*applied sciences*

MDPI

*Article*

# Agent Systems and GIS Integration in Requirements Analysis and Selection of Optimal Locations for Energy Infrastructure Facilities

**Anna Kochanek [1,*], Tomasz Zacłona [2], Michał Szucki [3] and Nikodem Bulanda [1]**

[1] Faculty of Engineering, State University of Applied Sciences in Nowy Sącz, 33-300 Nowy Sącz, Poland; nbulanda@ans-ns.edu.pl
[2] Faculty of Economic Sciences, State University of Applied Sciences in Nowy Sącz, 33-300 Nowy Sącz, Poland; tzaclona@ans-ns.edu.pl
[3] Foundry Institute, Technische Universität Bergakademie Freiberg, Bernhard-von-Cotta-Str. 4, 09599 Freiberg, Germany; michal.szucki@gi.tu-freiberg.de
* Correspondence: akochanek@ans-ns.edu.pl

**Abstract**

The dynamic development of agent systems and large language models opens up new possibilities for automating spatial and investment analyses. The study evaluated a reactive AI agent with an NLP interface, integrating Apache Spark for large-scale data processing with PostGIS as a reference point. The analyses were carried out for two areas: Nowy Sącz (36,000 plots, 7 layers) and Ostrołęka (220,000 plots). For medium-sized datasets, both technologies produced similar results, but with large datasets, PostGIS exceeded time limits and was prone to failures. Spark maintained stable performance, analyzing 220,000 plots in approximately 240 s, confirming its suitability for interactive applications. In addition, clustering and spatial search algorithms were compared. The basic DFS required 530 s, while the improved one reduced the time almost tenfold to 54–62 s. The improved K-Means improved the spatial compactness of clusters (0.61–0.76 vs. <0.50 in most base cases) with a time of 56–64 s. Agglomerative clustering, although accurate, was too slow (3000–6000 s). The results show that the combination of Spark, improved algorithms, and agent systems with NLP significantly speeds up the selection of plots for renewable energy sources, supporting sustainable investment decisions.

**Keywords:** large language models (LLM); conversational agents; Geographic Information System (GIS); site selection; big data; ETL; biogas; renewable energy



Academic Editor: Salvador García-Ayllón Veintimilla

Received: 19 August 2025
Revised: 22 September 2025
Accepted: 23 September 2025
Published: 25 September 2025

**Citation:** Kochanek, A.; Zacłona, T.; Szucki, M.; Bulanda, N. Agent Systems and GIS Integration in Requirements Analysis and Selection of Optimal Locations for Energy Infrastructure Facilities. *Appl. Sci.* **2025**, *15*, 10406. https://doi.org/10.3390/app151910406



## 1. Introduction

The progressive development of information technologies, particularly artificial intelligence (AI) and decision support systems (DSS), is setting new directions in the field of data analysis and strategic decision-making. There exists a complex yet synergistic relationship between artificial intelligence (AI) and decision support systems (DSS), whose integration constitutes a powerful tool supporting decision-making processes by combining advanced analytical and informational technologies [1]. The application of artificial intelligence (AI) in decision support systems (DSS) enables the analysis of large datasets, the identification of hidden patterns and trends, and the prediction of potential outcomes of decisions, which translates into better optimization of decision-making processes [1].

In this context, one of the fastest-growing technological and operational trends is the use of AI agents, driven by artificial intelligence and machine learning, to carry out

complex tasks within decision-making processes [2]. However, for such systems to be effective and practical, it is essential to maintain a balance between transparency, accuracy, and user trust [3]. In practice, this means designing solutions that ensure not only high analytical efficiency but also clarity and the ability for decision-makers to verify the decision-making processes.

An increasingly important perspective in this area is the environmental one [4], which is crucial both for achieving organizational goals and for fulfilling social and ecological objectives [5,6]. Many decisions in the field of environmental management (EM) are spatial in nature and are related, for example, to the siting of investments, including the selection of the most appropriate location for various infrastructure elements [7,8].

Due to the spatial nature of environmental decisions, tools that enable the analysis of geographical and infrastructural data are of key importance. The fundamental solution in this area is the Geographic Information System (GIS), which constitutes an essential component of modern DSS. GIS supports investment planning, especially in the renewable energy sector (RES), by collecting, integrating, and visualizing spatial data [9,10].

In this context, solutions that enable the integration of analytical methods with domain- and regulation-based approaches become particularly important. The complexity of decision-making problems, especially in environmental and investment management, requires not only precise computational tools but also mechanisms that incorporate legal and spatial considerations. In the Polish legal framework, the siting of agricultural biogas plants is closely linked to spatial planning and environmental protection requirements. Both national and EU regulations impose specific quantitative criteria, including the exclusion of protected areas (e.g., Naturaz2000 sites, landscape parks), minimum siting distances of 100–200 m from residential buildings, 30 m from rivers and surface waters, and 50 m from drinking water intakes. Digestate storage tanks must be located at least 25 m from drainage ditches and surface waters, while the agricultural use of digestate is limited by the Nitrate Directive to 170 kg of nitrogen per hectare per year. From an economic and logistical perspective, access to infrastructure is also critical: connections to electricity or gas grids should be available within 1–2 km, and biomass sources are considered optimal within a radius of 5–10 km. These requirements strongly determine the decision-making process and highlight the importance of incorporating regulatory criteria into GIS-based models supporting investment site selection.

Therefore, innovative ways of automating data analysis and supporting the selection of optimal solutions—including investment site selection—are increasingly sought, particularly through the use of AI agents capable of interpreting complex datasets. The aim of the study is to explore the potential of agent-based systems combined with natural language processing (NLP) techniques in automating the analysis of legal requirements and the selection of investment plots. The goal is to lower the entry threshold in this field by eliminating dependence on specialized engineering, legal, and advanced geospatial data interpretation expertise. The proposed solution significantly accelerates the processes of selection, prediction, and classification of plots. Unlike methods based on manual inspection of GIS data or complex interfaces (e.g., ArcGIS, QGIS), reactive AI agents controlled by a large language model are employed. This solution offers a conversational interface enabling transcription, matching, evaluation, and classification of requirements as well as management of the plot selection process. The reactive AI agent automates the retrieval, filtering, and analysis of geospatial data in accordance with legal criteria. Such an architecture enhances control over the decision-making process and presents results in a clear and intuitive way, aligned with the natural form of communication.

In contrast to traditional rule-based GIS platforms, which demand user proficiency in tools, query languages, and Multi-Criteria Decision Analysis (MCDA), our approach

introduces a layer of intelligent, reactive conversational agents powered by Large Language Models (LLMs). Consequently, the process of defining criteria and interpreting outcomes occurs through natural dialog, eliminating the need for specialized engineering or legal expertise. The agent automatically translates user requirements into formal analysis rules and performs filtering, grouping, and evaluation of spatial data in accordance with legal and investment conditions. This significantly lowers the barrier to entry for spatial analyses and aligns with the current trend of integrating GIS with agent architectures and LLMs in intelligent decision support systems.

The implementation employing reactive agents governed by an LLM represents a fundamentally different methodology for conducting searches. It substantially reduces the reliance on specialist skills or familiarity with interface functionalities and database operational rules. This enables even non-expert users to articulate requirements in natural language, which the system automatically converts into a structured set of signals. The LLM is responsible for data extraction and mapping, while geospatial data processing is delegated to a specialized reactive agent, mitigating the inefficiencies inherent in direct spatial data operations by LLMs. Currently, two primary approaches exist for integrating LLMs with GIS: the first involves fine-tuning the model for direct comprehension of spatial structures, metrics, and functions; the second focuses on constructing specialized reactive agents. Our solution adheres to the latter approach, ensuring precise control over the formulation of legal requirements and facilitating algorithm optimization based on dataset scale.

The main contributions of this study can be summarized as follows:

1. Integration of agent-based systems with large-scale data processing platforms.

We implemented a reactive agent architecture combining Apache Spark and PostGIS to handle large spatial datasets. The evaluation used the execution time metric, showing that Spark processed 220,000 parcels in ~240 s, whereas PostGIS exceeded the predefined 15 min limit, confirming Spark's scalability for conversational and interactive applications.

2. Comparative analysis of clustering algorithms.

Five clustering methods were tested: DFS (basic and improved), K-Means (basic and improved), and agglomerative clustering. The analysis time metric revealed that improved DFS and improved K-Means achieved acceptable runtimes (~54–64 s), while agglomerative clustering required several thousand seconds (>3000 s), limiting its applicability for interactive use.

3. Improvement of spatial compactness and coherence.

The optimized K-Means algorithm increased the compactness index (C) to 0.61–0.76, compared to <0.50 in most baseline cases. Similarly, the improved DFS reduced execution time by nearly one order of magnitude compared to its basic version (530 s $\rightarrow$ ~54–62 s), demonstrating both efficiency and higher cluster quality.

4. Demonstration of large-scale applicability in conversational interfaces.

The system efficiently processed datasets ranging from 36,000 to 220,000 parcels, enabling their use in an LLM-driven NLP interface. This confirms the feasibility of integrating advanced spatial analysis into chat-based decision-support systems.

5. Inclusion of regulatory and spatial criteria.

Beyond computational performance, the system embedded legal and spatial constraints (e.g., buffer distances from water bodies and residential buildings), ensuring that investment recommendations meet both regulatory requirements and practical siting conditions.

## 2. GIS as the Foundation of Spatial Analyses

Geographic Information Systems (GIS) constitute the foundation of modern spatial analyses, enabling the collection, integration, processing, analysis, and visualization of spatially referenced data [11]. Thanks to their ability to combine data from various sources—ranging from satellite imagery, through field measurements, to IoT sensor data—GIS makes it possible to create comprehensive representations of spatial phenomena. Such integration enables more accurate decision-making in spatial planning, environmental management, and infrastructure investments, as well as in forecasting spatial changes and their impacts [12,13].

The importance of GIS is particularly evident in the renewable energy sector, especially in wind energy. It enables the creation of suitability maps for siting wind farms by integrating data on topography, land cover, wind conditions, transmission infrastructure, and environmental constraints [14]. Methods such as AHP, fuzzy logic, and TOPSIS make it possible to account for both quantitative and qualitative criteria [15]. Moreover, GIS supports the analysis of wind energy resources and the assessment of investment potential, thereby facilitating cost optimization and impact minimization [16].

