EX-DSS: An Explorative Decision Support System for Designing and Deploying Smart Plug Forecasting Pipelines

Giulia Rinaldi, Lola Botman, Oscar Mauricio Agudelo and Bart De Moor
KU Leuven, Department of Electrical Engineering (ESAT), STADIUS Center for Dynamical Systems, Signal Processing and Data Analytics, Kasteelpark Arenberg 10, 3001 Leuven, Belgium

Keywords: Decision Support System, Smart Plug Forecasting, Artificial Intelligence Pipeline.

Abstract: Artificial Intelligence pipelines are increasingly used to address specific challenges, such as forecasting smart plug loads. Smart plugs, which remotely control various appliances, can significantly reduce energy consumption in commercial buildings by about 20% when effectively scheduled using AI techniques. Designing these AI pipelines involves numerous steps and variables, requiring collaboration and shared knowledge among designers. A Decision Support System (DSS) can facilitate this process. This paper introduces the Explorative Decision Support System (EX-DSS), which extends the classical DSS framework. The EX-DSS integrates an Explorative Management Subsystem to provide project-specific recommendations and a Data Quality (DQ) module to validate user inputs, ensuring clarity and enhancing information sharing. The EX-DSS architecture framework was tested through a software prototype designed to create AI pipelines for forecasting smart plug loads. The study found that using the EX-DSS improves the quality of suggestions, making them more problem-specific and resulting in a more personalized and meaningful user experience, with a significant potential to reduce energy consumption in commercial buildings.

1 INTRODUCTION

Designing an Artificial Intelligence (AI) pipeline involves multiple steps, from data cleaning to implementing machine learning methods. This process can be complex and time-consuming, especially when trying to find the most efficient combination. In the context of smart plug forecasting, this challenge is significant, as plug loads account for over 40% of total energy consumption in commercial buildings, excluding lighting, HVAC, and water heating (Chia et al., 2023). Smart plugs, which remotely monitor and control electrical appliances, can save up to 20% of electricity through effective scheduling, making AI forecasting and scheduling methods promising solutions (Botman et al., 2024).

A Decision Support System (DSS) can expedite the design of these AI pipelines. A DSS is a flexible software tool that assists in decision-making processes and allows for shareability and reproducibility among users, necessitating collaboration functionalities and ways to evaluate user input.

This paper introduces an experimental Explorative Decision Support System (EX-DSS) aimed at designing and deploying industrial smart plug pipelines to solve forecasting problems. The novel contributions of this research include:

- Extending the conventional DSS with the Explorative Management Subsystem, offering project-specific insights and recommendations.
- Integrating a Data Quality (DQ) module to assess the quality of user inputs, ensuring reliable AI pipeline design.
- Maintaining a human-in-the-loop approach, giving users maximal control over every feature in the EX-DSS.
- Providing a practical demonstration of the EX-DSS through a software prototype for smart plug forecasting, validating the system's capabilities and showcasing its effectiveness in real-world applications, thereby highlighting its potential impact on reducing energy consumption in commercial buildings.

This paper is structured as follows. Section 2 presents related works. In Section 3, the EX-DSS architecture framework is introduced. Section 4 describes the application of this framework in the design of smart plug forecasting AI pipelines via a software prototype. In Section 5, results are discussed, and fi-

Rinaldi, G., Botman, L., Agudelo, O. and De Moor, B.
EX-DSS: An Explorative Decision Support System for Designing and Deploying Smart Plug Forecasting Pipelines.
DOI: 10.5220/0012925800003838

In *Proceedings of the 16th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2024) - Volume 3: KMIS*, pages 197-205
ISBN: 978-989-758-716-0; ISSN: 2184-3228

nally, in Section 6, conclusions are drawn, and future research lines are presented.

2 BACKGROUND

A Decision Support System (DSS) is software that helps users analyze data and make decisions. It generates insights and suggestions through a structured framework (Gonzalez-Andujar, 2020). The classical DSS includes a **Data Management Subsystem**, which stores and handles data; a **Model Management Subsystem**, which manages models for DSS tasks; a **Knowledge Management Subsystem**, which provides information to users; and a **User Interface**, which connects users with DSS subsystems (Turban et al., 2010).

DSS supports semi-structured or unstructured decisions, requiring evaluations beyond mathematical modeling (Duan and Xu, 2009). For example, buying a new tool (Jacquet-Lagreze and Shakun, 1984) or deciding on stock market entry/exit (Chandra et al., 2007). These systems use rule-based models and AI to predict scenarios and derive insights (Phillips-Wren, 2013).

