University of Waterloo Docker Images for OSIRRC at SIGIR 2019

Ryan Clancy, Zeynep Akkalyoncu Yilmaz, Ze Zhong Wu, and Jimmy Lin
David R. Cheriton School of Computer Science
University of Waterloo

1 OVERVIEW

The University of Waterloo team submitted a total of four Docker images to the Open-Source IR Replicability Challenge (OSIRRC) at SIGIR 2019. This short overview outlines the functionality of each image. As the READMEs in all our source repositories provide details on the technical design of our images and the retrieval models used in our runs, we intentionally do not duplicate this information here.

Our primary submission is a packaging of Anserini [11, 12], an open-source information retrieval toolkit built around Lucene to facilitate replicable research. This `anserini-docker` image resides at the following URL:

```
https://github.com/osirrc/anserini-docker
```

The Anserini project grew out of the Open-Source IR Reproducibility Challenge from 2015 [5] and reflects growing community interest in using Lucene for academic IR research [1, 2]. As Lucene was not originally designed as a research toolkit, Anserini aims to fill in the "missing parts" that allow researchers to run standard *ad hoc* retrieval experiments "right out of the box", including competitive baselines and integration hooks for neural ranking models. Given Lucene's tremendous production deployment base (typically via Solr or Elasticsearch), better alignment between research in information retrieval and the practice of building real world search engines promises a smoother transition path from the lab to the "real world" for research innovations.

In addition to our main Anserini image, we built two ancillary images for the OSIRRC exercise:

```
https://github.com/osirrc/solrini-docker
https://github.com/osirrc/elastirini-docker
```

In production environments, Lucene is most often used as a core search library that powers two widely-deployed "full stack" search applications: Solr and Elasticsearch. With "Solrini" and "Elastirini", we have integrated Anserini with Solr and Elasticsearch, respectively. The integration is such that we can use Anserini as a common frontend to index into a backend Solr or Elasticsearch instance. This allows unification of the document processing pipeline (tokenization, stemming, etc.) to support standard TREC *ad hoc* experiments, while allowing users to take advantage of the wealth of capabilities provided by Solr and Elasticsearch. In the case of Solr, users can interact with sophisticated searching and faceted browsing interfaces such as Project Blacklight[1] [10], as described in Clancy et al. [3]. In the case of Elasticsearch, we can gain access to the so-called ELK stack (Elasticsearch, Logstash, Kibana) to provide a complete data analytics environment, including slick visualization interfaces.

Solrini and Elastirini capabilities are exposed via the `interact` hook in the OSIRRC jig. Since both Solr and Elasticsearch are designed as web apps, the user can trigger the hook and then directly navigate to a URL to access system capabilities. The batch runs provided by the `solrini` and `elastirini` images are exactly the same as the `anserini` image.

The final image submitted by our group packages Birch, our newest open-source search engine[2] that takes advantage of BERT [4] for *ad hoc* document retrieval:

```
https://github.com/osirrc/birch-docker
```

BERT can be characterized as one instance of a family of deep neural models that make heavy use of pretraining [8, 9]. Application to many natural language processing tasks, ranging from sentence classification to sequence labeling, has led to impressive gains on standard benchmark datasets. The model has been adapted to passage ranking [7] and question answering [13], and Birch can be viewed as a continuation of this thread of research, alongside other recent models such as CEDR [6]. The central insight that Birch explores, as detailed in Yang et al. [14], is to aggregate *sentence-level* scores to rank documents. This image allows other researchers to replicate the results of our paper with the `search` hook.

REFERENCES

[1] L. Azzopardi, M. Crane, H. Fang, G. Ingersoll, J. Lin, Y. Moshfeghi, H. Scells, P. Yang, and G. Zuccon. 2017. The Lucene for Information Access and Retrieval Research (LIARR) Workshop at SIGIR 2017. In *SIGIR*. 1429–1430.
[2] L. Azzopardi, Y. Moshfeghi, M. Halvey, R. Alkhawaldeh, K. Balog, E. Di Buccio, D. Ceccarelli, J. Fernández-Luna, C. Hull, J. Mannix, and S. Palchowdhury. 2017. Lucene4IR: Developing Information Retrieval Evaluation Resources Using Lucene. *SIGIR Forum* 50, 2 (2017), 58–75.
[3] R. Clancy, T. Eskildsen, N. Ruest, and J. Lin. 2019. Solr Integration in the Anserini Information Retrieval Toolkit. In *SIGIR*. Paris, France.
[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *arXiv:1810.04805*.
[5] J. Lin, M. Crane, A. Trotman, J. Callan, I. Chattopadhyaya, J. Foley, G. Ingersoll, C. Macdonald, and S. Vigna. 2016. Toward Reproducible Baselines: The Open-Source IR Reproducibility Challenge. In *ECIR*. 408–420.
[6] S. MacAvaney, A. Yates, A. Cohan, and N. Goharian. 2019. CEDR: Contextualized Embeddings for Document Ranking. In *arXiv:1904.07094*.
[7] R. Nogueira and K. Cho. 2019. Passage Re-ranking with BERT. In *arXiv:1901.04085*.
[8] M. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep Contextualized Word Representations. In *NAACL*. 2227–2237.
[9] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. 2018. *Improving Language Understanding by Generative Pre-training*. Technical Report.
[10] E. Sadler. 2009. Project Blacklight: A Next Generation Library Catalog at a First Generation University. *Library Hi Tech* 27, 1 (2009), 57–67.
[11] P. Yang, H. Fang, and J. Lin. 2017. Anserini: Enabling the Use of Lucene for Information Retrieval Research. In *SIGIR*. 1253–1256.
[12] P. Yang, H. Fang, and J. Lin. 2018. Anserini: Reproducible Ranking Baselines Using Lucene. *JDIQ* 10, 4 (2018), Article 16.
[13] W. Yang, Y. Xie, A. Lin, X. Li, L. Tan, K. Xiong, M. Li, and J. Lin. 2019. End-to-End Open-Domain Question Answering with BERTserini. In *NAACL Demos*. 72–77.
[14] W. Yang, H. Zhang, and J. Lin. 2019. Simple Applications of BERT for Ad Hoc Document Retrieval. In *arXiv:1903.10972*.

[1] https://projectblacklight.org/

[2] https://github.com/castorini/birch