In the case of photovoltaics, GIS enables precise identification of the best PV installation sites by taking into account solar exposure, slope and orientation of the terrain, shading, and access to infrastructure [17,18]. GIS models integrated with machine learning allow for analyses at both local and regional scales, while integration with economic modeling tools supports investment feasibility assessments. GISs are also used in the design of hybrid RES systems, combining different energy generation technologies to enhance system stability [19,20].

In the context of biogas plants, GIS supports siting processes by analyzing substrate availability, proximity to transport and energy infrastructure, legal conditions, and social factors [21]. These issues are particularly interesting from the point of view of high-energy industries such as foundry, metallurgy, and glass industries. For many years, attempts have been made to completely or partially replace natural gas in production processes [22]. One of the key problems is the creation of a stable and efficient biogas supply system. The use of GIS through the analysis of spatial conditions, raw material availability, and infrastructure allows for optimal design of agricultural biogas plants [23]. Integration of GIS models with MCDA methods, such as AHP or the Best-Worst method, enables objective evaluation of locations in terms of both economic and environ-mental efficiency [24]. Moreover, GIS makes it possible to build biomass resource databases, taking into account supply seasonality, transport logistics, and opportunities for integration with other energy systems, which fosters the implementation of solutions aligned with circular economy principles [25].

For hydropower plants, GIS is used to assess hydroenergy potential by analyzing topography, river flows, elevation drops, and environmental constraints. The integration of hydrological models with spatial data enables the identification of sites with the greatest energy potential while minimizing environmental impact [26]. In small hydropower plants, such as run-of-river installations, GIS supports transmission infrastructure planning, construction cost analysis, and landscape impact assessment. The combination of GIS analyses with MCDA methods makes it possible to optimize site selection based on balancing energy production efficiency with the protection of natural resources [27,28].

GIS also finds wide application in the circular economy (CE). In the waste management sector, it allows for the optimization of waste collection routes, siting of processing facilities, analysis of material flows, and assessment of resource recovery potential [29]. This is particularly important in the circulation of materials that are critical to the economy and industry, access to which may be restricted, e.g., as a result of military conflicts, natural

disasters, political decisions, environmental regulations, etc. Examples include processed materials such as metal alloys. It should be remembered that the available waste may vary depending on the technological culture, relevant industrial processes, local standards and other regulations [30]. The use of GIS makes it possible to link data on waste generation sources with information on recycling, composting, or energy recovery options, thereby supporting effective resource management and minimizing landfilling. An example is the AHP-based siting of municipal solid-waste collection points in rural areas using GIS [31,32].

In the area of CE, the use of GIS in spatial analyses of secondary raw material logistics is also significant. By integrating data on collection points, processing facility capacity, and transport infrastructure, it is possible to design closed material loops with minimal carbon footprints [33]. Moreover, GIS supports industrial symbiosis modeling by identifying potential linkages between enterprises where by-products from one activity become resources for another [34,35].

As the foundation of spatial analyses, GIS is becoming a key element of the digital transformation of decision-making processes in the environment, energy, and circular economy sectors. By combining diverse data sources, modern analytical techniques, and visualization tools, it enables not only the diagnosis of current conditions but also the forecasting of changes and planning of adaptive actions [36]. As a result, GIS serves as a strategic instrument supporting the achievement of sustainable development goals, integrating technological, environmental, economic, and social perspectives [37,38].

## 3. The Role of Multi-Criteria Methods and Metaheuristic Optimization in Decision Support Systems

Decision-making is a fundamental function of management [39] and, as Pušeljić et al. note, a key determinant of organizational success or failure [40]. It enables managers to select one or more alternatives aimed at achieving a desired outcome [39]. In a rational decision-making process, choices are made objectively, after carefully considering the circumstances, alternative perspectives, and the potential consequences of each option [41]. However, the complexity of many decision contexts [42], particularly in environmental management, necessitates the use of multi-criteria decision analysis (MCDA), which provides a systematic framework for integrating diverse data, information, and stakeholder opinions to compare possible courses of action [43]. This approach enables consideration of both measurable factors and those difficult to quantify financially, allowing available options to be ranked by overall attractiveness and thereby supporting the selection of the solution that best meets established objectives [43].

As Bączkiewicz et al. observe, there are various MCDA methods that represent three traditions: the American, the European, and a mixed one based on sets of rules [44]. The main differences among particular MCDA methods include, among others: the level of complexity of the applied algorithms, the method of assigning weights to criteria, the form of presenting preferences and evaluation of criteria, the ability to account for uncertain data, and the type of data aggregation method used [45]. When applying MCDA in the decision-making process, a challenge arises in selecting and applying the appropriate method for solving a given problem.

The complexity of decision-making processes in management, including environmental management, obliges managers to employ not only MCDA methods but also decision support systems (DSS). This also results from the fact that traditional approaches to environmental management, based on manual data collection, are often inefficient and inaccurate [46]. Interactive information systems, such as decision support systems, assist decision-makers in making informed choices by collecting, processing, and analyzing data [47]. They aim to solve complex problems, provide valuable insights, and increase

the accuracy of decisions by combining data management, analytical models, and intuitive user interfaces [47].

As Ali et al. point out, modern DSS are often equipped with MCDA along with modules for their control [45]. Highlighting the relationship and differences between MCDA and DSS, it should be noted that the former is a methodology—that is, a set of methods that allow evaluating and comparing alternatives by considering multiple criteria. Decision support systems, in turn, are information systems (software). This means that multi-criteria decision-making is a technique, while decision support systems are the machine that may or may not employ this technique [48]. Keenan observes that in scientific literature, MCDA is considered a subset of DSS research, and that multi-criteria decision-making methods have always played a key role in decision support systems [48].

Currently, hybrid approaches are increasingly being applied, in which DSS integrate various MCDA methods in fuzzy environments [49]. The literature also highlights intelligent decision support systems that combine machine learning (ML) with MCDA [50]. As a result, intelligent decision support systems (IDSS) using ML and MCDA techniques are dynamically evolving. Within DSS, distributed decision support systems (DDSS) are also distinguished, enabling the integration and sharing of data from previously isolated silos, including in real time [47].

Complementing the MCDA-based perspective developed above, the literature also examines metaheuristic optimization as a direct search approach for related energy-planning problems. Metaheuristic optimization methods, including evolutionary, swarm-based, differential, and related algorithms, are widely applied in power system problems because they can effectively handle nonlinearity, high-dimensional variable spaces, and numerous operational constraints [51]. Literature reviews highlight their successful use in areas such as optimal power flow, reactive power dispatch (ORPD), combined economic–emission dispatch, Volt/Var control, and the size and placement of distributed generation (DG), confirming their value for typical grid optimization tasks [51].

During the design phase of energy systems, particularly hybrid renewable energy systems (HRES), hybrid metaheuristics are extensively employed for the optimal sizing and parameterization of system components, both in single-objective and multi-objective contexts [52]. At the operational stage, they also support microgrid management. For instance, the Multi-Verse Optimizer (MVO) has been used to schedule battery energy storage systems (BESS) in AC microgrids, in both grid-connected and islanded modes, with the aim of reducing power losses and carbon emissions while respecting network constraints [53]. Such approaches provide decision-makers with flexible tools for addressing operational challenges under dynamically changing requirements.

In spatial planning and site selection, metaheuristic methods enable the simultaneous consideration of environmental, social, and economic criteria. A notable example is the use of an improved NSGA-II for selecting photovoltaic farm locations, which illustrates how metaheuristics can help identify Pareto-optimal trade-offs and support strategic infrastructure planning [54].

Recent studies further emphasize the importance of integrating metaheuristics with Geographic Information Systems (GIS) and Multi-Criteria Decision-Making (MCDM) techniques. Together, these create coherent decision-making frameworks for requirements analysis and alternative evaluation, as demonstrated in the context of offshore wind farm planning. In this way, metaheuristics complement ranking-based techniques, enabling the incorporation of multiple perspectives and facilitating the selection of optimal sites for energy infrastructure development [55].

## 4. The Role of Machine Learning, Large Language Models, and Agents in Decision Support Processes

Decision support systems (DSS) play a key role in environmental management, infrastructure, and business processes. In recent years, their development has increasingly relied on the integration of artificial intelligence, particularly machine learning, large language models (LLMs), and agent-based architectures [56,57]. This enables not only the storage and analysis of large datasets but also their interpretation in a more "human-like" way—comprehensible and transparent for users.

Machine learning (ML) allows DSS to identify hidden patterns, perform prediction and classification, and forecast the outcomes of alternative decisions [58]. Large language models (LLMs)—such as GPT or BERT—open new possibilities for interaction with decision systems through a conversational interface, enabling users to ask questions in natural language and receive personalized recommendations [59]. The integration of ML and LLM supports the transparency of analytical processes and provides users with greater control over inputs and outputs.

AI agents are autonomous components capable of perceiving their environment, making decisions, and executing actions within complex processes [60]. They can be classified into several types: reactive agents (operating on a stimulus–response basis), deliberative agents (planning and goal-oriented), hybrid agents (combining both approaches), and learning agents [61]. In practice, agent models are key elements of AI architectures, serving as intermediaries between the analytical layer and end users.

Reactive agents are defined by simple structures and immediate stimulus–response behavior, which makes them well suited for dynamic environments such as environmental monitoring or the Internet of Things (IoT) [62]. Their main advantages are speed and low computational requirements, but the absence of memory and planning limits their adaptability in complex contexts [63]. As a result, they are most commonly employed today as components of larger hybrid systems.

In the era of Big Data, decision-related data processing requires scalable platforms. Apache Spark enables parallel, distributed processing of large datasets, which, when combined with agents, allows for the creation of intelligent analytical workflows [64]. PostGIS, the spatial extension of PostgreSQL, supports agents operating directly in the database, enabling automatic spatial queries and the integration of agent logic with geodata [65]. Thanks to this, AI agents can analyze billions of spatial records in real time, supporting decisions in urban planning, transportation, or environmental protection.

Agent-based systems supported by ML and LLM are increasingly applied in spatial analysis, investment planning, and environmental management [66]. However, this development comes with challenges—the need to ensure transparency, user trust, and compliance with regulations [67,68]. It is therefore essential to design such systems in a way that not only maximizes efficiency but also allows users to verify and understand decision-making processes.

The methodology grounded in reactive agents operating under the supervision of a Large Language Model (LLM) introduces a fundamentally novel paradigm for executing spatial search processes. This approach minimizes, to an almost negligible level, the requirement for users to possess advanced expertise or familiarity with the interfaces and operational rules of geographic databases. Instead, it enables non-expert users to retrieve relevant information through natural language queries. Within this architecture, the LLM is responsible for extracting user requirements and transforming them into a structured set of signals, while the computationally demanding tasks of spatial data processing are delegated to a dedicated reactive agent.

