On the Way to Diverse Datasets for Evaluating ABox Abduction Algorithms (Extended Abstract)

Janka Boborová, Jakub Kloc, Martin Homola and Júlia Pukancová

Comenius University in Bratislava, Mlynská dolina, 842 41 Bratislava, Slovakia

Abstract

We propose a method for generating evaluation datasets for ABox abduction algorithms, using diverse real-world knowledge bases, logical consequences as observations to ensure meaningfulness, justifications to guarantee explanations exist, and ontology modules to constrain the search space.

Keywords

Description logics, ABox abduction, evaluation dataset, test case

1. Introduction

Abduction [1, 2] is a form of inference that explains an observation by identifying its possible causes (explanations). We focus specifically on ABox abduction, where both the observation and its explanations consist of ABox assertions. To enable meaningful comparison and evaluation of ABox abduction algorithms, a suitable dataset of ABox abduction problems is needed. Such a dataset should include multiple real-world knowledge bases, each with meaningful observations. For each observation, it should provide abducibles (a set representing the search space) of varying sizes. Additionally, the ABox abduction problems should vary in the number and length of explanations. Datasets in existing evaluations suffer from several limitations, including use of only one knowledge base [3]; artificial automatically generated observations [3, 4, 5]; observations that may have no explanations [4]; limited diversity in explanation length and number [3]; and weak constraints on the search space [6, 7, 3, 4, 5]. Building on prior approaches, we have begun constructing an evaluation dataset that addresses these shortcomings.

2. Construction of a Robust Evaluation Dataset

To generate meaningful, non-artificial observations and reduce the search space without losing explanations, we propose two methods: one for generating observations (applicable to any knowledge base) and another for generating abducibles (applicable to any ABox abduction problem). For simplicity, the methods are defined for atomic concept assertions but extend easily

DL 2025: 38th International Workshop on Description Logics, September 3–6, 2025, Opole, Poland

boborova@fmph.uniba.sk (J. Boborová); kloc4@uniba.sk (J. Kloc); homola@fmph.uniba.sk (M. Homola); pukancova@fmph.uniba.sk (J. Pukancová)

https://dai.fmph.uniba.sk/~homola/ (M. Homola); https://dai.fmph.uniba.sk/~pukancova/ (J. Pukancová)

0009-0002-7487-9962 (J. Boborová); 0000-0001-6384-9771 (M. Homola); 0009-0001-3505-0716 (J. Pukancová)

to all atomic assertions and their complements. Applying these methods to diverse real-world knowledge bases enables the construction of a robust evaluation dataset.

Real-World Knowledge Bases: Koopmann et al. [4] proposed to use the *2015 OWL Reasoner Competition Corpus*[1][8], providing 1,920 diverse real-world ontologies, as a suitable set of knowledge bases for the evaluation of abduction algorithms. To focus only on relevant ontologies with the potential to produce interesting problems, we defined the following requirements: consistency; consistency check time $\leq$ 30 seconds; individual count $\geq$ 1. After applying these requirements, we obtained 865 ontologies as candidate knowledge bases.

Consequences as Observations: Although many knowledge bases are available, we are not aware of real-world use cases with predefined observations; therefore, observations must be generated separately. We aim to generate meaningful observations $\mathcal{O}$ by selecting *logical consequences* of a knowledge base $\mathcal{K}$. To ensure explanatoriness ($\mathcal{K} \not\models \mathcal{O}$), each $\mathcal{K}$ must be modified to no longer entail observation $\mathcal{O}$. This is done by removing at least one assertion from each *justification* of $\mathcal{O}$, i.e., from a minimal set of axioms responsible for the entailment of $\mathcal{O}$ [9]. Our approach is described in Algorithm 1.

The core idea is to "corrupt" $\mathcal{K}$ by removing assertions that can later be recovered as explanations through ABox abduction. As ABox abduction yields only ABox assertions, other axiom types cannot be removed during the modification of $\mathcal{K}$.

