knóldus

Smart Searching Through Trillion Research Papers with Apache Spark ML

Himanshu Gupta, Knoldus Inc.

#SAISEco3

About Me

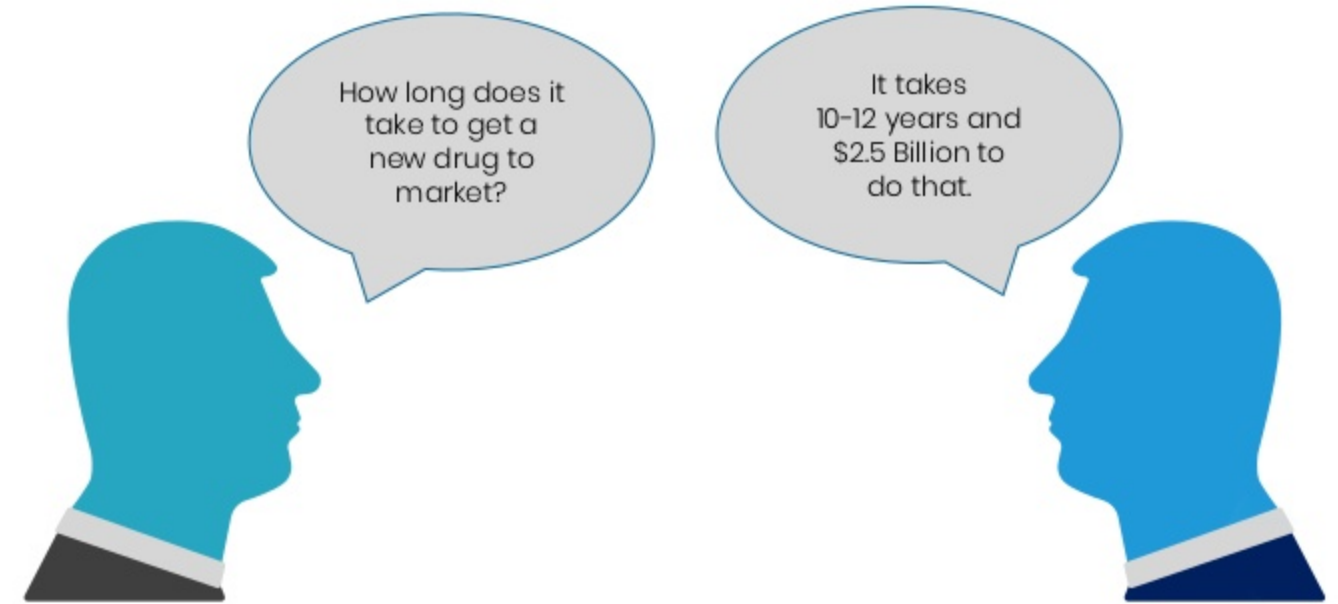
- Lead Consultant (Engineering) at Knoldus Inc.
- Work on reactive and streaming fast data solutions by leveraging Scala/Spark ecosystem.

Agenda

- The Need
- Challenges
- Our Solution
- Future work

The Need: Make Better Decisions Faster

The Need: Make Better Decisions Faster (contd.)



Surprises can be Costly

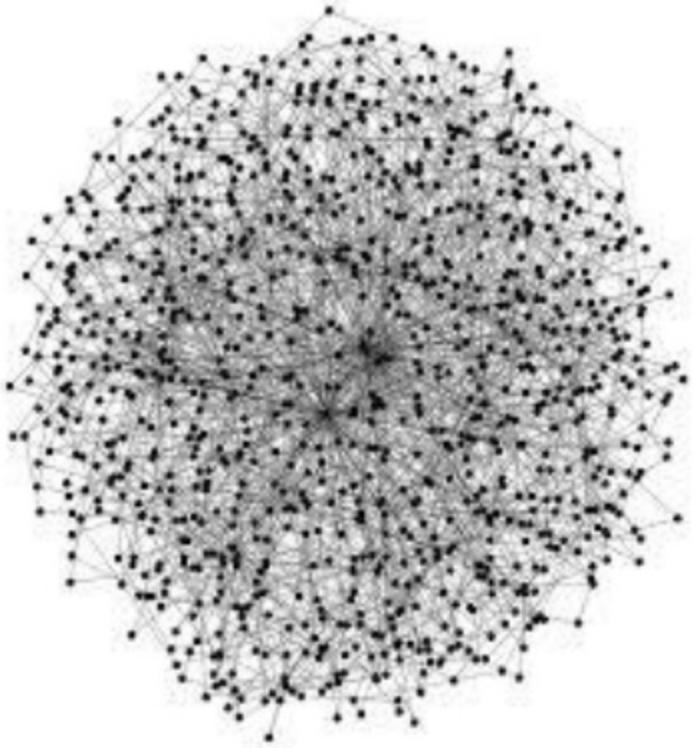
- In June, 2018, Tata motors produced just one unit of Nano (world's cheapest car).
- In case of few diseases the success rate of new drug being approved is less than 20%.

