Text2MQL: Fine-tuning Open-source Language Models for Model Query Languages Using ChatGPT

Máté Földiák, José Antonio Hernández López, Lena Buffoni, Dániel Varró

LINKÖPING UNIVERSITY

Background

Systems Modeling

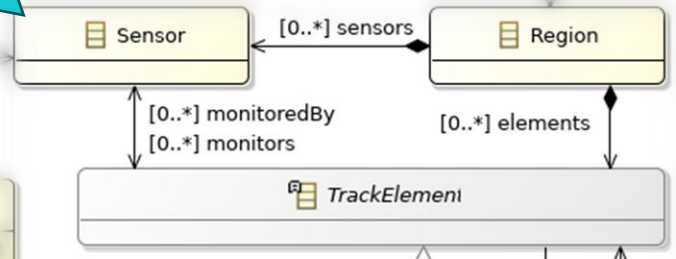
- High-level system models
  - UML, SysML, EMF
- Graph-based formalism
  - Labelled nodes and edges
  - Attributes
- Domain-specific terminology

Model Query Languages

- Core part of model-driven engineering
- Use cases
  - Check models for inconsistencies
  - Formalize model transformation
  - Support tooling
- Rely on formal semantics
- Used during development of the system

Model query example

Domain model (meta-model)



Requirement:
All tracks shall be monitored by a sensor.

Query description:
Retrieve all elements that are not monitored by any sensors.

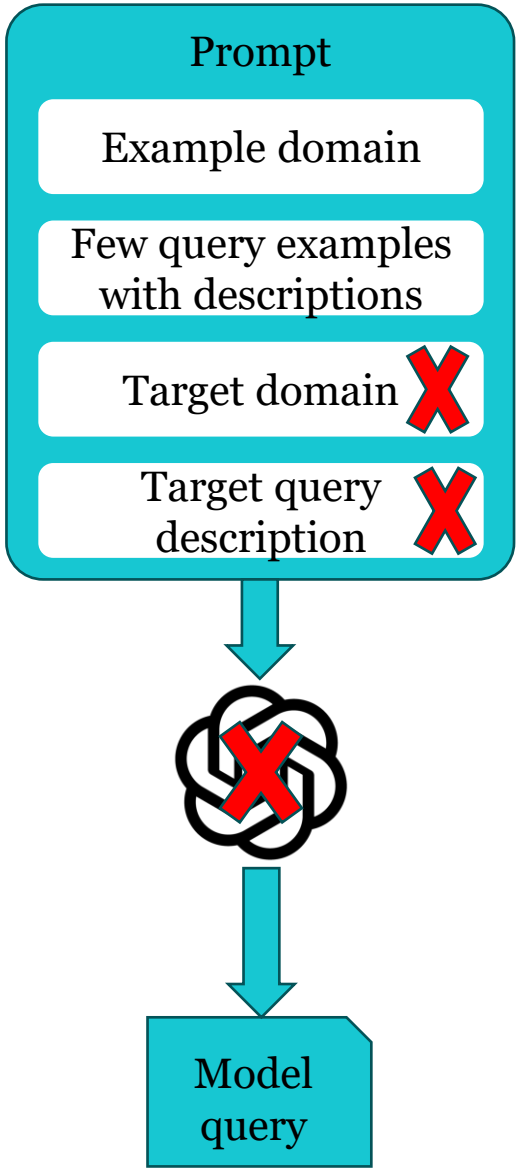
```
pattern elementsNotMonitored(element: TrackElement){
    neg TrackElement.monitoredBy(element, _);
}
```

