J. Adv. Res. in Information Technology, Systems and Management
Volume 8 Issue 1 - 2024, Pg. No. 19-26
Peer Reviewed Journal

Research Article

Unveiling Insights in the Iot Sector: Harnessing Power Bi for Data Mining And Visualization

Gourav[1], Shairy[2]

[1,2]Assistant Professor in Computer Science PCTE Institute of Engineering & Technology, Punjab, India.

I N F O

Corresponding Author:
Gourav , Computer Science PCTE Institute of Engineering & Technology, Punjab, India.
E-mail Id:
gouravcse@pcte.edu.in
Orcid ID:
https://orcid.org/0009-0009-7227-088X
How to cite this article:
Gourav , Shairy. Unveiling Insights in the Iot Sector: Harnessing Power Bi for Data Mining And Visualization. *J Adv Res Info Tech Sys Mgmt* 2024; 8(1):19-26.

Date of Submission: 2024-03-24
Date of Acceptance: 2024-04-27

A B S T R A C T

This exploratory investigation delves into a tool utilized for data mining with robust data visualization capabilities. The tool has the potential to transform ordinary data into sophisticated representations and engaging reports. To achieve the most comprehensible, efficient, effective, and accurate information visualization, various methods of data gathering, storing, processing, analysis, and real-time visualization are essential. The article conducts a comparative exploration of data mining, Power BI, highlighting their features, benefits, drawbacks, and their application in visualizing data within the IoT industry. This analysis aims to empower organizations to optimize their data, enabling data-driven decision-making and gaining a competitive edge in today's data-centric economy. Fundamentally, this exploration aims to decode the complexities surrounding the utilization of data mining tools and Power BI for optimal data visualization within the IoT domain. A comprehensive understanding of their nuanced features and potential pitfalls empowers businesses to strategically harness these capabilities, positioning themselves as pioneers in leveraging data for innovation.

Keywords: Power Bi, Iot Industry, Data Mining, Data Visualization, Decision Making

Introduction

Data Visualization and Data Mining constitute processes employed to refine vast amounts of data into a clear format, facilitating visual representation through tools such as Power BI. In today's IoT industries, copious data is generated every minute and second through numerous sensors, posing challenges for analysis due to its diverse sources.[1]

As an integral part of analytics, various techniques, including interactive and dynamic visualization, play a crucial role in presenting data through plots, graphs, slicers, stacked column charts, histograms, bar charts, tables, matrices, and other visual contexts. This paper specifically focuses on interactive IoT data visualization using the Microsoft Power BI tool, a cutting-edge business intelligence and data visualization platform developed by Microsoft.

It empowers organizations and individuals to convert raw data into insightful reports and visually compelling dashboards.[2] At its core, Power BI ensures robust data connectivity, allowing users to link to a wide range of data sources, be it in cloud-based services, databases, spreadsheets, or other on-premises locations. Once connected, Power Query is employed to clean, transform, and model the data, ensuring it is analysis-ready.[12] Power BI's essence lies in its data visualization capabilities and data mining functionalities, enabling users to create interactive and dynamic visuals, explore data comprehensively, and

effortlessly identify trends, patterns, and outliers. The Data Analysis Expressions language further empowers users to perform complex calculations.[4] The cloud-based Power BI Service streamlines collaboration and sharing of reports, data, and dashboards across teams and organizations, enabling stakeholders to access real-time data insights and make data-driven decisions collaboratively.

In the realm of IoT, Power BI is utilized to extract valuable insights from data, sourced through APIs, Microsoft Azure, and diverse methods. With its mobile apps, Power BI ensures users can access their reports and dashboards on the go, staying informed and connected to critical business metrics at all times.

Methodology of Power BI

The instructional guide "Power BI for Data Analysis" adopts a systematic, step-by-step methodology for acquiring and applying Power BI to achieve effective data analysis.[3][4] The session commences with a comprehensive introduction to Power BI, exploring its functionalities and key features, outlined below and depicted in Figure 03:

- **Get Data:** In this phase, end-users acquire data from diverse sources such as files, databases, Microsoft Azure, online services, and others, as detailed in the data source specifications.
- **Fetch:** This step involves selecting and retrieving data either through browsing data locations or using queries, considering various types of data sources and connectors.[13]
- **Process:** Data is manipulated and edited during this stage using various operations as it is loaded into Power BI.
- **Analyse:** The data is scrutinized using filters, conditional queries, and other operations, including the addition of columns based on conditions and filtering out erroneous data.
- **Visualize:** Processed and analyzed data is presented[4] visually using tools such as Microsoft Power BI Visuals and Custom Visuals, encompassing plots, graphs, slicers, KPIs, etc. Subsequently, the report is published to Power BI services.
- **Editing:** This step involves finalizing the published report on the web, addressing any errors or making necessary adjustments,[6] such as resolving issues with filters or visuals. Once the editing process is complete, the report is ready for publication on the web.
- **Web:** As mentioned in the preceding step, this phase involves converting the report into a dashboard, making it shareable through URLs, websites, etc.

