Update-Aware Information Extraction

Besat Kassaie[*]

David R. Cheriton School of Computer Science
University of Waterloo
bkassaie@uwaterloo.ca

Abstract

Information extraction programs (extractors) can be applied to documents to isolate structured versions of some content by creating tabular records corresponding to facts found in the documents. Most optimization techniques deployed in information extraction systems assume that source documents are static. Instead, extracted relations can be considered to be materialized views defined by a language built on regular expressions. Using this perspective, we provide an efficient verifier that can be used to avoid the high cost of re-extracting information after a batch update. In particular, we propose an efficient mechanism to identify updates for which we can autonomously compute an extracted relation. We present experimental results that support the practicality of this mechanism in real world extraction systems.

1 Introduction

When extracted relations or source documents are updated, we wish to ensure that those changes are propagated correctly. That is, we recommend that extracted relations be treated as materialized views over the document database. Within this context, we tackle two research challenges; I) Because extraction is prohibitively expensive, efficiently maintaining extracted relations up-to-date is crucial [5, 4]. II) To maintain system consistency, it is essential to translate updates on extracted views into corresponding document updates [3].

In this talk, I start by exploring update-aware information extraction, shedding light on the aforementioned critical issues that arise when dealing with updates. Next, I delve into our research on autonomously computable information extraction. Additionally, I highlight key findings from our experimental results for this problem, demonstrating whether our approach can be used effectively in realistic update and extraction scenarios.

2 Autonomously Computable Information Extraction

We apply static analysis to programs that specify extractors and updates in order to determine whether re-extraction can be avoided or reduced. Given a program defined as a document spanner [2] and an update specification, we determine sufficient conditions for autonomously re-computing extracted spans of an updated document. In particular, we propose three sufficient conditions for updates with respect to an extraction program. We prove that we require time and space that are polynomial in the size of the extraction program and the update specification to perform five required tests to determine that the revised extracted relation can be computed autonomously. Finally, we describe experiments with realistic extractors conducted on two real-world datasets to conclude that the runtime overhead imposed by our verification is small in practice when compared to re-evaluating extractors, even if the re-evaluation is performed incrementally [1].

References

[1] Fei Chen, AnHai Doan, Jun Yang, and Raghu Ramakrishnan. Efficient information extraction over evolving text data. In *Proc. 24th ICDE*, pages 943–952. IEEE Computer Society, 2008.

[2] Ronald Fagin, Benny Kimelfeld, Frederick Reiss, and Stijn Vansummeren. Document spanners: A formal approach to information extraction. *J. ACM*, 62(2):12:1–12:51, 2015.

[3] Besat Kassaie and Frank Wm. Tompa. Predictable and consistent information extraction. In *Proc. DocEng '19: ACM*, pages 14:1–14:10. ACM, 2019.

[4] Besat Kassaie and Frank Wm. Tompa. A framework for extracted view maintenance. In *Proc. DocEng '20: ACM*, pages 16:1–16:4. ACM, September 2020.

[5] Besat Kassaie and Frank Wm. Tompa. Autonomously computable information extraction. *Proc. VLDB Endow.*, 16(10):2431–2443, 2023.

[*] This talk is based on joint work with Frank Wm. Tompa: fwtompa@uwaterloo.ca