Contemporary research outlines two principal directions for integrating LLMs with GIS. The first emphasizes fine-tuning language models to comprehend data structures, metrics, and functions directly within spatial datasets. The second prioritizes the development of specialized agents that cooperate with LLMs. Our preference for the latter approach stems from the necessity of ensuring robust control over the generation of legislative requirement lists, as well as enabling algorithmic optimization that scales with dataset size.

## 5. Clustering Algorithms

Clustering is one of the fundamental techniques of data mining, aimed at grouping objects in such a way that elements within the same group (cluster) are more similar to each other than to those in other groups. Unlike supervised methods, clustering does not require prior data labeling—hence it belongs to unsupervised learning methods. It has wide applications in many fields, such as image analysis, customer segmentation, bioinformatics, and spatial data processing. This chapter presents the most important clustering algorithms, their characteristics, advantages and limitations, as well as examples of practical applications.

### *5.1. Depth-First Search Algorithm*

The depth-first search algorithm (DFS) is one of the fundamental methods for exploring graph structures. It uses a stack mechanism that allows visiting consecutive vertices as long as there is a path to an unvisited neighbor. Once the path is exhausted, the algorithm backtracks and continues searching in other directions. DFS is applied in various areas of computer science, including topological sorting, cycle detection, finding bridges, connected components, and as a component of more complex graph algorithms [69,70].

In the context of clustering, DFS can be used to determine connected components of a graph, which are then treated as independent clusters. This approach works particularly well when data is represented as graphs and relationships between objects are defined by a similarity metric or spatial structure. Using DFS for this purpose enables fast and efficient identification of related object groups, even in large datasets. Connected-component clustering is deterministic, and its results are fully replicable, which is a significant advantage compared to stochastic methods [71,72].

DFS is also employed in hybrid approaches, forming the basis for processing more complex data structures. For example, it can be used in the analysis of spatial trajectories, with vertices representing stop locations and edges corresponding to possible connections or spatiotemporal neighborhoods. DFS can then be applied to detect clusters of such points, reflecting real-world concentrations of user activity [73].

Beyond spatial analysis, DFS is also applied in recommender systems and graph pattern mining. In these cases, users, products, or events are represented as nodes, and their relationships as edges. DFS enables the grouping of nodes into clusters based on mutual reachability and graph structure. It also supports structural pattern searches in graph datasets, as in the gSpan algorithm, which systematically explores graphs by generating DFS codes for subgraphs to efficiently detect frequently occurring structures [74,75].

### *5.2. Agglomerative Clustering Algorithm*

Agglomerative hierarchical clustering is a bottom-up data segmentation method. The process begins by treating each data point as a separate cluster. In successive iterations, pairs of the most similar clusters are merged until one group covering the entire dataset is obtained, or until a predefined stopping level is reached. This process represents relationships between points in the form of a hierarchical tree, known as a dendrogram [76].

A key aspect of the algorithm is how distances between clusters are measured. The most common linkage methods include: single linkage (nearest neighbor), complete linkage (farthest neighbor), average linkage (average distance), and Ward's method, which minimizes the increase of within-cluster variance at each merge. The choice of linkage strategy strongly affects the structure, number, and shape of the resulting clusters [77].

Agglomerative clustering is particularly useful when the expected number of clusters is unknown. Unlike methods such as k-means, it does not require a predefined number of groups. Users can explore the hierarchical structure and decide on the number of clusters based on dendrogram analysis. This flexibility makes the method useful in molecular biology (gene expression analysis), social sciences (respondent segmentation), and spatial data analysis [78].

One of the main advantages of agglomerative clustering is its interpretability. The dendrogram provides insight into relationships between points at different levels of hierarchy. The method also performs well with irregularly shaped clusters and clusters of varying density. On the downside, its main limitation is high computational cost—the standard implementation has time complexity of $O(n^3)$ and space complexity of $O(n^2)$, which can be a serious barrier for very large datasets [79].

In response to these limitations, various optimizations have been developed. Algorithms such as SLINK (for single linkage) and CLINK (for complete linkage) reduce runtime to $O(n^2)$ or less while maintaining accuracy. Approximate hierarchical methods based on sampling or data compression can also be used, making them applicable in Big Data environments [80].

Modern implementations of agglomerative clustering are provided in many analytical libraries, such as SciPy, scikit-learn, and R (e.g., the hclust and agnes functions). The algorithm is also employed as a component of more complex solutions—for example in image analysis for object segmentation or in recommender systems, where it groups users according to behavioral profiles [81].

### 5.3. K-Means Algorithm

The k-means algorithm belongs to the family of non-hierarchical clustering methods, also known as partitional clustering. Its goal is to divide a dataset into $k$ non-overlapping clusters in such a way that within-cluster variance is minimized—that is, the sum of squared distances between points and the centroid of their assigned cluster [82].

The algorithm operates using an iterative optimization scheme. First, $k$ points are randomly selected as initial centroids. Each data point is then assigned to the nearest centroid according to a chosen distance metric (typically Euclidean). Once all points are assigned, centroids are updated as the arithmetic mean of the coordinates of points in each cluster. The assignment and update steps are repeated until stability is achieved—that is, no changes in assignments occur, or a maximum number of iterations is reached [83].

One of the main advantages of k-means is its simplicity and low computational cost. In many practical cases, convergence is reached very quickly, even for large datasets. As a result, k-means has become a standard tool in data mining and is widely applied in fields such as market analysis, image segmentation, bioinformatics, and natural language processing [84].

However, the classic k-means algorithm has important limitations. Most notably, it requires the number of clusters $k$ to be specified in advance, which in practice can be difficult. Moreover, k-means is sensitive to centroid initialization—different initializations may lead to different final results. To address this, variants such as k-means++ were developed, introducing improved initialization methods that increase the likelihood of finding a global minimum [85].

K-means also assumes that clusters are convex, roughly equal in size, and spherical. For datasets with irregular shapes or clusters of varying density, results may be unreliable. Additionally, the method is sensitive to outliers, which can distort centroids and negatively affect segmentation outcomes [86].

To evaluate clustering quality, several indices are used, such as the Silhouette coefficient, Calinski–Harabasz index, or Davies–Bouldin index. Choosing the right metric allows not only the assessment of clustering performance but also the selection of the optimal number of clusters $k$, often using the "elbow method" [87].

Despite its limitations, k-means remains one of the most widely used tools in data analysis. Its simplicity, interpretability, and effectiveness in many applications make it a crucial starting point for more advanced and hybrid clustering methods.

## 6. Materials and Methods

The system architecture was designed with the goal of maximizing user interaction efficiency while effectively managing potentially large datasets. A key assumption was the use of an interface enabling natural language processing, the selection of a specialized agent, and the separation of a one-time, computationally expensive process of data retrieval and preprocessing from lightweight, dynamic analyses performed in response to user actions. To achieve the objectives of this article, two parallel reactive agents were implemented. The first agent uses PostgreSQL/PostGIS as a stable and widely adopted environment for storing and analyzing spatial data. The second agent explores the application of Big Data solutions based on the distributed processing framework Spark, which enables scalable operations on large datasets.

### *6.1. Application Concept*

The application architecture was based on the assumption of maximum performance and responsiveness of the user interface while efficiently managing large datasets. The general operation of the application can be described in the following stages:

- Step 1—Data intake and preprocessing
- Step 2a—Use of Apache Spark
- Step 2b—Use of PostGIS
- Step 3—Aggregation of results

#### 6.1.1. Data Intake and Preprocessing (Step 1)

In this process, the user interface sends queries to the LLM, which executes an iterative procedure of gathering the information necessary to precisely define requirements. The model engages in follow-up questioning and interaction in order to obtain detailed input from the user. Based on the collected data, the model translates knowledge into a set of clear criteria that form the basis for triggering a specialized agent.

The implementation is based on a REST API serving as the communication layer between system components, enabling the automatic transformation of investor requirements expressed in natural language into formalized operational criteria for the agent's reactive execution engine. The process begins with user interaction through a conversational interface, where requirements are specified and passed to the Ollama environment. There, the selected large language model (LLM) processes them and generates a structured set of operational signals, which are subsequently forwarded to the agent's execution engine.

The current implementation of the clustering algorithm is characterized by a selective approach, in which the analysis is restricted exclusively to objects that meet the defined surface and quality criteria. During the preprocessing stage, incomplete records containing missing values in key attributes were first removed, followed by the elimination of

duplicates. In addition, spatial projections were standardized to the EPSG:2180 coordinate system to ensure consistency across the dataset and improve the reliability of the results.

Objects that failed to satisfy quantitative requirements—such as maintaining minimum spatial distances, belonging to designated areas, or meeting basic geometric correctness—were excluded from the clustering process. Geometric defects included, among others, unclosed boundaries and irreparable geometry errors, which prevented proper execution of subsequent analyses.

The geometry validation process relied on the analysis of spatial coherence indicators, including the number of connected components (K), compactness index (C), and elongation index. These measures were used for the hierarchical evaluation and comparison of the qualitative properties of clusters. To increase the precision of the analysis, the set of indicators was further extended with the fragmentation index (F) and the share of the largest component (U), both of which contributed to a more comprehensive assessment of spatial coherence. These geometric measures were also applied in identifying the dominant island within individual clusters, thereby enabling more representative and homogeneous results.

In addition to the automatic translation of requirements into operational criteria, a weighting procedure was introduced to reflect the relative importance of individual factors. At the current stage of research, deterministic expert weighting was applied. Opinions were collected from specialists in design, construction, GIS, and environmental engineering, all of whom had practical experience in comparable projects. Their assessments were consolidated through a consensus process, ensuring that the assigned weights accurately represented the expected influence of each criterion on the overall quality and effectiveness of the final solution. The resulting weights are presented below:

- Substrate availability: 0.45
- Ecosystems and biodiversity: 0.25
- Spatial and landscape pressure: 0.05
- Distances, statutory thresholds, and compactness: 0.15

A sensitivity analysis was subsequently carried out, indicating that the most influential factors were substrate levels, biodiversity, and parameters related to network infrastructure and cluster compactness. Although fuzzy or interval weighting schemes were not employed at this stage, future work will extend the methodology with fuzzy and probabilistic approaches in order to more adequately represent decision-making uncertainty.