DSS applications are used in various fields like business, healthcare, and agriculture. They can help medical staff plan and monitor medications (Sloane and J. Silva, 2020) and manage production costs in agriculture (Rupnik et al., 2019). Tools like Rapidminer[1] and Knime[2] support the design of AI pipelines but lack assistance during the configuration phases.

The problem considered in the EX-DSS software prototype involves forecasting the load of electrical appliances, monitored and controlled by smart plugs to optimize energy consumption (Chia et al., 2023). Smart plugs can significantly reduce energy usage in commercial buildings by scheduling appliance operations. Accurate load forecasting is crucial for creating effective schedules. Current methods, including time series analysis and AI-based techniques like neural networks and ensemble methods (Botman et al., 2024), are time-consuming and complex to implement. This underscores the need for a robust DSS framework like EX-DSS to streamline the process and improve accuracy.

This paper introduces a knowledge-driven DSS framework that incorporates detailed domain knowledge. This makes it especially useful for complex decision-making processes that require specific expertise. Specifically tailored for time series forecasting, this framework focuses on input quality assessment and fosters a collaborative research community by enabling users to share insights, models, and results.

[1]https://altair.com/altair-rapidminer

[2]https://www.knime.com/

3 EX-DSS FRAMEWORK

As mentioned in Section 2, a traditional Decision Support System (DSS) has three subsystems and a User Interface (UI). The new Explorative Decision Support System (EX-DSS) adds a Data Quality (DQ) Module (Fig.1C) and an Explorative Management Subsystem (Fig.1D.4). The DQ Module (Fig.1C) improves data quality and promotes information sharing. The Explorative Management Subsystem (Fig.1D.4) provides project-specific insights for better analysis.

The EX-DSS architecture framework has two levels: the User (Fig.1A) and the EX-DSS (Fig.1B). The User interacts with the system to solve problems using semi-automated steps. The EX-DSS analyzes problems, provides extra information, and helps create the AI pipeline. A Graphical User Interface (Fig.1E) facilitates these interactions.

The EX-DSS has two main parts: the DQ Module (Fig.1C) and the Pipeline Design Module (Fig.1D). The DQ Module includes the Intake Phase (Fig.1C.1) for uploading data and the Assessment Phase (Fig.1C.2) for evaluating data quality by generating a report. The Pipeline Design Module (Fig.1D) has four subsystems: Data Management (Fig.1D.1), which stores essential information; Model Management (Fig.1D.2), which manages the pipeline structure (Fig.1D.2.a), configuration (Fig.1D.2.b) and training (Fig.1D.2.c); Knowledge Management (Fig.1D.3), which provides general information; and Explorative Management (Fig.1D.4), which derives project-specific insights from the keywords extracted from the DQ report (Fig.1D.4.a).

The general knowledge in the Knowledge Management Subsystem (Fig.1D.3) is not tied to specific problems. For example, it answers questions like “What is forecasting?”. Project-specific knowledge in the Explorative Management Subsystem (Fig.1D.4) is detailed and tailored to specific projects, like “What are the best methods for forecasting smart plugs?”. The DSS is called “explorative” because it helps users dive into and explore specific problems, such as forecasting smart plugs for energy savings in buildings.

By adding an extra subsystem, the EX-DSS improves modularity, maintainability, and reusability. It separates general knowledge from project-specific information, allowing better customization and system performance.

