RelVis: Benchmarking OpenIE Systems

Rudolf Schneider, Tom Oberhauser, Tobias Klatt, Felix A. Gers, and Alexander Löser

Beuth University of Applied Sciences, Berlin, Germany
{ruschneider, toberhauser, tklatt, gers, aloeser} @beuth-hochschule.de

Abstract. We demonstrate RelVis, a toolkit for benchmarking Open Information Extraction(OIE) systems. RelVis enables the user to perform a comparative analysis among OIE systems like ClausIE, OpenIE 4.2, Stanford OpenIE or PredPatt. It features an intuitive dashboard that enables a user to explore annotations created by OIE systems and evaluate the impact of five common error classes. Our comprehensive benchmark contains four data sets with overall 4522 labeled sentences and 11243 binary or n-ary OIE relations.

1 Introduction

Open Information Extraction (OIE) is an important intermediate step for many text mining tasks, such as summarization, relation extraction or knowledge base construction [4][9]. OIE systems are designed for extracting n-ary tuples from diverse and large amounts of text as found in the web and without being restricted to a fixed schema.

Often users desire to select a OIE system suitable for their specific application domain. Unfortunately, there is surprisingly little work on evaluating and comparing results among different OIE systems. Worse, most OIE methods utilize proprietary and unpublished data sets.

Demonstration and contribution. Ideally, one could compare different OIE systems with a unified benchmarking suite. As a result, the user could identify "sweet spots" of each system but also weaknesses for common error classes. The benchmarking suite should feature a diverse set of gold annotations with several thousands of annotated sentences. By exploring results and errors in dashboards, the user can learn how to design the next generation of OIE systems.

We demonstrate RelVis[1], a web based open source OIE benchmarking suite, which fulfills these requirements, see also our work in [7]. Our contributions are: (1) We initially support four commonly used OIE systems. In addition, we permit the users to benchmark additional OIE systems via standardized interfaces. (2) We provide an integrated benchmark for OIE systems consisting of three news data sets and a large OIE Benchmark from Newswire and Wikipedia. Overall, our benchmark includes *4522* sentences and *11243* n-ary tuples. (3) Our system permits an in-depth analysis on five different error classes, different matching strategies and standard measures, such as f-measure, precision or recall.

[1] video demonstration: https://www.youtube.com/watch?v=Hs87hIe-HEs

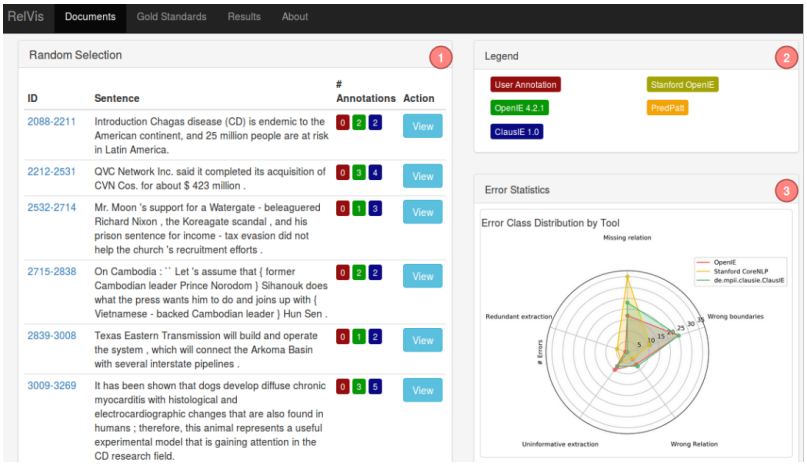

Fig. 1. Sentence selection view of RelVis. (1) For each sentence in the document we show text and number of extractions by system. (2) Denotes various OIE systems with different colors. (3) The lower right hand side visualizes error evaluation statistics.

2 Demo Walkthrough and Exploration

Startup. On system initialization, RelVis reads gold-annotations and performs a quantitative evaluation. Next, the system stores extraction- and gold annotations in a RDBMS.

Dashboards for exploring annotations. Now, the user can start exploring results and understanding the behaviour of each system. Figure 1 visualizes in a web-based dashboard sentences, precision, recall and F scores for each OIE system and for each error class. RelVis plots error distributions as a Kiviat diagram and draws bar charts for comparing error class impacts for each OIE system. In addition, the user can export results as tables and CSV files from the database.

Understanding and adding a single annotation. RelVis visualizes OIE extractions on sentence level. For each hit by a system, the user can drill down into a single sentence and can understand extraction predicates, in green, or arguments, in blue color, as shown on Figure 2.

Next, she can dive down into correct or incorrect annotations, can add labels for error classes of incorrect annotations or may leave a comment, see also Figure 2. We permit the user to apply multiple error classes to each subpart of an annotation. Next, she can focus on a sentence of interest and can compare extractions between different OIE systems. If no gold annotations are available the user can create them using RelVis. Note that such a process is also feasible with standard annotation tools, such as BRAT [10]. However in practice, we noted that such standard tools require a lot of configuration steps to adapt to OIE-relations. The user selects a sentence to annotate and starts with the first annotation by clicking on the "Add new OIE Relation" button. Next, she marks

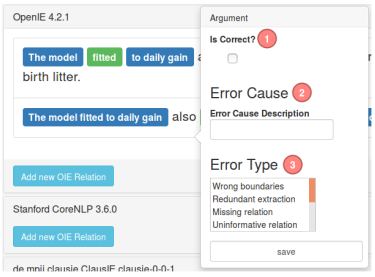

Fig. 2. Specifying correctness (1), error (3) and commenting on an cause (2).

