Grenze International Journal of Engineering and Technology, June Issue

IntelliStruct: An Automated Tool for Data Structuring

Vinita Kakani[1], Divya Mahajan[2], Kalpesh Marbate[3], Srushti Bandewar[4], Tanmayee Dube[5] and Aman Maan[6]
[1-6]Yeshwantrao Chavan College of Engineering, Nagpur, India
Email: {kakanivinita24} gmail.com, divyamahajan5113, kalpeshkk442, bandewarsrushti, tanmayeedube18, amanmeersinghmaan@gmail.com

***Abstract*—With the rapid growth of today's organizations, enormous amounts of data are being used like resumes, feedback, emails, multimedia, etc. Which plays a vital role in recruitment, project feedback, social media posts, etc., which are in the unstructured format, for using this data text extraction is a crucial task in the field of data analytics and information retrieval. Systems like AI and ML need data in a well-organized, structured format because unstructured data are inherently difficult to analyze and extract insights from. However, converting the data manually is a significant challenge. This process is time-consuming and prone to human error. Thus by using NLP libraries, FRONTEND and BACKEND parts converting unstructured data to structured data is more efficient and can improve decision-making.**

I. INTRODUCTION

Unstructured data, which comprises a significant portion of the information generated from sources like social media, emails, customer reviews, and multimedia content, is pervasive in today's data-driven economy. This type of data is rich in insights, but because it lacks a preset framework, analysis and decision-making are greatly impeded. It is therefore imperative to transform unstructured data into structured representations. By categorizing this data, businesses may use advanced analytics techniques, more easily integrate this data with existing systems, and ultimately achieve better strategic outcomes.

Text extraction from both organized and unstructured data is crucial for data processing and analysis. It is crucial to extrapolate pertinent textual information from the vast amounts of data that businesses have gathered from various sources. This makes it easier for them to make decisions, automate processes, and gain insights. To put it simply, effective text extraction is required to transform raw data into usable knowledge [1]. Thanks to developments in data gathering techniques, there is a significant growth in the amount of data that is collected and kept every day. It's usual to refer to this exponential growth in data as "Big Data." With so much data, it's imperative to derive meaningful conclusions from it. Put another way, when more data are received, the importance of finding pertinent information increases [4].

Data mining and natural language processing (NLP) are two methods used to transform data into a structured format. Although natural language processing (NLP) focuses on understanding and interpreting human language, data mining helps identify patterns and trends in large datasets. When integrated, these methods provide several perspectives for comprehending unstructured data, supporting the interpretation and organization of the information for enhanced understanding [5]. Modern applications have led to an evolution in the requirements for ETL (Extract, Transform, Load) processes. The conventional ETL approach involved reading data from files and writing it to files in multiple phases, which was a sluggish batch process that was manageable by computers with older hardware.

Grenze ID: 01.GIJET.11.2.106_20

However, newer applications—like those found in the Internet of Things (IoT)—need real-time data in order to support timely decision-making, which calls for expedited and more efficient ETL processes[6].

II. Literature Review

The article, titled "Text extraction from structured and unstructured data sources,"[1] focuses on techniques for extracting important text from various types of data, including structured ones such as databases and document files, web pages and multimedia content, and other parameters. Authors Edwin Frank, Joseph Oluwaseyi, and Godwin Olaoye delve into approaches ranging from simple rules-based approaches using regular expressions to advanced machine learning such as NLP and OCR for low-quality materials. The materials include.
Introduction to Text Extraction: Discusses the importance of text extraction for tasks such as information retrieval, decision-making, and analytics job automation (comparative context) and machine learning-based approaches (supervised and unsupervised learning). In data search, data analysis, compliance management, and customer management.
The article "Transforming unstructured and semi structured data into knowledge"[2] focuses on extracting useful information from unstructured and semi-structured data using knowledge discovery (KDD) and data mining techniques. It emphasizes the importance of seamlessly transforming diverse information, such as text and multimedia messages, into models that can be used for decision making in areas such as business copy and processing. It includes steps such as data selection, prioritization, transformation, and evaluation. Technology: Text mining, syntactic and semantic analysis, and similar techniques in data classification are used to derive insights.