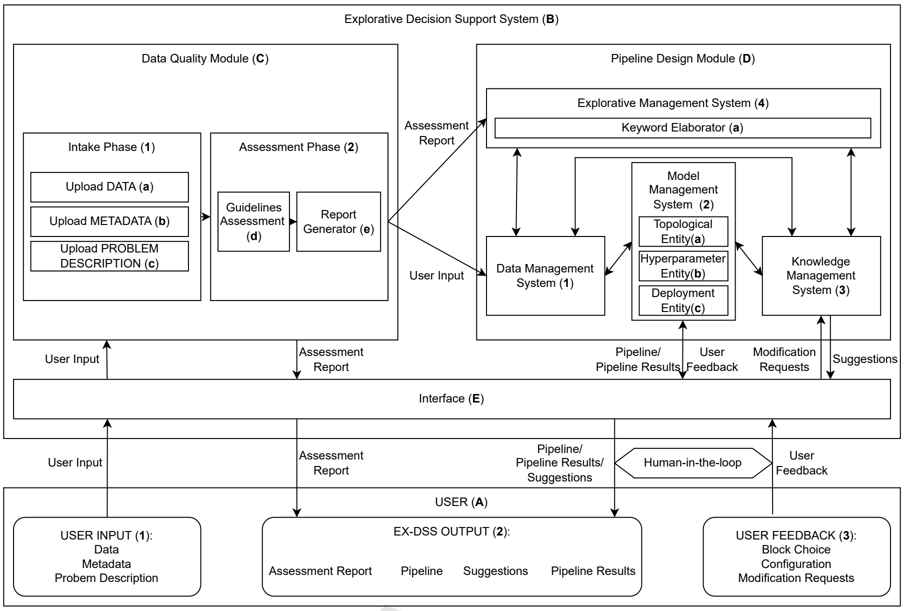


Figure 1: **EX-DSS architecture framework.** This illustrates the proposed EX-DSS framework. (A) User-level input and outputs. (B) Blocks composing the EX-DSS: (C) Block to assess the quality of the user's input and (D) Block to aid the users in designing, configuring, and deploying the pipeline. (E) User Interface that allows the user to dialogue with the EX-DSS. The arrows represent the flow of information inside the systems.

4 EX-DSS FOR SMART PLUG FORECASTING PIPELINES

This Section presents the application of the Explorative Decision Support System (EX-DSS) framework in the design of a smart plug forecasting AI pipeline via a software prototype. Section 4.1 describes the use case and the context to implement such EX-DSS, while Section 4.2 discusses in detail the EX-DSS implementation.

4.1 The Context

The EX-DSS framework described in Section 3 addresses the design of Smart Plug Forecasting pipelines. Smart plugs are devices that fit between appliances' power cords and wall sockets, enabling remote control. Smart plugs convert standard appliances into smart ones. Various appliances like printers, copiers, and TVs are monitored. Studies show that plug loads (excluding lighting, HVAC, and water heating) account for over 40% of total energy consumption in commercial buildings, and that this consumption is increasing over time (Chia et al., 2023; Tuttle et al., 2020). Smart plugs can significantly reduce energy consumption by automatically determining if a plug is idle or active and forecasting its usage. This allows scheduling devices to turn on or off accordingly, leading to potential electricity savings of up to 20%, optimizing building energy efficiency.

AI is crucial for determining plug usage, forecasting consumption, and scheduling devices (Botman et al., 2024). However, finding the best-performing method is time-consuming. The software prototype proposed in this paper aims to guide researchers in designing pipelines for smart plug usage forecasting and scheduling, thereby optimizing building energy consumption. The trained model from the design phase can also be downloaded for further use.

4.2 Description of the Software Prototype

The EX-DSS software prototype is a web application written in Python, implemented using the Dash framework[3]. It assists users by providing descriptions and suggestions for configuring each block of the AI pipeline, based on issue-specific keywords for the Smart Plug problem described in Section 4.1 and listed in the Appendix. Additionally, a chatbox allows users to ask general questions. It is integrated with the Natural Language Processing (NLP) model developed by Cohere[4].

[3]https://dash.plotly.com/
[4]https://cohere.com/

The functionalities of the EX-DSS software prototype are illustrated through an experiment using methods and datasets collected by Botman et al. (2024). While Botman et al. (2024) proposed many alternatives[5], a subset of these methods is included in this prototype. After configuring the pipeline, the EX-DSS connects to an external server for workflow management using the Airflow platform[6].

4.2.1 New Dataset

The EX-DSS software prototype's experiment began with uploading a new dataset (Fig.1C.1). The user provided a title, data source, and a short description (Fig.1C.1.b) to facilitate understanding and improve support in subsequent phases. Next, the user selected keywords from a set proposed by the EX-DSS (see Appendix for the full list). These keywords are crucial for the Explorative Management Subsystem to generate insights and target specific pipeline characteristics. Finally, the user submitted the main data file (Fig.1C.1.a) and could also upload supplementary files such as metadata or additional information (Fig.1C.1.b).