#### 6.1.2. Scenario Involving the Use of Apache Spark (Step 2a)

The scenario envisions the use of Apache Spark as the central component for processing and analyzing large spatial datasets, combined with tools such as GeoPandas, Scikit-learn, and Streamlit. This enables efficient execution of the ETL process, in-memory data buffering, as well as interactive analysis and clustering, which significantly improves work with distributed geospatial data. The diagram (Figure 1) illustrates the workflow of automation and selection of plots designated for the construction of agricultural biogas plants.

The proposed system architecture was designed as a sequence of four key stages of data processing. The first stage involves the ETL (Extract, Transform, Load) process, managed by Apache Spark. At this step, all necessary data—such as parcels, roads, and utilities—are loaded using Parquet packages, the GeoPandas library, binary files, or JDBC drivers. A preliminary transformation is also performed, primarily converting geometries from binary to columnar formats, which facilitates subsequent processing but may cause memory issues with very large datasets.

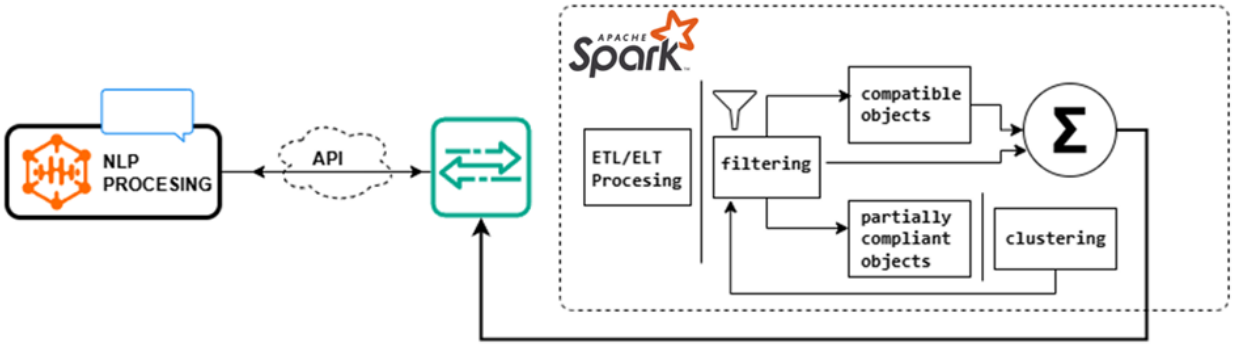


**Figure 1.** Diagram of the automation and investment plot selection process.

In the second stage, the processed data are converted into the GeoDataFrame format and stored in the application's cache memory using the @st.cache_data mechanism of Streamlit. This ensures that the data are loaded only once per session, significantly reducing system response time, although it may lead to memory overflow in the case of extensive datasets.

The third stage focuses on interactive in-memory analysis. All user initiated operations such as filtering, clustering, or distance calculations are executed with optimized functions from the GeoPandas library and clustering algorithms from Scikit-learn.

Finally, in the fourth stage, parcels that do not fully meet the defined criteria undergo clustering. Selected algorithms, including depth-first search (DFS), agglomerative clustering, and k-means, are applied to group parcels into larger, coherent areas that satisfy minimal investment requirements. Moreover, classical algorithms were enhanced with mechanisms to ensure cluster consistency, such as edge-contact analysis and the application of a minimum rotated rectangle algorithm, which optimize the shape and geometry of the final clusters.

#### 6.1.3. Scenario Involving the Use of Apache Spark and PostGIS (Step 2b)

Figure 2 presents the diagram of a PostGIS-based agent, which serves as a reference point for describing the scenario of integration with the Apache Spark platform.

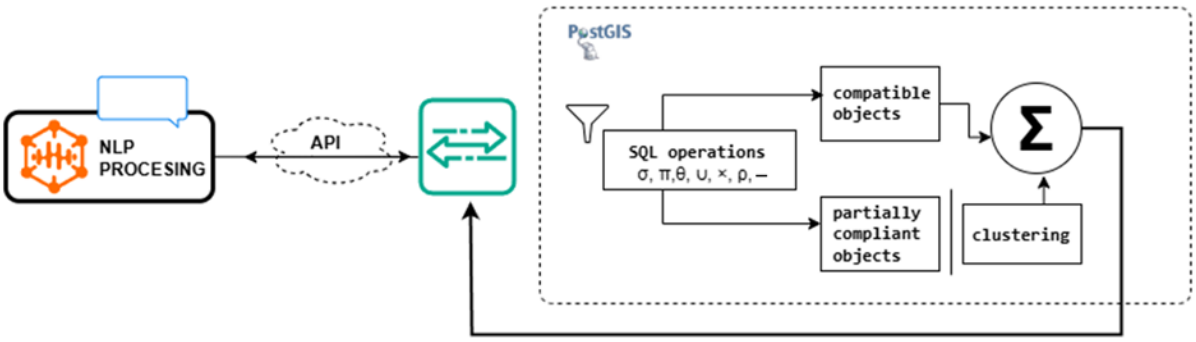


**Figure 2.** Diagram of the use of a PostGIS-based agent.

The data preparation and analysis phase constitutes a key distinction between the discussed approaches to agent architecture. In the traditional model, the process begins with the preliminary selection of potential investment plots based on defined criteria (e.g., building status, land classification). Next, using SQL queries and the ST_Distance function implemented in PostGIS, the geometric distance is calculated between the examined plot and the union of infrastructure representations (roads, power lines, water supply and sewage networks, etc.). The obtained results are then converted into a Pandas DataFrame for integration with the original plot GeoDataFrame based on an identifier.

Subsequent stages of data processing, such as buffer generation or clustering, can be treated as equivalents of the processes performed using the Apache Spark platform.

#### 6.1.4. Aggregation of Results (Step 3)

The final stage of the proposed solution is the aggregation of results, aimed at standardizing the data presentation interface. The system provides both spatial visualizations in the form of maps and area drawings, as well as tabular data characterizing plots and identified clusters that meet specific criteria. After the analysis is complete, the data saved in the session state is used to dynamically generate visualizations:

1. Map visualization—the _display_map() function from the ui.py module is responsible for generating an interactive map using the PyDeck library. Separate visual layers are created:
    a. GeoJsonLayer for individual plots (fits), marked in green by default.
    b. GeoJsonLayer for plots belonging to valid clusters, with each cluster assigned a random, unique color.
    c. PathLayer layers for specific types of infrastructure (roads, power lines, water, sewage), each marked with a different color.

The map is interactive and displays detailed information about an object (plot or cluster) in a tooltip when hovered over with the cursor.

2. Tabular presentation—the function _display_summary_tables() generates two separate summary tables. The first contains detailed data on the qualified individual plots, and the second provides analogous information for the clusters that met all the criteria.

### *6.2. Software Environment*

The choice of technologies was driven by the aim of creating an efficient, scalable, and easy-to-maintain system. Below are the key components of the technology stack along with the rationale for their selection (Table 1).

**Table 1.** Main components of the software environment.

| Type of Software | Description of Operation | Ref. |
| --- | --- | --- |
| Python (version 3.11.13) | Chosen as the main programming language due to its versatility, rich ecosystem of data analysis libraries (Pandas, Scikit-learn) and GIS (GeoPandas), as well as excellent integration with the other components of the project. | [88] |
| Streamlit (version 1.48.1) | Used for rapid prototyping and building an interactive user interface. Its simplicity and script-based model allowed the focus to remain on analytical logic rather than the complexity of developing web applications. | [89] |
| PostgreSQL (version 16.4-1) with PostGIS (version 3.4.3) | Used as the database system. The PostGIS extension is the industry standard for storing, indexing, and querying geospatial data, which was crucial for effective management of plot geometries and infrastructure data. | [90] |
| QGIS (Quantum GIS version 3.44.1-Solothurn) | Applied at the stage of data preparation and preprocessing. This desktop GIS software was used for manual editing, validation, and preparation of vector layers (plots, water, power lines) before importing them into the target PostGIS database. | [91] |

**Table 1.** *Cont.*

| Type of Software | Description of Operation | Ref. |
| --- | --- | --- |
| Apache Spark (version 3.5.1) | Used to implement the ETL process. Its ability to perform distributed data processing makes it an ideal tool for efficiently loading and preprocessing large volumes of data from the database, offloading this task from the main application. | [92] |
| GeoPandas (version 1.1.1) | Serves as the foundation of the project's analytical layer. This library extends the popular Pandas package with geospatial data handling, providing an intuitive and efficient interface for geometric operations, which was essential for implementing filtering and clustering logic. | [93] |
| Scikit-learn (version 1.7.1) | Used to implement standard machine learning algorithms, which served as the basis for clustering mechanisms (e.g., K-Means, Agglomerative Clustering). | [94] |
| PyDeck (version 0.9.1) | Applied to create interactive, multi-layered map visualizations. Its ability to render large datasets client-side (in the browser) using WebGL ensures high performance and smooth navigation. | [95] |
| Docker Engine version 24.0 & Docker Compose vervion 2.39.4 | These tools were used to containerize the entire application and its dependencies (database, Python environment). This enabled the creation of a consistent, portable, and easily reproducible runtime environment, significantly simplifying deployment and development. | [96] |
| Ollama & Gemma-3n-e4b | Designed as a toolkit providing an API interface for communication with the large language model (LLM). The choice of the Gemma-3n-e4b model was motivated by its optimal balance between performance quality and computational resource consumption (RAM, VRAM, CPU). The model supports both text and image inputs. A distinctive feature of Gemma-3n-e4b is its extended context window of 128K, allowing the processing of longer information sequences. | [97] |

### *6.3. Implementation of Clustering Algorithms*

The central analytical component of the application is the clustering module, responsible for merging smaller neighboring plots into larger, coherent investment areas. Within the project, several approaches were implemented and analyzed, ranging from neighborhood graph-based methods to advanced hybrid techniques. The following sections provide a detailed description of each algorithm.

#### 6.3.1. Depth-First Search (_Cluster_Plots_NN)

The clustering algorithm is based on graphical representation and the depth-first search (DFS) method. The process consists of three main stages, as shown in Figure 3.

