COntExt: Towards Context-Aware Ontology Extension from Operational Metrics

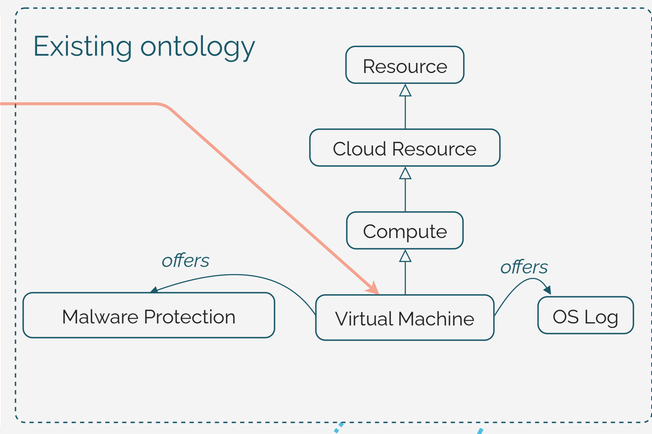
Hussain Hussain[1], Stefan Schöberl[2], Angelika Schneider[3], and Verena Geist[2]

[1]Know Center Research GmbH, Graz, Austria, [2]Software Competence Center Hagenberg GmbH, Hagenberg im Mühlkreis, Austria, [3]Fraunhofer AISEC, Garching bei München, Germany

Organizations increasingly define **operational metrics** in structured formats encoding domain knowledge that extends what is captured in **formal ontologies** → ***Can we use ontology annotations and metric annotations to extend the existing ontology?***

Input: Metric definitions
Structured file (e.g., YAML, JSON, XML), referencing *domain concepts* through, e.g., typed class, property references, configuration parameters, or textual descriptions.

```
# =========== METRIC DEFINITION =========== #
name: AutomaticUpdatesEnabled
description: This rule assesses whether a [VirtualMachine]
that offers the property [AutomaticUpdates], has [p1:enabled]
configured correctly.
category: EndpointSecurity
version: "v1"
comments: Resources like virtual machines often offer a
simple configuration that automatically installs updates,
e.g. critical security updates.
configuration:
  p1:
    operator: "=="
    targetValue: True
```



Assumption: Existing ontology
A set of classes $\mathcal{C}$, a set of relation types $\mathcal{R}$, and a set of data properties (attributes) $\mathcal{A}$, with a class hierarchy, typed class relations and class-attribute association.

Approach
We develop unsupervised heuristics that can incorporate additional textual context using pretrained language models (BERT, Sentence Transformers).

Usecase: CertGraph ontology
Automated assessment of certification-relevant security metrics [1].
Associated with a catalogue of security metrics [2].

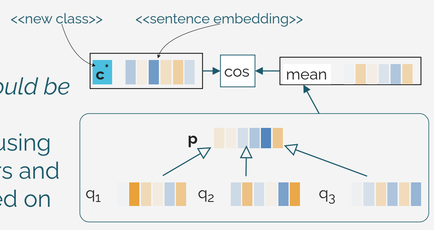
Task: using the information in the metric file, extend the class hierarchy, relations between classes, and the properties associated with them.

Tasks and Methods

TASK 1 — PARENT CLASS PREDICTION

Given a new concept name, rank all existing classes $p \in \mathcal{C}$ by their suitability as parent (c, is-a, p) in the hierarchy.

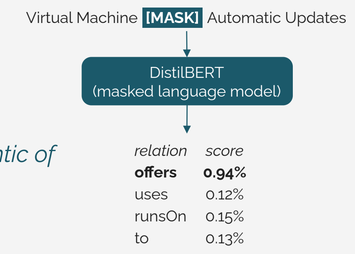
Ad-hoc algorithm: ChildAgg
Motivation: a node should be similar to its siblings
Embed class names using sentence transformers and aggregate. Rank based on cosine similarity.



TASK 2 — RELATION TYPE PREDICTION

Given a query of two classes $c_i, c_j \in \mathcal{C}$, rank all candidate relation types $r \in \mathcal{R}$ by the plausibility of the triple (c_i, r, c_j).

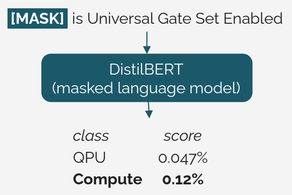
Ad-hoc algorithm: MLM prompting
Motivation: harness the semantic of concept names.