For this experiment, the input was based on the work proposed by Botman et al. (2024)::

- **Title:** Smart Plug Data,
- **Data Source:** "https://gitlab.esat.kuleuven.be/Lola.Botman/smart-plug-pipeline/-/tree/main/Dataset"
- **Data Description:** "The dataset is collected through smart plug sockets between the wall plugs and the electric appliance as detailed in Chia et al. (2023). Smart plugs from Best Energy Reduction Technologies (BERT) are deployed in fifteen buildings on the campus of the University of California, San Diego (UCSD). The power level of each appliance is recorded in mW at fifteen-minute intervals. The dataset consists of 169 high-quality smart plug time series spanning 498 days, from November 18^{th}, 2021, to March 31^{st}, 2023. The 169 plug loads include 146 printers, 16 copiers, 4 TVs, and 3 fax machines. This dataset is openly accessible; see Botman et al. (2024) for more details."
- **Keywords Selected:** "Smart Plugs, High Performance, Non-intrusive"
- **Main File:** "SmartPlugPreprocessedData.csv"
- **Additional Metadata Files:** ["SmartPlugMetaData.csv" "SmartPlugHolidayData.csv"]

[5] https://gitlab.esat.kuleuven.be/Lola.Botman/smart-plug-pipeline

[6] https://airflow.apache.org/

Table 1 shows a snapshot of the main dataset, "SmartPlugPreprocessedData.csv." Rows indicate the time at which power values are recorded, and columns represent monitored electrical appliances with power values in mW. The goal is to predict these power values. In the illustrated subset, recordings start at 11:45 on the 18^{th} of November 2021 until 17:00 on the 31^{st}, 2023, capturing the power load of 169 smart plugs at fifteen-minute intervals.

4.2.2 Assessment Report

The Assessment Phase (Fig.1C.2) begins once the system receives the data and the complementary input. In this phase, the Data Quality module in EX-DSS runs an internal analysis to assess the quality of the inputs used in subsequent steps to generate suggestions based on the work by Rinaldi. et al. (2023).

Although the Assessment Phase can be set up to run different guidelines, for the software prototype, three guidelines were chosen (Fig.1C.2.d):

- **FAIR Paradigms** (Wilkinson et al., 2016): The system checks dataset uniqueness and corrects saving (findability), attempts to open the dataset in a pandas dataframe[7] (reusability), evaluates data encoding (interoperability), and audited security rules (accessibility).
- **Data Quality Analysis:** This included checking for missing values (completeness), assessing value types (consistency), measuring distance between input and reference datasets (accuracy), and verifying if the data is up-to-date (timeliness).
- **Data Cleaning Automation:** This involved routines like evaluating missing values, identifying columns with a single value, and checking for row duplication.

The clarity of the dataset description is also evaluated. At the end of this phase, the system provides two scores: one for data quality analysis and another for data cleaning assessment, as shown in Rinaldi. et al. (2023).

Although this analysis is done in the background, users could configure it manually. They can choose whether to analyze only the main file or include metadata files, select the weight of each quality metric, and decide if any analysis should be skipped. All this information is compiled into a report (Fig.1C.2.b), which users can download.

Listing 1 shows a summary of the report that contains the analysis with the above-mentioned guidelines.

[7] https://pandas.pydata.org/

Table 1: **Smart Plugs Preprocessed Dataset snapshot.** The dataset contains power loads of 169 smart plugs.

Timestamp	Plug_0	Plug_1	Plug_2	...	Plug_167	Plug_168
2021-11-18 11:45:00	NaN	NaN	NaN	...	NaN	NaN
...	...	...	...	...	...	...
2021-11-18 13:15:00	NaN	10.413	19.656	...	NaN	NaN
...	...	...	...	...	...	...
2023-03-31 17:00:00	55.809	10.03	19.44	...	2.64	4.176

Listing 1: Extract of the JSON report containing the findings of the DSS after analyzing the information uploaded by the user.

```
{"SCORES" {
    "Data Quality": 0.56,
    "Data Cleaning": 0.99},
"FAIR" {
    "Reusability": "The dataset was
        rightly converted to a
        dataframe."
    "Data_description_clarity_score":
        "Input text clarity score:
        85%", ...},
"DATA QUALITY" {
    "Timeliness": "The dataset is not
        up to date -> parameter is
        0. The user set up the
        threshold to 0 years." ...},
"DATA CLEANING" {
    "Time Column": "Identified."
    "Single Columns": "There are no
        columns with only one value."
        ...}
```

4.2.3 New Project

The user needed to define the project's title and description, upload any additional documents that could help understand the project's final goal, and select the most pertinent keywords from a subset proposed by the EX-DSS (see Appendix for the full keywords list). These keywords assist in formulating proper suggestions and clarifying the research intention. An example of a new project record is as follows:

- **Title:** Smart Plug Project,
- **Project's Description:** "This project implements smart plug active operating mode detection, plug-level load forecasting, and a plug scheduling methodology. A pipeline integrates the detected operating modes with forecasting and scheduling, aiming at reducing building energy consumption (Botman et al., 2024)"
- **Additional Documents:** "No additional description documents"
- **Keywords Selected:** "High Performance, Non-intrusive"

4.2.4 Pipeline Configuration and the EX-DSS Guidance

When a project is initialized, the user can start the design and configuration of the pipeline. Figure 2 shows a screenshot of the configuration page of the EX-DSS prototype software. Figure 2a lists the available blocks developed in the EX-DSS software prototype, while Figure 2b displays the suggestions generated by the Knowledge Management Subsystem with support from the Explorative Management Subsystem.

