White Paper: Kvasir's Enterprise Search Knowledge Management Technology

1. Overview

Enterprises produce and consume data from many sources. Much of this data is messy, unstructured text, and it is organised and stored using a wide range of tools from the ad hoc intranets, chat systems, and email, to purpose-built proprietary tools such as Jira, Sharepoint and Confluence. Many organisations must also reference external data over which they have no control such as public archives, research papers, and patents.

Fast, effective access to data is therefore a major challenge and represents a significant cost to businesses in time lost finding information as well as the business risk of simply failing to find vital information at all. Research conducted by IDC[1] in 2019 indicated that this inability to access the right data costs businesses $430 billion/year, and that 66% of enterprises are currently investigating and investing in ways to address this important problem.

The likes of Google make excellent search tools to help people find information on the Internet, but these tools fall short in satisfying enterprise requirements. Google's search engine was created with the consumer in mind to use keywords to search the interlinked pages of the World Wide Web. It does not cater for the needs of enterprises who must search across multiple and disparate sources where data comes in complex and varied formats and is typically unstructured, with little or nothing linking one document to another.

Kvasir's knowledge management solutions directly address these enterprise needs, by creating a data technology that is fast, efficient, relies on document contents to provide the best possible matches, and can work with unstructured data stored drawn from multiple data silos.

[1] IDC 2019 https://www.idc.com/getdoc.jsp?containerId=US45900020

2. Technology Core

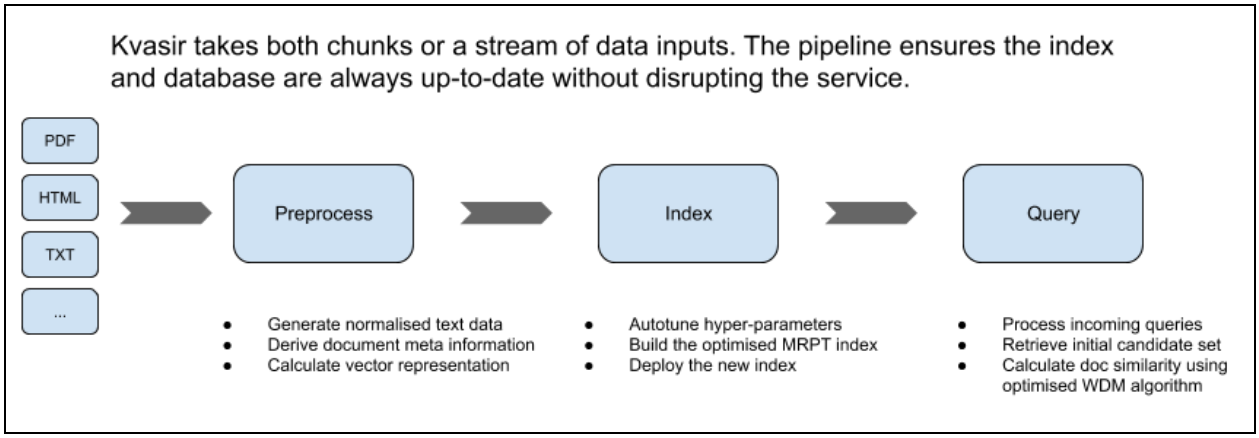
At its heart, Kvasir's technology provides for document matching: given a document-as-a-query, we find the top-N best matches among one or more of the data sources we have processed. This technique relies on two core components: first, means to acquire a document to use as a query, and second, means to process a data source to enable fast and effective document matching.

The first component is relatively straightforward in the context of enterprise search: in many workflows - whether searching for prior art for a patent filing, or accessing corporate memory of past successful projects when putting together a new bid - the user will already have a document ready to use as a query. Indeed, traditional search tools usually require the user to do extra work to read that document or documents, understand the content to some extent, extract the significant keywords, and then input those keywords into the search tool. Worse, this laborious process is often iterative, particularly where the user does not have deep familiarity with the domain and so may struggle to happen upon the best keywords to use.

The second is more intricate, and comprises a pipeline of three steps given a collection of documents from the data source:

1. Project the collection into a Vector Space Model (VSM).
2. Index that VSM.
3. Enable search using those indexes.

By using a richer input source (an entire document rather than just a few keywords), far more accurate results can be obtained and requires considerably less work by the user. However, all three steps face a major challenge in making them perform well enough that such an approach can scale to realistic sized input data in an enterprise environment.



