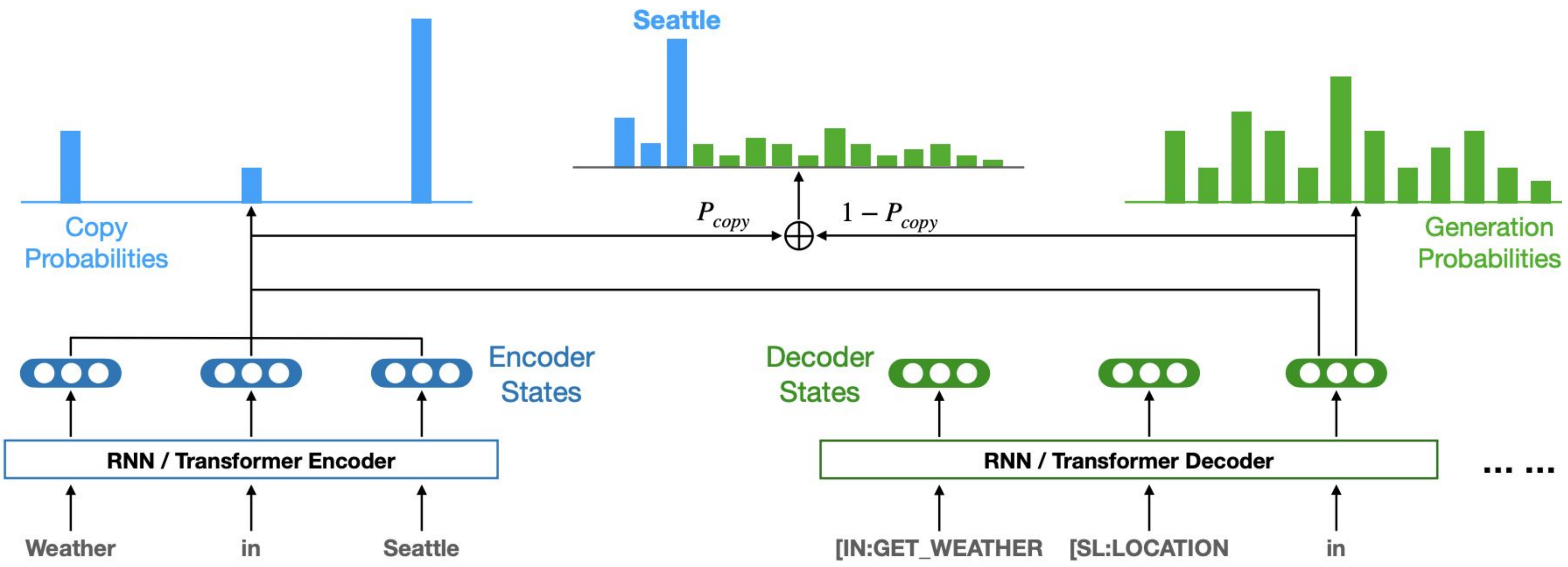
Low Resource Domain Adaptation

Train 20 ex/per intent/slot

Validation 10%, Test 20%

[1] Chen, X., Ghoshal, A., Mehdad, Y., Zettlemoyer, L., & Gupta, S. (2020). Low-Resource Domain Adaptation for Compositional Task-Oriented Semantic Parsing. *arXiv preprint arXiv:2010.03546*. EMNLP 2020

COPY TRANSFORMER MODEL



[1] Chen, X., Ghoshal, A., Mehdad, Y., Zettlemoyer, L., & Gupta, S. (2020). Low-Resource Domain Adaptation for Compositional Task-Oriented Semantic Parsing. *arXiv preprint arXiv:2010.03546*. EMNLP 2020

QUESTIONS

In this work we are focusing on the following questions:

(a) ***Data Efficiency:*** Can retrieval based on non-parametric external knowledge alleviate reliance on parametric knowledge typically acquired via supervised training on large labeled datasets?

RESULT AND ANALYSIS

1. Performance Analysis
   a. Supervised Setting (lot's labeled data)
   b. Unsupervised Setting (limited label data)

SUPERVISED SETTING

1-NN FA vs without NN FA:

Micro Average = 85.74 vs 84.43 (Δ 1.31)

Macro Average = 85.82 vs 84.66 (Δ 1.16)

Per-Domain FA:

alarm = 88.57 vs 86.67 (Δ 1.90)

event = 84.77 vs 83.83 (Δ 0.94)

messaging = 94.65 vs 93.50 (Δ 1.15)

music = 80.71 vs 79.80 (Δ 0.91)

navigation = 85.20 vs 82.96 (Δ 2.24)

timer = 81.00 vs 81.21 (▽0.21)

No duplicate in NN (No-exact match)

increasing # of nn help (marginal **Δ**)

#neighbors	k = 1	k = 2	k = 3
without-nn	84.43	84.43	84.43
utterance-nn	85.28	85.35	85.40
semparse-nn	**85.74**	**85.81**	**85.80**

possible issues

- many similar nn (no diversity)

UNSUPERVISED SETTING

initial: utterance$_p$ → $\boxed{\textbf{nn-context: } \text{utterance}_2}$ → **final** : utterance$_2$ | utterance$_p$

NN Index	
utterance$_1$	~~semprase$_1$~~
utterance$_2$	~~semparse$_2$~~
….	….
….	….
utterance$_n$	~~semparse$_n$~~

Initial Problem

→ utterance$_p$ → semparse$_p$

After Retrieval Augmentation

→ utterance$_2$ | utterance$_p$ → semparse$_p$

NN index → Using pre-train BART Model

Why Unsupervised Setting Work

Quasi Symmetric Property of NN (training)

utterance$_1$ <neighbour> utterance$_2$
utterance$_2$ <neighbour> utterance$_1$

utterance$_2$ | utterance$_1$ → semparse$_1$

utterance$_1$ | utterance$_2$ → semparse$_2$

input 1 & input 2 only position changed

Contrastive Learning (Similar Examples)

please add 20 minutes on the lasagna timer | add ten minutes to the oven timer

→ [in:add_time_timer [sl:date_time 20 minutes] [sl:timer_name lasagna] [sl:method_timer timer]]

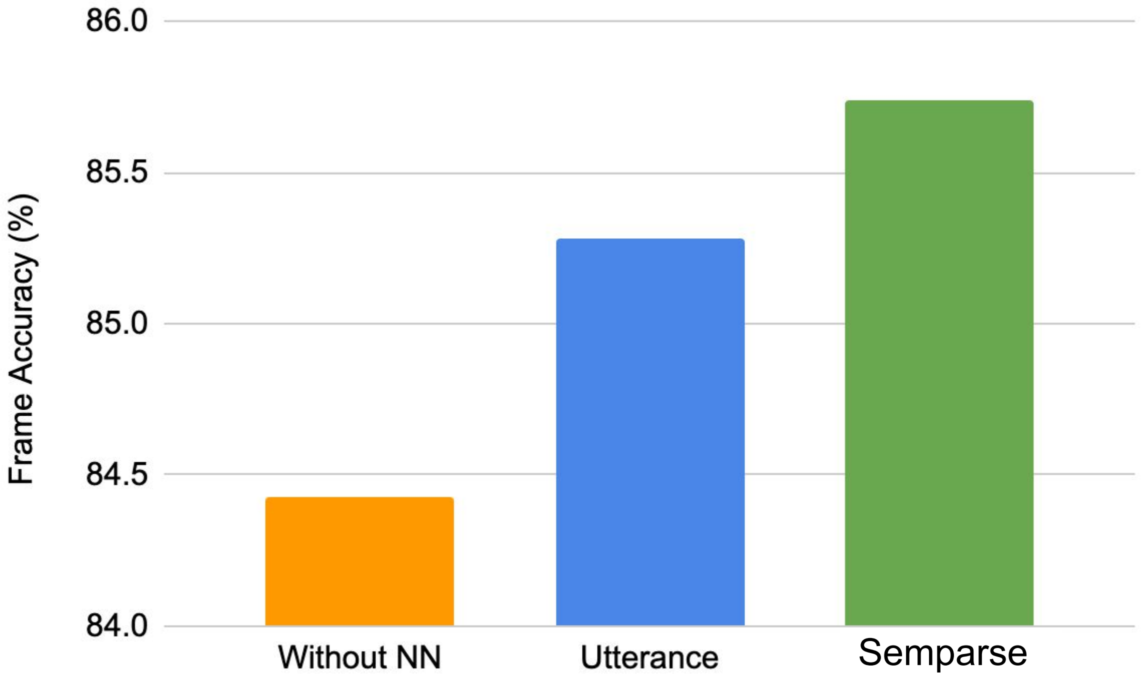
add ten minutes to the oven timer | please add 20 minutes on the lasagna timer

