DBpedia Group at the 1st Text2SPARQL Challenge

4th International Workshop on LLM-Integrated Knowledge Graphs @ ESWC 2025

Mehrzad Shahinmoghadam, Tommaso Soru, Saurav Joshi, and Sanju Tiwari

Question Answering over DBpedia

with Pre-trained Autoregressive Models

Question Answering over DBpedia

with Pre-trained Autoregressive Models

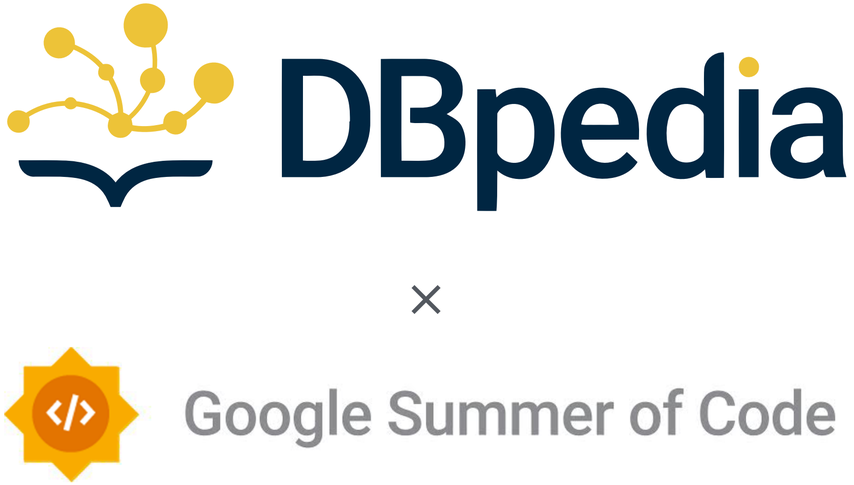
×

Google Summer of Code

Question Answering over DBpedia

with Pre-trained Autoregressive Models

contributor	role
Mehrzad Shahinmoghadam	mentee
Tommaso Soru	main mentor
Sanju Tiwari	mentor
Anand Panchbhai	mentor
Saurav Joshi	junior mentor



Question Answering over DBpedia

with Pre-trained Autoregressive Models

contributor	role
Mehrzad Shahinmoghadam	mentee
Tommaso Soru	main mentor
Sanju Tiwari	mentor
Anand Panchbhai	mentor
Saurav Joshi	junior mentor