System model

element
Track_41
Track_75
Track_1373

AI assistance

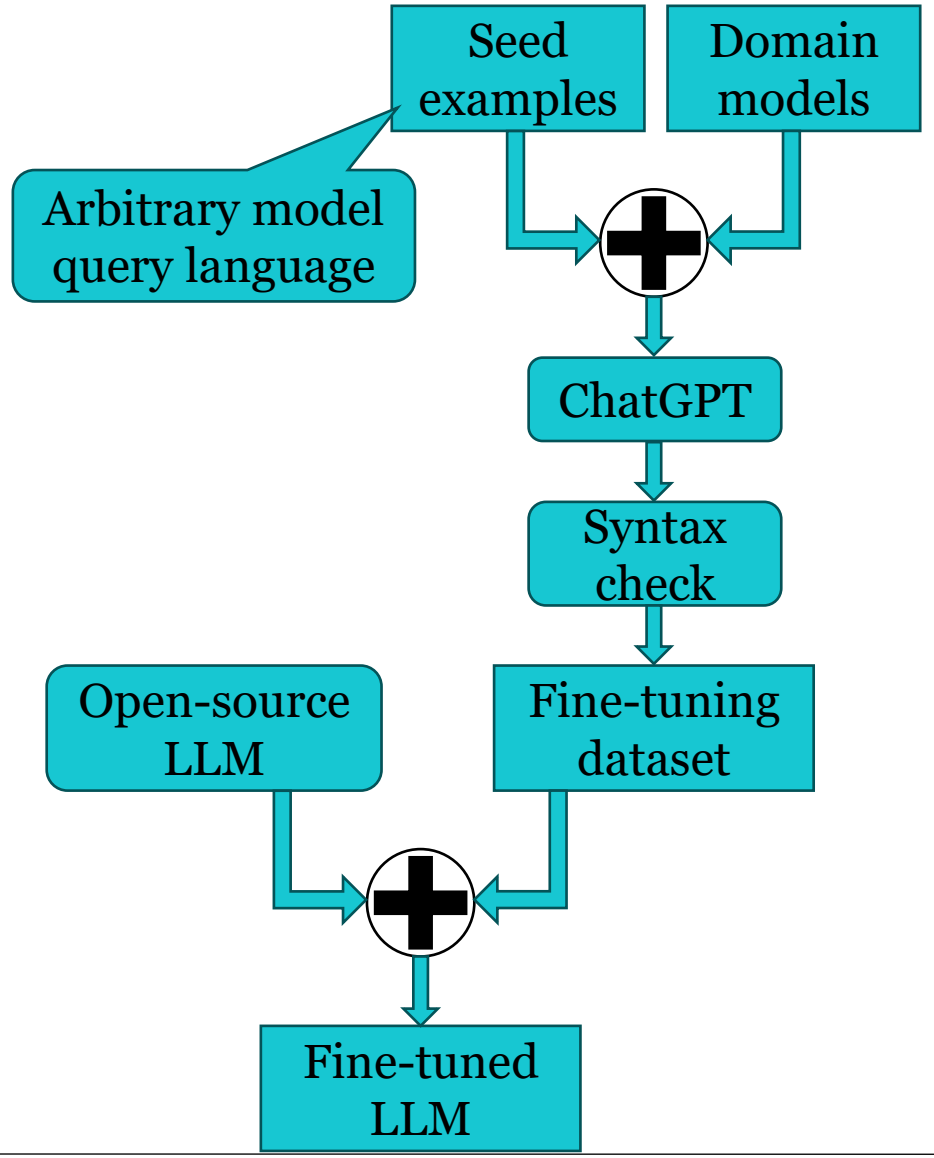
- Sensitive industrial applications
  - Domain models and queries might be confidential
- State of the art LLMs are proprietary
- Lack of public examples
  - Excluded from LLM training data
- State of the art LLMs are large and expensive