A suggestion consists of a block's description, insight generated by the Explorative Management for the specific project, and a chatbox for communication with the Knowledge Management Subsystem. The Explorative Management Subsystem (Fig.1D.4) uses the project's keywords to find the closest matching dataset and generate suggestions based on stored expert knowledge. These suggestions include a list of blocks and methods, such as the appropriate forecasting method or technique to preprocess the dataset. This modular approach provides accurate and specialized knowledge, which can be modified to enhance specific subject knowledge or reused for other purposes.

The Knowledge Management Subsystem (Fig.1D.3) maintains a general knowledge view. In the EX-DSS software prototype, it uses a Generative AI model developed by Cohere[8]. The system connects to the Cohere platform via an API, sends a request with a question, and displays the AI-generated response to the user. Additionally, it ensures the correct interconnections between blocks, such as preventing the "Forecast" block from being followed by the "Dataset" block.

Additionally, the Knowledge Management Subsystem ensures that the interconnections between blocks are correct. For example, in the EX-DSS software prototype, the "Forecast" block could not be followed by the "Dataset" block. Figure 2c represents the configuration area, where users choose methods and configure parameters. For example, in the "Forecast" block, users can select the forecasting method and the prediction horizon. In the example shown

[8]https://cohere.com/

(Fig. 2c), “Global XGBoost” was chosen with a prediction horizon of “1” day.

Figure 2d shows the area where the user can organize and connect the chosen blocks. In the example, the tasks were organized as follows: “Dataset” for selecting the dataset, “Operating Mode” for determining the appliance’s operating mode, “Forecast” for predicting the appliance loads, “Schedule” for planning when the appliances should be turned on or off, and “Evaluate” for assessing the performance of the other blocks. “Data Cleaning” and “Preprocessing” were excluded since the dataset was already preprocessed

Once the configuration for each block is saved, the pipeline is ready for training.

4.2.5 Pipeline Training and Results

Once the pipeline is ready, the user initiates the Model Management Subsystem (Fig.1B.2), specifically the Deployment Entity (Fig.1B.2.C), to launch the deployment process. In the EX-DSS software prototype, this was implemented by an external server running Airflow[9], a platform designed for executing and monitoring a chain of tasks. The Deployment Entity transferred the necessary files to the Airflow server. Figure 3 displays the pipeline’s runtime, which took 9 minutes and 24 seconds to train and generate results.

Upon completion, the Airflow server communicates the results back to the EX-DSS, and users can download these results through the EX-DSS interface. Additionally, users can download the forecasting model, operating mode model, predictions, and final schedule. Table 2 provides an example of the final schedule. Additionally, Table 3 displays the numerical values of the metrics used to assess smart plug scheduling following the application of the “Global XGBoost” method.

Listing 2: Example of static suggestion stored inside the EX-DSS software prototype.

```
{"Pipeline Steps"  {
   "Data Cleaning"  "Imputation"
   "Data Cleaning"  "Normalization"
        ...},
"Operating Mode Detection"  {
   "Method"  "GMM"
   "Method"  "Ensemble"
"Forecasting"  {
   "Method"  "Global XGBoost"
   "Method"  "Global FNN"}
```

[9] https://airflow.apache.org/

Table 2: **Schedule Overview.** The table provides an example of the schedule for plugs 0 and 168. Both plugs are off at night and turn on in the morning, with plug 0 turning on earlier. Plug 0 turns off around lunch while plug 168 turns off later in the afternoon. The schedule can be summarized with the turn-on and turn-off times: Plug 0: {2023-01-13 05:15:00: ”Turn on”; 2023-01-13 12:30:00: ”Turn off”}, Plug 168: {2023-01-13 09:30:00: ”Turn on”; 2023-01-13 16:45:00: ”Turn off”}

Timestamp	Plug_0	...	Plug_168
2023-01-12 00:00:00	OFF	...	OFF
...	...	...	...
2023-01-13 05:15:00	ON	...	OFF
...	...	...	...
2023-01-13 09:30:00	ON	...	ON
...	...	...	...
2023-01-13 12:30:00	OFF	...	ON
...	...	...	...
2023-01-13 16:45:00	OFF	...	OFF
...	...	...	...