×

GSoC projects: 2018–2023
This project: May–November 2023

Approach

Fine-tuning of pre-trained code LLMs

Approach

Fine-tuning of pre-trained code LLMs

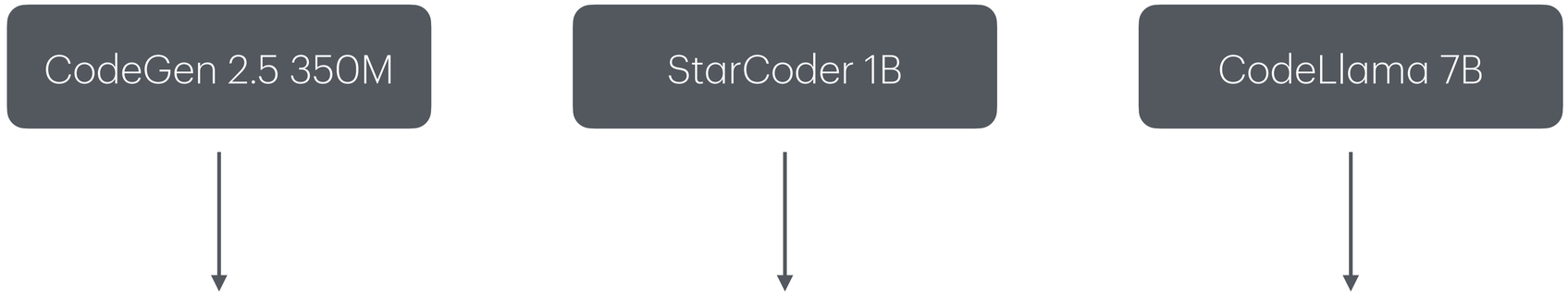
CodeGen 2.5 350M

StarCoder 1B

CodeLlama 7B

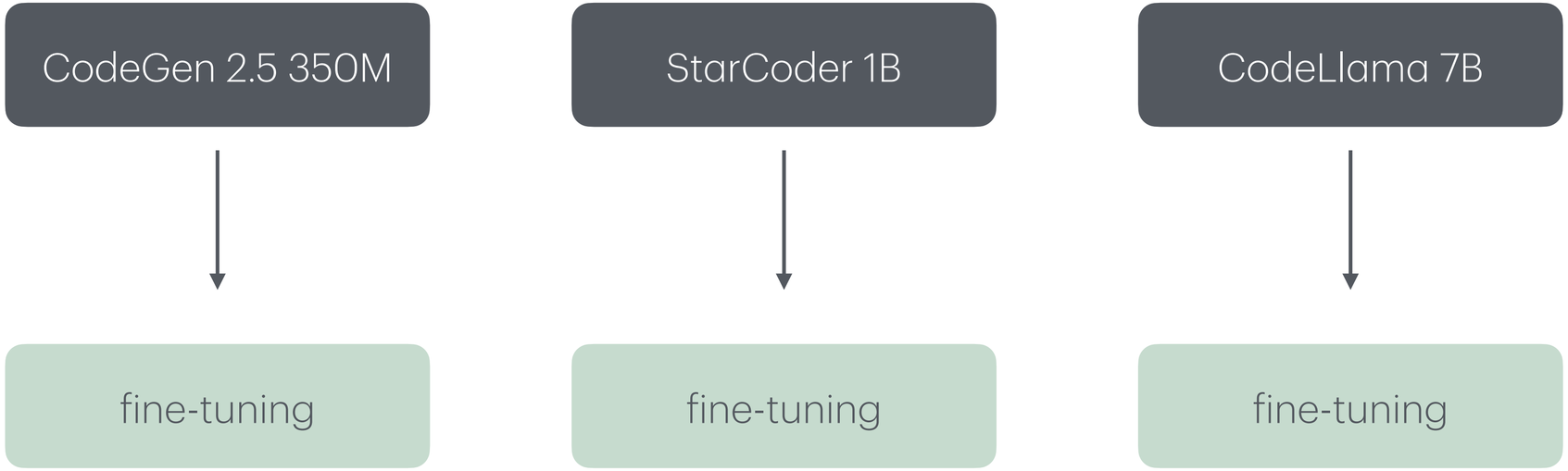
Approach

Fine-tuning of pre-trained code LLMs