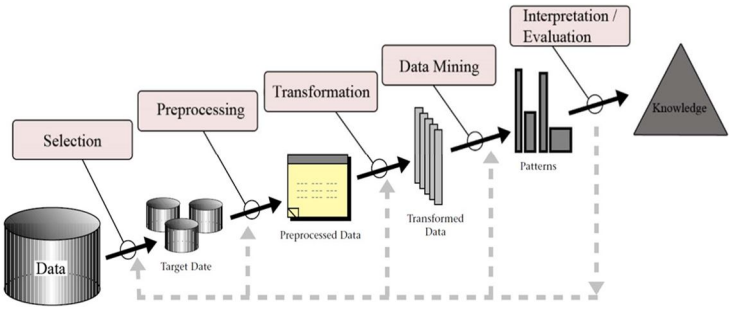

Figure.1. Steps of the KDD process [2]

The paper" Current Trends in Data Ingestion and Integration [3]: A Comprehensive Review" provides insightful information about how to manage large and diverse data types using data ingestion and integration techniques. It emphasizes the critical difference between data ingestion, which is just storing data, and integration, which is transforming data into a structured format and organizing unstructured data.
The research "A Proposed Technique for Conversion of Unstructured Agro-Data to Semi-Structured or Structured Data"[4] Tackles the difficulties in organizing and deriving valuable insights from the vast amounts of unstructured agricultural data produced by IoT, sensors, and social media. It suggests a way to transform this unstructured data into a more manageable semi-structured or structured format by utilizing Couchbase, a NoSQL database.
Given that the majority of agricultural data is unstructured and challenging to evaluate with conventional techniques, the study emphasizes the significance of transforming unstructured data. It goes over a number of current methods and talks about their drawbacks, like the requirement for particular setups or the inability to analyze a single text. The suggested method makes use of Couchbase to improve performance, scalability, and accessibility of data. This strategy could benefit agricultural academics and practitioners by enhancing data management and giving a way to store and analyze huge datasets effectively, ultimately helping in making data-driven decisions based on historical agricultural data[4]
The article, titled "Conversion of unstructured data to structured data with a profile handling application."[5] explores the process designed to create redundant data, with a specific focus on starting the recovery from scratch. The main goal of the research is to transform many uninformative resumes into standard methods to improve literacy and improve the hiring process. This article addresses the problems caused by inadequate recovery, which often manifests itself in different ways and contains different information. This makes it difficult for job seekers to get good information on the subject. The author publishes an application that uses

Natural Language Processing (NLP) technology to convert illegal text into documents. The system relies on Python's and other tools like Apache PDFBox to extract information like name, email, phone number, and educational content from the written record. - Upload your resume in PDF format and use PDFBox to convert it to text. Regular expressions are used to extract data, such as finding email addresses, phone numbers, and CGPA. The following is an example of conversion of an unstructured resume into a structured form which is mentioned in this paper [5].

The article "Device Data Ingestion for Industrial Big Data Platforms with a Case Study"[8] presents a model for managing heterogeneous big data consumption from various devices in the context of Industry 4.0. The authors focus on three main challenges: heterogeneous data, multiple data, and big data (big data). Big Data Platform (IBDP).

This model supports data objects from various sources (such as streaming data, archived data, and relational data) and uses four data processing methods: data synchronization, slicing, segmentation, and indexing. Two companies, Big Food Group and Jinan District Heating, are taken as examples to demonstrate the effectiveness of the model. In this study, temperature and energy consumption data from air conditioners are extracted and analyzed to identify deficiencies and reduce costs, including optimization strategies for refrigerators. Models for acquiring and combining heterogeneous data from multiple sources.

Case Study. This statement shows that this model can improve the enterprise's ability to analyze and use business information to make better decisions.

Another paper titled "Big data preprocessing: methods and prospects"[15]. Indicates the preprocessing of big data and the tasks included in it. This paper mainly focuses on different methods and techniques necessary for handling large amounts of data. The paper indicates the importance of preprocessing and ensures the quality and usability of data before applying methods, highlighting problems like noise, data dimensionality, and missing values.

III. METHODOLOGY

This review paper takes the methodical approach to assess current technologies and methods used for the conversion of unstructured data into structured formats. The approach involves outlining search methods, choosing pertinent research, and setting up criteria for evaluating various techniques. The methodology consists of the project planning phases, the chosen technology, and the process for implementing the tool.