Best Solution:
Leverage the Work Done

Pharma companies partner with Research Organizations and Academic Institutes to reduce R&D cost up to 30%.

The Challenge:

It is Difficult

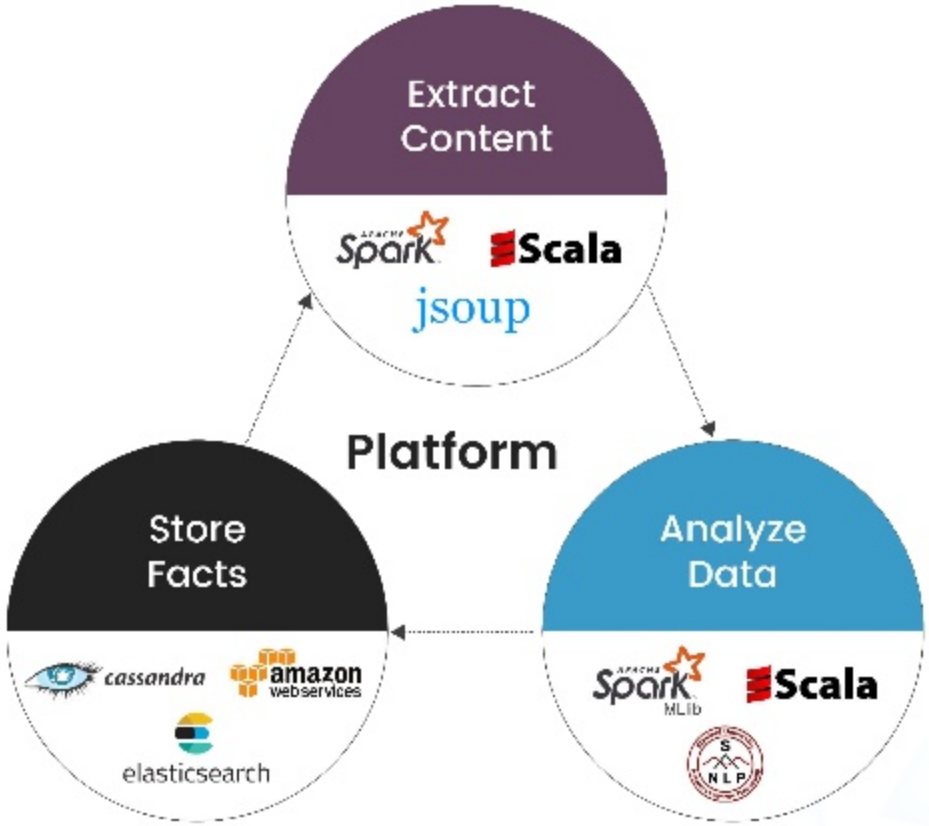


- R&D data is extremely complex.
- Each and every research work have a specific Aim which can overlap with other research work or not.
- The test environment of R&D work is different than actual world.
- There are many factors which are either assumed or ignored while conducting research.
- Facts are scattered over multiple research work.