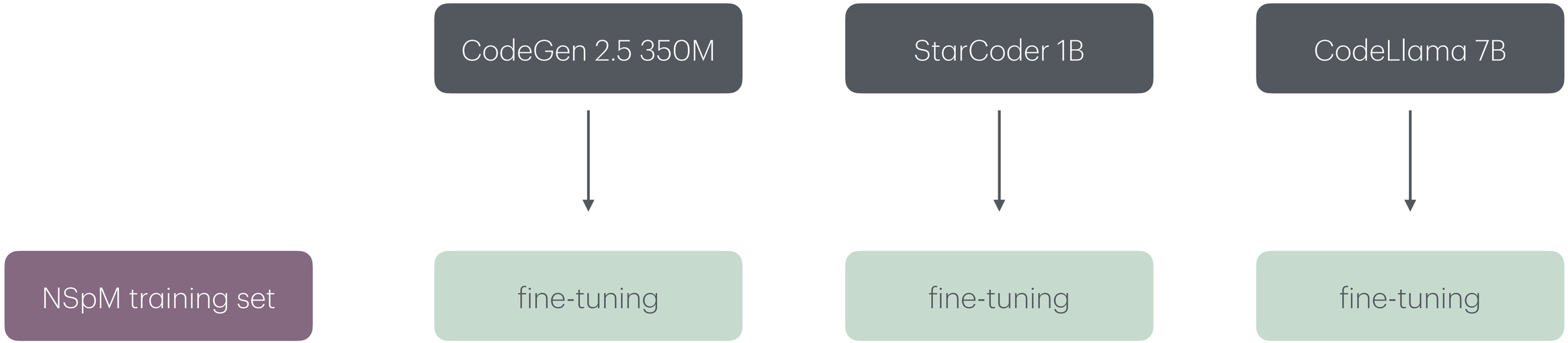
Approach

Fine-tuning of pre-trained code LLMs



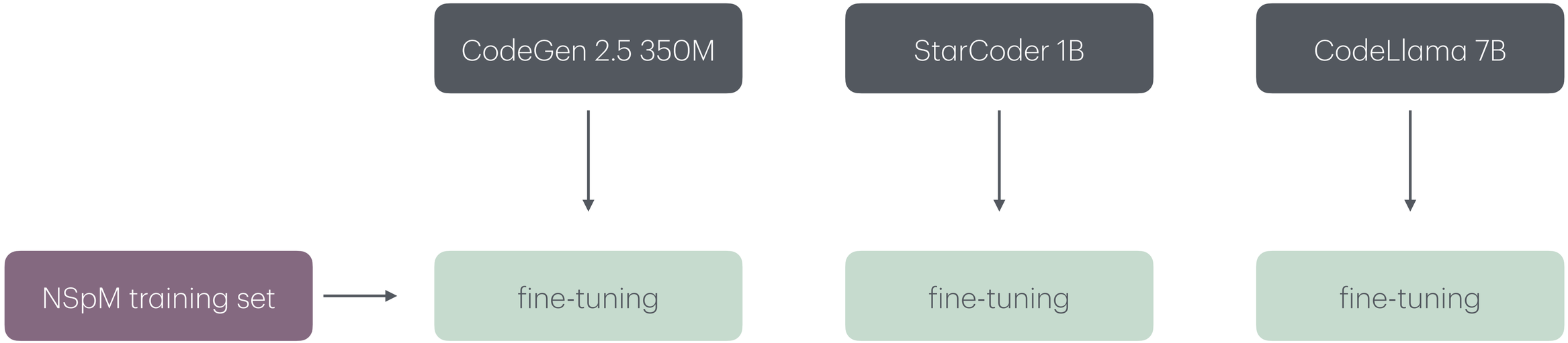
Approach

Fine-tuning of pre-trained code LLMs



Approach

Fine-tuning of pre-trained code LLMs

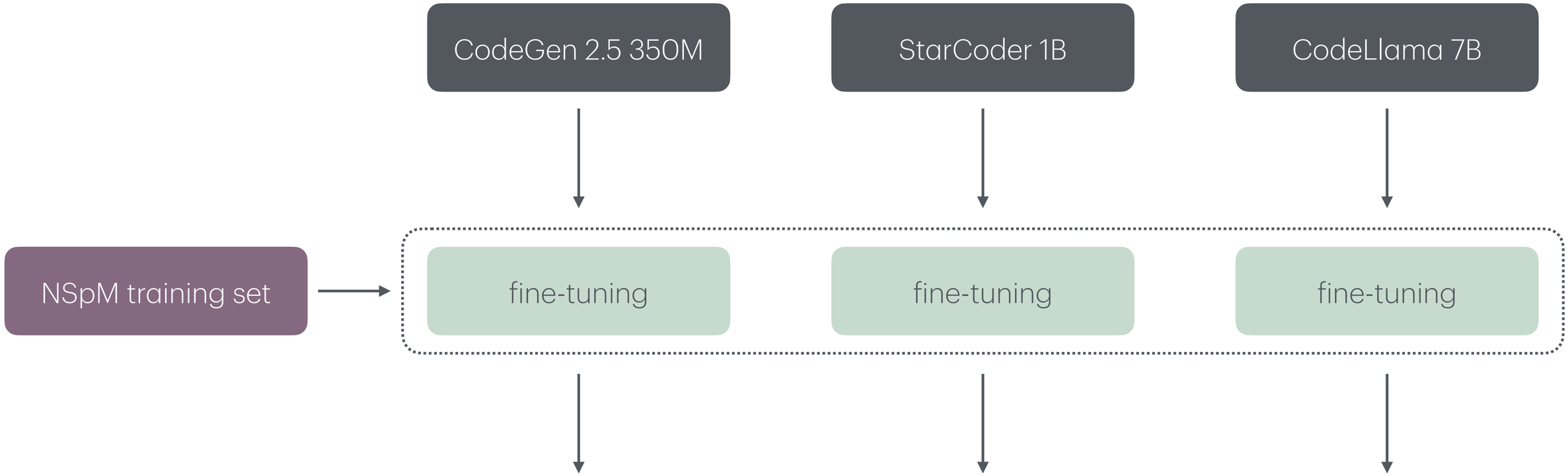


Approach

Fine-tuning of pre-trained code LLMs