→ [in:add_time_timer [sl:date_time ten minutes] [sl:timer_name oven] [sl:method_timer timer]]

Unsupervised Setting

Unsupervised		Supervised	
1-NN FA vs without NN FA:		**1-NN FA vs without NN FA:**	
Micro Average = 85.28 vs 84.43	**(Δ 0.8)**	Micro Average = 85.74 vs 84.43	**(Δ 1.31)**
Macro Average = 85.56 vs 84.66	**(Δ 0.9)**	Macro Average = 85.82 vs 84.66	**(Δ 1.16)**
Per-Domain FA:		**Per-Domain FA:**	
alarm = 87.17 vs 86.67	**(Δ 0.50)**	alarm = 88.57 vs 86.67	**(Δ 1.90)**
event = 85.03 vs 83.83	**(Δ 1.20)**	event = 84.77 vs 83.83	**(Δ 0.94)**
messaging = 94.52 vs 93.50	**(Δ 1.02)**	messaging = 94.65 vs 93.50	**(Δ 1.15)**
music = 80.73 vs 79.80	**(Δ 0.93)**	music = 80.71 vs 79.80	**(Δ 0.91)**
navigation = 84.16 vs 82.96	**(Δ 1.20)**	navigation = 85.20 vs 82.96	**(Δ 2.24)**
timer = 81.75 vs 81.21	**(Δ 0.54)**	timer = 81.00 vs 81.21	**(∇0.21)**

UNSUPERVISED SETTING



lesser gains than supervised

similar trend with 2-NN & 3-NN

gap decrease marginally with more NN

(sup vs unsup)

Questions

In this work we are focusing on the following questions:

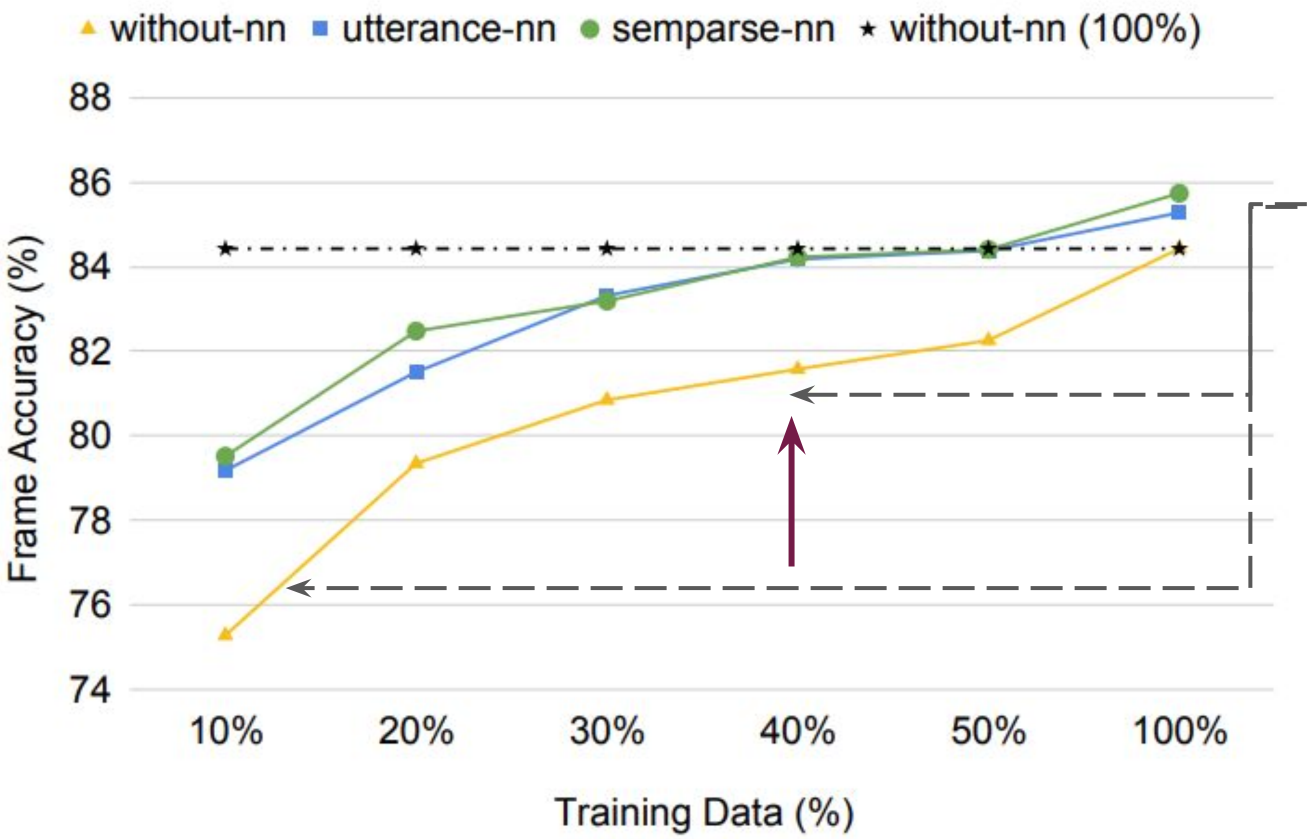
(a) ***Data Efficiency:*** Can retrieval based on non-parametric external knowledge alleviate reliance on parametric knowledge typically acquired via supervised training on large labeled datasets?

(b) ***Limited Supervision:*** Can we enhance models by using abundant and inexpensive unlabeled external non-parametric knowledge rather than structurally labeled knowledge?

RESULT AND ANALYSIS

1. Performance Analysis
    a. Supervised Setting (lot’s labeled data)
    b. Unsupervised Setting (limited label data)
    c. Semi-Supervised
        i. incremental update (limited training)
        ii. unlabeled data (limited label data)