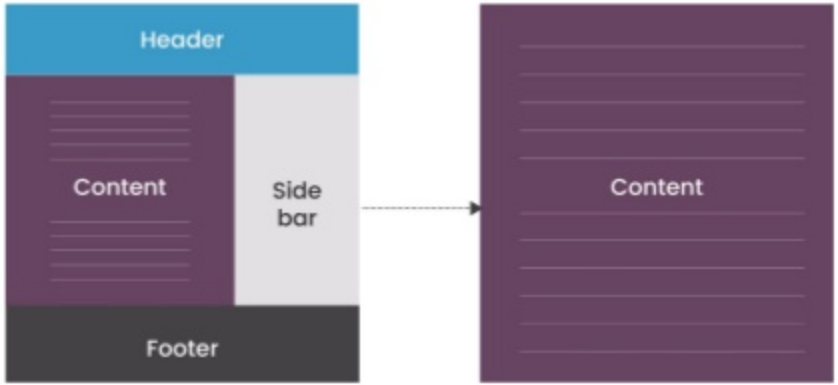
Our Solution: Build a Platform

- Where all the work done (research papers/articles) are collated.
- Allow easy access to the relevant research work.
- Discover new fields and concepts.

Design Philosophy



Step 1: Extract Content



- Extracting content from Research papers/articles is a time consuming and tiring process.
- Requires expertise of SME(s).
- However, if done by systems, can become blazingly fast and cost efficient.
- Systems extract content from research papers/articles and store them into a database from where it can be explored

Step 1: Extract Content (Process)

To scale the extraction process we leveraged Apache Spark's distributed computing feature

Read Documents

Read Research Papers /Articles from S3 / HDFS

01

Index

Index the content in to:

- Title
- Structure
- Special Objects

02

Enrich

- Prepare N-Grams
- Create MxN matrix
- Index the Matrix

03

Save

Save the enriched data into Database (Cassandra)

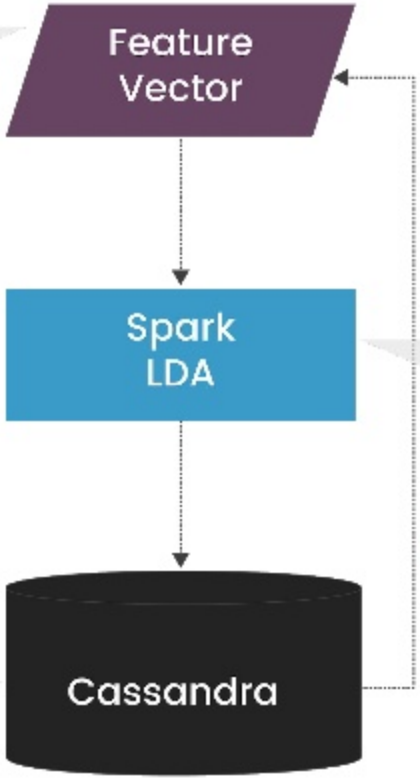
04

Step 1: Extract Content (Output)

	Word1	Word2	Word3
Doc1	Count	Count	Count
Doc2	Count	Count	Count
Doc3	Count	Count	Count
...	...	...	...
...	...	...	...
DocM	Count	Count	Count

Step 2: Analyze Content (First Iteration)

The input was a Feature Vector of Word Counts (for each word in a bag of words)



It takes in a collection of documents as vectors of word counts along with parameters: *k*, *optimizer*, *docConcentration* & *topicConcentration*, *MaxIterations*, & *checkpointInterval*