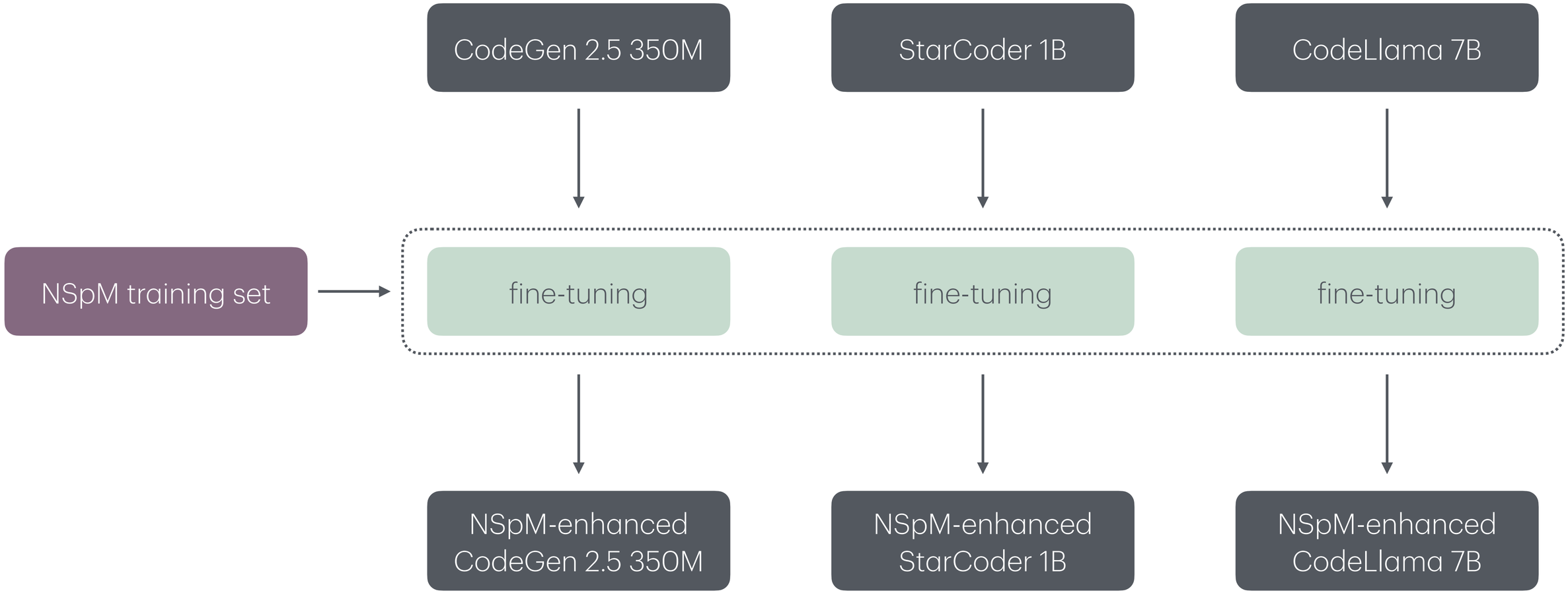
Approach

Fine-tuning of pre-trained code LLMs



Approach

Fine-tuning of pre-trained code LLMs



The NSpM training set

Created during Google Summer of Code 2021–22

Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

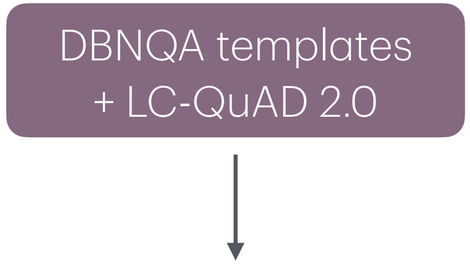
Created during Google Summer of Code 2021–22