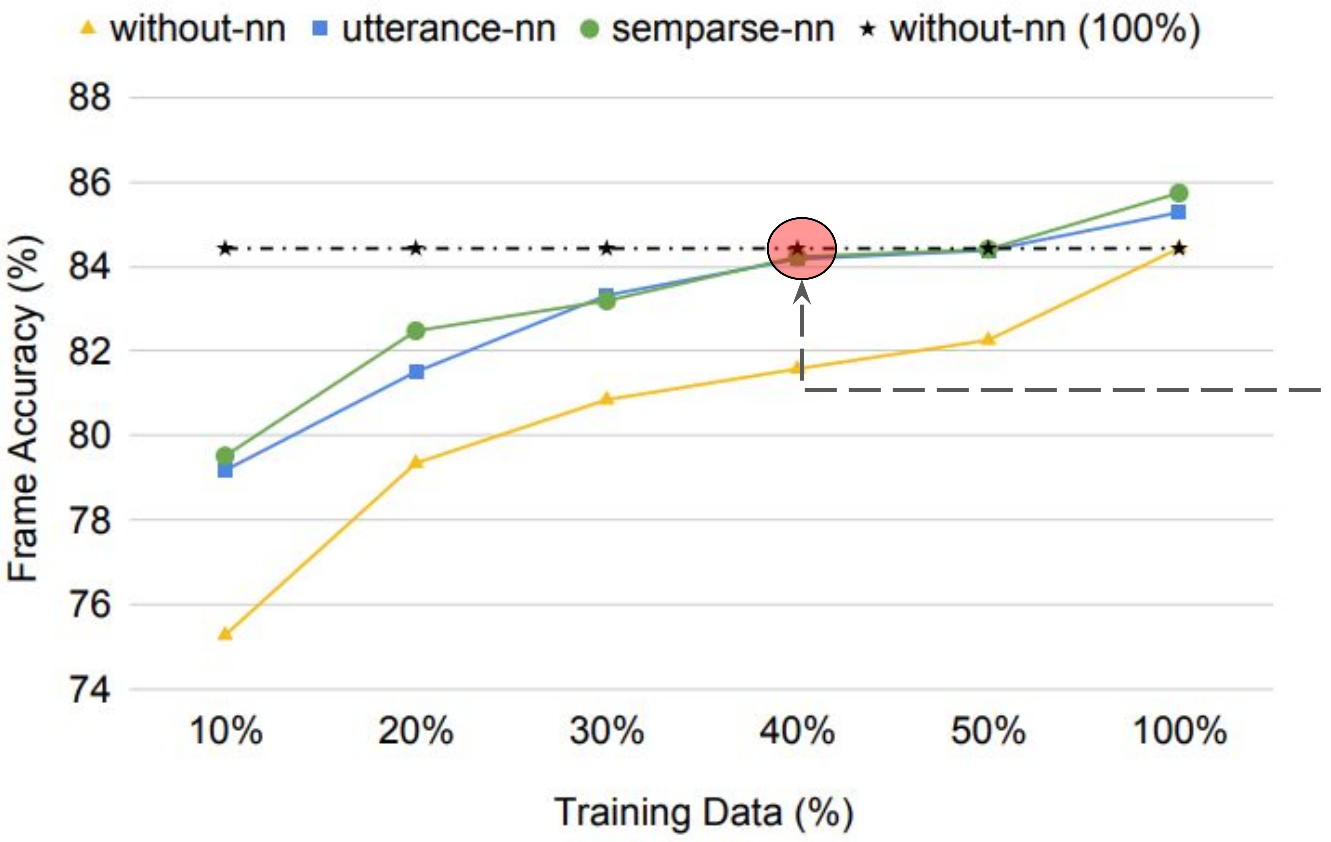
SEMI SUPERVISED SETTING



NN is index of full data

Less data → Less accuracy (75 vs 84)

SEMI SUPERVISED SETTING

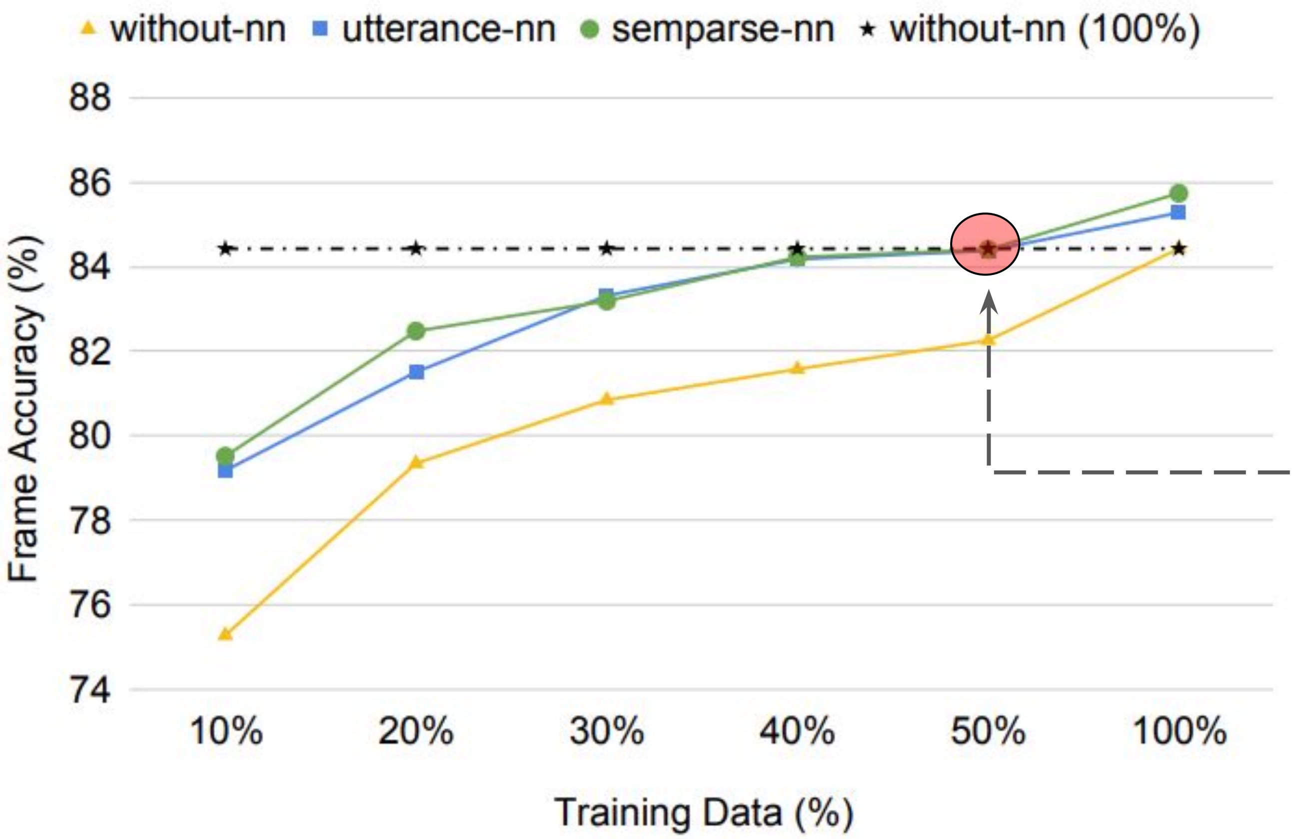


NN is index of full data

Less data → Less accuracy (75 vs 84)