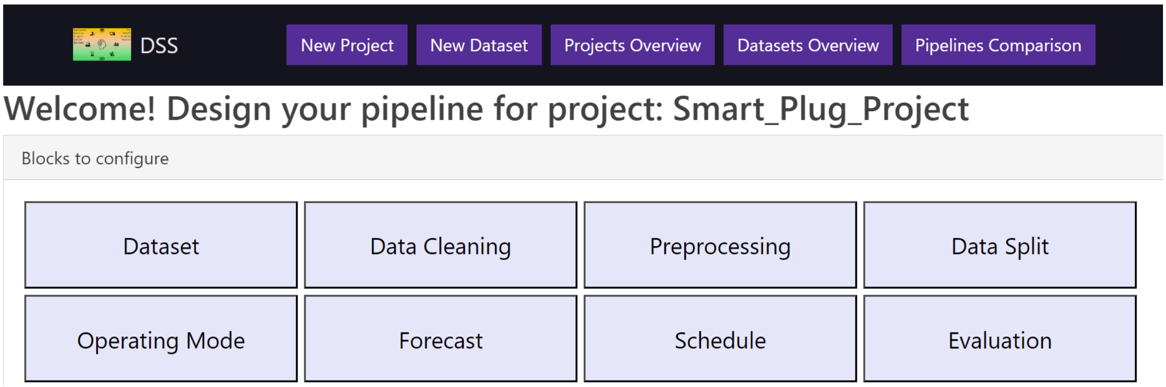
Table 3: **Forecasting Evaluation.** The table illustrates how the chosen forecasting method performed in the pipeline.

	Global XGBoost Schedule
Number of violations (%)	3.27
Missed chances (%)	27.67
Energy saved (%)	25.84
Number of turn on/off commands per plug per day	3.33
Energy Efficiency (%)	47.92

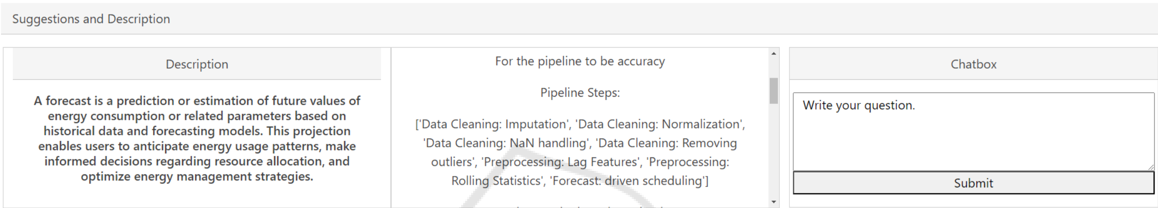
5 RESULTS AND DISCUSSION

The study’s findings indicate that the Explorative Decision Support System (EX-DSS) architecture framework enhances the design and implementation of DSS for designing smart plug pipelines, optimizing data forecasting, and, consequently, aiding in reducing energy consumption. By extending the classical Decision Support System (DSS) framework, the EX-DSS incorporates a module for assessing user input quality, facilitating collaboration among users, and promoting reproducibility. This prevents the repetition of previous errors and accelerates the pipeline creation process.

The EX-DSS allows users to upload new datasets, along with descriptions and additional documents, providing a comprehensive overview. The system internally evaluates these inputs and generates a report

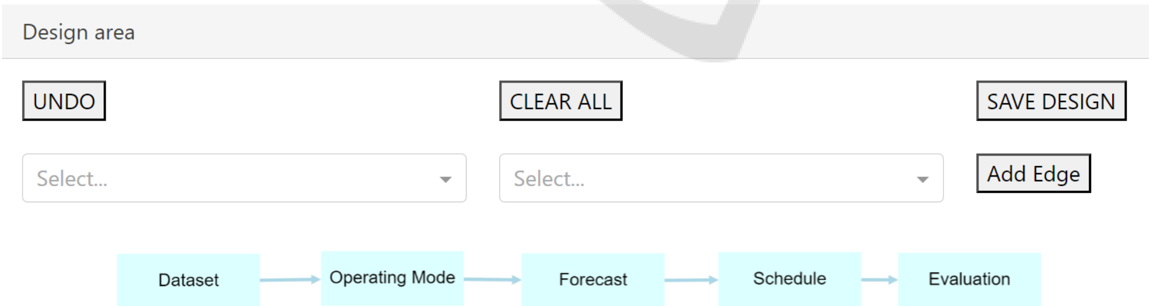


(a)



(b)

(c)



(d)

Figure 2: **Configuration page screenshots.** They were taken from the EX-DSS software prototype. (a) The list of the blocks developed inside the EX-DSS software prototype. (b) The structure of the suggestions provided to the user by the system. (c) This part of the page is dedicated to configuring the blocks. In the image, it is possible to visualize the configuration for the Forecast block. (d) This is the design area where the user can add and connect the blocks.