Process of Data Mining & Visualizations

Data preprocessing is a crucial step in the data analysis process, and Power BI provides various tools and features to help users clean, transform, and shape their data.[7] Here are some common data preprocessing techniques in Power BI:

Figure 1.Methodology of Power BI

Data Loading

- **Import Data:** Use the Power BI Desktop to import data from various sources such as Excel, SQL Server, SharePoint, web services, and more.
- **Direct Query:** Connect to a data source without importing the data, allowing for real-time analysis.

Data Cleaning

- **Remove Duplicates:** Eliminate duplicate rows from your dataset.
- **Fill or Replace Values:** Replace null or missing values with specific values or use algorithms to fill missing data.
- **Remove Columns:** Eliminate unnecessary columns from the dataset.
- **Remove Rows:** Exclude rows based on certain conditions or criteria.

Data Transformation

- **Merge Queries:** Combine data from multiple queries using various join types.[6]
- **Split Columns:** Divide a single column into multiple columns based on delimiters or patterns.
- **Format Data Types:** Change the data type of a column (e.g., from text to date).
- **Extract Data:** Extract relevant information from a column using text, date, or number functions.
- **Aggregate Data:** Summarize data using aggregation functions (e.g., sum, average, count).

Data Shaping

- **Pivot and Unpivot:** Reshape data by pivoting columns into rows or vice versa.
- **Group By:** Grouping the data is based on one or more columns and apply aggregation functions.
- **Sort and Filter:** Arrange data in a specific order and filter out irrelevant information.

Data Modeling

- **Define Relationships:** Establish relationships between different tables to create a unified data model.
- **Create Calculated Columns and Measures:** Add new columns or calculations to enhance the analysis.

Data Cleansing and Profiling

- Use the Query Editor to clean and profile data by identifying patterns, outliers, and inconsistencies.

Data Loading Optimization

- Optimize data loading by selecting only the required columns and filtering rows early in the transformation process.

Advanced Data Transformation

- Use Power Query M language for advanced data transformation scenarios that cannot be achieved through the user interface alone.[4]

Data Quality Monitoring

- Utilize tools like the Power BI Data Quality Indicators to assess the quality of your data and identify potential issues.

Data Refresh

- Set up automated data refresh to keep your Power BI reports up-to-date with the latest data from the source.

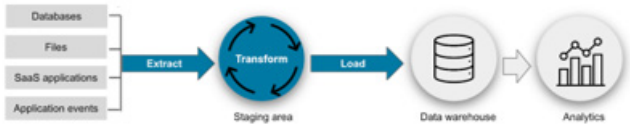

Figure 2.Process of Data Mining & Visualizations

- Power BI provides a user-friendly interface for performing these data preprocessing tasks, and the Power Query Editor is particularly powerful for data transformation and shaping.

Steps to implement Data Pre-Processing Query in Power BI Query Editor

In Power BI, data preprocessing queries are typically created and managed using the Power Query Editor. The Power Query Editor is a powerful tool that allows you to perform various data transformation and cleaning operations on your dataset before loading it into the Power BI model.[5] Here are some common data preprocessing queries you might perform in Power BI:

Removing Duplicates

- In Power Query Editor, select the column or columns where you want to remove duplicates.
- Go to the "Home" tab and click on "Remove duplicates."

Handling Missing Values

- To replace or fill missing values, select the column with missing values.
- Use the "Transform" tab to choose options like "Fill Down" or "Replace Values."

Filtering Rows

- Use the "Home" tab to filter rows based on specific conditions.
- You can use options like "Number Filter," "Text Filter," or custom filter conditions.

Transforming Data Types

- Choose the column for which you wish to modify the data type.[7]
- Use the "Transform" tab and choose the appropriate data type from the dropdown.

Splitting Columns

- Select the column you want to split.
- Use the "Transform" tab and choose "Split Column." Specify the delimiter or the number of characters.