Original performance at 60% less data

SEMI SUPERVISED SETTING



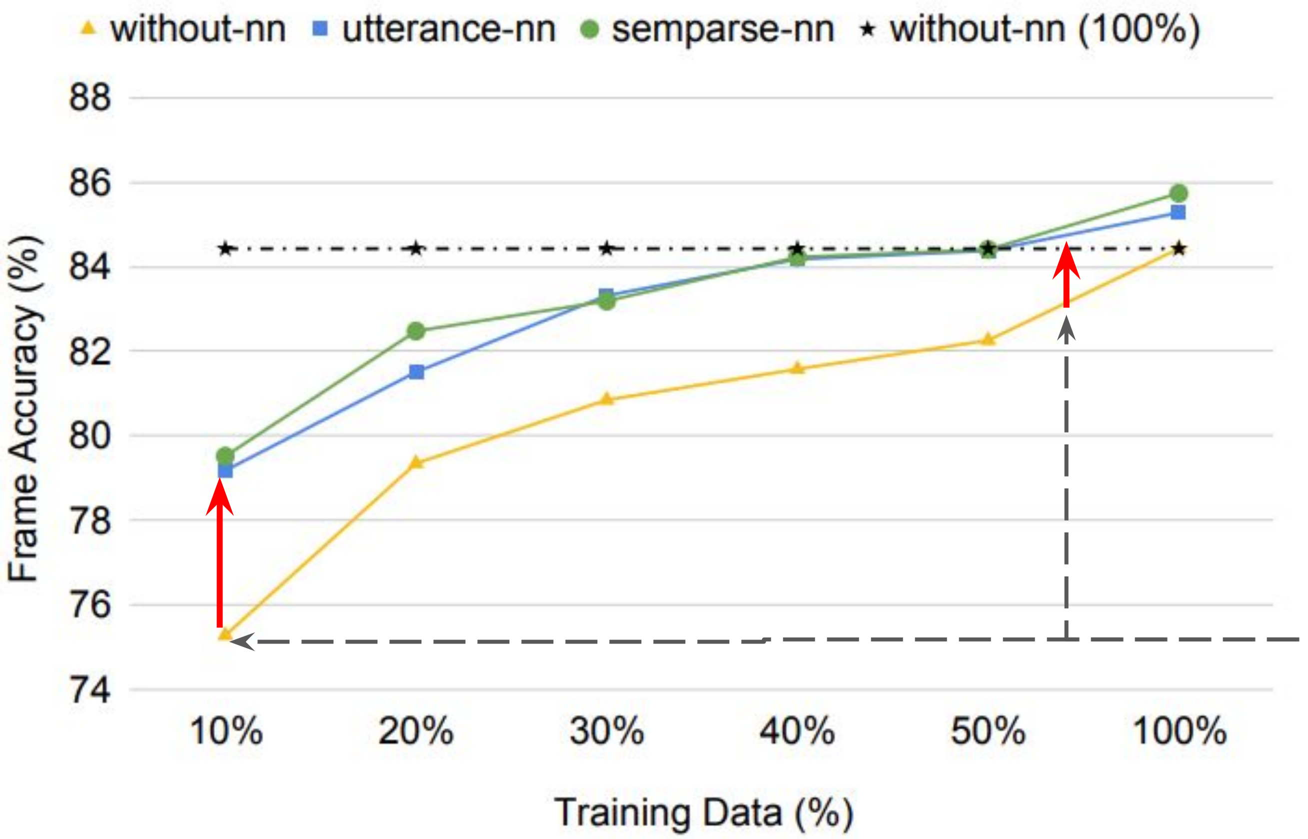
NN is index of full data

Less data → Less accuracy (75 vs 84)

Original performance at 60% less data

Utterance only perform as good as semparse

SEMI SUPERVISED SETTING



NN is index of full data

Less data → Less accuracy (75 vs 84)

Original performance at 60% less data

Utterance only perform as good as semparse

Relative gain decrease with more data(4.2 vs 1.3)

RESULT AND ANALYSIS

1. Performance Analysis
    a. Supervised Setting (lot’s labeled data)
    b. Unsupervised Setting (limited label data)
    c. Semi-Supervised
        i. incremental update (limited training)
        ii. unlabeled data (limited label data)
2. Retrieval Analysis
    a. Retrieval Quality (nn quality)
    b. Simple vs Complex (query complexity)
    c. Frequent vs Rare (nn frequency)

QUESTIONS

In this work we are focusing on the following questions:

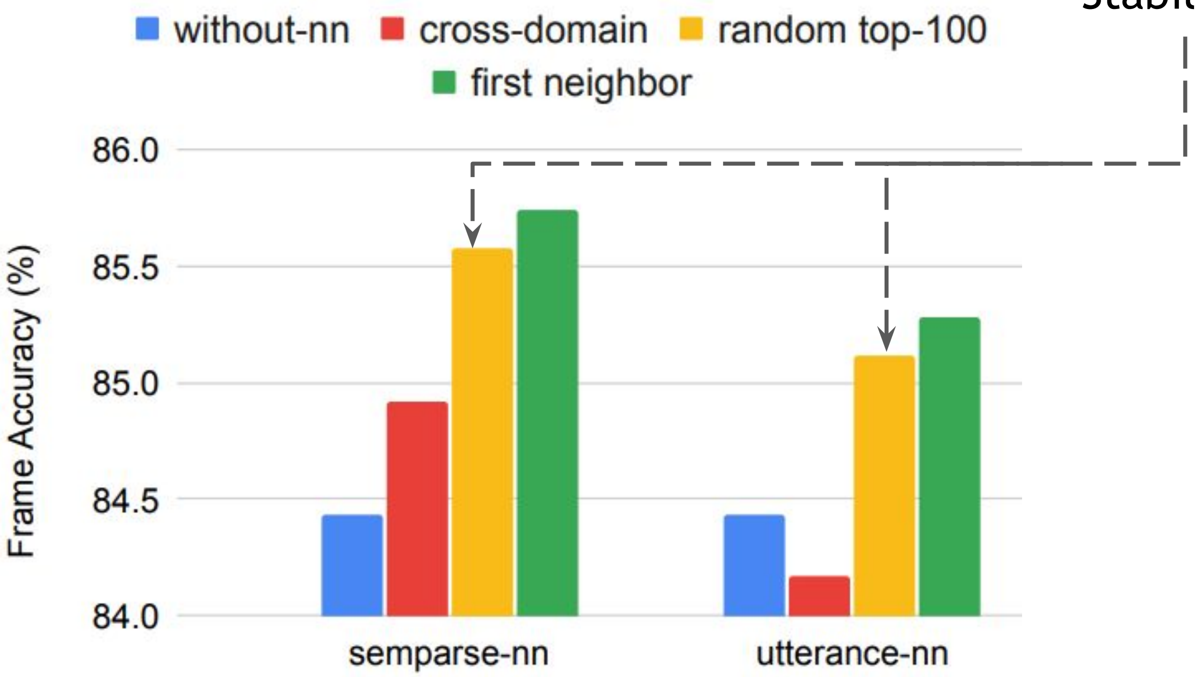
(a) ***Data Efficiency:*** Can retrieval based on non-parametric external knowledge alleviate reliance on parametric knowledge typically acquired via supervised training on large labeled datasets?

(b) ***Limited Supervision:*** Can we enhance models by using abundant and inexpensive unlabeled external non-parametric knowledge rather than structurally labeled knowledge?

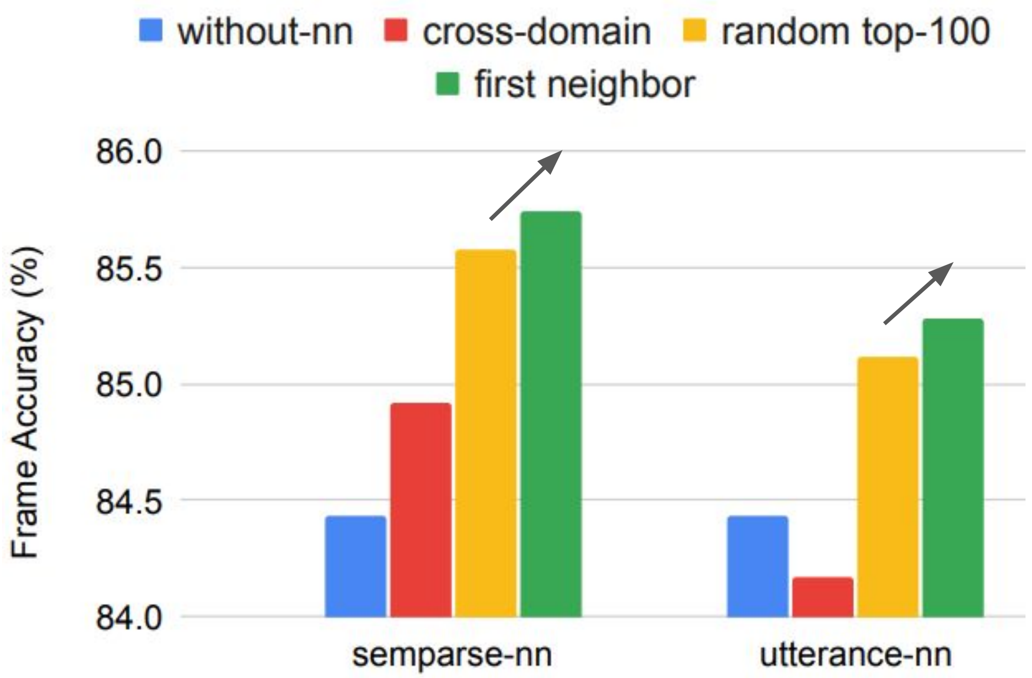
(c) ***Noise Robustness:*** Can a model opt to employ parametric knowledge rather than non-parametric knowledge in a resilient manner, e.g. when the non-parametric information is unreliable?

