| Fields | Values |
| --- | --- |
| Team Name | Cloud-9 |
| Team Member #1 | Sanjay Nithin (20BIT0150) |
| Team Member #2 | Ishani Chowdhury (20BIT0175) |
| Team Member #3 | Saudamini Shail (20BIT0117) |
| Topic | DA Sharing platform review-2 |

New

New

# 1) Introduction

## 1.1) Background

There hasn't been an authentic assignments sharing platform in the market. Chegg and Coursehero are the major players but they are paid services. A community driven sharing forum is much needed among college peeps.

## 1.2) Problem Statement

DA sharing platform is a web-app where students can login into our site using their student IDs and can access as well as upload assignments. To make it more accessible, it is required that only digitally typed assignments should be uploaded. Duplication of assignments is prevented to maintain the authenticity.

## 1.3) Literature Survey

**1) J. Zhao, M. A. Rodriguez and R. Buyya, "High-Performance Mining of COVID-19 Open Research Datasets for Text Classification and Insights in Cloud Computing Environments," 2020 IEEE/ACM 13th International Conference on Utility and Cloud Computing (UCC), 2020, pp. 302-309, doi: 10.1109/UCC48980.2020.00048.**

The global epidemic of COVID-19 is a health emergency without precedent. Since the epidemic, numerous researchers from all around the world have

created a substantial body of literature. It takes a lot of processing power to analyse this data, extract knowledge from it, and quickly give insightful information. Cloud computing platforms are made to deliver this processing capacity in an elastic and on-demand way. Particularly well suited for the deployment of computationally intensive workloads in a cost-effective yet scalable way are hybrid clouds made up of both private and public data centres. In this research, we created a system to speed up the data pipeline and article categorization process using machine learning on a hybrid cloud by utilising the Aneka Platform as a Service middleware with parallel processing and multi-cloud capacity. AWS Kendra is then used to persist the findings for further use in searching, referencing, and visualisation. The system can aid in reducing processing time and attaining linear scalability, according to the performance evaluation. The programme could be immediately applied to broader scholarly article indexing and analysis outside of COVID-19.

**2) J. Wang, R. Zhang, J. Li and Y. Xiao, "Owner-Enabled Secure Authorized Keyword Search Over Encrypted Data With Flexible Metadata," in IEEE Transactions on Information Forensics and Security, vol. 17, pp. 2746-2760, 2022, doi: 10.1109/TIFS.2022.3163886.**

The ability to organise, retrieve, and comprehend data is greatly aided by metadata, which also contains a wealth of sensitive information about the data and the people who use it, such as the location where a photo was taken. When sensitive data is encrypted before being uploaded to unreliable public clouds, a practical conundrum arises: if the metadata is completely encrypted, its capabilities are lost; otherwise, sensitive material may leak. It is therefore desirable to have a secure and adaptable mechanism for processing many fields of metadata at once, either tagged as private or public depending on the needs of the scenario. We looked through the literature for strategies to accomplish such a goal, but it turned out that this had never been openly thought of or rationally resolved. Therefore, in this work, the issue of creating flexible, tamper-resistant metadata settings and owner-enabled secure search authorization with explicit metadata is considered. A novel Authorized Keyword Search over Encrypted Data with Metadata scheme (MD-AKS) that, in the first place, effectively addresses the aforementioned requirements is provided. It is based on the idea of public key encryption with keyword search (PEKS). We define the security model and demonstrate the MD-AKS scheme's

security. In two ways, the method maximises the flexibility of metadata setting: the related metadata can be set as any string, and client costs are independent of the complexity of the explicit metadata. MD-AKS is used, and the results of the experiments and theoretical comparisons further show its usability and scalability.

**3) A. Keck, M. Romero, R. Sandor, D. Woodbridge and P. Intrevado, "Predicting Unethical Physician Behavior At Scale: A Distributed Computing Framework," 2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2019, pp. 110-116, doi: 10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00061**

Creating a strong pipeline to stream, store, and analyse data is essential as the volume of publicly shared data rises, as casual users frequently lack the software, hardware, and/or expertise required to handle such a large amount of data. In this study, the authors use Amazon EC2, EMR, and Apache Spark to analyse 28.5 GB of CMS Open Payments data in an effort to pinpoint doctors who may be particularly likely to act unethically as a result of substantial wealth transfers from pharmaceutical corporations. The top decile of doctors who are most likely to engage in unethical activity in the upcoming year are predicted using a Random Forest Classifier, yielding an F-Score of 91%. The authors also use an algorithm for detecting anomalies, which successfully discovered a well-known instance of a doctor quitting his prominent job while omitting to reveal unusually huge transfers of wealth from pharmaceutical companies.