TASK 3 — DATA PROPERTY ASSIGNMENT

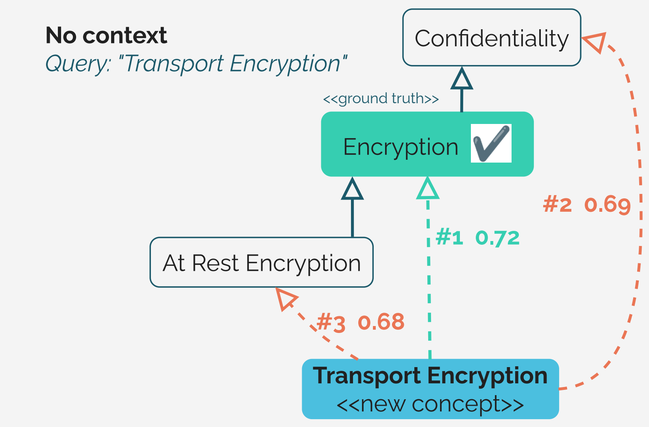
Given a new data property $a \in \mathcal{A}$ and a set of candidate owner classes $\{c_1, \ldots, c_m\} \subset \mathcal{C}$, predict which class c^* from the candidates should own data property a.

Ad-hoc algorithm: MLM prompting
Motivation: (also) harness the semantic of concept names.

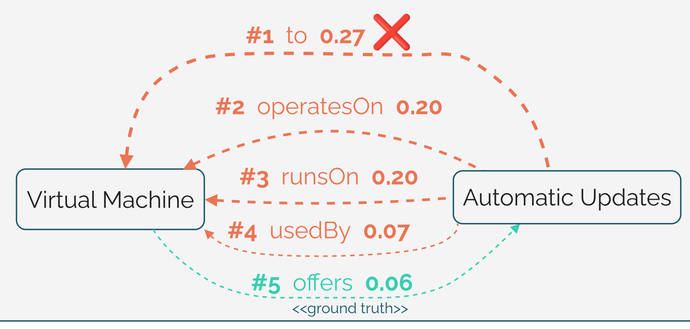


No Context

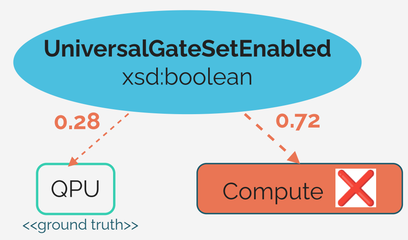
No context
Query: "Transport Encryption"



No context
MLM Prompt: "Virtual Machine [MASK] Automatic Updates"
or: "Automatic Updates [MASK] Virtual Machine"



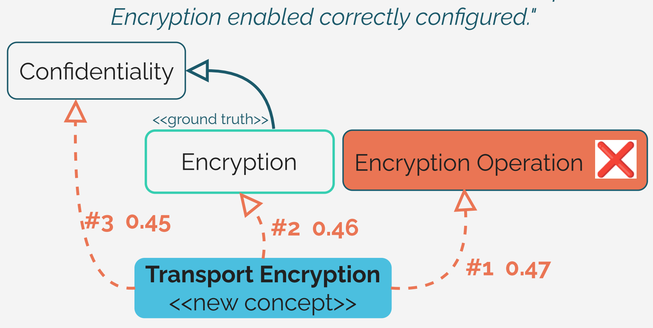
No context
MLM Prompt: "[MASK] is Universal Gate Set Enabled"



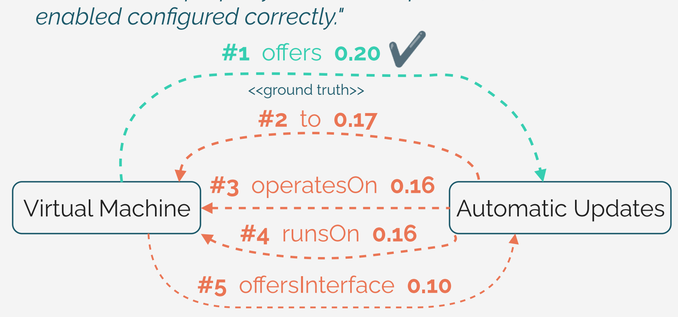
Context Enrichment

<<Example: metric description context>>

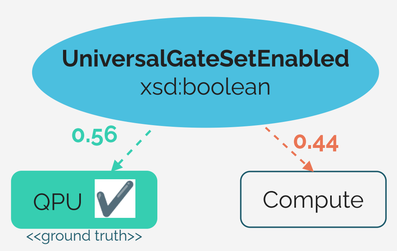
+ metric description
Query: "Transport Encryption This rule assesses whether a Resource has its Transport Encryption enabled correctly configured."