DBNQA templates
+ LC-QuAD 2.0

Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

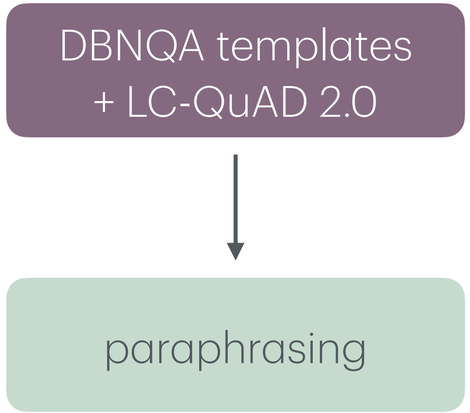
Created during Google Summer of Code 2021–22



Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

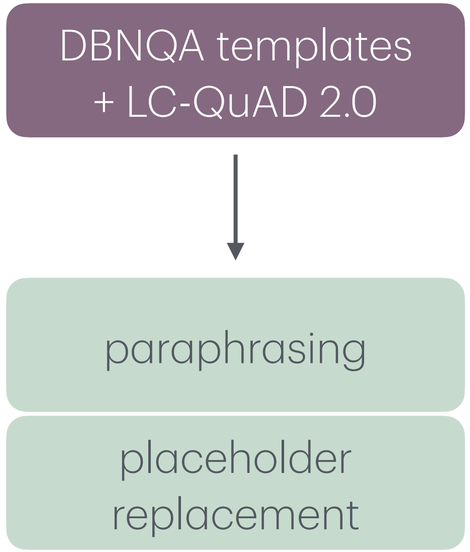
Created during Google Summer of Code 2021–22



Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

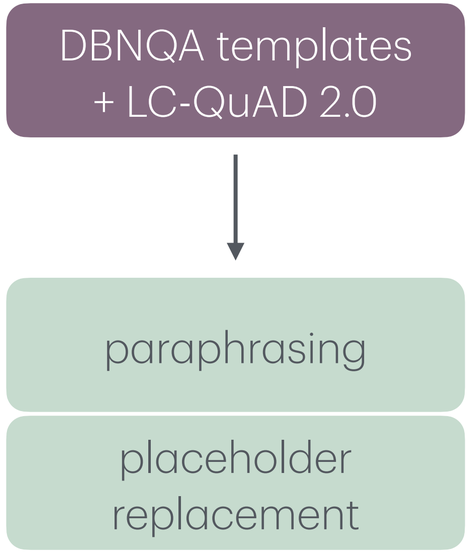
Created during Google Summer of Code 2021–22



Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

Created during Google Summer of Code 2021–22

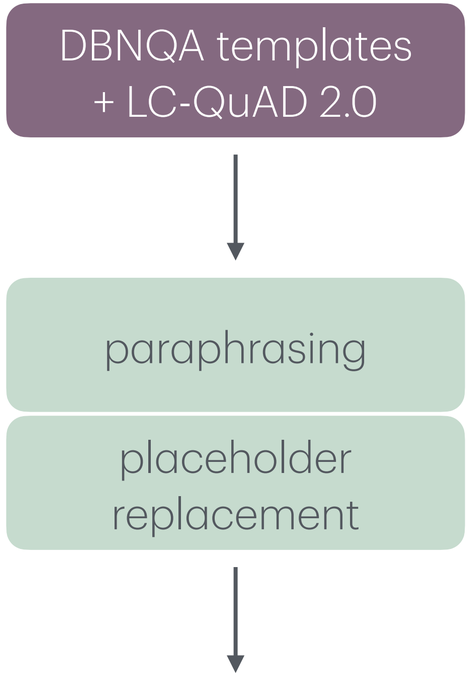


select ?x { ?x rdf:type <A> }

Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

Created during Google Summer of Code 2021–22

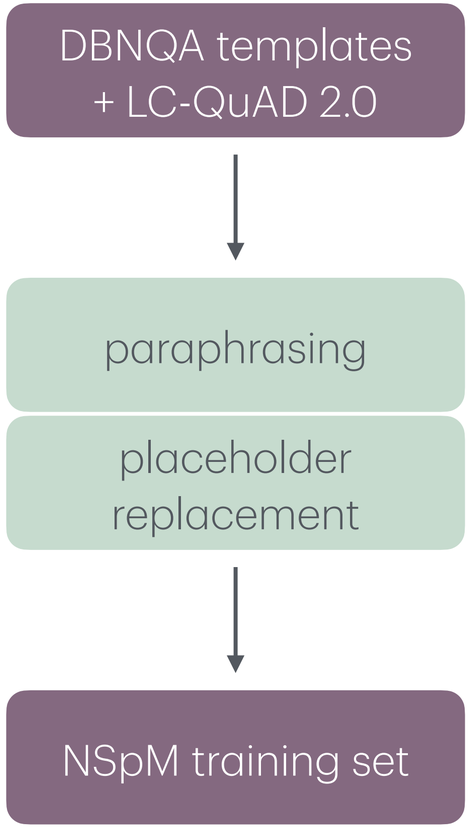


select ?x { ?x rdf:type <A> }

Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

Created during Google Summer of Code 2021–22

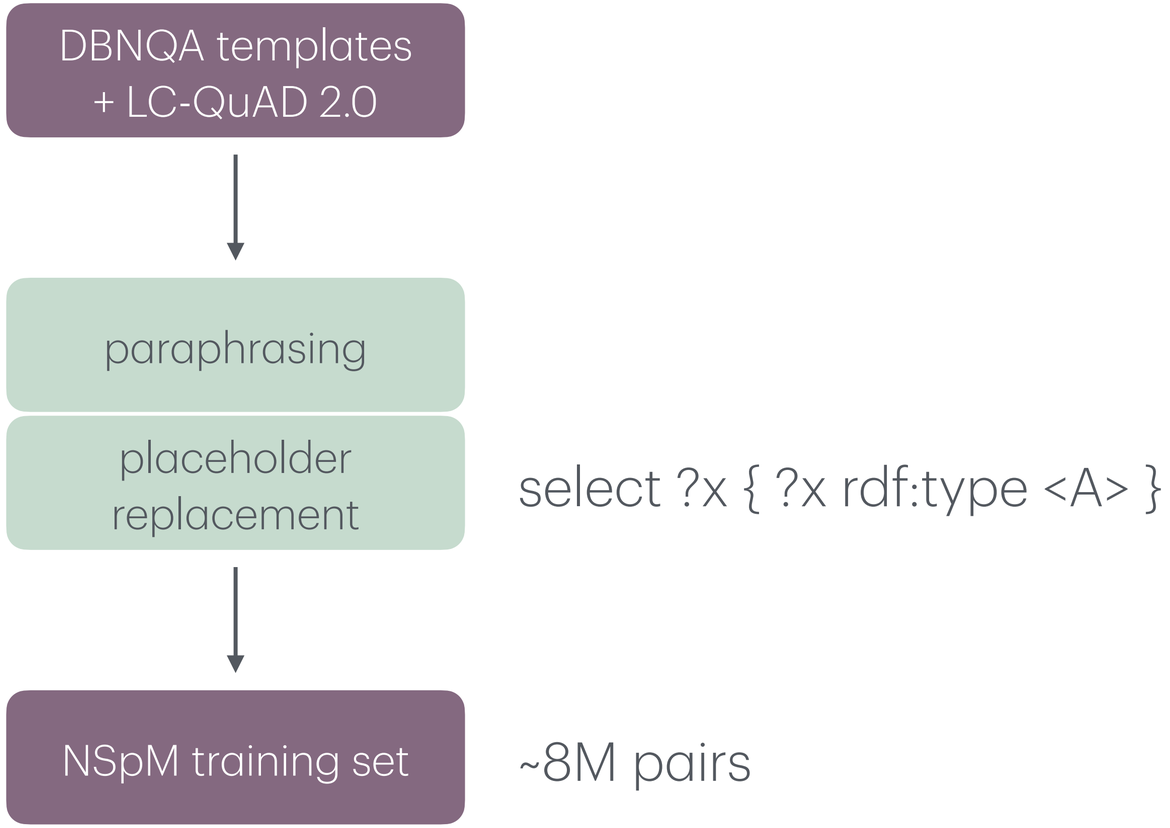


select ?x { ?x rdf:type <A> }

Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

The NSpM training set

Created during Google Summer of Code 2021–22



select ?x { ?x rdf:type <A> }

~8M pairs

Hartmann, A.K., Marx, E. and Soru, T., 2018, April. Generating a large dataset for neural question answering over the DBpedia knowledge base
Dubey, M., Banerjee, D., Abdelkawi, A. and Lehmann, J., 2019. Lc-quad 2.0: A large dataset for complex question answering over Wikidata and DBpedia