2.1 Building the Vector Space Model

The classic and most widely used method is to use topic modelling, commonly Latent Semantic Analysis (LSA) or Latent Dirichlet Allocation (LDA), to convert each document in the collection into a vector. The collection of these vectors then forms the VSM. Unfortunately, both LDA and LSA are expensive, require substantial computational resources, and so take considerable time to construct a VSM given a document collection of any realistic enterprise size. Even worse, it is difficult to incrementally update the VSM as new documents arrive or old documents are deleted, which severely limits the practicality of these approaches in enterprise search where new content is continually being generated and updated.

Instead, we adopt state-of-the-art deep neural network technology, using word embeddings to project a collection of documents into a VSM of 300 dimensions. In comparison to the classical methods noted above, it has the following benefits:

- Very small memory footprint even with large vocabularies.
- Extremely fast, and scales linearly as more CPUs are available.
- Straightforward to incrementally update models.
- Enables support for multiple different languages.

2.2 Indexing the Vector Space Model

Before the VSM can be searched, it must be indexed. However, the classical K-Dimensional Tree algorithm simply cannot scale up to the high-dimensional spaces common in information retrieval applications such as enterprise search. Instead, we use our state-of-the-art Multiple Random Projection Tree (MRPT) algorithm which uses random projection as its backbone technology[2]. The main challenge that such an indexing algorithm must solve is the three-way tradeoff between search speed, result accuracy, and index size. MRPT is the fastest indexing algorithm that supports approximate k-nearest-neighbours search at the time of writing, as well as generating the smallest index structure, greatly improving scalability when processing large collections of documents. It is also inherently parallelisable, further improving scalability.

While the vanilla MRPT algorithm described above easily beats competing algorithms, it requires manual tuning of several hyper-parameters to obtain the best speed, accuracy, and query performance. The value of these hyper-parameters depends on the content being indexed, which implies extensive and expensive tuning whenever a new collection of documents is processed. To address this, our most recent innovation adds autotuning functionality so MRPT can automatically optimise all the hyper-parameters given targets for accuracy (in terms of recall) and query latency. The result is that we have a fully automated pipeline that can ingest a document collection at 15,000 documents/second to build the optimal index for subsequent queries.

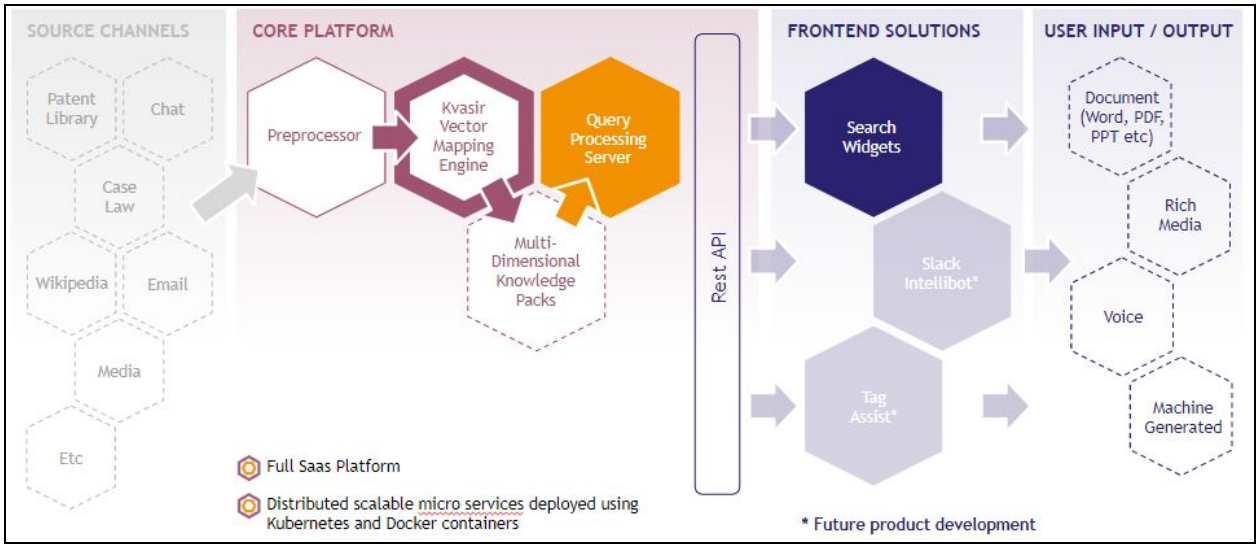
[2] Wang et al, IEEE Trans. Big Data, 2016, https://kvasira.com/papers/transbd16-kvasir.pdf

