The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Constructing Domain-Specific Search Engines with No Programming

Mayank Kejriwal, Pedro Szekely
Information Sciences Institute
USC Viterbi School of Engineering
4676 Admiralty Way 1001
Marina Del Rey, California 90292
{kejriwal,pszekely}@isi.edu

Abstract

We propose a demonstration of myDIG (my Domain-specific Insight Graphs), a system that allows non-technical domain experts, including those with no programming experience, to construct a domain-specific search engine over a raw corpus of webpages. myDIG has been developed and refined over multiple years under the DARPA MEMEX program, and has undergone rigorous user testing with actual domain experts from investigative agencies like the Securities and Exchange Commission (SEC). All components of myDIG are open-source, and the product of fundamental research.

Introduction

As engineering complex systems, especially those based on machine learning, becomes ever more complicated, there is a need to build interactive systems with powerful capabilities that can be accessed and used by non-technical domain experts. Such capabilities are especially useful on crawled Web data, since many interesting phenomena worthy of social or investigative interest (like fraud), have a significant Web presence. We propose a demonstration of myDIG, a system that ingests a corpus of webpages stored in a distributed file system, and allows a user to construct a domain-specific knowledge graph and search engine without any programming. A high-level architectural description of myDIG is provided in the next section, along with details on the actual demonstration. We provide screenshots and visualizations of myDIG in the slides accompanying this demonstration proposal. myDIG is currently in a relatively mature stage, having already been evaluated by DARPA in multiple domains with enormous investigative potential, including securities and penny stock fraud, illegal weapons sales, counterfeit electronics, narcotics and mail shipment fraud.

Significance. To the best of our knowledge, myDIG is among the first systems (and the first open-source system[1]) to allow non-programmers to set up, and search, a domain expressed in raw, heterogeneous Web documents using an interactive interface. Attendees of the demonstration will be able to explore this capability first-hand. myDIG contains state-of-the-art interacting components (see Related Work) that were developed over years of research, and is already being transitioned to combat important problems like fraud and human trafficking.

High-level Architecture

At a high level, the myDIG architecture operates in two high-level phases. In the first *domain setup* phase, users construct and iteratively refine a knowledge graph and vocabulary from the raw corpus of webpages. Users can try multiple options on a sampled subset of the corpus, and decide where they want to invest more effort. Users can also customize the appearance (icons, colors) and algorithmic elements of the search GUI without actual programming, including deciding aspects such as the importance of an attribute for search, and whether a given attribute is textual or entity-centric. Users can also input their glossaries to seed knowledge graph construction for certain attributes. For example, one could input a glossary of stock ticker symbols to seed the extractions for an attribute ‘Stock Tickers’.

In the second *domain exploration* phase, domain experts use the search engine for gaining further insight into domain properties and characteristics, and in the case of investigative domains, both generating and investigating leads. Search in myDIG can be both fine-grained and coarse-grained, depending on user preferences. For example, myDIG supports basic keyword search, but also supports filtering on fields and facets, and ‘form-based’ querying. Many of our users tend to start their explorations with keyword search, followed by more sophisticated explorations once they have located an item of interest. In some cases, however, users already have a ‘lead’ and commence search with the sophisticated options available to them. myDIG provides users with options to guide the search in customized ways.

Demo

In the exhibit, we will take our users through important, representative steps in a workflow that will involve constructing a domain and a domain-specific search engine in a matter of minutes on a small, but interesting (and real-world) corpus of webpages describing penny stocks. Users will also be given a chance to explore the search capabilities of myDIG on a much larger dataset that was crawled over the Open Web and has been evaluated by the SEC for its potential in pinpointing evidence of securities fraud.

[1] https://github.com/usc-isi-i2/dig-etl-engine

Related Work

The myDIG system depends on a number of advances in both knowledge graph construction and information retrieval. Rather than provide an exhaustive review herein, we synthesize the primary trends that have proved influential.

Knowledge Graph Construction (KGC). KGC is an umbrella term that primarily includes information extraction (IE), but can also include post-extraction steps that try to improve the quality of the data in a semi-automatic fashion (Niu et al. 2012a), (Craven et al. 2000). IE has been surveyed by several authors; see, for example, (Sarawagi and others 2008), (Aggarwal and Zhai 2012). A particular category of IE that is relevant to myDIG is Web IE (Chang et al. 2006). We note that, while myDIG uses a *set* of IE technologies, the user is not required to train a system or tune algorithmic parameters. In general, interactive KGC systems, even involving technical expertise, are still quite uncommon. A good exception is the DeepDive system (Niu et al. 2012b), which allows customized, interaction-driven information extraction. Another system, Snorkel, that relies on weak supervision, thereby easing the burden of acquiring and manually annotating large amounts of training data, requires users to code functions and like DeepDive, has no facilities for supporting search and analytics (Ratner et al. 2016).