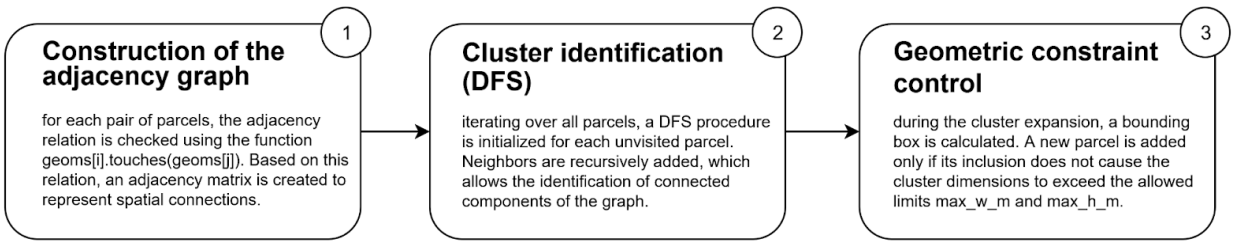


**Figure 3.** Main stages of the depth-first search.

6.3.2. Improved Depth-First Search (_Cluster_Plots_NN_Improved)

This version of the algorithm represents a significant improvement over the basic method, eliminating its key drawbacks. The improved algorithm performs clustering in two main steps:

1. Optimized graph construction—instead of an $n^2$ loop, the algorithm first creates a spatial index (R-tree) for all plots using the attribute gdf.sindex. Then, for each plot, it searches only for potential neighbors (whose bounding boxes intersect), which drastically reduces the number of costly .touches() checks that need to be performed.
2. Precise dimension control—instead of a bounding box, the algorithm applies a much more accurate method. When considering the addition of a new plot, it creates a temporary merged cluster geometry (unary_union) and computes its minimum rotated rectangle (minimum_rotated_rectangle). This method perfectly fits the rectangle to the shape of the cluster, regardless of its orientation, allowing for precise verification of its actual width and height.

6.3.3. Basic K-Means (_Cluster_Plots_KMeans)

This is an implementation of the classic k-means algorithm. The algorithm works on the principle of partitioning and proceeds as shown in Figure 4.

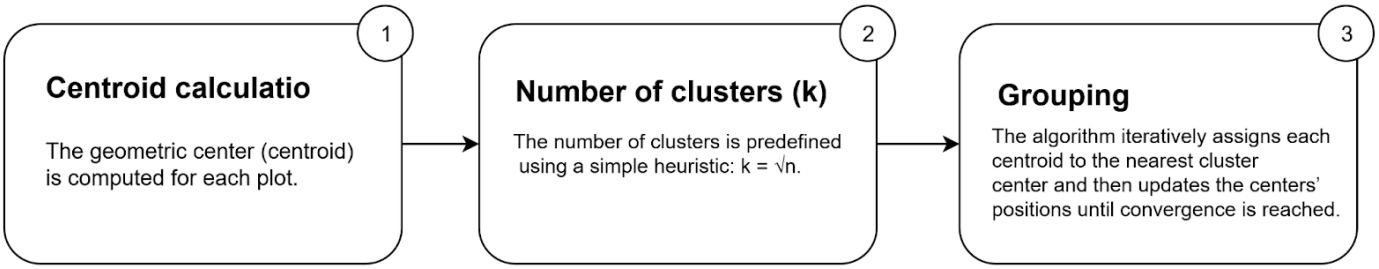


**Figure 4.** Steps of the k-means algorithm.

6.3.4. Improved K-Means (_Cluster_Plots_KMeans_Improved)

This is an advanced hybrid method that combines the speed of k-means with the accuracy of the neighborhood graph-based approach. The algorithm works in two stages:

1. Stage 1—Initial clustering (optimization): The fast k-means algorithm is run first. Its output is not treated as final but only as a way to coarsely divide the entire dataset into smaller "zones" or "neighborhoods."
2. Stage 2—Building coherent clusters: Next, the algorithm iterates through each plot, building coherent clusters in a way very similar to the _cluster_plots_NN_improved method. The key difference is that neighbor search is restricted only to plots within the same k-means "zone." Dimension control is performed precisely using the minimum rotated rectangle.

6.3.5. Agglomerative Clustering (_Cluster_Plots_Agglomerative_Clustering)

This method uses a standard hierarchical clustering algorithm from the Scikit-learn library. The algorithm works from the bottom up and proceeds in the steps shown in Figure 5.

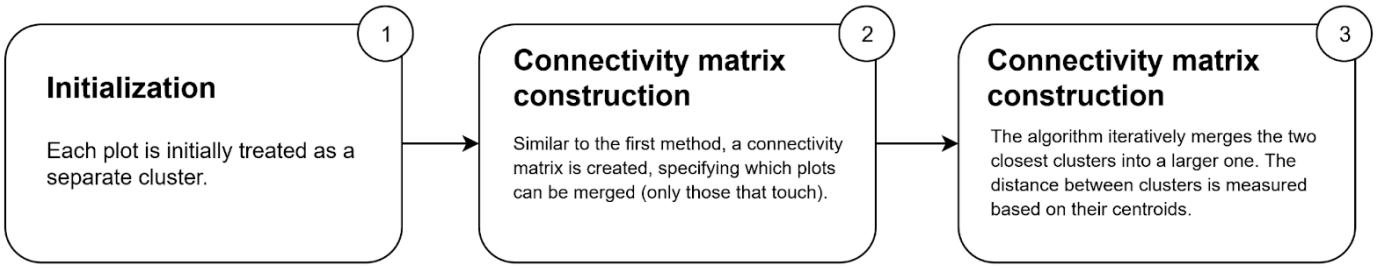


**Figure 5.** Stages of the hierarchical agglomerative clustering algorithm.

## 7. Results

### *7.1. Evaluation Metric*

This chapter focuses on the performance and quality analysis of the implemented system. Particular attention was given to the evaluation of two key architectural components: the data loading mechanism and the clustering algorithms. To objectively assess system performance, a set of metrics based on execution time and resource consumption was defined.

#### 7.1.1. Execution Time Metric

Execution time was recorded using the *time* module in Python, with special attention to two stages: data loading and analysis.

Data loading time is defined as the period from application startup until all data is fully loaded into memory and ready for analysis. In practice, this corresponds to the execution time of the load_initial_data function. This parameter is a key metric, as it enables performance comparisons between the two architectural variants—Spark and PostGIS—and allows assessment of the impact of the caching mechanism on overall system performance.

The second key indicator is the analysis time. It is measured from clicking the "Filter and analyze" button until the results are saved in the session state and the user interface is refreshed. This metric was used to evaluate and compare the efficiency of individual clustering algorithms, facilitating the clear identification of the most effective solutions.

#### 7.1.2. Resource Consumption Metric

In addition to execution time, an important aspect of performance evaluation is the analysis of system resource consumption. These parameters were monitored using the docker stats command for the application container, which enabled real-time tracking of system load during program execution.

The first measured indicator was peak RAM usage [MB], understood as the maximum amount of memory consumed by the container during both the data loading process and the actual analysis. The second parameter was maximum CPU load [%], reflecting the highest level of processor utilization by the application container. Together, these indicators allow for a comprehensive assessment of system efficiency, not only in terms of execution time but also in terms of hardware resource management.

### *7.2. Experimental Plan*

#### 7.2.1. Test Environment

To ensure repeatability and reliability of the results, all tests were carried out in a unified research environment, the specifications of which are presented in Table 2.

**Table 2.** Specification of the test environment.

| Component | Sample Specification |
|---|---|
| Operating system | Windows 11/Ubuntu 22.04 |
| Processor (CPU) | Intel Core i7-12700H |
| Memory (RAM) | 32 GB DDR5 |
| Disk | NVMe SSD |
| Software | Docker and Docker Compose, QGIS (wersja 3.44.1) |
| Dataset | 36,000 plots and infrastructure layers |

Based on the prepared research environment, two main test scenarios were planned, covering all the specified configuration and analysis variants.

#### 7.2.2. Test Scenarios

Based on the defined research environment, two test scenarios were planned to enable a comprehensive evaluation of the system's performance. To obtain a more complete picture of efficiency, each scenario was carried out in two different configuration variants:

- Variant I—Spark—covering tests using the Spark mechanism, enabling the analysis of distributed data processing efficiency.
- Variant II—PostGIS—covering tests based on the PostGIS database, allowing results to be compared with a more traditional database solution.

In each case, measurements were repeated five times, and the final analysis considered the average values. This approach ensures not only the reliability of the results but also their comparability between variants. On this basis, two main test scenarios were distinguished, covering the key aspects of system performance:

- Scenario 1—comparison of data loading performance.
- Scenario 2—comparison of clustering algorithm performance.

The first scenario focuses on evaluating the time and resources required to prepare the application for operation, depending on the loading mechanism (Spark vs. PostGIS) and the use of caching. The analysis includes two configuration variants that examine the combination of both technologies.

The second scenario aims to compare analysis time and resource consumption for the five implemented clustering algorithms. The tests are carried out using the most efficient data loading configuration identified in the first scenario, while keeping filter values constant.

## 8. Discussion

### *8.1. Data Loading Results*

The first element of the analysis concerns the results of the data loading process. Table 3 presents the average application loading time as well as the peak values of RAM usage and CPU load recorded for the four configuration variants.

**Table 3.** Comparison of data loading performance.

| Application Variant | Average Loading Time [s] | Peak RAM [MB] | Peak CPU [%] |
|---|---|---|---|
| Variant I—Spark | 0.0372 | 1040.38 | 195.2 |
| Variant II—PostGIS | 0.03375 | 760.06 | 163.38 |

The experimental results show that, on the 36,000-parcel dataset, the PostGIS-based implementation had a marginal time advantage over the Spark-based solution, while Spark consumed approximately 27% more RAM and 16% more CPU. Dataset size strongly affected the data-loading stage; nevertheless, the response time of individual agents remained below 1 s, which is excellent for most applications. The study pursued two objectives: accelerating the identification of parcels meeting the siting constraints for renewable energy facilities (RES) and comparing Spark and PostGIS within a specialized reactive agent to determine the option most optimal in terms of execution time and resource consumption—a factor critical for interactions with conversational systems. The approaches are fundamentally different: Spark is a distributed Big Data engine that provides high scalability for very large datasets, whereas PostGIS is a classical geospatial environment within a monolithic PostgreSQL database. The experiments covered two regions: the city of Nowy Sącz and the county and city of Ostrołęka. For medium-sized datasets (approximately 36,000 parcels with seven thematic layers), both technologies achieved comparable analysis times, with PostGIS using fewer resources.