A. Project Planning and Phases

- Data Ingestion: The main goal of this stage is to gather and bring unorganized data from different sources into the system. The data categories under consideration encompass textual documents, PDFs, and images containing embedded text. Multiple channels are utilized by the system to receive inputs, which include user-uploaded files, API integrations for accessing real-time data, and batch uploads for handling large datasets.
- Data Preprocessing: After the data is received, it goes through a preprocessing phase to get it ready for analysis, which is essential for removing unwanted elements and ensuring data uniformity. Methods like getting rid of special characters, extra spaces, and stop words were used to remove irrelevant details from the text data. Practices such as converting to lowercase and lemmatization were employed to ensure the consistency of different word forms throughout the dataset. When handling image inputs, OpenCV and Pytesseract were used for Optical Character Recognition (OCR) to extract text from images. The extracted text was then considered part of the textual data and cleaned accordingly
- NLP Processing: The preprocessed text undergoes NLP techniques to extract valuable information, including identifying entities like names, locations, dates, and keywords through named entity recognition (NER) and keyword extraction using tools such as SpaCy and NLTK. For sentiment analysis of the text data, models pre-trained by Hugging Face Transformers are used to classify content as positive, negative, or neutral. To enable organization and storage, the text is converted into numerical representations using methods like TF-IDF (Term Frequency-Inverse Document Frequency) and Word2Vec embeddings.
- Data Structuring: In this stage, the emphasis is on converting the processed data information into organized formats that are easily storable and can be easily retrieved. A mapping schema was developed to convert the identified text attributes into structured formats like tables or JSON files. Tailored scripts were employed to convert the identified attributes into structured formats including CSV, JSON, and SQL tables. The structured data was stored in MySQL for relational data, enabling complex queries, and in MongoDB for the more flexible, document-oriented storage of semi structured data.

- Frontend Integration: The last phase of planning is to create a user-friendly interface using Flask for the backend and React for the frontend to ensure that the organized data is easily accessible to users. Users can upload files for processing, view structured data outputs, and download results in CSV or JSON formats through the interface. Flask was used to develop RESTful APIs, enabling seamless communication between the UI and the backend processing logic, allowing users to request data processing and receive real-time results. Using Visualization tools like Chart.js were integrated into frontend, providing users with graphical representations of key insights such as the frequency of named entities or sentiment analysis results.

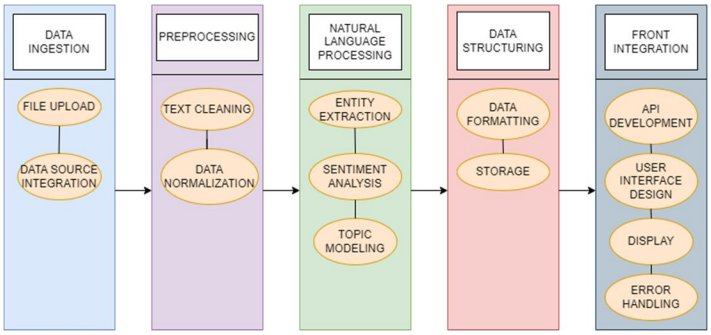

Figure.2. Methodology

B. Technologies and Tools Selections

To create this tool there are some technologies and tools which guarantee the effective handling, organization, and control of the unstructured data. This part details the primary decisions for backend development, natural language processing (NLP), machine learning, and data management, explaining the reasoning behind each selection.

- Backend Development: Python was selected as the primary programming language because of its adaptability and wide range of libraries for machine learning, data processing, and connecting to various tools. Then we choose Flask for building an effective and lightweight API interface, for proper and smooth integration with other applications, and for providing users with an easy way to interact with tools through a web-based interface.
- NLP and ML libraries: The NLP tasks were managed using libraries like SpaCy, NLTK, and Hugging Face Transformers. SpaCy's high-speed performance and pretrained models made it the preferred choice, while NLTK's comprehensive NLP functions were also utilized. Hugging Face Transformers was employed for its advanced language models like BERT which provided deeper meaning into textual data. For implementing feature extraction methods like TF-IDF and training basic classification models, the Scikit-learn library was used. Its ease of use, integration with Python, and extensive documentation made it a reliable choice for prototyping and testing models.
- Data Management: We made use of MySQL for its strong support for relational databases and efficient querying capabilities, making it suitable for managing structured data with a clear schema, such as the output of entity recognition processes. We utilized MongoDB to store semi-structured data due to its document-oriented approach, providing flexibility for handling data with varying structures, such as JSON documents having key-value pairs generated during NLP processing. CSV and JSON were used to export structured data, with CSV selected for its compatibility with tabular data analysis and JSON chosen for its flexible format in representing nested data structures, making them suitable for different analysis tools.