Algorithm 1 Generating ABox Abduction Problems

Input: knowledge base $\mathcal{K}$
Output: a set of ABox abduction problems $\mathcal{P}_s$

$\mathcal{P}_s \leftarrow \{\}$
$concequences \leftarrow \{A(i) \mid i \in N_I, A \in N_C, \mathcal{K} \models A(i), A(i) \notin \mathcal{K}\}$ ▷ generate observations
for $A(i)$ in $concequences$ **do** ▷ compute observation justifications
    $J(A(i)) \leftarrow$ get the ABox parts of justifications for $A(i)$ using `OWLExplanation`
end for
for $A(i)$ in $concequences$ **do** ▷ generate ABox abduction problems
    $n \leftarrow$ get the size of the *largest* justification in $J(A(i))$
    for x in $\langle 1, n \rangle$ **do**
        $\mathcal{K}_x \leftarrow \mathcal{K}$
        for $just$ in $J(A(i))$ **do**
            $toDelete \leftarrow$ randomly select $min(x, |just|)$ assertions from $just$
            $\mathcal{K}_x \leftarrow \mathcal{K}_x \setminus toDelete$ ▷ modify $\mathcal{K}$
        end for
        $\mathcal{P}_s \leftarrow \mathcal{P}_s \cup \{\mathcal{P} = (\mathcal{K}_x, A(i))\}$
    end for
end for
return $\mathcal{P}_s$

To explore a wider range of possibilities, we generated multiple modified knowledge bases for each observation by progressively removing more assertions, aiming to produce different explanations for the same observation.

Module-Based Abducibles: During evaluation, it is useful to examine how algorithms perform with search spaces of varying sizes. However, reducing the search space requires a

[1] https://zenodo.org/record/18578#.Y3tygXbMJPb

careful strategy to preserve explanations.

We propose using *module extraction* [10, 11], a technique that extracts a meaningful fragment of an ontology while preserving all axioms relevant to the complete meaning of a given signature. This technique can be used to generate *module abducibles*, i.e., assertions relevant to an observation $\mathcal{O}$, excluding symbols unlikely to appear in explanations: $\mathrm{Abd}_{\text{module}} = \{A(i) \mid i \in N_I, A \in N_C$ from $\Sigma(module)\}$. Specifically, the $\top$-module, which includes all subclasses of the atomic classes in the signature of $\mathcal{O}$, as explanations typically involve concepts subsumed by the concept in $\mathcal{O}$.

To differentiate within $\mathrm{Abd}_{\text{module}}$ when generating abducibles of a given size, we prioritise assertions involving individuals from $\mathcal{O}$, as they are more likely to appear in explanations.

3. Analysis of Generated Inputs

Generating ABox Abduction Problems: The observation generation process applied to 865 knowledge bases resulted in 37,042 ABox abduction problems. The largest ABox part of a justification contained 12 assertions, and the maximum number of justifications for an observation was 38 (all with a single-assertion ABox part). Over 90% of observations had one justification with a single-element ABox part.

In theory, more justifications should lead to more explanations, and justifications with more assertions should lead to longer explanations. In practice, justifications may contain complex assertions that cannot be reconstructed by algorithms limited to atomic assertions and their complements. Additionally, observations may have explanations beyond those found in the justifications. Therefore, given the large number of generated problems, identifying those with interesting properties is challenging without additional information.

Generating Abducibles: To analyse abducible generation, we selected a sample of ABox abduction problems (Table 1) by applying MergeXplain (MXP)[2] [13] with $\mathrm{Abd}_{\text{default}} = \{A(i) \mid A \in N_C, i \in N_I$ from $\Sigma(\mathcal{K} \cup \mathcal{O})\}$ to a random subset of the generated problems. MergeXplain returns a set of explanations, $S_{\mathcal{E}}$, containing all explanations of length 1 and, if present, at least one additional explanation of a greater length. We selected the final sample based on the number and length of explanations. Notably, only one problem (ont934_obs01) in the subset produced explanations of varying lengths, including some longer than one.