Merging Queries

- If you have multiple queries and want to merge them, use the "Home" tab and click on "Merge Queries."

Grouping and Aggregating Data

- To group data based on specific columns, use the "Transform" tab and select "Group By."
- Specify the grouping columns and define aggregations.[10]

Pivoting and Unpivoting

- For pivoting or unpivoting columns, select the columns you want to transform.
- Use the "Transform" tab and choose either "Pivot Column" or "Unpivot Columns."

Sorting Data

- Select the column you want to sort.
- Use the "Sort Ascending" or "Sort Descending" options in the "Home" tab.

Adding Custom Columns

- Use the "Add Column" tab to add custom columns based on formulas or expressions.
- Advanced Data Transformations (M Language):
- For complex transformations,[9] you can use the Power Query M language by clicking on the "Advanced Editor."

Renaming Columns

- Right-click on a column header and select "Rename" to change the column name.

Applying Conditional Transformations

- Use the "Add Conditional Column" feature in the "Add Column" tab to create columns based on conditional logic.

Removing or Reordering Columns

- Right-click on a column header to remove or reorder columns.

Data Quality Checks

- Use the "View" tab to enable the "Data Quality" view, which helps identify potential data quality issues.

Remember that these are just some of the common operations, and the Power Query Editor provides a wide range of tools and functions to cater to[11] various data preprocessing needs in Power BI.

DAX Query in Power BI & Optimization Techniques

DAX, or Data Analysis Expressions, serves as a formula language within Power BI, enabling users to generate personalized calculations and perform aggregations on their data.DAX queries are expressions written in this language, allowing users to define calculated columns, measures, and calculated tables. These queries are essential for transforming raw data into meaningful insights within Power BI reports.

- **Formulas and Functions:** DAX queries consist of formulas and functions that operate on tables and columns. Functions like SUM, FILTER, and CALCULATE are commonly used to perform calculations and manipulations.
- **Row and Filter Context:** DAX operates in either row context or filter context. Row context is used for calculations at the row level, while filter context is applied when filters are set on the data.
- **Data Model Relationships:** DAX queries leverage relationships established in the Power BI data model. Understanding these relationships is crucial for creating effective queries that span multiple tables.

Optimizing Dax Queries is Essential for Improving Report Performance, Especially when Dealing with Large Datasets. Here are Key Optimization Techniques:

Use of Relationships Wisely

- Leverage relationships efficiently by understanding the cardinality and direction of relationships between tables.
- Avoid unnecessary relationships that might introduce ambiguity.

Minimize the Use of Calculated Columns

- While calculated columns can be useful, excessive use may impact performance.
- Prefer creating calculated measures instead, as they are only evaluated when used in a visualization.

Selective Use of DAX Functions

- Choose DAX functions wisely based on the specific requirements.
- Some functions, like EARLIER and FILTER, can be resource-intensive. Consider alternatives when possible.

Avoid Nested Iterations

- Nested iterations in DAX expressions can lead to performance degradation.
- Use vectorized operations and set-based logic to minimize the need for row-by-row iterations.

Partitioning and Segmentation

- Partition large datasets into manageable segments for analysis.
- Utilize techniques such as query folding to push some computations back to the data source.

Indexing and Sorting

- Sort data appropriately in your model to optimize the effectiveness of DAX queries.
- Create indexes on columns that are frequently used in filters or sorting.

Cache Utilization

- Leverage the Power BI cache to store intermediate results when possible.
- Reuse measures and calculated tables to avoid redundant computations.

Model Simplification

- Simplify your Power BI data model by removing unnecessary tables and columns.
- Keep only the columns and tables needed for the analysis to reduce the data model's complexity.

DAX Query Implementation and Analysis in Power BI

The FILTER function to filter the 's_data' table based on conditions related to temperature and humidity. The PERCENTILE.INC function is used to calculate the 1st and 99th percentiles for each group of devices.

- **Filter Query for temperature sensor**

```
FILTER(
    's_data',
```

```
  's_data'[temp] > PERCENTILE.INC('s_data'[temp], 0.01)
&&
  's_data'[temp] < PERCENTILE.INC('s_data'[temp], 0.99)
)
```

- **Filter Query for humidity sensor**

```
FILTER(
  's_data',
  's_data'[humidity] > PERCENTILE.INC('s_data'[humidity],
0.01) &&
  's_data'[humidity] < PERCENTILE.INC('s_data'[humidity],
0.99)
)
```

The FORMAT function to extract the date and time components from the 'ts' column. The "YYYY-MM-DD" format is used for the date, and "HH:mm:ss UTC" format is used for the time. Adjust the formats based on your specific requirements.