The Text2MQL framework

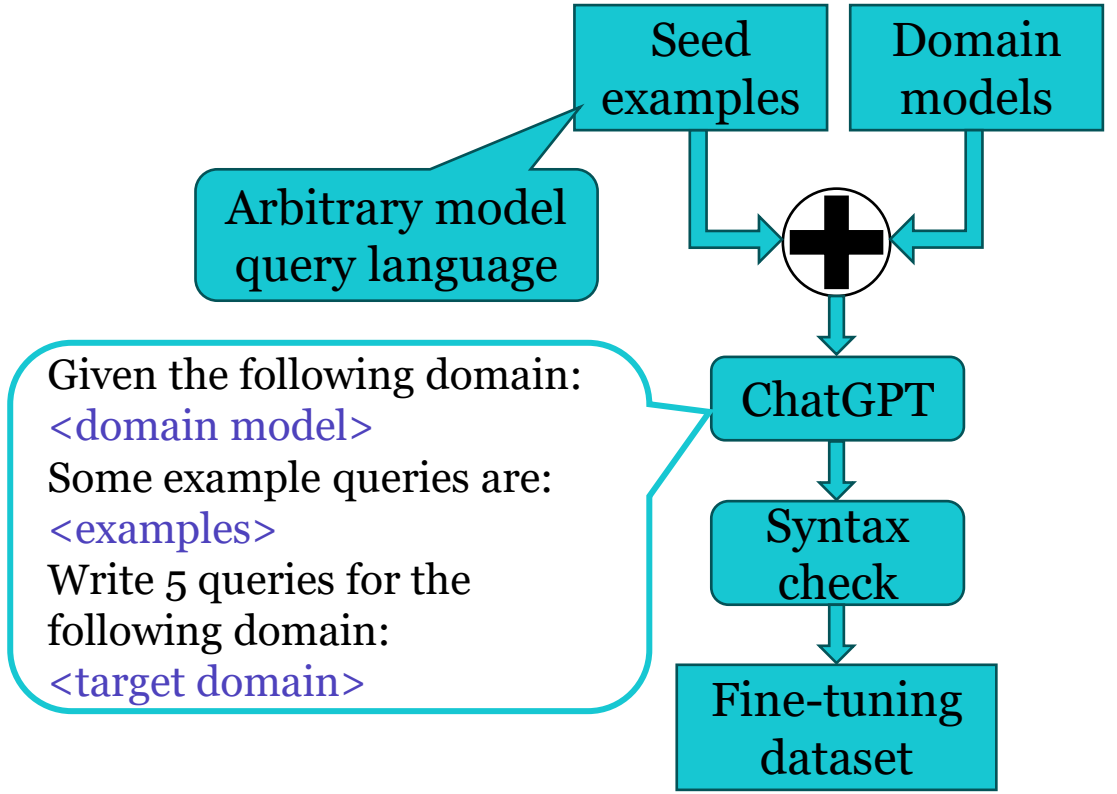
Text2MQL framework

- Lack of examples
  - Generate datasets for model query languages for fine-tuning
- Very large external LLM
  - Fine-tune open-source models
  - Use small LLMs
- Evaluation framework



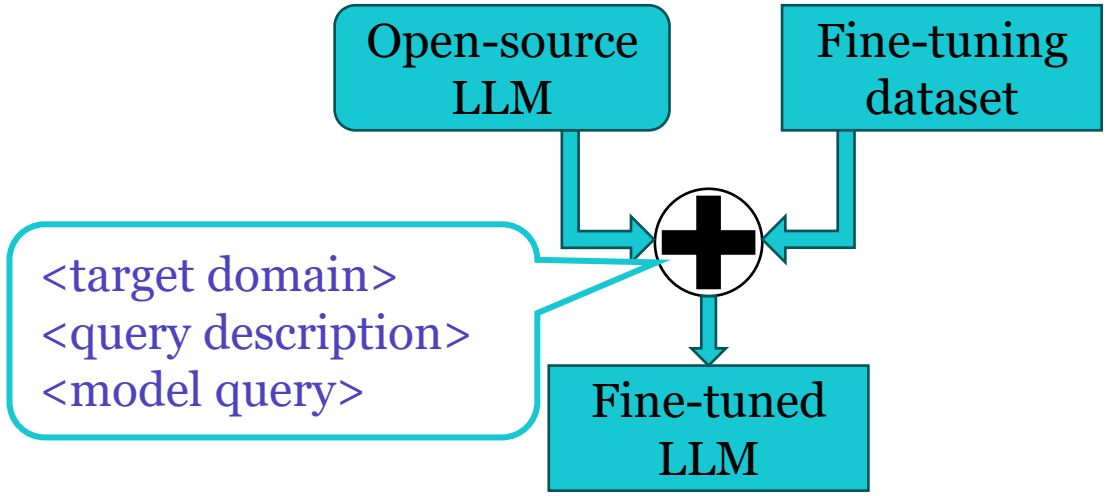
Dataset construction

- Goal: Generate examples for an arbitrary model query language
  - With minimal work
- Process
  - Select a seed domain (public)
  - Design examples for language features
    - 2-5 example for each feature
  - Expand the examples for different domains
    - Public domains from GitHub
    - Expansion with ChatGPT
  - Keep syntactically correct examples