Name	Type	Domain	Sent.	# Tuple
NYT-nary	n-ary	News	222	222
WEB-500	binary	Web	500	461
PENN-100	binary	Mixed	100	51
OIE2016	n-ary	Wiki	3200	10359

Table 1. Data sets in RelVis

the predicate and arguments in the sentence for her first annotation by selecting them with the cursor.

Predefined common error classes. Over the years, different error classes have been defined for evaluating OIE systems. We identified the following five types of errors as most relevant in our previous work [7]. *(1) Wrong Boundaries* indicate too large or too small boundaries for an argument or predicate of an OIE extraction. A downstream application has to filter out or correct incorrect boundaries which may cause a drastic recall loss. *(2) Redundant Extraction* appear if the OIE system does not filter out these tuples. *(3) Uninformative Extraction* are tuples without any reasonable value. This error type causes additional processing effort without delivering any value. *(4)Missing Extraction* describes relations which were not found by a system. *(5)Wrong Extraction* are tuples emiting a wrong information. It is not possible to recover from a error of this class and it emits a wrong signal.

3 System Design

RelVis currently supports the following **OIE Systems**: StanfordOpenIE [1], OpenIE 4.2 [2,5], ClausIE [3] and PredPat [11]. To add a new OIE system, the user can either implement a Java interface or upload results in RelVis' data format. The system is compatible with four **datasets**, see Table 1, of which two feature only binary relations with two arguments. Data sets *NYT-nary* and *OIE2016* also contain n-ary relations. These labeled data sets origin from [6] and [8]. RelVis supports *equal matching* of boundaries in text to a gold standard. This matching strategy delivers exact results for computing precision. However, this strategy penalizes other, potentially correct, boundary definitions beyond the gold standard. Dealing with multiple OIE systems and their different annotation styles requires a less restrictive matching strategy. As second strategy we focus on a *containment match*. Here an argument or predicate is considered correct if it at least contains a gold standard annotation, hence spans from the gold standard may be contained (fully) inside the spans of the annotation from the OIE system. However, this strategy may label over-specific tuples as correct

and may lead to a lower precision. A containment strategy still penalizes binary systems on n-ary data sets. Therefore we introduce as third strategy a *relaxed containment strategy* which removes a penalty for wrong boundaries especially for over specific extractions. This strategy counts an extraction correct even when the number of arguments doesn't match the gold standard.

4 Conclusion

To our best knowledge, RelVis is the first attempt integrating four different OIE systems and four different data sets in a single comprehensive benchmark system for OIE systems. It provides dashboards for in-depth qualitative evaluations, classifies errors in five common expendable classes and supports user defined annotations or data sets. In our future work we will obtain output compatibility with BRAT annotations [10]. RelVis enables the community exploring existing and adding home grown OIE systems and is available as open source at *https://github.com/SchmaR/RelVis*.

Acknowledgement. Our work is funded by the German Federal Ministry of Economic Affairs and Energy (BMWi) under grant agreement 01MD16011E (Medical Allround-Care Service Solutions), grant agreement 01MD15010B (Smart Data Web) and H2020 ICT-2016-1 grant agreement 732328 (FashionBrain).

References

1. Angeli, G., Premkumar, M.J.J., Manning, C.D.: Leveraging linguistic structure for open domain information extraction. In: ACL. pp. 344–354 (2015)
2. Christensen, J., Soderland, S., Etzioni, O., others: An analysis of open information extraction based on semantic role labeling. In: K-CAP. pp. 113–120
3. Corro, L.D., Gemulla, R.: Clausie: clause-based open information extraction. In: WWW. pp. 355–366 (2013)
4. Mausam: Open information extraction systems and downstream applications. In: IJCAI. pp. 4074–4077 (2016)
5. Pal, H., Mausam: Demonyms and compound relational nouns in nominal open IE. In: AKBC at NAACL
6. de Sá Mesquita, F., Schmidek, J., Barbosa, D.: Effectiveness and efficiency of open relation extraction. In: EMNLP. pp. 447–457 (2013)
7. Schneider, R., Oberhauser, T., Klatt, T., Gers, A.F., Löser, A.: Analysing errors of open information extraction systems. In: BLGNLP at EMNLP (2017)
8. Stanovsky, G., Dagan, I.: Creating a large benchmark for open information extraction. In: EMNLP, 2016. pp. 2300–2305 (2016)
9. Stanovsky, G., Dagan, I., Mausam: Open IE as an intermediate structure for semantic tasks. In: ACL. pp. 303–308 (2015)
10. Stenetorp, P., Pyysalo, S., Topic, G., Ohta, T., Ananiadou, S., Tsujii, J.: brat: a web-based tool for nlp-assisted text annotation. In: EACL. pp. 102–107 (2012)
11. White, A.S., Reisinger, D., Sakaguchi, K., Vieira, T., Zhang, S., Rudinger, R., Rawlins, K., Durme, B.V.: Universal decompositional semantics on universal dependencies. In: EMNLP. pp. 1713–1723 (2016)