For substantially larger datasets, the differences became pronounced. Processing approximately 220,000 parcels with seven thematic layers in PostGIS exceeded the predefined 15 min time limit. The workflow encompassed computing distance metrics for all parcels, clustering and grouping, filtering, and preparing data for the graphical interface (conversion to GeoJSON). Similar behavior was observed in QGIS under the same conditions, with frequent time overruns and application crashes during processing and export. In contrast, the Spark-based pipeline maintained stable performance as the number of parcels increased and did not exceed the 15 min limit in any experiment; the average processing time for the 220,000-parcel scenario was about 240 s. An additional scalability stress test with approximately 230,000 parcels and seven thematic layers used a longer, 2 h ceiling to observe degradation effects. In this regime, the PostGIS pathway again exceeded the time limit, and QGIS exhibited similar limitations. The Spark-based architecture showed no deterioration as the number of parcels increased and never breached the time ceiling in the conducted experiments. These results indicate that, while both approaches are viable at medium scale, Spark offers more stable and scalable behavior for analyses involving hundreds of thousands of objects and, by extension, for nationwide datasets.

In future stages of research, the analysis will be extended to larger datasets (0.5 and 1 million parcels), which will allow the presentation of scalability curves and a more comprehensive evaluation of system performance under big data conditions.

### *8.2. Clustering Algorithms Performance Results*

The next stage of the analysis focused on comparing the performance of the implemented clustering algorithms. Similarly to the data loading stage, both the execution time of the analysis and the peak utilization of system resources were evaluated. Five algorithms were considered: two variants of the depth-first search method (basic and optimized), agglomerative clustering, the classical K-Means, and its optimized version. The results are presented in Table 4.

The clustering stage proved to be the most resource-intensive in terms of computational time, memory consumption, and processor load, which highlights the importance of providing users with the option to deactivate this functionality based on natural language interface (NLP) commands. A qualitative analysis of the applied methods revealed a tendency for the optimized algorithms to generate more efficient spatial development proposals. This improvement results from the selective selection of neighboring parcels and the implementation of the minimum bounding rectangle inversion algorithm, which enables better spatial layout optimization.

**Table 4.** Comparison of clustering algorithm performance.

| Clustering Algorithm | Avg. Spark Analysis Time [s] | Avg. PostGIS Analysis Time [s] | Peak RAM Usage Spark [MB] | Peak CPU Usage Spark [%] | Peak RAM Usage PostGIS [MB] | Peak CPU Usage PostGIS [%] |
|---|---|---|---|---|---|---|
| Depth-First Search (DFS) | 530.97 | 526.77 | 1038.3 | 110.21 | 760.06 | 109.48 |
| Improved DFS | 54.19 | 61.994 | 1005.05 | 109.3 | 515.58 | 96.20 |
| K-Means (basic) | 5.884 | 17.478 | 1040.38 | 195.2 | 620.16 | 174.00 |
| K-Means (improved) | 56.346 | 63.994 | 703.94 | 142.9 | 531.36 | 123.90 |
| Agglomerative clustering | 3056.2 | 6032.5 | 1267.82 | 326.71 | 1023.34 | 254.46 |

The time analysis confirmed that both the optimized DFS implementation and the K-Means method achieved execution times of approximately 60 s. This constitutes a key factor in determining their potential applicability in chat-based interfaces. Although this duration exceeds the commonly cited 10 s threshold of acceptability in human–system interactions—after which user frustration typically increases—it can nonetheless be considered acceptable, aligning with the notion of a "creative process of analysis" [98]. In contrast, despite the high quality of its results, agglomerative clustering is characterized by significantly longer computation times (stemming from its high computational complexity), which effectively disqualifies it from rapid prototyping scenarios. Despite their high computational cost, slower methods may be preferred in strategic analyses where quality and spatial consistency take precedence over response time; a potential solution is to use approximate variants (e.g., sampling) or hierarchical approaches, in which detailed algorithms are applied only to preselected areas.

The introduction of algorithmic improvements through preliminary neighbor analysis and the application of the minimum rotated rectangle (MRR) aimed to enhance the qualitative coherence of the generated clusters. An expert validation process was adopted as a supervisory method, confirming that these improvements reduced parcel splits and preserved appropriate cluster proportions. The improvements introduced an additional computational overhead—for the K-Means algorithm, execution time increased by approximately one order of magnitude compared to the baseline version, while for DFS the improved implementation reduced execution time by nearly an order of magnitude. Despite these differences, in comparative terms both improved methods provided similar and acceptable analysis times (~60 s for 36,000 parcels and 7 infrastructure layers, as shown in Table 4

The results of the agent's work were evaluated both by domain experts and by selected spatial coherence metrics, which included the number of connected components, the compactness index, and the elongation index. These measures were used to rank and assess the clusters. In addition, they were applied in the process of extracting the dominant island within a cluster, which was then transformed and became a separate cluster tending towards K = 1 (improved methods).

The K metric was further extended to include the fragmentation index (F) and the share of the largest component (U). In the basic versions of clustering algorithms (non-improved), the number of connected components ranged from 1 up to as many as 14, with clusters where K > 3 being predominant. This indicated discontinuity and the possible presence of "holey" areas.

Quantitative results should also be interpreted in the context of location—for example, the number of possible clusters directly affects the availability of potential investment

locations, and the distance from the power or gas grid is a key factor in assessing their actual suitability.

The introduced modification, consisting of extracting the most compact island within the cluster, proved effective, leading to achieving K = 1. Furthermore, for each cluster, the compactness index (C) was calculated. For improved algorithms, compactness ranged between 0.61 and 0.763, whereas in the basic version of the algorithm it most often (in 60% of cases) fell significantly below 0.50, suggesting the need for cluster boundary revision.

On this basis, an improvement in the spatial coherence of the improved algorithms was confirmed. It should be remembered that the system proposes parcels by ranking them according to a weighted average $W = (0.5 \times K) + (0.5 \times C)$; however, the final decision on siting always belongs to the user.

The sensitivity analysis indicated that the final ranking is most influenced by the threshold values for distance from network infrastructure and the weight of the compactness indicator (C); slight changes in these parameters can lead to a significant change in the order of clusters, suggesting the need for careful calibration of thresholds and weights depending on the regional context.

To better illustrate the discussed differences, maps were used to present example results of the clustering algorithms, showing characteristic patterns of spatial parcel grouping. Map (Figure 6) demonstrates the basic operation of the K-Means algorithm.


**Figure 6.** Map illustrating the result of the K-Means algorithm (baseline implementation).

The clusters obtained with the baseline K-Means algorithm are characterized by lower spatial coherence and therefore reduced practical applicability. In contrast, the same map after applying the improved version of the algorithm (Figure 7) presents more coherent and functionally useful clusters.

The presented illustrations show an interactive map generated by the agent based on user feedback, visualizing land parcels that meet specific criteria. A significant improvement in result quality was observed when applying the enhanced algorithm, understood as increased operational usefulness and better alignment with the defined parameters. Through selective grouping, merging, and hierarchical organization of qualifying parcels, the system expands the range of available options and optimizes the selection process.

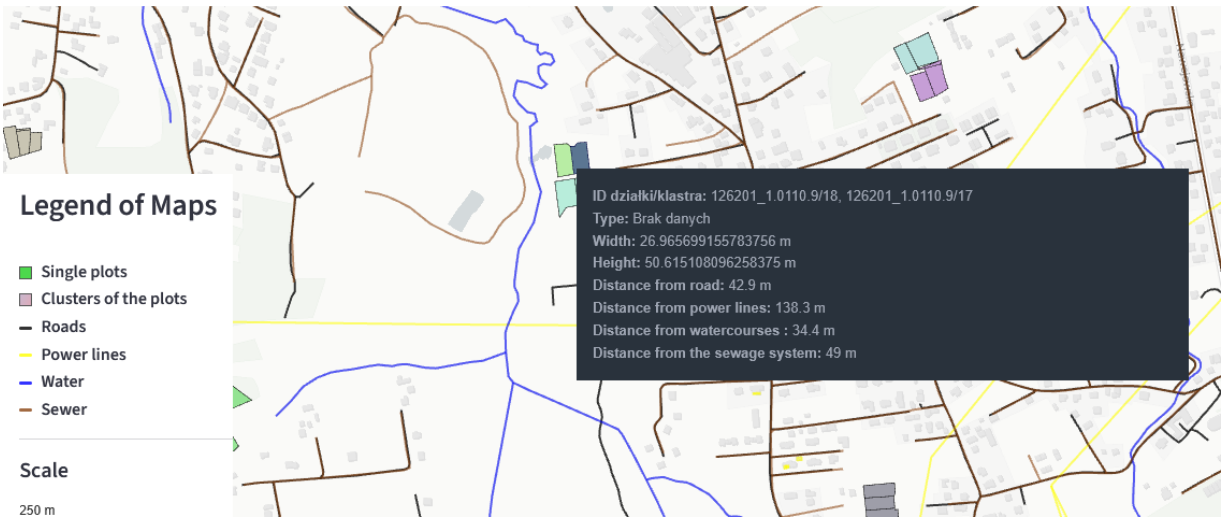


**Figure 7.** Map illustrating the result of the improved K-Means algorithm.

A key distinction of the proposed architecture from conventional GIS approaches lies in the organization and control of the analytical workflow. Traditional GIS platforms, predicated on rule-based operations and Multi-Criteria Decision Analysis (MCDA), necessitate advanced user expertise in query formulation, parameter configuration, and spatial result interpretation. In contrast, our approach utilizes reactive agents operating under the guidance of a large language model (LLM). The LLM is tasked with extracting and transforming user requirements into a structured set of formal signals, while the actual processing of spatial data is handled by a dedicated GIS agent. This division of labor overcomes the inefficiency of using LLMs for direct spatial computations and ensures control over legal requirement generation, as well as enabling algorithm optimization based on dataset size. This methodology not only automates site selection but also enhances its transparency and scalability, aligning with current research trends in integrating GIS with agent architectures and LLMs.

Comparable approaches to the application of advanced optimization models have been reported in the waste-to-energy sector, where polyoptimization of combustible fractions of municipal waste was used to improve the energy efficiency of biomass fuel production [99]. The results confirm that multi-criteria optimization methods and intelligent algorithms can significantly increase resource efficiency, which is consistent with the potential for integrating agent systems and GIS tools in biogas plant planning. Both approaches emphasize the importance of advanced analytical models for achieving higher efficiency and sustainable development in circular economy processes.