- **Format Query for date & Time**

```
Date = FORMAT('s_data'[ts], "YYYY-MM-DD")
Time = FORMAT('s_data'[ts], "HH:mm:ss UTC")
```

The EVALUATE statement, along with the SUMMARIZE function, to group by the 'device' column and calculate the average temperature for each device. The result is displayed with the 'device' column and the calculated 'average_temperature' column.

```
EVALUATE
SUMMARIZE(
  's_data',
  's_data'[device],
  "average_temperature", AVERAGE('s_data'[temp])
)
```

AVERAGE function to calculate the average temperature from the 'temp' column in the 's_data' table. You can create this measure in the Power BI Desktop interface by going to the "Modeling" tab, selecting "New Measure," and then entering the DAX expression.

- **Average Query**

```
AverageTemperature = AVERAGE('s_data'[temp])
```

The uses the EVALUATE statement, along with the SUMMARIZE and ADDCOLUMNS functions, to achieve the desired result. It first calculates the maximum temperature for each device in the 'MaxTempTable', and then filters the 's_data' table to select rows where the temperature matches the maximum temperature for each device. Finally, it selects the relevant columns and displays the result.

- **Maximum Temperature Calculation Query**

```
EVALUATE
VAR MaxTempTable =
  SUMMARIZE(
    's_data',
    's_data'[device],
    "max_temp", MAX('s_data'[temp])
  )
VAR ResultTable =
  ADDCOLUMNS(
    FILTER(
      's_data',
      's_data'[device] IN VALUES('s_data'[device]) &&
       's_data'[temp] = RELATED('MaxTempTable'[max_
temp])
    ),
    "max_date", MAX('s_data'[date]),
    "max_time", MAX('s_data'[time])
  )
SELECTCOLUMNS(
  ResultTable,
  's_data'[device],
  "max_date", [max_date],
  "max_time", [max_time],
  's_data'[temp]
)
```

Calculate the exponential moving average of the temperature for each device and fetch the device ID, timestamp, temperature, and EMA temperature for the devices in the table on the based on the worldwide installed sensors in the different countries based on their timestamp.

- **Moving Temperature calculation Query**

```
TopDeviceAverageTemperature =
CALCULATE (
  AVERAGE ( 's_data'[temp] ),
  ALLEXCEPT ( 's_data', 's_data'[device] ),
  VALUES ( 's_data'[device] )
)
```

Analyze sensor data for all countries, examining variations based on timestamp and date. Evaluate the average temperature across all temperature sensors to identify trends over time.

- **Average Temperature across all the sensors**

Date = FORMAT('s_data'[date] & ' ' & 's_data'[time], "YYYY-MM-DD")

Hour = FORMAT('s_data'[date] & ' ' & 's_data'[time], "HH")

AverageTemperature = AVERAGE('s_data'[temp])

The exploration of sensor data extends beyond temperature, encompassing humidity sensors and various other sensor types deployed across different countries. [9] The analysis follows a systematic approach, examining timestamp and date variations to discern patterns in the data. This comprehensive investigation aims to uncover trends and fluctuations specific to each sensor category, providing valuable insights into environmental conditions globally. Similar to the temperature sensors, the process involves creating calculated columns for date and hour, allowing for a detailed breakdown of sensor readings over time. By calculating the average humidity for all humidity sensors and applying analogous procedures for other sensor types, the analysis captures a holistic view of environmental parameters. The resulting data insights contribute to a better understanding of sensor behavior, facilitating informed decision-making in areas such as climate monitoring and resource management.

- **Humidity sensor DAX query Example**

```
EVALUATE
SUMMARIZE(
  's_data',
  's_data'[device],
  "highest_humidity", MAX('s_data'[humidity])
)
```

A Power BI telemetry dashboard for sensors offers real-time insights into sensor data, featuring dynamic visualizations for easy monitoring. Users can track sensor readings, detect trends, and address anomalies promptly. The interactive interface allows exploration of data across time and locations. With predictive analytics capabilities, the dashboard enables forecasting and proactive management of sensor behavior. This user-friendly tool empowers organizations to make informed decisions and optimize system performance based on sensor-generated insights.