SENSITIVITY TO NN

Stability with rand-NN



Sensitivity To NN

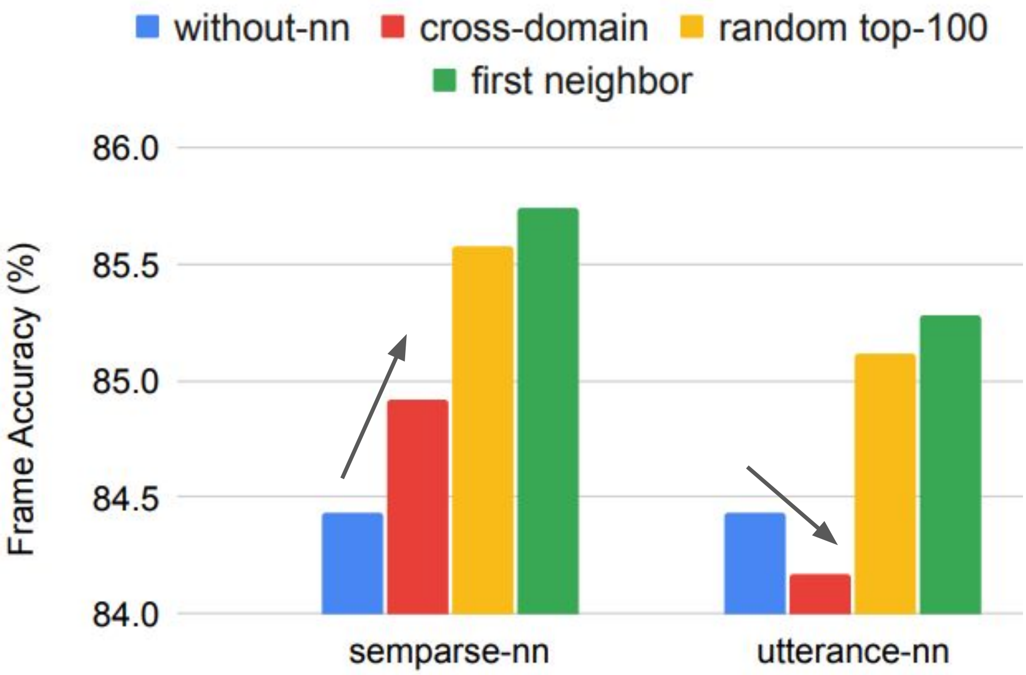


Stability with rand-NN

Improvement much better with better NN

- Top 1 > Top 100 (Random)

Sensitivity To NN



Stability with rand-NN

Improvement much better with better NN

- Top 1 > Top 100 (Random)

Different Domain Random NN

- improvement semparse-nn
- marginally hurts utterance-nn

QUESTIONS

In this work we are focusing on the following questions:

(a) ***Data Efficiency:*** Can retrieval based on non-parametric external knowledge alleviate reliance on parametric knowledge typically acquired via supervised training on large labeled datasets?

(b) ***Limited Supervision:*** Can we enhance models by using abundant and inexpensive unlabeled external non-parametric knowledge rather than structurally labeled knowledge?

(c) ***Noise Robustness:*** Can a model opt to employ parametric knowledge rather than non-parametric knowledge in a resilient manner, e.g. when the non-parametric information is unreliable?

(d) ***Utterance Complexity:*** Is nonparametric external knowledge addition effective for both uncommon and complex structured (hierarchical) examples?

SIMPLE VS COMPLEX UTTERANCE

complex utterance (v1 → v2) , more domains

a. hierarchical nesting
b. multiple intent
    i. [sl:] can have also have [in:
    ii. depth 2 to 7

example with depth
1 : 22409 (81.9%)
2 : 4190 (15.3%)
>= 3 : 737 (2.7%)

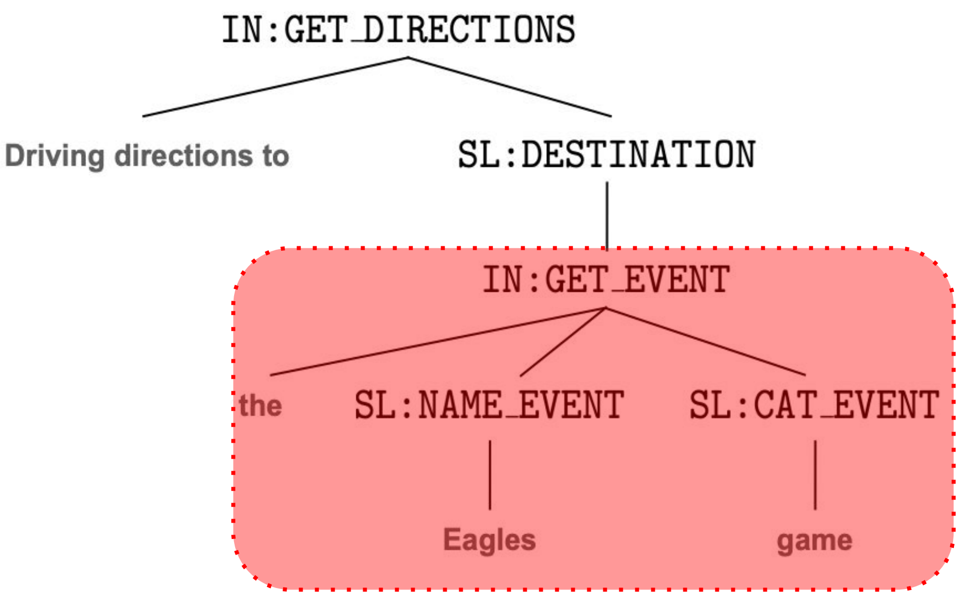
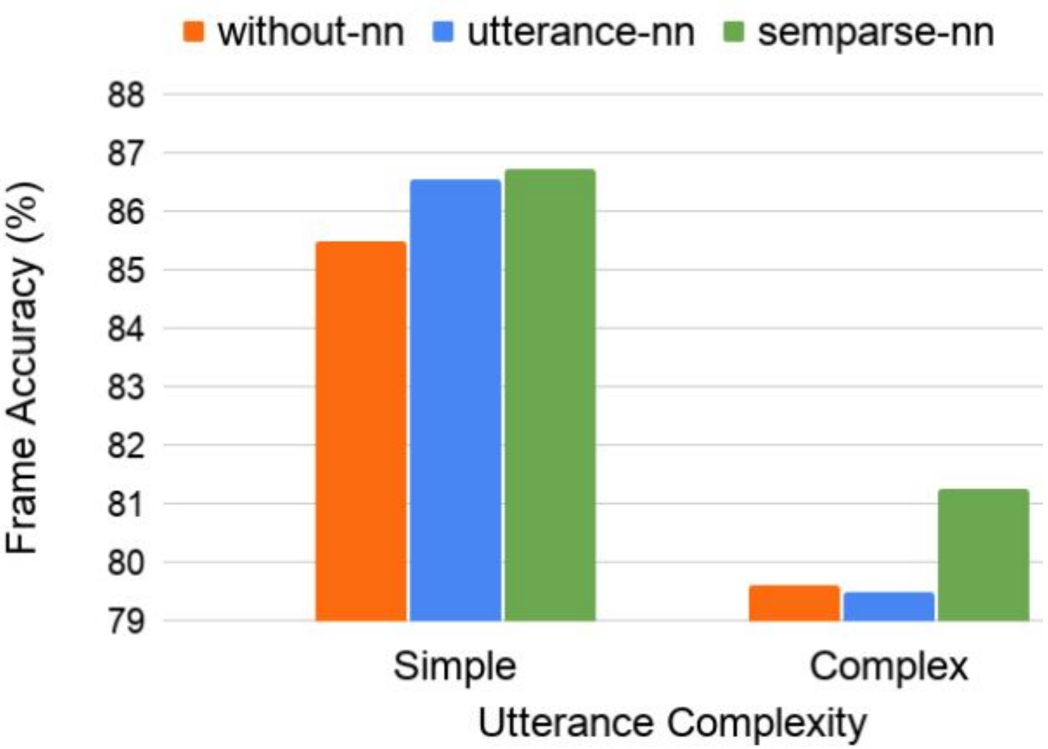


Figure 1: An compositional query from TOP dataset.