For each problem in the sample, we generated module abducibles, Abd_{module}, which on average reduced the search space to 46%. Abd_{module} consistently included all explanations found by MergeXplain, ensuring none were lost.

To generate abducibles of varying sizes, three methods were applied: (a) *module abducibles prioritising assertions with individuals from the observation* (our proposed approach), (b) *module abducibles without prioritisation*, and (c) *completely random selection*. Each method was used to generate abducibles of sizes 10, 25, 50, 100, 250, and 500, and was run three times per size to obtain averaged results. For each method and size, we report the percentage of explanation assertions covered by the generated Abd sets, relative to their size, computed as $\frac{|\texttt{expl. assertions in Abd}|}{\min(|\mathrm{Abd}|, |\texttt{expl. assertions}|)}$ (e.g., a set of size 10 can cover at most 10 explanation assertions, and

[2]MXP was run using CATS [12]: https://github.com/Comenius-Abduction-Team/CATS-Abduction-Solver

Table 1
ABox Abduction Problem Test Sample

ABox abd. problems	$\lvert\mathrm{Abd}_{\text{default}}\rvert$	$\lvert\mathrm{Abd}_{\text{module}}\rvert$	$\lvert S_{\mathcal{E}}\rvert$	(length $n : \lvert\{\mathcal{E} \mid \mathcal{E} \in S_{\mathcal{E}}, \lvert\mathcal{E}\rvert = n\}\rvert$)
ont155_obs46	31,078	6,396	58	(1:58)
ont155_obs167	31,078	20,090	192	(1:192)
ont394_obs27	3,760	2,544	90	(1:90)
ont568_obs61	40,132	8,509	17	(1:17)
ont934_obs01	16,023	15 876	4	(1:1), (3:2), (4:1)
ont1117_obs45	599,844	36	5	(1:5)

Table 2
Explanation Coverage for Different Abducibles Generation Methods

Generation method	$\lvert\mathrm{Abd}\rvert$ 10	25	50	100	250	500
(a)	55%	58%	67%	79%	88%	88%
(b)	4%	12%	18%	18%	20%	23%
(c)	1%	1%	0%	1%	2%	4%

the maximum possible coverage is bounded by the total number of explanation assertions). The results (Table 2) were averaged over all problems and runs. Method (a) was the most successful, consistently generating sets that covered the highest number of explanation assertions across all sizes. At sizes 250 and 500, it achieved full coverage for all problems except ont934_obs01, where it generated no more than two explanation assertions per Abd set. Still, even on this problem, it outperformed the other methods, which on average generated none. Out of 108 runs, the generated set contained no explanation assertions in 5 cases for method (a) (all for ont934_obs01), 57 for method (b), and 78 for method (c).

4. Discussion and Outlook

The dataset generation process needs refinement, especially in producing ABox abduction problems. To narrow down the generated problems and focus on the most relevant ones, we plan to analyse the ABox parts of justifications. Since many observations produce only single-assertion explanations, we aim to use observations composed of multiple assertions.

In contrast, for generating abducibles, the module-based approach prioritising individuals from the observation seems promising.

Acknowledgments

This research was sponsored by the Slovak Republic under the SRDA grant APVV-23-0292 (DyMAX) and VEGA grant no. 1/0630/25 (eXSec), and it is also based on the work from COST Action CA23147 GOBLIN. J. Boborová was supported by Grant no. UK/1283/2025 awarded by Comenius University in Bratislava.

References

[1] C. Elsenbroich, O. Kutz, U. Sattler, A case for abductive reasoning over ontologies, in: Proceedings of the OWLED*06 Workshop on OWL: Experiences and Directions, Athens, GA, US, volume 216 of *CEUR-WS*, CEUR-WS.org, 2006. URL: https://ceur-ws.org/Vol-216/submission_25.pdf.