**4) Park, G., Heo, Y.S., Lee, K. et al. A parallel and accurate method for large-scale image segmentation on a cloud environment. J Supercomput 78, 4330–4357 (2022).**

In this paper, PSLIC-on-Spark is introduced. It is a parallel approach for SLIC on Apache Spark. To achieve this, the original SLIC algorithm is enhanced to make use of Apache Spark's features, enabling parallel processing across several Apache Spark cluster executors. Then, they examined how the partitioning of the original image datasets affects the trade-off between processing speed

and accuracy for PSLIC-on-Spark. They tested the relationship between the trade-offs through experiments. They demonstrated that PSLIC-on-Spark with 8 CPU cores decreases the processing time of SLIC by 2.24–2.93 times while increasing under-segmentation error (UE) and decreasing boundary recall (BR) of SLIC by 1.54–6.32%. Then, they suggested PASLIC-on-Spark, an enhanced PSLIC-on-Spark algorithm that increases PSLIC-on- Spark's accuracy. We use two crucial components for PASLIC-on-Spark. It has two primary characteristics: (1) image partitioning that takes into account the position and structure of the clusters rather than using an evenly distributed method; and (2) controlled duplication for the border between image partitions. We demonstrate through trials the precision and effectiveness of PASLIC-on-Spark in a real cloud setting with 8 worker nodes utilising Amazon AWS. According to the experimental findings, PASLIC-on-Spark increases PSLIC-on-accuracy Spark's by 3.66–3.77% of BR and 1.39–1.96% of UE. With 8 CPU cores configured on a single node, PASLIC-on-Spark still considerably reduces processing time SLIC by 1.5–1.67 times, and by 1.18–1.26 times in a cloud system with 8 compute nodes.

**5) MOHAMED, R. ., EL-BASTAWISSY, A. ., NASR, E. ., & GHEITH, M. . (2021). Comparative Study of Record Linkage Approaches for Big Data. Walailak Journal of Science and Technology (WJST), 18(2), Article 7221 (22 pages).**

Record linking is a difficult Big Data task. Thus, by contrasting three aspects including record linkage stages, dataset features, and a parallel processing technique for big data, thisresearch aims to shed light on record linkage procedures for big data. Only comparative studies of different record linkage strategies have been done so far, according to the state of the art. There has only been one comparative research that has looked at the relational database's entire record linkage system. It is thought that the current study's emphasis on the characteristics of datasets and the dimensions of parallel processing algorithms for big data was worthwhile to investigate. First, despite the importance of exploring the dataset under study, data exploration was almost nonexistent. Second, methods for handling the first dimension's data standardisation and preparation phase were not thoroughly covered in the literature. Third, record linkage in unstructured data was not yet explored in literature. Fourth, MapReduce was used in about 50% of the selected studies.

Due to its support for in-memory processing, Apache Spark has just recently been modified to resolve duplicates, making the entire linking process more effective. Although Apache Spark is supported by a large number of current studies in the comparison study, further research must be done before using Apache Spark to address the issue of record linkage. Additionally, Apache Flink is still infrequently utilised to address the Big Data record linking challenge. Fifth, despite its impact on lowering the search space and resulting in a more effective Record Linkage process, pruning strategies, employed to reduce pointless comparisons, are not successfully implemented in the studied research.

**6) Max Petrov, Nikolay Butakov, Denis Nasonov, Mikhail Melnik, Adaptive performance model for dynamic scaling Apache Spark Streaming, Procedia Computer Science, Volume 136, 2018, Pages 109-117, ISSN 1877-0509,**

Data volumes are skyrocketing today, and a variety of sources, including sensors, traffic, mobile phones, and other devices, provide a wealth of data. Each piece of information from these sources can be shown as a data stream, whose size can change over time. In the first scenario, data processing must be optimised by dynamic resource allocation in order to reduce processing time. In the second scenario, data processing must be optimised through resource deallocation because cutting out unused resources can lower overall costs. How can I determine the ideal resource allocation to meet the necessary processing latency under a specific workload volume? Existing models and the way Apache Spark Streaming is currently implemented prevent us from having this possibility. Adaptive performance model, which can dynamically scale up and down Apache Spark Streaming platform on the AWS, is proposed in this work.