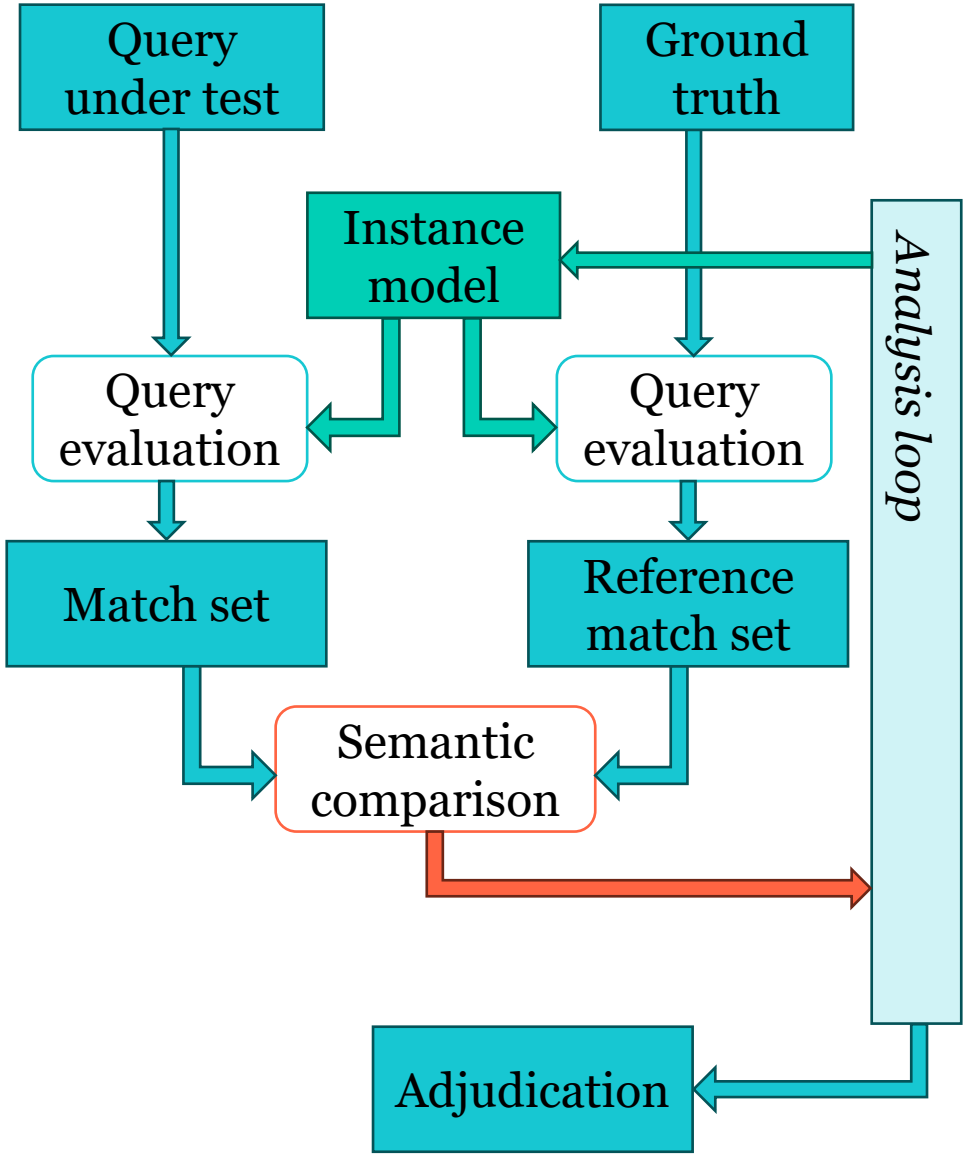
Finetuning

- Fine-tune LLMs to predict the query based on
  - Domain model
  - Query description
- Open-source models
  - CodeLLama (8B)
  - DeepSeekCoder (1.3B, 7B)
  - Qwen2.5 Coder (1.5B, 7B)
  - Qwen3 (1.7B, 8B)



Evaluation

- Differential testing
  - Ground truth query (reference)
  - Identical matches on a model
    - Semantic equivalence
- Evaluate it on many models
  - Generated from domain model and additional constraints



Results

Results

- RQ1: To what extent fine-tuning with our datasets improve performance?
- RQ2: How well does fine-tuned models perform compared to a state-of-the-art LLM?

Evaluation setup

- Model query languages
  - VQL: VIATRA Query Language
  - OCL: Object Constraint Language
  - Java (Eclipse Modeling Framework)
- Benchmark domains
  - TB: Train Benchmark
  - DLT: Blockchain
  - CPS: Cyber-physical Systems
- 400 synthetic instance models

RQ1: To what extent fine-tuning with our datasets improve performance?