Setup

Setup

- 2x Nvidia GeForce RTX 3090

Setup

- 2x Nvidia GeForce RTX 3090
- ~5 days per 1B parameters

Setup

- 2x Nvidia GeForce RTX 3090
- ~5 days per 1B parameters
- Track BLEU score between output SPARQL and ground truth

Setup

- 2x Nvidia GeForce RTX 3090
- ~5 days per 1B parameters
- Track BLEU score between output SPARQL and ground truth
- Weight optimisation done through typical, straightforward, ‘ignorant’ Instruction Tuning

Evaluation

Evaluation

- BLEU score during and after fine-tuning

Evaluation

- BLEU score during and after fine-tuning
- Only participated in 2nd challenge (DBpedia)

Evaluation

- BLEU score during and after fine-tuning
- Only participated in 2nd challenge (DBpedia)
- No evaluation against result-sets

Evaluation

- BLEU score during and after fine-tuning
- Only participated in 2nd challenge (DBpedia)
- No evaluation against result-sets
- Transfer learning towards a different test set:

Evaluation

- BLEU score during and after fine-tuning
- Only participated in 2nd challenge (DBpedia)
- No evaluation against result-sets
- Transfer learning towards a different test set:

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches
- Consider the question: “What is the capital of Germany?”

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches
- Consider the question: “What is the capital of Germany?”
- Translated BGP: dbr:Germany dbo:largestCity ?x

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches
- Consider the question: “What is the capital of Germany?”
- Translated BGP: dbr:Germany dbo:largestCity ?x
- Result-set: dbr:Berlin

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches
- Consider the question: “What is the capital of Germany?”
- Translated BGP: dbr:Germany dbo:largestCity ?x
- Result-set: dbr:Berlin
- Correct result-set, wrong query

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches
- Consider the question: “What is the capital of Germany?”
- Translated BGP: dbr:Germany dbo:largestCity ?x
- Result-set: dbr:Berlin
- Correct result-set, wrong query
- Even weirder with empty result-sets

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Result-set is necessary to evaluate against non-SPARQL approaches
- Consider the question: “What is the capital of Germany?”
- Translated BGP: dbr:Germany dbo:largestCity ?x
- Result-set: dbr:Berlin
- Correct result-set, wrong query
- Even weirder with empty result-sets
- Benchmarks have a massive selection bias towards non-empty queries

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Idea: Use ARQ algebra from SPARQL

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Idea: Use ARQ algebra from SPARQL

```
select ?p (count(*) as ?c)
where { ?s ?p ?o }
group by ?p
order by desc(?c)
```

```
(project (?p ?c)
  (order ((desc ?c))
    (extend ((?c ?.0))
      (group (?p) ((?.0 (count)))
        (bgp (triple ?s ?p ?o))))))
```

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Idea: Use ARQ algebra from SPARQL
- Can deal with different queries with same semantics

```
select ?p (count(*) as ?c)
where { ?s ?p ?o }
group by ?p
order by desc(?c)
```

```
(project (?p ?c)
  (order ((desc ?c))
    (extend ((?c ?.0))
      (group (?p) ((?.0 (count)))
        (bgp (triple ?s ?p ?o))))))
```

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Idea: Use ARQ algebra from SPARQL
- Can deal with different queries with same semantics
- Ignores formatting, unlike BLEU

```
select ?p (count(*) as ?c)
where { ?s ?p ?o }
group by ?p
order by desc(?c)
```

```
(project (?p ?c)
  (order ((desc ?c))
    (extend ((?c ?.0))
      (group (?p) ((?.0 (count)))
        (bgp (triple ?s ?p ?o))))))
```

By the way, about evaluation…

Is Result-Set matching still the best way to perform it?

- Idea: Use ARQ algebra from SPARQL
- Can deal with different queries with same semantics
- Ignores formatting, unlike BLEU

```
select ?p (count(*) as ?c)
where { ?s ?p ?o }
group by ?p
order by desc(?c)
```

```
(project (?p ?c)
  (order ((desc ?c))
    (extend ((?c ?.0))
      (group (?p) ((?.0 (count)))
        (bgp (triple ?s ?p ?o))))))
```

Python wrapper: https://github.com/LiberAI/arq-algebra/

Thank you!