[1] Chen, X., Ghoshal, A., Mehdad, Y., Zettlemoyer, L., & Gupta, S. (2020). Low-Resource Domain Adaptation for Compositional Task-Oriented Semantic Parsing. *arXiv preprint arXiv:2010.03546*. EMNLP 2020

PERFORMANCE: SIMPLE VS COMPLEX UTTERANCE



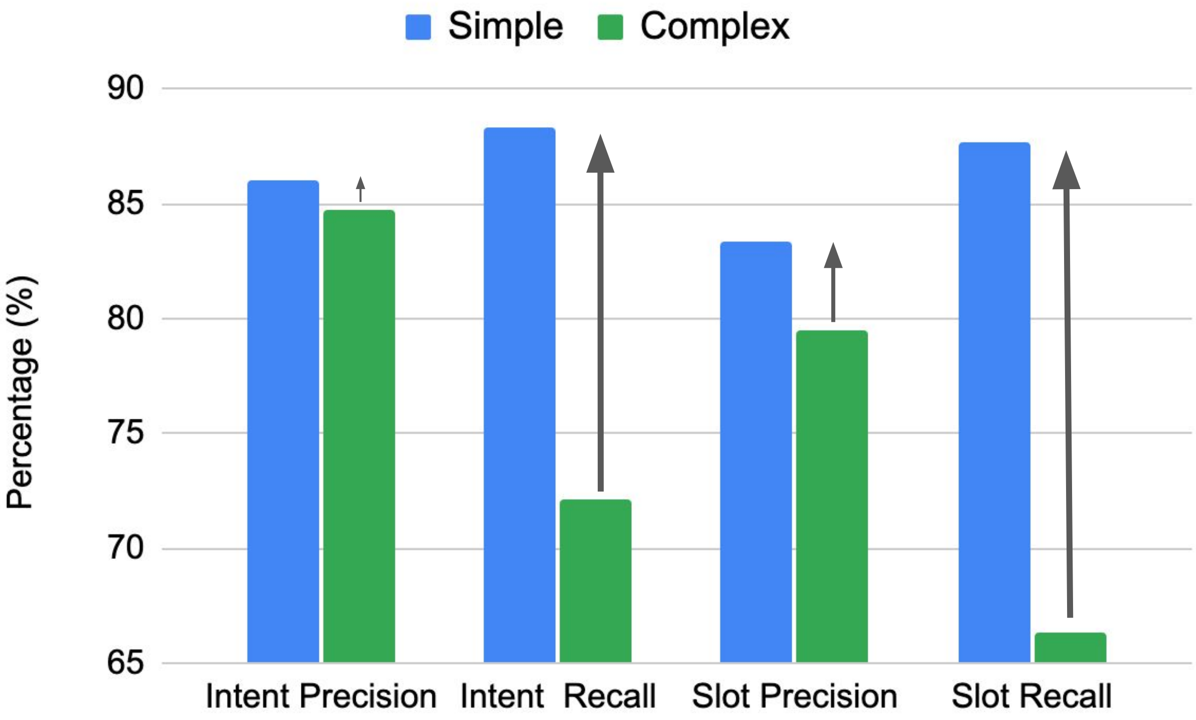
depth >1 is tougher than depth 1

1. Compositional
2. Lesser Data

lesser improvement on depth >1 vs depth 1

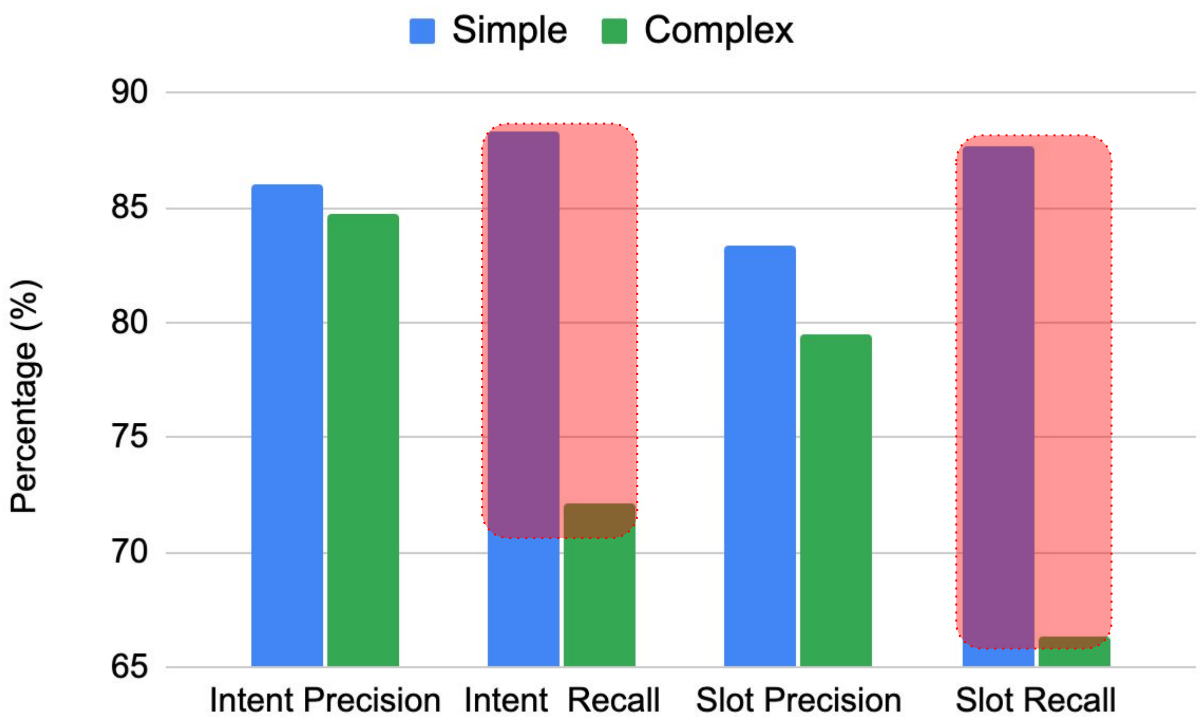
1. lesser data
2. diversity in [in:] & [sl:]
3. complexity in query