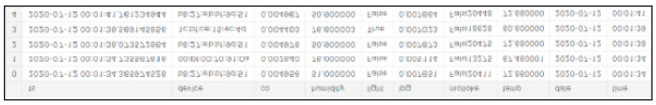

Figure 3.IOT Telemetry Sensors Dataset Sample

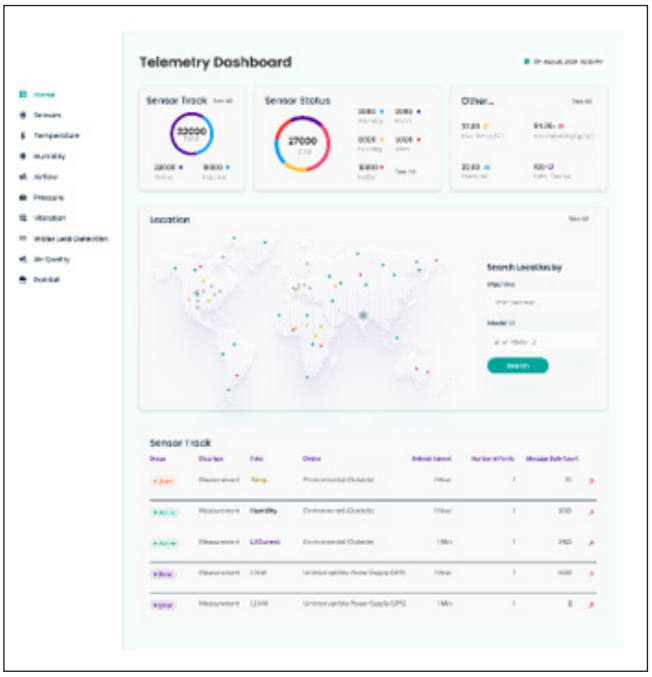

Figure 4.Power BI Telemetry Dashboard

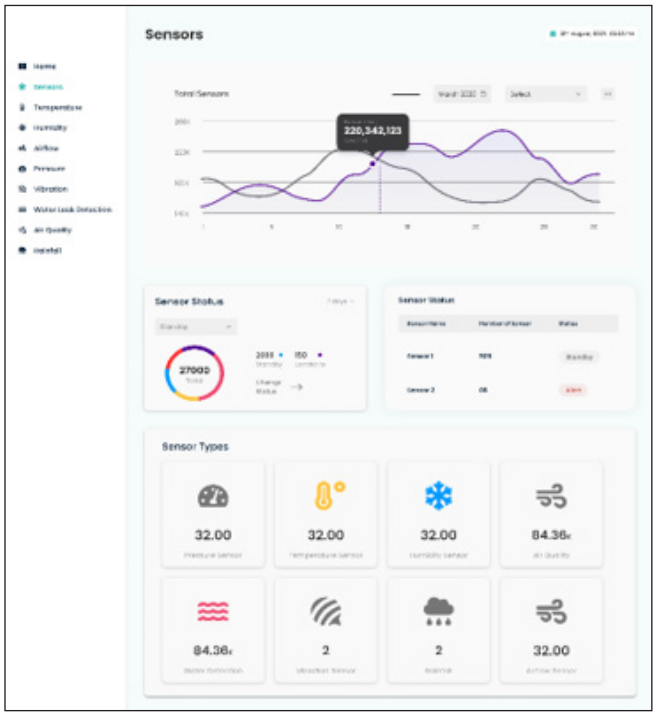

Figure 5.Telemetry Dashboard for Temperature & Humidity Sensors

Conclusion

In conclusion, this research paper has delved into the intricate realm of IoT data visualization and mining, focusing on the utilization of Microsoft Power BI as a powerful tool in this context. The exploration highlighted Power BI's robust data connectivity, processing capabilities, and its role in transforming raw data into insightful reports and visually compelling dashboards. The paper emphasized the significance of data visualization in the IoT industry, where copious amounts of data are generated from diverse sources.

Through a systematic methodology, the paper illustrated the step-by-step process of employing Power BI for effective data analysis, covering data acquisition, processing, analysis, visualization, and publication. The discussion also touched upon data mining techniques within Power BI, showcasing its capabilities in data preprocessing, shaping, modeling, and optimization. The comprehensive exploration of Power BI's functionalities provides organizations and individuals with a roadmap for harnessing this tool to derive valuable insights from IoT data.

This research contributes to the broader understanding of how Power BI can be strategically employed in the IoT sector for optimal data-driven decision-making. By decoding the complexities surrounding data mining tools and Power BI, businesses can position themselves as pioneers in leveraging data for innovation and gaining a competitive edge in today's data-centric economy.