[2] C. S. Peirce, Illustrations of the logic of science VI: Deduction, induction, and hypothesis, Popular Science Monthly 13 (1878) 470–482.

[3] M. Homola, J. Pukancová, I. Balintová, J. Boborová, Hybrid MHS-MXP ABox abduction solver: First empirical results, in: Proceedings of the 35th International Workshop on Description Logics (DL 2022) co-located with Federated Logic Conference (FLoC 2022), Haifa, Israel, August 7th to 10th, 2022, volume 3263 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022. URL: https://ceur-ws.org/Vol-3263/paper-13.pdf.

[4] P. Koopmann, W. Del-Pinto, S. Tourret, R. A. Schmidt, Signature-based abduction for expressive description logics, in: Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, 2020, pp. 592–602. URL: https://doi.org/10.24963/kr.2020/59. doi:10.24963/KR.2020/59.

[5] J. Du, K. Wang, Y. Shen, A tractable approach to ABox abduction over description logic ontologies, in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada, AAAI Press, 2014, pp. 1034–1040. URL: https://doi.org/10.1609/aaai.v28i1.8852. doi:10.1609/AAAI.V28I1.8852.

[6] J. Pukancová, M. Homola, ABox abduction for description logics: The case of multiple observations, in: Proceedings of the 31st International Workshop on Description Logics, Tempe, Arizona, US, volume 2211 of *CEUR-WS*, CEUR-WS.org, 2018. URL: https://ceur-ws.org/Vol-2211/paper-31.pdf.

[7] D. Mrózek, J. Pukancová, M. Homola, ABox abduction solver exploiting multiple DL reasoners, in: Proceedings of the 31st International Workshop on Description Logics co-located with 16th International Conference on Principles of Knowledge Representation and Reasoning (KR 2018), Tempe, Arizona, US, October 27th - to - 29th, 2018, volume 2211 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2018. URL: https://ceur-ws.org/Vol-2211/paper-24.pdf.

[8] B. Parsia, N. Matentzoglu, R. S. Gonçalves, B. Glimm, A. Steigmiller, The OWL reasoner evaluation (ORE) 2015 competition report, J. Autom. Reason. 59 (2017) 455–482. URL: https://doi.org/10.1007/s10817-017-9406-8. doi:10.1007/S10817-017-9406-8.

[9] A. Kalyanpur, B. Parsia, M. Horridge, E. Sirin, Finding all justifications of OWL DL entailments, in: The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea, November 11-15, 2007, volume 4825 of *Lecture Notes in Computer Science*, Springer, 2007, pp. 267–280. URL: https://doi.org/10.1007/978-3-540-76298-0_20. doi:10.1007/978-3-540-76298-0_20.

[10] C. D. Vescovo, D. Gessler, P. Klinov, B. Parsia, U. Sattler, T. Schneider, A. Winget, Decomposition and modular structure of bioportal ontologies, in: The Semantic Web - ISWC 2011 - 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I, volume 7031 of *Lecture Notes in Computer Science*, Springer, 2011, pp. 130–145. URL: https://doi.org/10.1007/978-3-642-25073-6_9.

[11] B. C. Grau, I. Horrocks, Y. Kazakov, U. Sattler, Just the right amount: Extracting modules from ontologies, in: Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007, ACM, 2007, pp. 717–726. URL: https://doi.org/10.1145/1242572.1242669. doi:10.1145/1242572.1242669.

[12] J. Kloc, J. Boborová, M. Homola, J. Pukancová, CATS solver: The rise of hybrid abduction algorithms, in: Proceedings of the 38th International Workshop on Description Logics (DL 2025), Opole, Poland, September 3–6, 2025, 2025. To appear.

[13] K. M. Shchekotykhin, D. Jannach, T. Schmitz, MergeXplain: Fast computation of multiple conflicts for diagnosis, in: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015, AAAI Press, 2015, pp. 3221–3228. URL: http://ijcai.org/Abstract/15/454.