### *8.3. Limitations and Challenges*

Although the proposed method delivers promising results and accelerates the decision-making process, its practical application involves several limitations that must be considered:

1. Limitations of clustering-based approaches.

The main limitation of the method is the inability to generate new parcels beyond those defined in the cadastral database. In scenarios where individual plots fail to meet zero-level requirements (e.g., minimum distance from water resources), clustering is not performed, even if their combination could potentially form an area suitable for investment. This constraint narrows the spectrum of feasible locations. A potential direction for overcoming this limitation is the modification of the agent architecture to incorporate nesting algorithms, which would allow the creation of optimized, artificial parcels that meet regulatory and spatial criteria.

2. Regulatory and social acceptance issues.

While clustering provides a computationally efficient tool for spatial optimization, it does not inherently address legal frameworks or social constraints. Investment siting is strongly influenced by land-use regulations, environmental protection requirements, and public acceptance. In practice, even technically optimal clusters may be excluded due to zoning restrictions, Natura 2000 areas, or community opposition. Therefore, the presented method should be treated as a decision-support tool rather than a substitute for comprehensive regulatory and stakeholder analyses.

3. Risks of NLP-based legal interpretation.

The application of natural language processing (NLP) to interpret legal requirements introduces the risk of misclassification or oversimplification of regulations. Automatic parsing of legal texts may overlook exceptions, ambiguous clauses, or recent amendments, which could result in non-compliance. To mitigate this risk, the system should be combined with a validation stage involving legal experts and continuously updated rule sets. This hybrid approach would ensure greater transparency and reliability in translating legal provisions into operational constraints.

In the Polish legal framework, the siting of agricultural biogas plants is strictly linked to spatial planning and environmental protection requirements. According to national guidelines and EU directives, several quantitative criteria must be fulfilled. Facilities cannot be located within protected areas (e.g., Natura 2000 sites, landscape parks), while minimum buffer distances are required: typically 100–200 m from residential buildings, 30 m from surface water bodies, and 50 m from drinking water intakes. Storage of digestate must respect at least 25 m distance from drainage ditches or watercourses, and its agricultural use is limited by the Nitrate Directive to 170 kg N/ha/year. Moreover, access to the electricity or gas grid within a radius of 1–2 km and proximity to biomass sources (ideally less than 5–10 km) are considered decisive for economic and logistic feasibility. These regulatory and jurisdictional constraints significantly shape the decision-making criteria and are therefore crucial to incorporate into GIS-based site selection models.

In summary, while the proposed methodology accelerates site selection and enhances decision-making efficiency, its applicability requires careful consideration of spatial, regulatory, and social dimensions, as well as safeguards against potential misinterpretations of legal frameworks.

4. Transferability to other regions.

Although the method has been validated on datasets from two Polish case studies, its broader application requires caution. Assumptions regarding data completeness, cadastral structure, and regulatory thresholds may not hold in other regions or countries. Therefore, successful transfer depends on adapting regulatory layers, recalibrating distance thresholds, and validating the results against local infrastructural and environmental conditions.

5. Weighting and uncertainty representation.

At the current stage of research, only deterministic expert weighting was applied. While this provided a practical foundation for the evaluation process, it does not fully capture uncertainty in decision-making. Future work will incorporate fuzzy and probabilistic approaches, which are expected to better reflect uncertainty, enhance flexibility in stakeholder input interpretation, and improve the overall robustness of the evaluation framework.

### 8.4. Extension with Additional Evaluation Criteria

The proposed method has the character of a flexible "skeleton" that can be expanded with additional data layers and analytical criteria. Proper geospatial tagging enables their integration into the analysis process, the calculation of dedicated metrics, and their application within the agent algorithm. Examples of potential extensions include:

1. Greenhouse gas (GHG) emission reduction potential.

The database can be supplemented with emission indicators related to the use of substrates and the replacement of fossil fuels with biogas-based energy. This would allow for the calculation of potential $CO_2$eq savings for each siting scenario, which is particularly relevant in the context of climate policies and renewable energy support schemes.

2. Transport logistics.

Road network layers, together with information on substrate sources and energy consumers, can be used to assess transport costs and environmental footprint. Network analysis (e.g., shortest path) enables the evaluation of infrastructure accessibility and the minimization of transport distances for feedstock and energy products.

3. Environmental impact assessment (EIA).

The system can be enriched with layers representing protected areas (e.g., Natura 2000 sites, landscape parks), soil classes, or ecologically sensitive zones. This makes it possible to automatically exclude conflict-prone locations and to evaluate the degree of potential environmental interference.

Extending the method with these criteria will increase its practical usefulness and provide more comprehensive support for decision-making in sustainable energy investment planning.

### 8.5. Comparison of Plot Evaluation by the Traditional Method and the Agent-Based Model

To validate the proposed approach, a comparison was carried out between the plots assessed using the traditional expert-based method and those identified by the agent-based model. Due to the very large number of parcels in the database, a case study was conducted on a selected set of parcels located in district 110 in the city of Nowy Sącz—plots no. 9/17 and 9/18, with a total area of 0.2701 ha (Figure 8).

**Figure 8.** Presentation of the plot evaluated by the expert compared with the results obtained using the agent-based model. The area of the planned investment is marked in red.

The expert evaluation was based on criteria consistent with the materials presented in the article, including: width (50 m), height (54 m), distance from the road (43.2 m), distance from power lines (138 m), distance from watercourses (34 m), and distance from the sewage network (50 m). The analysis showed that the plots indicated by the expert strongly correlated with those identified by the agent-based system. The obtained results confirm the correctness of the proposed model and its practical applicability in supporting investment decision-making for biogas plant siting.

## 9. Conclusions and Future Research

The conducted study confirms that the integration of agent-based systems with GIS tools and large-scale data processing platforms such as Apache Spark and PostGIS provides effective support for the site selection of agricultural biogas plants. The results demonstrate that both architectures—the distributed (Spark) and the database-centered (PostGIS)—offer comparable performance when analyzing datasets of tens of thousands of plots, while maintaining acceptable response times (6–60 s). The clustering stage remains the most resource-intensive in terms of computational demand, yet it enables more optimal and coherent spatial planning proposals.

The inclusion of improved algorithms—such as enhanced DFS and K-Means variants—significantly increased the quality of the results by enabling more precise selection of neighboring plots and applying geometric optimization methods. As a result, the proposed solution not only automates the selection process but also enhances its transparency and usability for designers and decision-makers, eliminating the need for time-consuming manual work in classical GISs. In this way, agent-based systems combined with GISs are thus becoming fully functional instruments of environmental management, supporting both spatial planning and sustainable development policies [100,101].

The developed architecture, supported by a conversational interface powered by a large language model, lowers the entry barrier to geospatial and legal analysis, making it more accessible to users without advanced technical knowledge. In this way, the project aligns with the broader trend of digital transformation in decision-making processes within renewable energy and circular economy domains.

In conclusion, the presented agent-based system constitutes a viable alternative to traditional design procedures by combining speed, scalability, and interactivity with the ability to verify and visualize outcomes. Future research should focus on improving clustering efficiency, expanding the scope of analysis to include additional environmental and social criteria, and conducting pilot implementations in real-world investment processes.

In this context, particular attention should be given to the agglomerative clustering method due to the high qualitative usefulness of the obtained results. Furthermore, an important direction for future development is the concept of a hybrid agent which, based on changing legal conditions, could adapt its operations in an evolutionary manner while simultaneously significantly reducing the maintenance overhead of the information environment.

Another promising area of research is the use of biogas not only in the distributed energy sector, but also in industrial processes requiring high temperatures, such as metallurgy and foundry work. The use of biogas as a fuel in foundry furnaces or in metal heat treatment processes could significantly reduce the dependence of these industries on fossil fuels, while contributing to the reduction of greenhouse gas emissions. The integration of biogas plant location models with an analysis of potential industrial customers opens up the possibility of creating local energy-industrial clusters, in which agricultural and municipal waste becomes a resource supporting the sustainable transformation of metallurgical processes. Research in this area could include both technical and economic simulations as

well as experimental tests of the quality of biogas combustion under industrial conditions, which represents another step towards the fuller integration of the circular economy with traditional industrial sectors. Additionally, the development of technologies based on biogas and circular economy models may also contribute to reducing the demand for plastics and the pollution associated with them [102–104].

This study is subject to several limitations. Firstly, the experiments were conducted on two case-study regions in Poland, and the applicability of the approach may vary in other geographical or regulatory contexts. Secondly, the implemented clustering algorithms, although effective for medium-sized datasets, remain computationally intensive for very large-scale scenarios. Thirdly, the integration of legal rules through NLP carries the risk of misinterpretation of ambiguous clauses or recent amendments. Future research should therefore focus on improving algorithmic scalability, validating the system with diversified datasets from different regions, and incorporating hybrid legal–expert validation stages to enhance reliability and compliance.

In the production workflow, each algorithm plays a distinct role under specific time and resource constraints. The improved DFS provides a near real-time solution for ensuring spatial coherence of clusters at moderate computational cost, while the basic K-Means offers a very fast approximation useful for rapid prototyping. The improved K-Means balances execution time (~60 s) with significantly higher compactness, making it suitable for decision-making interfaces. Finally, agglomerative clustering, despite its long runtime, remains valuable for offline strategic planning, where accuracy and hierarchical structure are prioritized over response time. This diversity allows the agent-based system to adapt the choice of algorithm to the scale, purpose, and temporal requirements of the analysis.