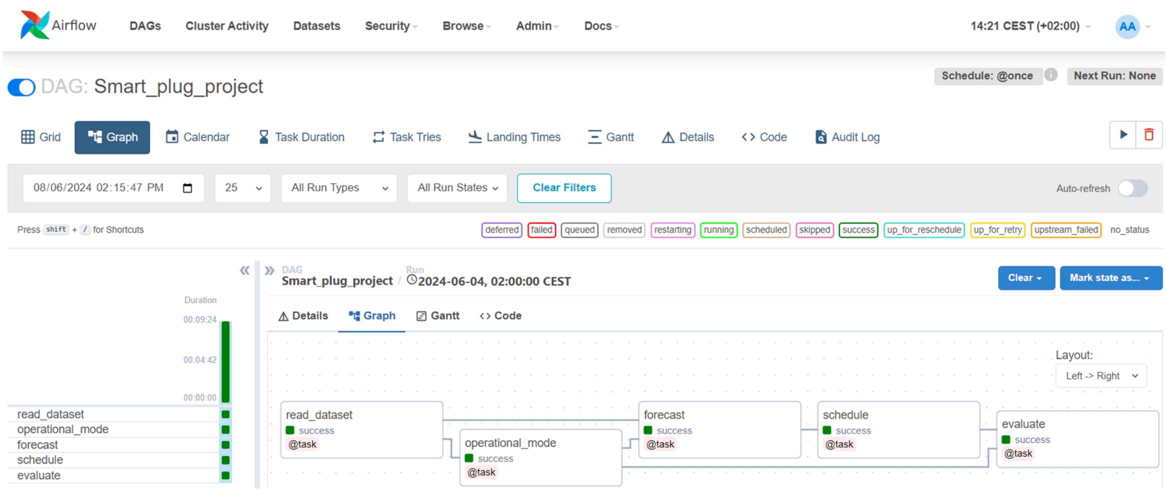


Figure 3: **Screenshot of the Airflow server.** The image shows the service used by the EX-DSS to run the pipeline.

on data quality and description clarity. By maintaining the essential human-in-the-loop characteristic of a DSS, users retain control over the types of analyses the system should perform, ensuring that the information received is of high quality and increasing the likelihood of high-performance outcomes from the deployed pipeline.

A key innovation in the EX-DSS is the inclusion of the Explorative Management Subsystem, which supports the Knowledge Management Subsystem by analyzing user inputs and providing relevant insights. By using a system of keywords tailored to the smart plug forecasting problem, the EX-DSS offers specialized support while maintaining general intelligence within the Knowledge Management Subsystem, thanks to the integration of the Cohere generative AI model. This dual support (general and specific) significantly aids users in designing and configuring forecasting pipelines, allowing for customization and catering to various needs and requirements.

The study has shown that the pipeline design, configuration, and deployment process can be carried out without coding, making it accessible to a broader range of users. However, future work should include extensive testing with diverse datasets to further validate the system's robustness and flexibility. Additionally, user feedback should be collected to refine the interface and improve the overall user experience. Integrating real-time data processing and adaptive learning capabilities could also enhance the EX-DSS, making it more responsive to changing conditions and user needs.

6 CONCLUSIONS

The Explorative Decision Support System (EX-DSS) framework, implemented through a software prototype, has proven effective in designing pipelines for scheduling smart plugs to reduce building energy consumption. By extending the classical DSS with the Explorative Management Subsystem, the EX-DSS provides specialized suggestions alongside general support, enhancing the design process.

The introduction of the data quality module has enabled the EX-DSS to assess the quality of input information, providing users with a clear overview of data status and ensuring higher performance when training models. The system also promotes the shareability of pipeline information and dataset knowledge by requiring users to describe new projects and datasets, thus promoting reproducibility and avoiding duplication.

Despite its benefits, the study faced limitations. The software is not production-ready and was developed for demonstration purposes only. Additionally, the study included only the methods proposed by Botman et al. (2024), and the development relied on input from a single expert due to time constraints and limited availability of specific profiles.