**7) G. Satyanarayana, J. Bhuvana and M. Balamurugan, "Sentimental Analysis on voice using AWS Comprehend," 2020 International Conference on Computer Communication and Informatics (ICCCI), 2020, pp. 1-4, doi: 10.1109/ICCCI48352.2020.9104105.**

These days, sentimental analysis is crucial since many startups have user-driven content as their foundation [1]. An essential field of study in natural language processing is sentiment analysis. There are several uses for natural

language processing, including text classification, aspect-oriented product analysis, sentiment analysis, voice recognition, machine translation, and product reviews [2]. By analysing the conversation's emotions, this procedure will enhance the company's operations. Author will use Amazon Comprehend to conduct sentimental analysis for this project. In order to extract the document's information, Amazon Comprehend uses machine learning and natural language processing (NLP). You canextract unstructured data, such as photos, audio, and other types, by using this service. As a result, it will be possible to tell whether a dialogue is positive, negative, neutral, or mixed by identifying its emotions. To complete this task, the author will use several AWS services, including S3 (used for data storage), Transcribe (used for text-to-audio conversion), Aws Glue (used to create metadata from comprehend files), Aws Comprehend (used to create sentiment files from audio), Lambda (used to trigger from s3 data store), Aws Athena (used to turn text into structured data), and finally Quick Sight (used to visualise the data from s3 files).

**8) Discover Insights and Relationships in Text Using Amazon Comprehend by Raj Kumar Mohanta, International Journal Available Online @ www.ijtiir.com of Trend in Innovative Research (IJTIIR) | ISSN: 25820354**

A new, quickly evolving, and highly promising technology is cloud computing. It speaks of computing, software, data access, and storage services that don't need the end user to be aware of the system's physical location or configuration in order to be used. The growing use of virtualization, services, autonomic computing, and utility computing has led to the natural progression of cloud computing. One of the many services supported by Amazon Web Services (AWS), which provides dependable, scalable, and affordable cloud computing services, is Amazon Comprehend.

**9) S. M. Zobaed, M. A. Salehi and R. Buyya, "SAED: Edge-Based Intelligence for PrivacyPreserving Enterprise Search on the Cloud," 2021 IEEE/ACM 21st International Symposium on Cluster, Cloud and Internet Computing (CCGrid), 2021, pp. 366-375, doi: 10.1109/CCGrid51090.2021.00046.**

Big data owners have been drawn to cloud-based enterprise search services (like AWS Kendra) because they provide them with practical and immediate

search solutions. The issue is that people and organisations with private big data are reluctant to use these services because of legitimate data privacy concerns. These services also examine the user's search history in order to provide an intelligent search, thereby jeopardising the user's privacy. The primary goal of this research is to distinguish between the pattern matching and intelligence components of the search in order to solve the privacy issue. The shared cloud tier only acts as an exhaustive pattern matching search tool in this theory, and an on-premises edge tier provides the search intelligence. We put forth the Smartness at Edge (SAED) mechanism, which provides intelligence in the form of tailored and semantic search at the edge tier while protecting the privacy of the search on the cloud tier. In order to broaden the query and incorporate its semantics, SAED uses a knowledge-based lexical database at the edge tier. Through the use of an RNN model that can detect the user's interests, SAED customises the search. In order to obtain documents that are semantically relevant to the search query, a word embedding model is used. Without imposing any changes, SAED may be plugged into existing enterprise search systems to allow them to offer intelligent, privacy-preserving search. SAED can increase the relevancy of the retrieved results by an average of 24% for plain-text and 75% for encrypted generic datasets, according to evaluation results on two enterprise search systems conducted in real settings and validated by human users.

