CS 410 Technical Review: ElasticSearch and OpenSearch - An Exploration

Ivan Cheung (Icheung2@illinois.edu)

1. Background

ElasticSearch is a very popular, distributed search engine that is built on Apache Lucene. It's main uses include log analysis, machine learning, text search and document retrieval. Being distributed, the technology is able to construct a replicated, distributed inverted index which allows for fast text retrieval. ElasticSearch is supported on all three major cloud platforms: Google Cloud, Microsoft Azure and Amazon Web Services. However, from January 21, 2021, ElasticSearch modified its code licenses to proprietary, and in response, AWS created OpenSearch, an open-source implementation maintained by Amazon. This technology review will be to provide background on ElasticSearch and OpenSearch, and also list out the advantages of both, to aid a potential user on which branch to choose for their application.

2 Apache Lucene, OpenSearch, and ElasticSearch

Apache Lucene is a Java based library that ElasticSearch and OpenSearch are built on. It is the engine that runs the text search, storage, and retrieval analytics. ElasticSearch calls Lucene API functions, and provides additional features. The most important ones are high availability (through data replication), distributed computing, and data monitoring APIs. The ElasticSearch index is also worthy of mention here. It is comprised of chunks of documents, much like a relational data. These chunks are replicated over multiple servers, to maintain high availability. To search for specific terms that match the documents, ElasticSearch uses the Lucene index in something called a shard. The shard stores statistics about different terms to make a term-based search very efficient. It is built in the form of an inverted index. ElasticSearch also has a business model where users can pay for premium features – including security, which has been a controversial point in the userbase and a primary reason why OpenSearch was created.

OpenSearch is a forked copy of the last version of ElasticSearch that was open source – 7.10.2. It is open source but fully backed by Amazon teams. The goals of OpenSearch is to improve visibility, documentation and provide free version of ElasticSearch paid modules – the main one is security, but other features include data visualization modules, SQL integration and machine learning. [1]

4. OpenSearch advantages

Integration with existing AWS technologies

Being supported by AWS has major advantages for OpenSearch integration with the AWS infrastructure. AWS UltraWarm, for example, is a storage solution that allows the user to keep storage costs low while maintaining fast search performance. [2] Although the fastest speeds for indexing and retrieval come from on-board storage, it is recognized that the bulk of the data is

not as frequently accessed. UltraWarm uses a combination of both on-board storage for hot data (EBS volumes) and S3, which is more suitable for less frequently accessed data, or warm data. This makes UltraWarm an effective cost savings solution when the OpenSearch application is mainly used on large sets of analytic data, such as log analysis and web crawling.

AWS Auto-Tune is another integrated services with AWS that OpenSearch users can access. OpenSearch configurations and resource management is complex and a lot of the tuning parameters and metrics are internal. AutoTune can be enabled by enabling a plugin on OpenSearch, which then monitors and gathers the application's metrics. It then provides a suggested set of tuning parameters that can be used to optimize performance. These parameters vary from cache sizing to thread pool queue configurations. An example demonstrated by AWS shows that AutoTune can reduce JVM usage by 25%, and lower heap usages by almost 50%. [3]

3. ElasticSearch advantages

Platform Independence

Using OpenSearch confines the user's platform to AWS. For business applications, a business may need the additional flexibility of being able to choose different platforms apart from AWS, especially if they are already integrated into other cloud provides such as Azure and Google Cloud. ElasticSearch, and future version of it, are being supported through Elastic Stack on Azure and Elastic on Google Cloud. Different providers also offer differing services in terms of machine learning tools, observability through dashboard UI, and security considerations. Having the option to choose a different provider is a major benefit of picking ElasticSearch over OpenSearch.

Visualization tools

ElasticSearch supports a good amount of data visualization tools through user interfaces that OpenSearch has not matched yet. Many of these features are provided through the X-Pack, which is a paid module. While this is a deterrent to the open source community and individual users, the simplicity of having an out of the box solution for data visualization can be useful to the customer. For example, Kibana Maps allows users to build a map with "multiple layers and indices, embed it in dashboards, and animate spatial temporal data". [4] It can also integrate easily with the Elastic Dashboard client to provide an easy way for end users to interact with. Another example would be Kibana Lens. This is a tool that allows for users with no coding background to visualize data very quickly. They can drag and drop different data fields into Lens and instantly get a graphical visualization with customizable options. [5] From a business perspective, the technical department can use ElasticSearch to generate the data, and then pass it to the business analytics for analysis and visualization. OpenSearch, in its infancy, does not have a lot of these tools, and will have to rely on third party software and technical expertise to transform data into useful representations.

4. Conclusion

A quick analysis of both tools doesn't bring out a clear winner. OpenSearch, being freely supported and its integration to the AWS infrastructure, is good for users starting out their own

applications on a simple scale. However, ElasticSearch seems to be more tailored for business users, with its pipeline having a clear distinction for technical and non-technical users. Its platform flexibility is also a huge benefit to those who rely on other cloud services and providers. Picking the correct tool for text analysis will require analyzing one's own application and seeing which tool can provide the best cost to benefit ratios.

5. References

1. https://aws.amazon.com/blogs/aws/amazon-elasticsearch-service-is-now-amazon-opensearch-service-and-supports-opensearch-10/
2. https://docs.aws.amazon.com/opensearch-service/latest/developerguide/ultrawarm.html
3. https://aws.amazon.com/blogs/big-data/introducing-auto-tune-in-amazon-es/
4. https://www.elastic.co/guide/en/kibana/current/maps.html
5. https://www.elastic.co/kibana/kibana-lens