RETRIEVAL: SIMPLE VS COMPLEX UTTERANCE



Retrieval NN better for simple than complex

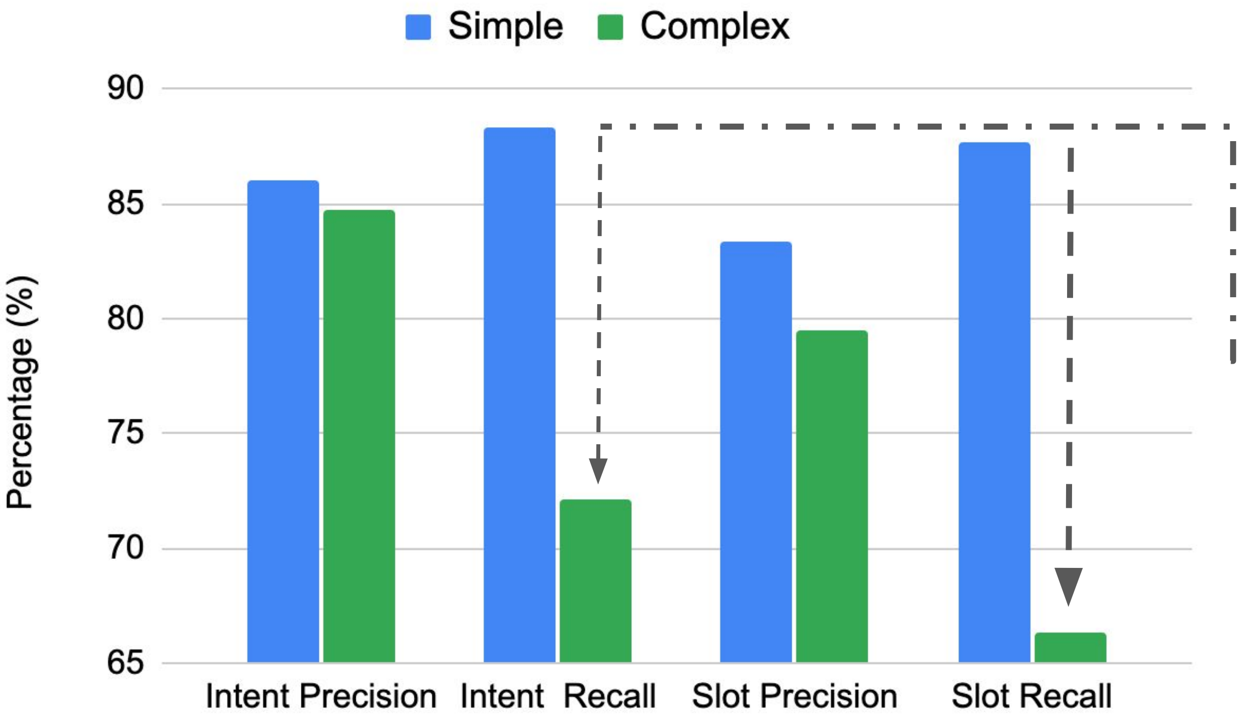
RETRIEVAL: SIMPLE VS COMPLEX UTTERANCE



Retrieval NN better for simple than complex

Recall suffer the most for complex queries

RETRIEVAL: SIMPLE VS COMPLEX UTTERANCE

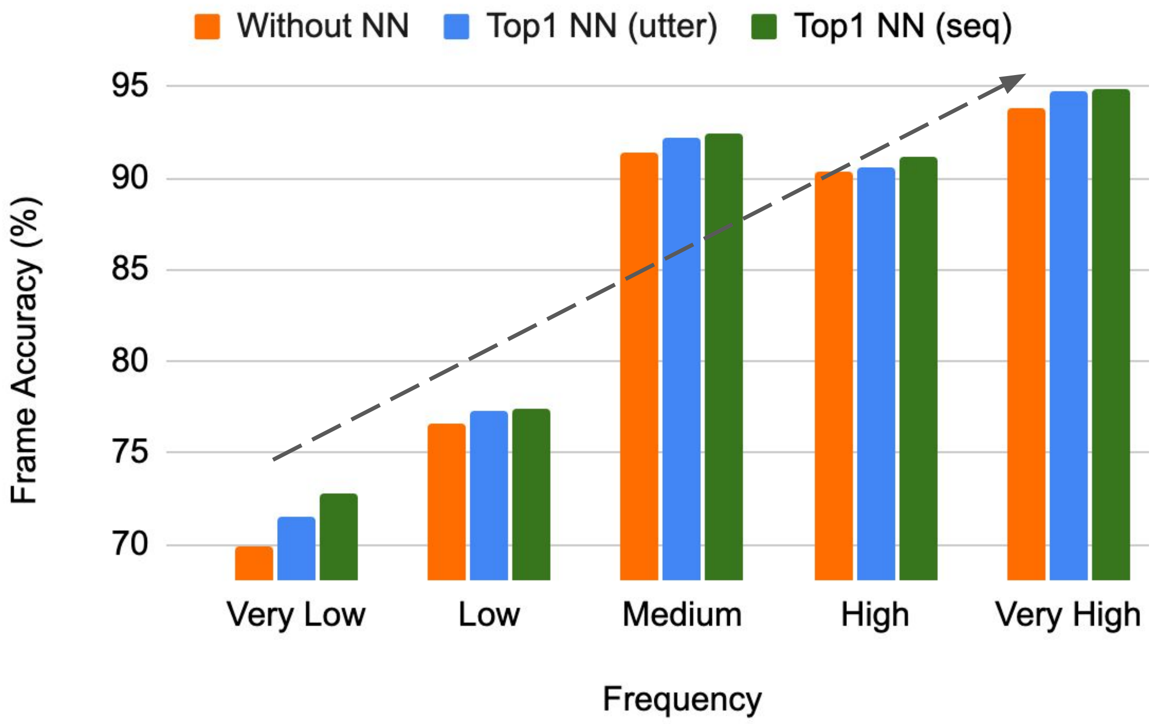


Retrieval NN better for simple than complex

Recall suffer the most for complex queries

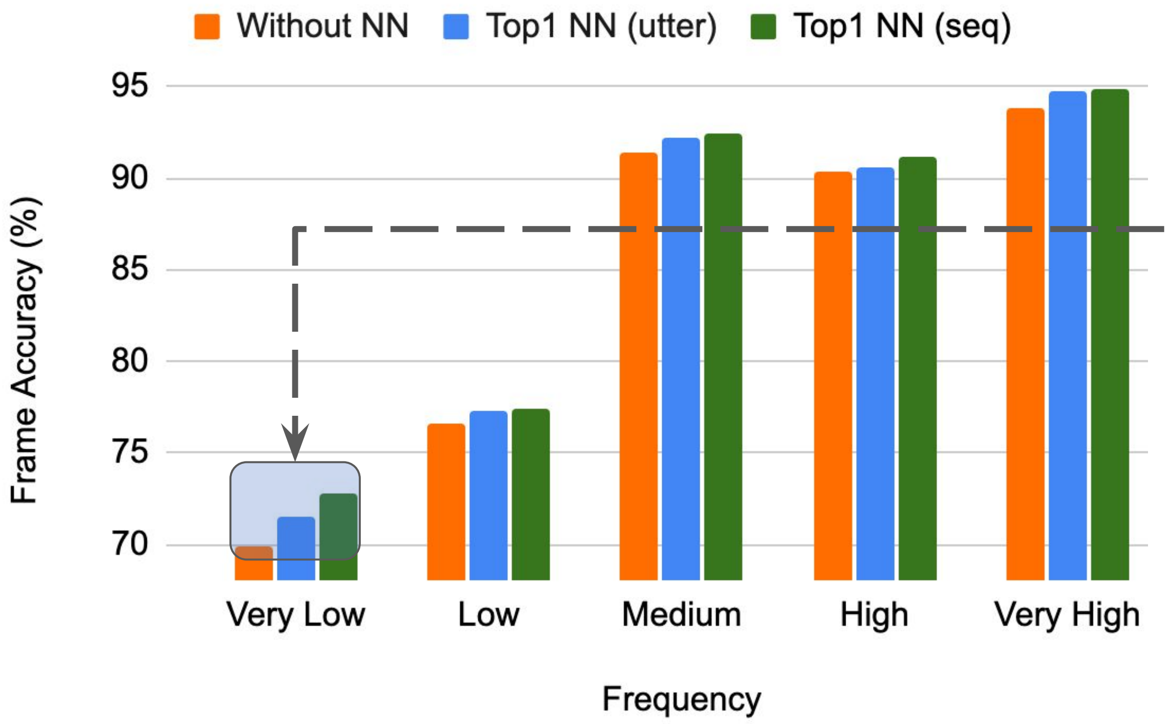
Slot retrieval decreases much more than intent