**10) Zero-Shot Open-Book Question Answering, Sia Gholami, Mehdi Noori, arXiv:2111.11520**

Open-book question answering is a subset of question-answering jobs in which the system looks for solutions in a set of papers and general information about a subject. This article suggests a method for responding to inquiries in natural language from a corpus of technical publications from Amazon Web Services (AWS) that lack domain-specific tagged data (zeroshot). Answers to these queries may be yes-no-none, brief, lengthy, or any combination of the three. This solution has a two-step design where a retriever locates the appropriate document and an extractor locates the solutions in the recovered document. A new test dataset based on actual customer queries on the AWS technical documentation is being introduced for open-book QA. The method tries to identify the yes-no-none responses and text answers in the same pass after testing out a number of information

retrieval systems and extractor models based on extractive language models. The Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2016) and Natural Questions (Kwiatkowski et al., 2019) datasets are used to train the model. Without any domain-specific training, we were able to attain an end-to-end exact match score (EM) of 39% and 49% F1.

## 2) Overview and Assumptions

### 2.1) Proposed System Overview

1. There's a login page where students can login to our site using their university mail IDs.
2. The document is checked for any duplication and if so it is rejected.
3. Upon passing the duplication check, it is stored in S3.
4. To search for a document, a query is sent and the relevant documents are sent back as the response.

### 2.2) Assumptions

1. The only one assumption we have is the document should be digitally typed as we are using NLP tools to extract the text from the document.

### 2.3) Challenges

1. Duplication of documents and searching throught the contents of the file are the two big challenges involved in this project.
2. Duplication of documents can be solved using NLP tools which extracts text from the document and compare it with the text extracted from other documents stored in S3.
3. Searching can also be handled using the same methodology.

### 2.4) Software Specifications

1. AWS comprehend
2. Apache Spark
3. AWS S3
4. Amazon Kendra

### 2.5) Hardware specifications

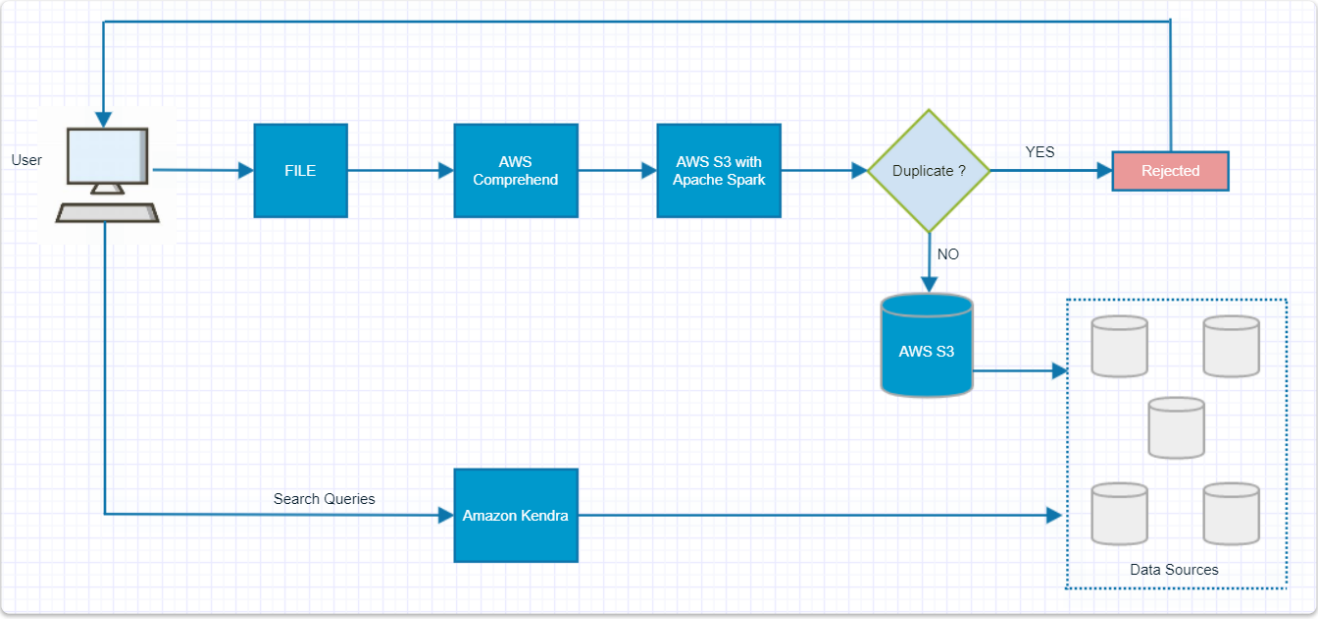
1. 5GB of S3 in the free tier.