Information Retrieval (IR). Search in myDIG draws upon several independently developed techniques in the IR community. Chief among these are *query reformulation* and *constraint relaxation* (Rieh and others 2006), (Viswanathan et al. 2017), (Muslea 2004), (Mirzadeh, Ricci, and Bansal 2004). These techniques are designed to be robust to erroneous or missing data, which is an unavoidable problem for semi-automatic KGC systems. Our search system uses advances in NoSQL databases like Elasticsearch, which use optimized inverted index querying for fast retrieval (Gormley and Tong 2015). NoSQL was surveyed by (Han et al. 2011). The myDIG system also supports faceting and filtering (Amitay et al. 2011), along with basic keyword search.

Acknowledgements

We gratefully acknowledge our collaborators and all (former and current) members of our team who contributed their efforts and expertise to DIG, particularly during the dry and final evaluation runs: Amandeep Singh, Linhong Zhu, Lingzhe Teng, Nimesh Jain, Rahul Kapoor, Muthu Rajendran R. Gurumoorthy, Sanjay Singh, Majid Ghasemi Gol, Brian Amanatullah, Craig Knoblock and Steve Minton. This research is supported by the Defense Advanced Research Projects Agency (DARPA) and the Air Force Research Laboratory (AFRL) under contract number FA8750- 14-C-0240. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA, AFRL, or the U.S. Government.

References

Aggarwal, C. C., and Zhai, C. 2012. *Mining text data*. Springer Science & Business Media.

Amitay, E.; Carmel, D.; Golbandi, N.; Har'el, N. Y.; Ofek-Koifman, S.; and Yogev, S. 2011. Information retrieval with unified search using multiple facets. US Patent 8,024,324.

Chang, C.-H.; Kayed, M.; Girgis, M. R.; and Shaalan, K. F. 2006. A survey of web information extraction systems. *IEEE transactions on knowledge and data engineering* 18(10):1411–1428.

Craven, M.; DiPasquo, D.; Freitag, D.; McCallum, A.; Mitchell, T.; Nigam, K.; and Slattery, S. 2000. Learning to construct knowledge bases from the world wide web. *Artificial intelligence* 118(1-2):69–113.

Gormley, C., and Tong, Z. 2015. *Elasticsearch: The Definitive Guide: A Distributed Real-Time Search and Analytics Engine*. " O'Reilly Media, Inc.".

Han, J.; Haihong, E.; Le, G.; and Du, J. 2011. Survey on nosql database. In *Pervasive computing and applications (ICPCA), 2011 6th international conference on*, 363–366. IEEE.

Mirzadeh, N.; Ricci, F.; and Bansal, M. 2004. Supporting user query relaxation in a recommender system. In *Ec-web*, 31–40. Springer.

Muslea, I. 2004. Machine learning for online query relaxation. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 246–255. ACM.

Niu, F.; Zhang, C.; Ré, C.; and Shavlik, J. 2012a. Elementary: Large-scale knowledge-base construction via machine learning and statistical inference. *International Journal on Semantic Web and Information Systems (IJSWIS)* 8(3):42–73.

Niu, F.; Zhang, C.; Ré, C.; and Shavlik, J. W. 2012b. Deepdive: Web-scale knowledge-base construction using statistical learning and inference. *VLDS* 12:25–28.

Ratner, A. J.; De Sa, C. M.; Wu, S.; Selsam, D.; and Ré, C. 2016. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems*, 3567–3575.

Rieh, S. Y., et al. 2006. Analysis of multiple query reformulations on the web: The interactive information retrieval context. *Information Processing & Management* 42(3):751–768.

Sarawagi, S., et al. 2008. Information extraction. *Foundations and Trends® in Databases* 1(3):261–377.

Viswanathan, A.; Michaelis, J. R.; Cassidy, T.; de Mel, G.; and Hendler, J. 2017. In context query reformulation for failing sparql queries. In *SPIE Defense+ Security*, 101900M–101900M. International Society for Optics and Photonics.