- In-context learning (**IC**, baseline)
  - Few examples in each prompt
  - Only a few correct solution
- After fine-tuning (**FT**)
  - All language features learned
  - >2x as many correct queries
  - Consistent improvements (**Δ**) across all query languages
    - With some exceptions in **DLT**

	VQL							OCL							JAVA						
Benchmark	DeepSeek Coder 1.3B	Qwen2.5 Coder 1.5B	Qwen3 1.7B	CodeLlama 7B	DeepSeek Coder 7B	Qwen2.5 Coder 7B	Qwen3 8B	DeepSeek Coder 1.3B	Qwen2.5 Coder 1.5B	Qwen3 1.7B	CodeLlama 7B	DeepSeek Coder 7B	Qwen2.5 Coder 7B	Qwen3 8B	DeepSeek Coder 1.3B	Qwen2.5 Coder 1.5B	Qwen3 1.7B	CodeLlama 7B	DeepSeek Coder 7B	Qwen2.5 Coder 7B	Qwen3 8B
IC (max 11)	0	1	1	2	3	3	3	1	2	3	5	5	7	4	2	3	3	3	7	6	7
FT (max 11)	10	7	7	9	9	9	10	9	10	10	10	10	9	9	9	9	9	10	10	8	9
IC (max 6)	1	2	3	2	2	3	2	0	1	1	2	2	3	2	1	2	0	2	2	2	2
TB Δ	+1	+1	-1	+2	+1	+1	+3	+3	+2	+1	+1	+1	+0	+1	+1	+0	+2	+2	+1	+0	+2
IC (max 9)	0	1	0	1	3	1	0	0	0	0	0	0	0	0	1	0	0	0	1	2	0
DLT Δ	+4	+2	+3	+4	+4	+5	+6	+1	+1	+1	+1	+1	+1	+1	+0	+1	+1	+1	+4	+0	+2
IC (max 12)	0	2	0	1	3	3	3	2	1	0	1	5	5	3	1	3	3	4	4	4	6
CPS Δ	+8	+7	+7	+8	+6	+6	+5	+5	+3	+5	+5	+2	+2	+2	+6	+3	+5	+4	+5	+3	+1
IC (max 38)	1	5	3	4	8	5	5	2	2	1	3	7	8	5	3	5	3	6	7	8	8
Total Δ	+13	+10	+9	+14	+11	+12	+14	+9	+6	+7	+7	+4	+3	+4	+7	+4	+8	+7	+10	+3	+5

IC: Baseline performance

Δ: Improvements

RQ2: How well does fine-tuned models perform compared to a state-of-the-art LLM?

- Baseline: ChatGPT/GPT-5
  - In-context configuration
  - Good performance
- Fine-tuned LLMs
  - At most 8B parameters
  - 4 fine-tuned LLMs perform at least as good as GPT-5
  - 5 fine-tuned LLMs perform almost (-1) as good as GPT-5

	VQL								OCL								JAVA							
Benchmark	GPT-5	DeepSeek Coder 1.3B	Qwen2.5 Coder 1.5B	Qwen3 1.7B	CodeLlama 7B	DeepSeek Coder 7B	Qwen2.5 Coder 7B	Qwen3 8B	GPT-5	DeepSeek Coder 1.3B	Qwen2.5 Coder 1.5B	Qwen3 1.7B	CodeLlama 7B	DeepSeek Coder 7B	Qwen2.5 Coder 7B	Qwen3 8B	GPT-5	DeepSeek Coder 1.3B	Qwen2.5 Coder 1.5B	Qwen3 1.7B	CodeLlama 7B	DeepSeek Coder 7B	Qwen2.5 Coder 7B	Qwen3 8B
TB (of 17)	(11)	+1	-1	-2	+2	+1	+2	+4	(13)	-1	+0	-1	+0	+0	-1	-1	(12)	-1	-1	-1	+2	+1	-2	+1
DLT (of 9)	(8)	-4	-5	-5	-3	-1	-2	-2	(2)	-1	-1	-1	-1	-1	-1	-1	(7)	-6	-6	-6	-6	-2	-5	-5
CPS (of 12)	(9)	-1	+0	-2	+0	+0	+0	-1	(6)	+1	-2	-1	+0	+1	+1	-1	(9)	-2	-3	-1	-1	+0	-2	-2
Total (of 38)	(28)	-4	-6	-9	-1	+0	+0	+1	(21)	-1	-3	-3	-1	+0	-1	-3	(28)	-9	-10	-8	-5	-1	-9	-6

Baseline performance

Relative performances

Conclusion and future work

- Conclusion – Text2MQL
  - Generalizes for model query languages
  - Extensible for many model query languages
- Future work
  - Improve performance on more complex queries
    - Improved dataset generation methods

Text2MQL: Fine-tuning Open-source Language Models for Model Query Languages Using ChatGPT

Publications

https://dl.acm.org/doi/10.1145/3640310.3674091

Text2VQL

Code

https://github.com/PELAB-LiU/Text2VQL

- *extension* branch for Text2MQL (VQL, OCL, Java)
- *main* branch for Text2VQL (VQL only)