### 2.6) Architecture Specifications

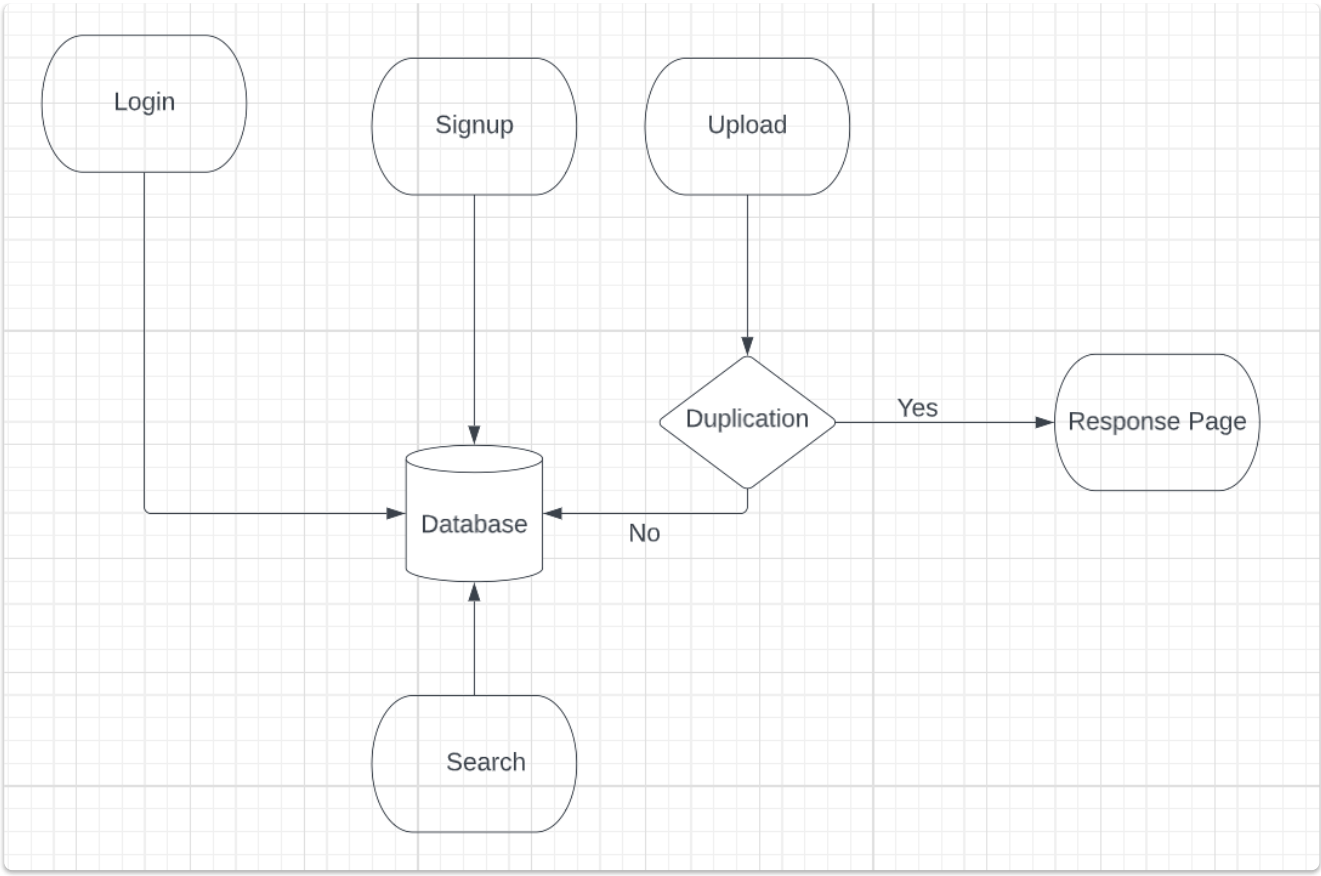
1. Document will be passed onto AWS comprehend, an NLP tool to extract keywords and to improve performance of search query Apache Spark is used.
2. S3 datastore is searched for the query and duplication is avoided by rejecting.
3. Otherwise, stored in S3.
4. Searching is done with the help of Amazon Kendra which is used to search through the contents of the file.

## 3) System Design

### 3.1) High level System Design




### 3.2) Low level System Design


Login
Signup
Upload
Duplication
Yes
Response Page
Database
No
Search