2.3 Searching the Vector Space Model

Finally, the most widely used industry solution to search the VSM is to calculate the Euclidean or Cosine similarity from the query documents' vector representation, and then ranking based on the similarity to points in the VSM. However, more advanced metrics such as Word Mover's Distance (WMD) produce better quality search results but present a performance challenge. WMD is essentially a convex optimization problem and requires heavy computation. Even with the most advanced approximation Sinkhorn, a query will still take several (3-5) seconds to calculate the best ordering given 125 calculated distances. Thanks to our work optimising this process, we bring this down from several seconds cost to less than 50ms.

3. Implementation

The above technology forms the core of our platform, processing document collections to produce indexed VSMs ready to be efficiently queried. We refer to each such indexed VSM as a Knowledge Pack. Our deployment follows a modern micro-services approach, wrapping each pipeline stage in a Docker container, and deploying an instance of the pipeline for each document collection as a Kubernetes pod. (Each pod actually contains more than three containers, as there may be one or two extra stages involved to clean and extract relevant text from the input documents.)



Our platform then presents a simple RESTful API supporting operations including:

- List all available public Knowledge Packs
- Find the *N* best matching documents in a list of Knowledge Packs

In both cases, results are returned as well-formatted JSON, with appropriate metadata. This super simple API thus makes it straightforward to integrate Kvasir's advanced enterprise search functionality with a wide range of tools and workflows. Integrations to date include our website, apps for Slack, Chrome, and Firefox, a plugin for Office 365 Microsoft Word, a Drupal "block", an Alexa skill, and a simple command line script.

4. Technology Benefits and USPs

Our approach provides several unique features, including:

- **Speed**. It is fast, both building and querying Knowledge Packs. We can ingest documents at around 15,000 documents/second when building a Knowledge Pack, and typical queries take around 50-150ms depending on the size of the Knowledge Pack and the size of the input document.
- **Multi-lingual**. We support querying a Knowledge Pack built from documents in one language using a query document written in a different language. For example, among our publicly accessible Knowledge Packs we currently have Wikipedia in English, Spanish, Hindi and Arabic; all can be searched using an input document written in English.
- **Single point of access**. We support querying of multiple Knowledge Packs representing data from different data sources simultaneously, with results presented either as separate lists, one per Knowledge Pack, or a single coherently merged list.
- **Secure and private**. Knowledge packs are divorced from the document collections used to create them, so they can be distributed without violating the privacy or security of the original documents.
- **Data efficient**. As each input document is represented as a 300-dimensional vector, the Knowledge Pack is typically several hundred or thousand times smaller than the document collection processed to generate it.
- **Any input**. Our platform can process input documents in a wide range of formats when building and querying Knowledge Packs. Currently supported formats include text, HTML, PDF, Microsoft Word (.DOCX), and Microsoft PowerPoint (.PPTX).
- **Immersive**. In combination, the above features allow us to provide truly immersive information retrieval - users need not leave what they are doing to execute a search for key content by typing keywords into a search engine, but can instead be presented proactively with the right information before they even ask for it.

5. Intellectual Property

All the code involved is the sole property of the company, with the exception of several open-source libraries that we use. All of these have been audited to ensure they have appropriately permissive licenses and no impediment exists to us using them in the ways that we do. While currently presented as a Software-as-a-Service solution, appropriate licensing

arrangements would be put in place to support enterprise sales of an on-premises version of our platform.

6. Future Proofing

Our platform is at the leading edge of the current state of the art, but this is a fast moving area. Our extensive and substantial academic heritage, plus our strong links into leading Computer Science research institutions such as the University of Cambridge Department of Computer Science & Technology ("The Computer Lab") and the UK's National Centre for AI, the Alan Turing Institute will ensure that we remain at the forefront of this developing field.

7. More Information

Email: contact@kvasira.com

Demos: https://kvasira.com/demo

Academic Papers: https://kvasira.com/technology