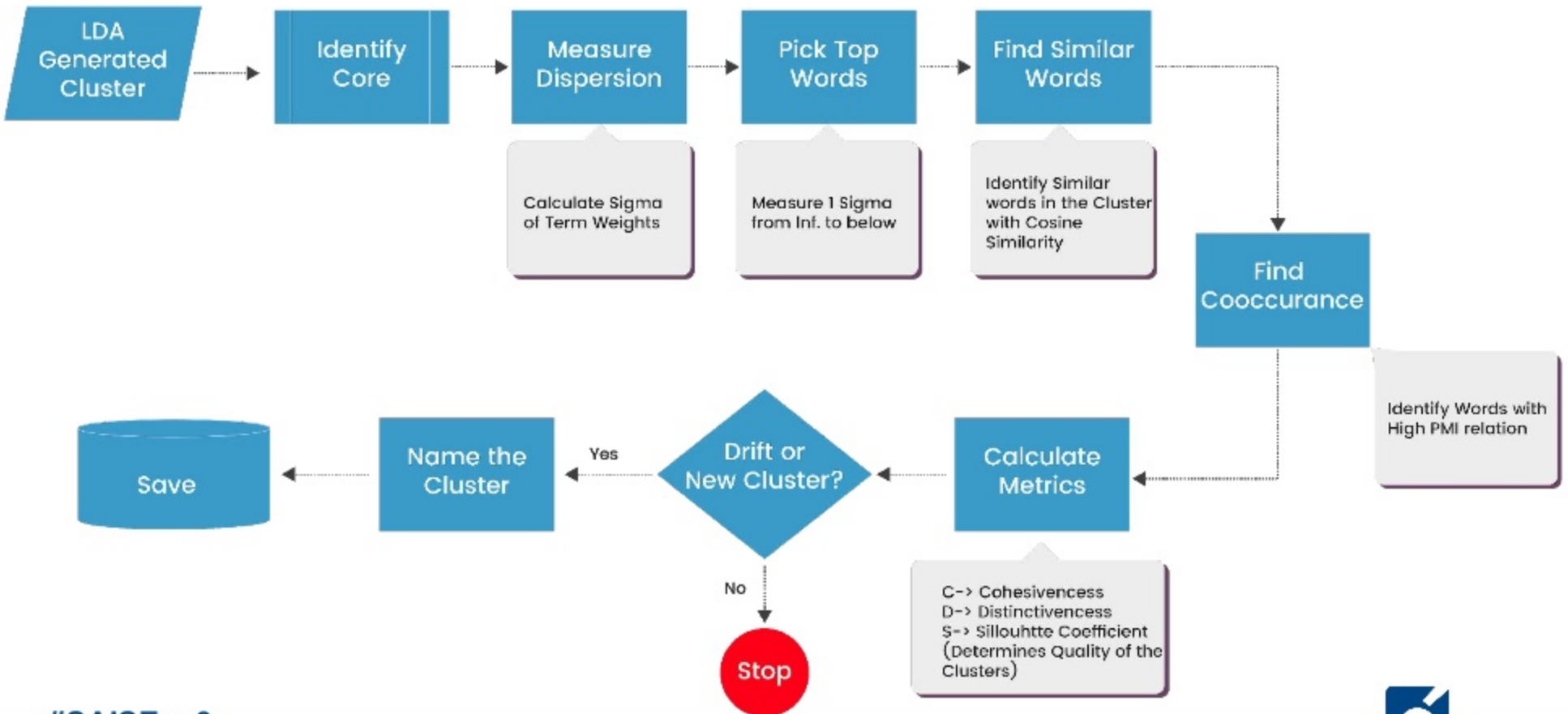
Store the results for future reference and tuning

Step 2: Analyze Content (LDA Output)

	Topic1	Topic2	Topic3
Word1	Term Weight	Term Weight	Term Weight
Word2	Term Weight	Term Weight	Term Weight
Word3	Term Weight	Term Weight	Term Weight
...	...	...	...
...	...	...	...
WordN	Term Weight	Term Weight	Term Weight

- Above words with term weights may not necessarily be the final chosen phrase to be identified as cluster(s).
- Because the number words that belong to cluster can be high (which is good, considering there will be several words that are ambiguous), one need to use different ways to identify phrases.