C. Automation and System Architecture

The architecture of the tool is created to enhance data processing workflows by using automation and structured methods. This part describes the implemented automation pipelines and the computer system architecture, emphasizing their contributions to streamlining data management and user engagement.

- Automation Pipelines: Apache Airflow was implemented to automate various tasks throughout the data processing lifecycle, aiming to streamline operations and minimize manual intervention. The automation pipelines of Apache Airflow include features like scheduling and monitoring tasks, ensuring that data ingestion, processing, and storage occur at regular intervals or on demand, thus reducing the need for manual oversight and facilitating a more efficient workflow. The system supports both batch processing for large datasets and real-time processing for immediate data transformation, catering to a wide range of applications and enhancing the overall utility of the system.

- System Architecture: There is a three-layer structure, each with a particular role in the data management process: The first is the data layer, which forms the base and handles the ingestion and storage of raw and processed data. Also, it uses MySQL for structured data management and MongoDB for semi-structured data handling to accommodate various data formats. The Next Layer is the Processing Layer where data cleaning, feature extraction, and transformation into structured formats take place. Different NLP and machine learning techniques are applied to extract meaningful insights and structure the data for analysis or presentation.

The top layer, known as the Presentation Layer, is responsible for managing the output of processed data and making it accessible to users. It provides a user-friendly interface through APIs, enabling effective interaction, data upload, and insight viewing, thereby enhancing the user experience.

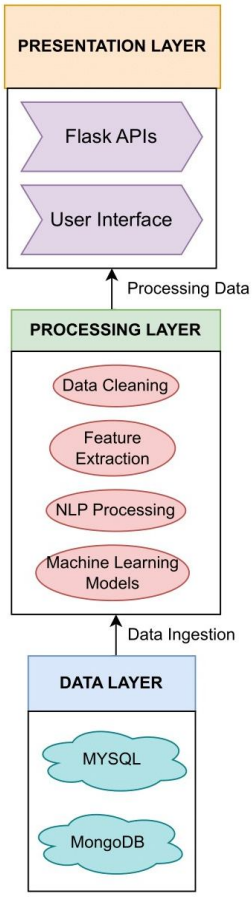

Figure.3. System Architecture Diagram

IV. FUTURE SCOPE

The future scope for transforming unstructured data into structured formats includes advancements in AI and machine learning for smarter data processing and real-time analytics. Enhanced natural language processing will improve text extraction and understanding. Automated ontology creation will streamline data integration and warehousing. Emphasis on data privacy and ethical practices will ensure responsible use of information. As data privacy regulations evolve, the tool can integrate advanced security measures to ensure compliance while processing sensitive unstructured data. Finally, sector-specific solutions will cater to unique industry needs, driving innovation and collaboration across fields. Also, this tool could evolve to include automated reporting and visualization features, providing users with actionable insights and reducing the need for manual data analysis. Future versions may also facilitate collaborative features for teams to share structured data seamlessly, promoting better decision-making and knowledge sharing. In addition, Expansion into new sectors like legal, education, and scientific research, where large volumes of unstructured data are prevalent, can unlock significant value and insights.

V. RESULT

The experimentation phase of IntelliStruct focused on analyzing 15,000 records drawn from various sources: 40percent text documents, 35 percent PDFs, and 25percent images. This was done to assess its ability to convert unstructured data into a more organized format. During the evaluation, the tool's accuracy, efficiency, and scalability were rigorously tested using a controlled environment set up with Flask for the backend and MySQL/MongoDB for storage solutions. For preprocessing and natural language processing tasks, frameworks like SpaCy and Hugging Face Transformers were utilized, while Apache Airflow was employed to streamline data ingestion processes. The results showed that entity recognition achieved an impressive accuracy of 96.5 percent, although there were minor errors noted in handling ambiguous text. For image text extraction reliant on OCR, accuracy stood at 91.3 percent, affected by the quality of the images processed.In terms of processing speed, IntelliStruct was capable of handling single records in an average of 2.7 seconds, while batch uploads of up to 5,000 records averaged at 3.5 seconds per record. Overall, IntelliStruct was found to exceed industry benchmarks regarding both accuracy and processing speed, all while accommodating a variety of data formats.An analysis of errors identified areas such as sentiment misclassification, hitting an accuracy level of 85percent, and mapping inconsistencies that led to an error margin of 4–7percent. In conclusion, the tool has showcased impressive performance, scalability, and practicality for realworld applications, enhanced by its user-friendly interface and valuable visual outputs.