Future Scope

The findings of this research lay the groundwork for several potential avenues for future exploration in the field of IoT data visualization and mining:

- **Advanced Power BI Features:** Investigate and analyze advanced features and functionalities of Power BI that may not have been covered in this research. Explore features like machine learning integration, advanced analytics, and the integration of emerging technologies.
- **Integration with Emerging Technologies:** Explore the integration of Power BI with emerging technologies such as artificial intelligence, edge computing, and blockchain to enhance the capabilities of data visualization and mining in the IoT domain.
- **Security and Privacy Concerns:** Delve deeper into the security and privacy aspects of utilizing Power BI in IoT applications. Examine potential vulnerabilities, data encryption methods, and privacy-preserving techniques to ensure secure handling of sensitive IoT data.
- **User Experience and Interface Design:** Investigate the user experience aspects of utilizing Power BI for data visualization in the IoT sector. Explore ways to enhance the user interface, interactivity, and accessibility for a diverse range of stakeholders.
- **Real-time Data Visualization:** Explore methods to enhance real-time data visualization capabilities within Power BI for IoT applications. Investigate how organizations can derive actionable insights from rapidly changing data streams in real-time.
- **Case Studies and Industry Applications:** Conduct in-depth case studies across various industries within the IoT sector to showcase successful implementations of Power BI for data visualization and mining. Analyze the specific challenges and benefits experienced by different organizations.

By pursuing these future research directions, scholars and practitioners can contribute to the ongoing evolution of IoT data visualization and mining, ensuring that organizations continue to leverage the full potential of tools like Power BI in an ever-changing technological landscape.

References

1. P. R. C. Patil, A. Desai, P. Jadhav, M. Jamadar, S. Koli, and S. Shirole, "Sales analysis using power bi," no. 04, pp. 1561–1563, 2022.
2. M. M. Abdulla, "Using Microsoft Power BI and Data Visualization Tools to Study How Online Shopping was Affected During COVID19 Period: A Comparison in Three Countries Between 2019 and 2020," pp. 3591–3600, 2023, doi: 10.46254/an12.20220651.
3. G. Shoaib and S. Nandi, "Power Bi Dashboard for Data Analysis," Int. Res. J. Eng. Technol., no. July, pp. 2881–2885, 2022.
4. Michael Hart, 2017, "Quick Insights with PowerBI", Accessed online at https://powerbi.microsoft.com/en-us/documentation/powerbi-service-auto-insights/
5. Microsoft Documentation, 2017, "What is PowerBI", Accessed online at https://powerbi.microsoft.com/en-us/what-is-power-bi/
6. Jignesh Shah," 8 Major benefits of Microsoft Power BI you must know",Saviant,July,2018.
7. Michael Diamond , Angela Mattia "Data visualization: an exploratory study into the software tools used by businesses", Journal of Instructional Pedagogies,Vol.18,2018,
8. Zion China uses Azure IoT Stream Analytics and Machine Learning to evolve its Intelligent Diabetes Management solution", June 2017.
9. Nagy J, Oláh J, Erdei E, Máté D, Popp J (2018) The role and impact of Industry 4.0 and the Internet of things on the business strategy of the value chain—the case of Hungary. Sustainability 10(10):3491
10. Ansari, M., Alam, M. (2023). Exploring the Role of Business Intelligence Systems in IoT-Cloud Environment. In: Chaudhary, K., Alam, M., Debnath, N.C. (eds) Intelligent Data Analytics in Business . ICDABM 2022. Lecture Notes in Electrical Engineering, vol 1068. Springer, Singapore.

11. Manzoor Ansari & Mansaf Alam,"Exploring the Role of Business Intelligence Systems in IoT-Cloud Environment", International Conference on Data Analytics in Business and Marketing, Conference paper, September 2023
12. holkapiyan, M., Ramadass, S., Seetha, J., Ravuri, A., Vidyullatha, P., Siva Shankar, S., & Gore Santosh(2023). Examining the Impacts of Climate Variability on Agricultural Phenology: A Comprehensive Approach Integrating Geoinformatics, Satellite Agrometeorology, and Artificial Intelligence.International Journal of Intelligent Systems and Applications in Engineering,11(6s),592598.
13. Gore Santosh, Dhindsa, G., Sujata Gore, Jagtap, N. S., & Nanavare, U. (2023, July). Recommendation of Contemporary Fashion Trends via AI-Enhanced Multimodal Search Engine and Blockchain Integration. In 2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC) (pp. 1676-1682). IEEE.