Future work should focus on investigating the interaction between the Knowledge Management Subsystem and the Explorative Management Subsystem to optimize their integration. Extending the software prototype to include result visualization directly within the EX-DSS would improve the user experience. Additionally, developing functionality to compare the results of similar pipelines would help identify the best-performing ones. Transitioning the software from a prototype to a production-ready system is essential, as is conducting extensive testing with diverse datasets and gathering user feedback to refine the interface and enhance overall usability. These steps will help validate the EX-DSS's robustness, improve its flexibility, and ensure it meets the needs of various users.

ACKNOWLEDGEMENTS

This research received funding from • KU Leuven: Research Fund (projects iBOF/23/064, C3/20/117, C3I-21-00316), Industrial Research Fund and several Leuven Research and Development bilateral industrial projects; • Flemish Government Agencies: ○ FWO: SBO project S005319N, TBM Project T001919N; FWO PhD Grant 11K5623N; ○ EWI: the Flanders AI Research Program; ○ VLAIO: CSBO (HBC.2021.0076) Baekeland PhD (HBC.20192204); • EU: This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 885682). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or ERC. Neither the European Union nor the granting authority can be held responsible for them.

REFERENCES

Botman, L., Lago, J., Fu, X., Chia, K., Wolf, J., Kleissl, J., and De Moor, B. (2024). Building plug load mode detection, forecasting and scheduling. *Applied Energy*.

Chandra, A., Krovi, R., and Rajagopalan, B. (2007). Risk visualization: A mechanism for supporting unstructured decision making processes. *The International Journal of Applied Management and Technology*.

Chia, K., Lebar, A., Agarwal, V., Lee, M., Ikedo, J., Wolf, J., Trenbath, K., and Kleissl, J. (2023). Integration of a Smart Outlet-Based Plug Load Management System with a Building Automation System. *2023 IEEE PES Grid Edge Technologies Conference and Exposition, Grid Edge 2023*.

Duan, Y. and Xu, M. (2009). Decision support systems in small businesses. *IGI Global - Business and Information Systems Research Centre (BISC)*.

Gonzalez-Andujar, J. (2020). *Introduction to Decision Support Systems*, pages 25–38.

Jacquet-Lagreze, E. and Shakun, M. F. (1984). Decision support systems for semi-structured buying decisions. *European Journal of Operational Research*.

Phillips-Wren, G. (2013). *Intelligent Decision Support Systems*.

Rinaldi., G., Crema Garcia., F., Agudelo., O. M., Becker., T., Vanthournout., K., Mestdagh., W., and De Moor., B. (2023). A framework for a data quality module in decision support systems: An application with smart grid time series. In *Proceedings of the 25th International Conference on Enterprise Information Systems - Volume 1: ICEIS*. INSTICC, SciTePress.

Rupnik, R., Kukar, M., Vračar, P., Košir, D., Pevec, D., and Bosnić, Z. (2019). Agrodss: A decision support system for agriculture and farming. *Computers and Electronics in Agriculture*.

Sloane, E. B. and J. Silva, R. (2020). Artificial intelligence in medical devices and clinical decision support systems. In Iadanza, E., editor, *Clinical Engineering Handbook (Second Edition)*. Academic Press, second edition edition.

Turban, E., Sharda, R., and Delen, D. (2010). *Decision Support and Business Intelligence Systems*. Prentice Hall Press, 9th edition.

Tuttle, R., Trenbath, K., Maisha, K., and LeBar, A. (2020). Assessing and reducing plug and process loads in office buildings. *Technical Report*.

Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J., Santos, L. O. B. d. S., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T. W., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., González-Beltrán, A., Gray, A. J. G., Groth, P., Goble, C., Grethe, J. S., Heringa, J., Hoen, P. A., Hooft, R., Kuhn, T., Kok, R., Kok, J. N., Lusher, S. J., Martone, M. E., Mons, A., Packer, A. L., Persson, B. L., Rocca-Serra, P., Roos, M., Schaik, R. v., Sansone, S., Schultes, E., Sengstag, T., Slater, T., Strawn, G., Swertz, M. A., Thompson, M., Lei, J. v. d., Mulligen, E. M. v., Velterop, J., Waagmeester, A., Wittenburg, P., Wolstencroft, K., Zhao, J., and Mons, B. (2016). The fair guiding principles for scientific data management and stewardship. *Scientific Data*.

APPENDIX

Keywords used to define datasets and projects: time series, energy, smart plugs, fast, high speed, less computation, accuracy, high performance, precise, reliable, user convenient, minimal disruption, user-friendly, non-intrusive.