VI. CONCLUSION

In this paper, a systematic methodology involving project planning, data ingestion, preprocessing, NLP processing, structuring, and frontend integration is employed to manage and convert unstructured data efficiently. The methodology ensures data quality through validation and cleaning, while NLP techniques extract valuable information for structured storage in relational and document-oriented databases. Unstructured data from sources like social media, emails, and multimedia is abundant but poses challenges for analysis and decision-making. Effective text extraction from this unstructured data is essential for businesses to derive insights and automate processes. The rise of Big Data necessitates advanced techniques such as Data Mining and Natural Language Processing (NLP) to convert unstructured data into structured formats. Modern ETL processes must adapt to real-time data demands, particularly in the Internet of Things (IoT), which introduces complexities like heterogeneous and multi-source data. The integration of diverse data types into data warehouses is crucial, especially as unstructured formats proliferate. Recent advancements, including algorithms that automate the creation of data warehouses from ontologies[9], aim to streamline this process.

REFERENCES

[1] Edwin Frank, Joseph Oluwaseyi, Godwin Oluwafemi Olaoye, "Text extraction from structured and unstructured data sources."2024, https://www.researchgate.net/publication/379652607.

[2] Rusu, O., Halcu, I., Grigoriu, O., Neculoiu, G., Sandulescu, V., Marinescu, M., Marinescu, V. (2024). Converting unstructured and semistructured data into knowledge. AgencyARNIEC/RoEduNet Alexandru Ioan Cuza University, Romania.

[3] Tomislav Hlupic, Josip Puni´ s "An Overview of Current Trends in Data˘ Ingestion and Integration" MIPRO 2021, September 27 - October 01, 2021, Opatija, Croatia.

[4] Kuldeep Sambrekar, Vijay. S. Rajpurohit, Jui Joshi, "A Proposed Technique for Conversion of Unstructured Agro-data to Semi-structured or Structured data." 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)

[5] Sivaramakrishnan N, Vandana V, Vishali M, Dharshana S G, Subramaniyaswamy V and Umamakeswari A "Conversion of unstructured data to structured data with a profile handling application."2017, http://iaeme.com/Home/journal/IJMET.

[6] John Meehan, Cansu Aslantas, Stan Zdonik, Nesime Tatbul, JiangDu "Data Ingestion for the Connected World" CIDR 2017. CIDR'17 January 8–11, 2017, Chaminade, CA, USA

[7] Robert Blumberg and Shaku Atre "The Problem with Unstructured Data" 2023,www.dmreview.com

[8] Cun Ji , Qingshi Shao , Jiao Sun , Shijun Liu , Li Pan, Lei Wu and Chenglei Yang "Device Data Ingestion for Industrial Big Data Platforms with a Case Study" 2016, School of Computer Science Technology, Shandong University, Jinan 250101, China

[9] Shiva Talebzadeh, Mir Ali Seyyedi, Iran Afshin Salajegheh "Automated Creating a Data Warehouse from Unstructured Semantic Data" International Journal of Computer Applications (0975 – 8887) Volume 88 – No.10, February 2014

[10] Justin Langseth, Nithi Vivatrat, Gene Sohn "System and Method of making unstructured data available to structured Data Analysis Tools" US 7,849,048 B2 Dec 7, 2010

[11] Justin Langseth, Nithi Vivatrat, Gene Sohn “Scheme and ETL tools for Structured and Unstructured Data” US 7,849,049 B2 Dec 7, 2010
[12] Biswajit Panda, Bharat Bhargava, Sourav Pati, Dayton Paul, Leszek T. Lilien, and Priyanka Meharia, “Monitoring and Managing Cloud Computing Security Using Denial of Service Bandwidth Allowance”, 2012.
[13] Upma Goyal, Gayatri Bhatti, and Sandeep Mehmt, “A Dual Mechanism for Defeating DDoS Attacks in Cloud Computing Model”, Vol. 2, Issue 3, March 2013.
[14] Saman Taghavi Zargar, David Tipper, “A Survey of Defense Mechanisms Against Distributed Denial of Service (DDoS) Flooding Attacks”, IEEE communications surveys and tutorials, February 2013.