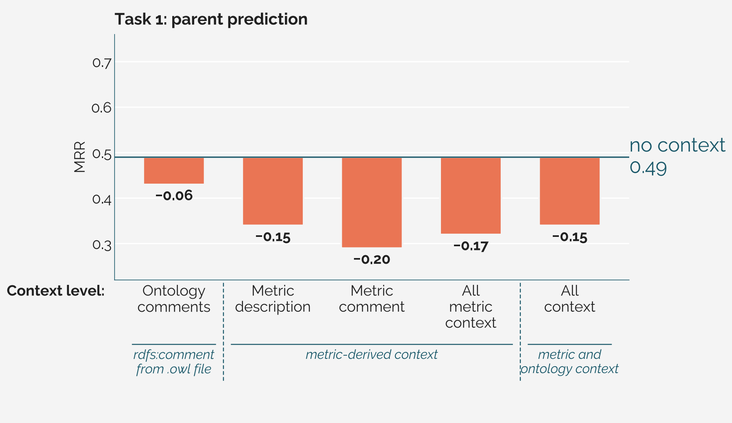
+ metric description
"This rule assesses whether a Virtual Machine that offers the property Automatic Updates, has enabled configured correctly."

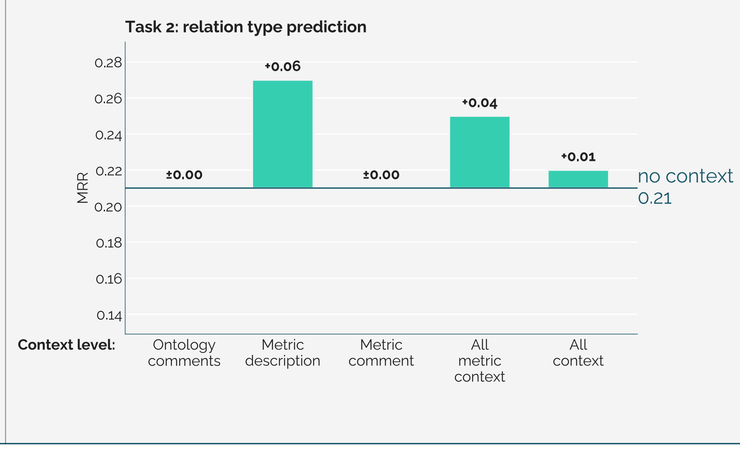


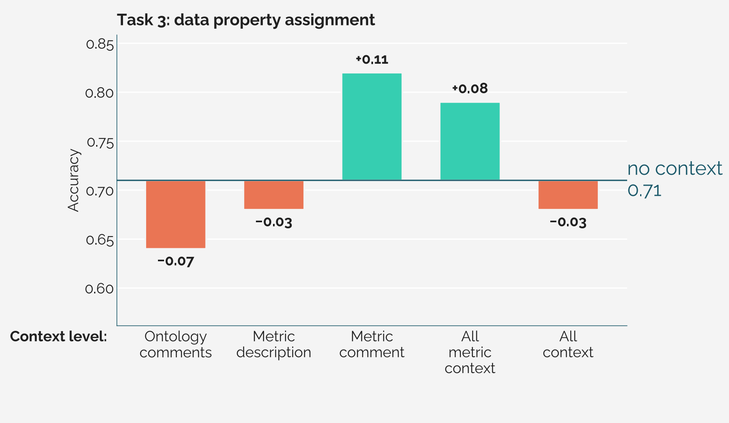
+ metric description
Prompt: "This rule assesses whether a QPU has Universal Gate Set Enabled correctly configured. [MASK] is Universal Gate Set Enabled"



COntExt Gain







Conclusions

Results suggest that metric-derived context significantly improves relation type prediction and data property assignment over prediction without context.
Parent class prediction requires a focused investigation to understand how to improve it.
Practical implications: Organizations maintain operational metric catalogues (YAML, JSON, RDF) - especially cybersecurity-related. COntExt makes these immediately reusable as ontology-extension inputs with no additional annotation effort.

References

[1] Schöberl, Stefan, et al. "Certgraph: Towards a comprehensive knowledge graph for cloud security certifications." Proceedings of the ACM/IEEE 27th International Conference on Model Driven Engineering Languages and Systems. 2024.

[2] Banse, Christian, et al. "A semantic evidence-based approach to continuous cloud service certification." Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing. 2023.

Code on Github

EMERALD · Know Center · scch · Fraunhofer AISEC

Funded by the European Union