PERFORMANCE: RARE vs FREQUENT



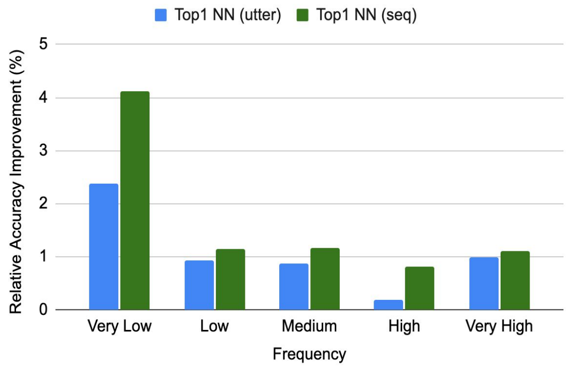
performance increase with frequency

PERFORMANCE: RARE vs FREQUENT

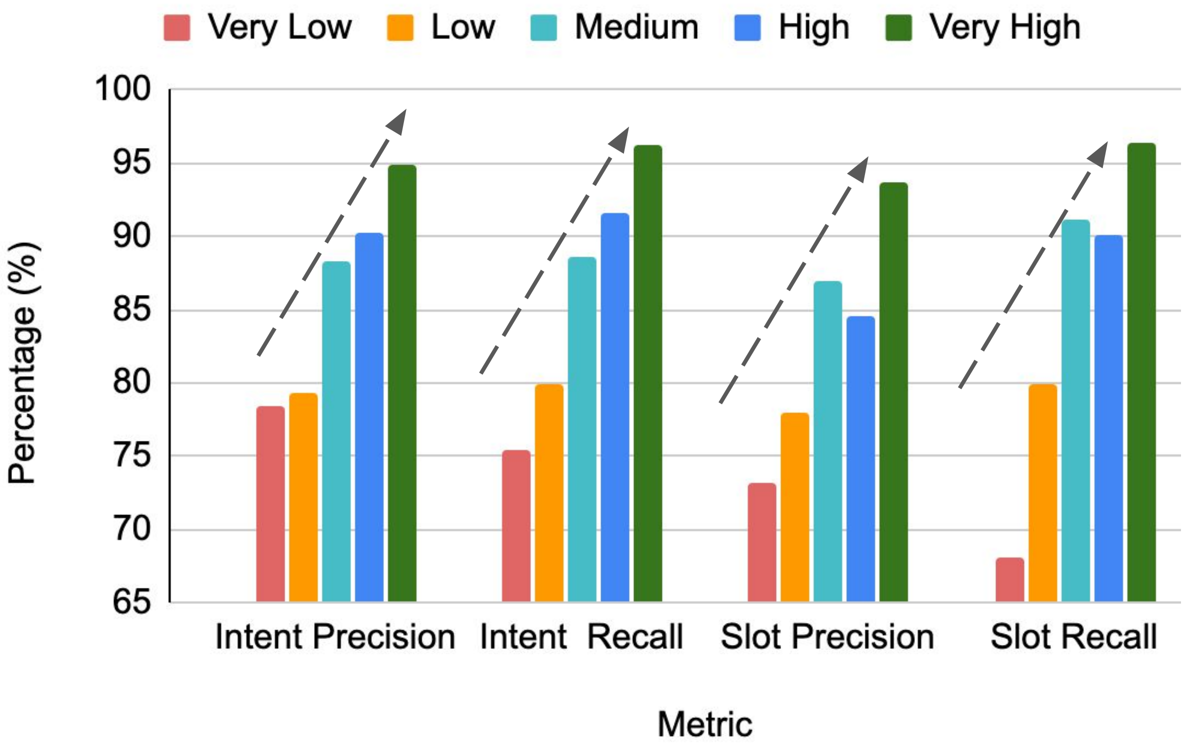


performance increase with frequency

performance improve more for very lower frequency



RETRIEVAL QUALITY (RARE VS FREQUENT)



Retrieval NN better with high frequency

expected at more examples of similar frame structure

Similar trend for intent and slot for precision and recall

QUESTIONS

In this work we are focusing on the following questions:

(a) ***Data Efficiency:*** Can retrieval based on non-parametric external knowledge alleviate reliance on parametric knowledge typically acquired via supervised training on large labeled datasets?

(b) ***Limited Supervision:*** Can we enhance models by using abundant and inexpensive unlabeled external non-parametric knowledge rather than structurally labeled knowledge?

(c) ***Noise Robustness:*** Can a model opt to employ parametric knowledge rather than non-parametric knowledge in a resilient manner, e.g. when the non-parametric information is unreliable?

(d) ***Utterance Complexity:*** Is nonparametric external knowledge addition effective for both uncommon and complex structured (hierarchical) examples?

(e) ***Knowledge Efficiency:*** Is it beneficial to continue adding external information, or are there certain boundaries and challenges?

RETRIEVAL QUALITY

pre-train BART model Index ; Format : {#nn : (precision, recall)}

Train

avg_intent {3: (81.39, 81.81), 2: (82.07, 82.50), 1: (84.84, 85.04)}
avg_slot {3: (75.02, 79.56), 2: (76.06, 80.37), 1: (80.05, 83.19)}

Valid

avg_intent {3: (80.46, 81.10), 2: (82.12, 82.39), 1: (87.59, 87.93)}
avg_slot {3: (73.46, 79.77), 2: (76.80, 81.61), 1: (82.38, 85.81)}

Test

avg_intent {3: (79.09, 79.35), 2: (81.19, 81.34), 1: (86.23, 86.22)}
avg_slot {3: (74.59, 79.51), 2: (77.68, 81.39), 1: (83.21, 85.11)}

Good Quality Retrieval

Pre-train bart embedding is good

Decrease with farther neighbour

- More in valid/test

TAKEAWAY

1. In this work, we explore RETRONLU: retrieval based modeling approach for task-oriented semantic parsing problem.
2. RETRONLU makes explicit use of memory of retrieve examples of semantic parses that the model learn to adapt for other similar input utterance.
3. We analyse the robustness and sensitivity of RETRONLU in several dimensions as follows:
   a. Data Efficiency
   b. Limited Supervision
   c. Noise Robustness
   d. Utterance Complexity
   e. Knowledge Efficiency

Examples

Incorrect after nearest neighbour

Input ⇒ set a timer for 5 minutes at 4 : 30 pm
Target ⇒ [in:create_timer [sl:date_time for 5 minutes at 4 : 30 pm] [sl:method_timer timer]]
Prediction ⇒ [in:create_timer [sl:method_timer timer] [sl:date_time for 5 minutes at 4 : 30 pm]]

Input ⇒ [in:create_timer set [sl:method_timer timer] [sl:date_time for 15 minutes at 2 : 00 pm]] | set a timer for 5 minutes at 4 : 30 pm
Target ⇒ [in:create_timer [sl:method_timer timer] [sl:date_time for 5 minutes at 4 : 30 pm]]
Prediction ⇒ [in:unsupported_timer]

Correct after nearest neighbour

Input ⇒ does the traffic get better after 5 p.m
Target ⇒ [in:get_info_traffic [sl:date_time after 5 p.m]]
Prediction ⇒ [in:unsupported_navigation]

Input ⇒ [in:get_info_traffic [sl:date_time before 5 p.m to 6:00 pm]] | does the traffic get better after 5 p.m
Target ⇒ [in:get_info_traffic [sl:date_time after 5 p.m]]
Prediction ⇒ [in:get_info_traffic [sl:date_time after 5 p.m]]

See More Examples

Extra slides

Follow Up (Target ACL 2021)

1. Improving Retrieval Quality
   a. indexing focus on capturing the semparse structure
   b. diversifying NN by grouping on structural similarity
2. Joint training to Improve modeling
   a. Joint training model with indexing (alt. indexing & training)
   b. Similar to MARGE or ReaLM model
3. Zero-shot setting - just updating index for out-of-domain structure
4. Applications: **Multilingual** / **Conversational** / FB-Marketplace

LITERATURE AND RELEVANT LINKS

1. Proposed Project Proposal
2. Model Literature Review
3. Best Coverage vs Pre-trained Bart NN
4. Experimental Results