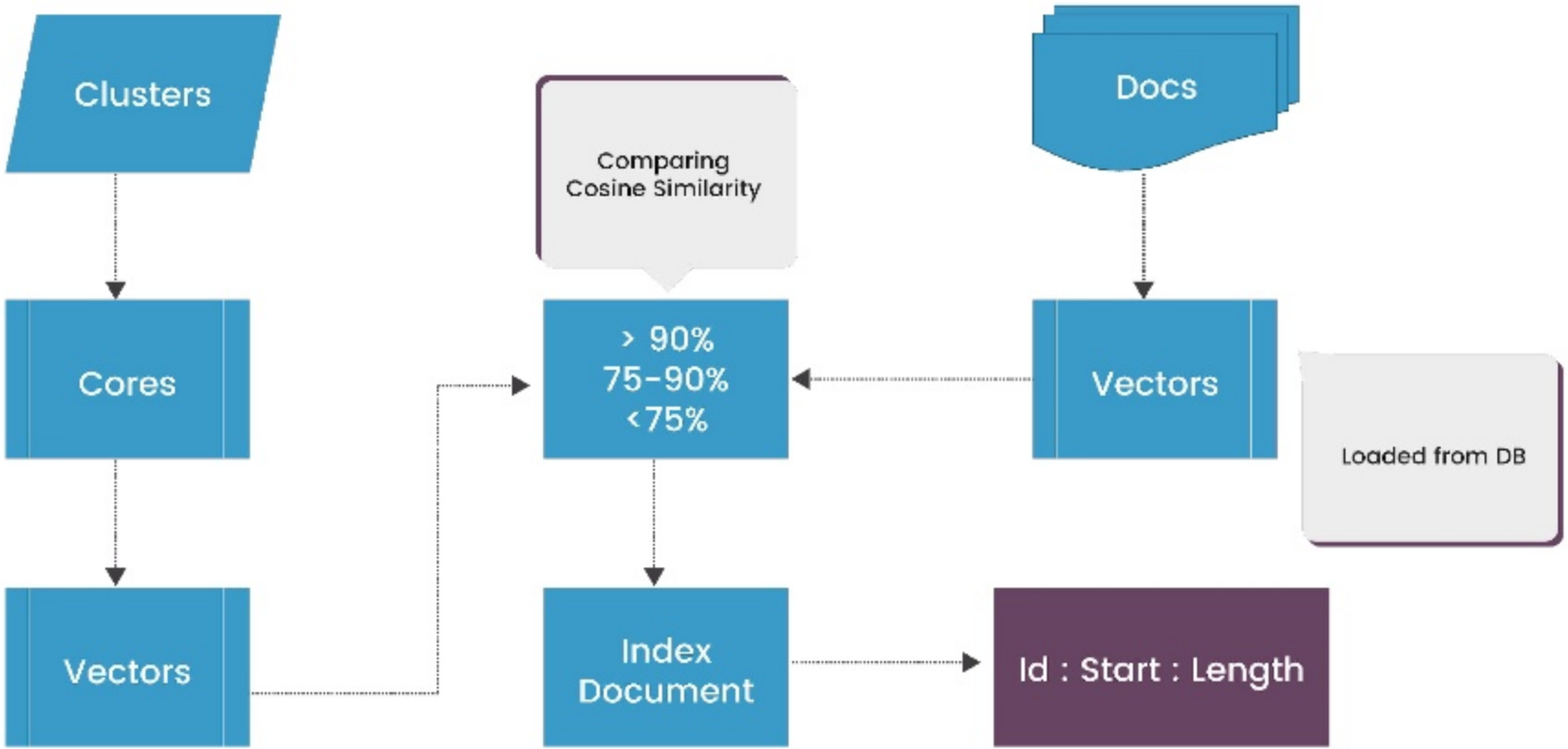
Step 2: Analyze Content (Identify Clusters)



Step 2: Analyze Content (Output)

Cluster of words formed from the research papers on Tuberculosis

Step 3: Store Facts (Indexing Documents)



Step 3: Store Facts (Output)

Doc Id	Content	Cluster ID	Core Terms	Similarity	Index
Doc1	Content1	Cluster Id1	Cluster1 Terms	Between (0-1)	Cluster Id1:Start:Length
Doc2	Content2	Cluster Id2	Cluster2 Terms	Between (0-1)	Cluster Id2:Start:Length
..	..	..	..	..	..
DocN	ContentN	Cluster Id1	Cluster1 Terms	Between (0-1)	Cluster Id1:Start:Length

Now we can search documents on the basis of terms we want to:
select * from facts where coreterms like 'metallurgy'

Future Work

Semantic Search

- Index Data in Elasticsearch/Solr
- Run semantic query over indexed data
- Like, How Can we Separate Gold From Mercury? Or Which are the compounds which have recursive bonding with Carbon and Iron?

Quality Workbench

- To measure the relevance of search.
- To tune the performance of ML algorithms.

knóldus

Thank You!

Stay in Touch

https://www.facebook.com/KnoldusSoftware/

@himanshug735

+(1) 647-467-4396