**Author Contributions:** Conceptualization, A.K. and N.B.; methodology, A.K.; software, N.B.; validation, M.S., T.Z. and A.K.; formal analysis, A.K.; investigation, T.Z.; resources, A.K. and N.B.; data curation, N.B.; writing—original draft preparation, A.K., T.Z., N.B. and M.S.; writing—review and editing, A.K.,T.Z., N.B. and M.S. visualization, A.K. and N.B.; supervision, A.K. and M.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| AHP | Analytic Hierarchy Process |
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| APC | Article Processing Charge |
| BWM | Best–Worst Method |
| CE | Circular Economy |
| CLINK | Complete Linkage (hierarchical clustering algorithm) |
| CPU | Central Processing Unit |
| DDSS | Distributed Decision Support System |
| DFS | Depth-First Search |
| DSS | Decision Support System |

| | |
|---|---|
| EM | Environmental Management |
| ETL | Extract, Transform, Load |
| F-AHP | Fuzzy Analytic Hierarchy Process |
| GIS | Geographic Information System |
| IDSS | Intelligent Decision Support System |
| IoT | Internet of Things |
| JDBC | Java Database Connectivity |
| K-Means | K-Means Clustering Algorithm |
| LCA | Life-Cycle Assessment |
| LLM | Large Language Model |
| MCDA | Multi-Criteria Decision Analysis |
| ML | Machine Learning |
| MRR | Minimum Rotated Rectangle |
| NLP | Natural Language Processing |
| PV | Photovoltaics |
| QGIS | Quantum GIS (Free and Open-Source Geographic Information System) |
| RAM | Random Access Memory |
| RES | Renewable Energy Sources |
| REST | Representational State Transfer |
| R-Tree | Rectangle Tree (Spatial Index) |
| SLINK | Sequential Linkage (single-link hierarchical clustering algorithm) |
| SQL | Structured Query Language |
| TOPSIS | Technique for Order Preference by Similarity to Ideal Solution |
| VRAM | Video Random Access Memory |
| WebGL | Web Graphics Library |

## References


1. Kostopoulos, G.; Davrazos, G.; Kotsiantis, S. Explainable Artificial Intelligence-Based Decision Support Systems: A Recent Review. *Electronics* **2024**, *13*, 2842. [CrossRef]
2. Vélez Bedoya, J.I.; González Bedia, M.; Castillo Ossa, L.F. Intelligent Agents and Causal Inference: Enhancing Decision-Making through Causal Reasoning. *Appl. Sci.* **2024**, *14*, 3818. [CrossRef]
3. Kovari, A. AI for Decision Support: Balancing Accuracy, Transparency, and Trust Across Sectors. *Information* **2024**, *15*, 725. [CrossRef]
4. Fernandez, V. Environmental Management: Implications for Business Performance, Innovation, and Financing. *Technol. Forecast. Soc. Change* **2022**, *182*, 121797. [CrossRef]
5. do Amaral, M.R.; Willerding, I.A.V.; Lapolli, É.M. ESG Practices: The Key to Organizational Sustainability (Práticas ESG: A Chave para a Sustentabilidade Organizacional). *Concilium* **2024**, *24*, 85–107. [CrossRef]
6. Li, C.; Zhang, T.; Wang, X.; Lian, Z. Site Selection of Urban Parks Based on Fuzzy-Analytic Hierarchy Process (F-AHP): A Case Study of Nanjing, China. *Int. J. Environ. Res. Public Health* **2022**, *19*, 13159. [CrossRef]
7. Farsi, H.; Dizene, R.; Flamant, G.; Notton, G. Multi-Criteria Decision Making Methods for Suitable Site Selection of Concentrating Solar Power Plants. *Sustainability* **2024**, *16*, 7673. [CrossRef]
8. Rekik, S.; Khabbouchi, I.; El Alimi, S. A Spatial Analysis for Optimal Wind Site Selection from a Sustainable Supply-Chain-Management Perspective. *Sustainability* **2025**, *17*, 1571. [CrossRef]
9. Kochanek, A.; Ciuła, J.; Cembruch-Nowakowski, M.; Zacłona, T. Polish Farmers' Perceptions of the Benefits and Risks of Investing in Biogas Plants and the Role of GISs in Site Selection. *Energies* **2025**, *18*, 3981. [CrossRef]
10. Ciuła, J.; Gaska, K.; Siedlarz, D.; Koval, V. Management of Sewage Sludge Energy Use with the Application of Bifunctional Bioreactor as an Element of Pure Production in Industry. In *E3S Web of Conferences*; EDP Sciences: Les Ulis, France, 2019; Volume 123, p. 01016. [CrossRef]
11. Department of Forestry and Rural Development. *An Introduction to the Geo-Information System of the Canada Land Inventory*; Department of Forestry and Rural Development: Ottawa, ON, Canada, 1967; p. 46. Available online: https://gisandscience.wordpress.com/wp-content/uploads/2014/02/3-an-introduction-to-the-geo-information-system-of-the-canada-land-inventory_complete.pdf (accessed on 5 August 2025).
12. Zhu, J.; Wu, P. Towards Effective BIM/GIS Data Integration for Smart City by Integrating Computer Graphics Technique. *Remote Sens.* **2021**, *13*, 1889. [CrossRef]

13. Bordbar, M.; Shahabi, H.; Chapi, K.; Kariminejad, N.; Deo, R.C.; Ahmad, A.; Rahmati, O.; Pham, B.T.; Bui, D.T.; Tien Bui, D.; et al. Multi-Hazard Spatial Modeling via Ensembles of Machine Learning Algorithms (Earthquake, Flood, Landslide). *Sci. Rep.* **2022**, *12*, 2115–2134. [CrossRef] [PubMed]
14. Miller, A.; Li, R. A Geospatial Approach for Prioritizing Wind Farm Development in Northeast Nebraska, USA. *ISPRS Int. J. Geo-Inf.* **2014**, *3*, 968–979. [CrossRef]
15. Moltames, R.; Naghavi, M.S.; Silakhori, M.; Noorollahi, Y.; Yousefi, H.; Hajiaghaei-Keshteli, M.; Azizimehr, B. Multi-Criteria Decision Methods for Selecting a Wind Farm Site Using a Geographic Information System (GIS). *Sustainability* **2022**, *14*, 14742. [CrossRef]
16. Demir, A.; Dinçer, A.E.; Çiftçi, C.; Gülçimen, S.; Uzal, N.; Yılmaz, K. Wind Farm Site Selection Using GIS-Based Multicriteria Analysis with Life-Cycle Assessment Integration. *Earth Sci. Inform.* **2024**, *17*, 1591–1608. [CrossRef]
17. Kwaśnicki, P.; Gronba-Chyła, A.; Generowicz, A.; Ciuła, J.; Makara, A.; Kowalski, Z. Characterization of the TCO Layer on a Glass Surface for PV IInd and IIIrd Generation Applications. *Energies* **2024**, *17*, 3122. [CrossRef]
18. Ashraf, H.A.; Li, J.; Li, Z.; Sohail, A.; Ahmed, R.; Butt, M.H.; Ullah, H. Geographic Information System and Machine Learning Approach for Solar Photovoltaic Site Selection: A Case Study in Pakistan. *Processes* **2025**, *13*, 981. [CrossRef]
19. Adhikari, M.D.; Yune, C.-Y. Geospatial-Based Risk Analysis of Solar Plants Located in the Mountainous Region of Gangwon Province, South Korea. *Renew. Energy* **2025**, *251*, 123408. [CrossRef]
20. He, Z.; Xu, W.; Sun, Y.; Zhang, X. A GIS-Based Techno-Economic Comparative Assessment of Offshore Fixed and Floating Photovoltaic Systems: A Case Study of Hainan. *Appl. Energy* **2025**, *391*, 125854. [CrossRef]
21. Chukwuma, E.C.; Onyesolu, F.C.O.; Ani, K.A.; Nwanna, E.C. GIS Bio Waste Assessment and Suitability Analysis for Biogas Power Plant: A Case Study of Anambra State of Nigeria. *Renew. Energy* **2021**, *163*, 1182–1194. [CrossRef]
22. Fiehl, M.; Leicher, J.; Giese, A.; Görner, K.; Fleischmann, B.; Spielmann, S. Biogas as a Co-Firing Fuel in Thermal Processing Industries: Implementation in a Glass Melting Furnace. *Energy Procedia* **2017**, *120*, 302–308. [CrossRef]
23. Mesthrige, T.G.; Kaparaju, P. Decarbonisation of Natural Gas Grid: A Review of GIS-Based Approaches on Spatial Biomass Assessment, Plant Siting and Biomethane Grid Injection. *Energies* **2025**, *18*, 734. [CrossRef]
24. Uyan, M.; Ertunç, E. GIS-Based Optimal Site Selection of the Biogas Facility Installation Using the Best-Worst Method. *Chem. Eng. Res. Des.* **2023**, *192*, 1003–1011. [CrossRef]
25. Michalski, K.; Kośka-Wolny, M.; Chmielowski, K.; Bedla, D.; Petryk, A.; Guzdek, P.; Dąbek, K.A.; Gąsiorek, M.; Grübel, K.; Halecki, W. Examining the Potential of Biogas: A Pathway from Post-Fermented Waste into Energy in a Wastewater Treatment Plant. *Energies* **2024**, *17*, 5618. [CrossRef]
26. Petryk, A.; Czop, M.; Pohrebennyk, V. The Assessment of the Degree of Pollution of Fallow Vegetation with Heavy Metals in Rural Administrative Units of Psary and Płoki in Poland. In Proceedings of the 18th International Multidisciplinary Scientific Geoconference SGEM, Albena, Bulgaria, 2–8 July 2018; Volume 18, pp. 921–928. [CrossRef]
27. Korkovelos, A.; Mentis, D.; Siyal, S.H.; Arderne, C.; Rogner, H.; Bazilian, M.; Howells, M.; Beck, H.; De Roo, A. A Geospatial Assessment of Small-Scale Hydropower Potential in Sub-Saharan Africa. *Energies* **2018**, *11*, 3100. [CrossRef]
28. Chiu, Y.-R.; Tsai, Y.-L.; Chiang, Y.-C. Designing Rainwater Harvesting Systems Cost-Effectively in an Urban Water-Energy Saving Scheme by Using a GIS-Simulation Based Design System. *Water* **2015**, *7*, 6285–6300. [CrossRef]
29. Ciuła, J.; Sobiecka, E.; Zacłona, T.; Rydwańska, P.; Oleksy-Gębczyk, A.; Olejnik, T.P.; Jurkowski, S. Management of the Municipal Waste Stream: Waste into Energy in the Context of a Circular Economy—Economic and Technological Aspects for a Selected Region in Poland. *Sustainability* **2024**, *16*, 6493. [CrossRef]
30. Piatkowski, J.; Nowinska, K.; Matula, T.; Siwiec, G.; Szucki, M.; Oleksiak, B. Microstructure and Mechanical Properties of AlSi10MnMg Alloy with Increased Content of Recycled Scrap. *Materials* **2025**, *18*, 1119. [CrossRef]
31. Malinowski, M.; Guzdek, S.; Petryk, A.; Tomaszek, K. A GIS and AHP-Based Approach to Determine Potential Locations of Municipal Solid Waste Collection Points in Rural Areas. *J. Water Land Dev.* **2021**, *51*, 94–101. [CrossRef]
32. Kochanek, A.; Ciuła, J.; Generowicz, A.; Mitryasova, O.; Jasińska, A.; Jurkowski, S.; Kwaśnicki, P. The Analysis of Geospatial Factors Necessary for the Planning, Design, and Construction of Agricultural Biogas Plants in the Context of Sustainable Development. *Energies* **2024**, *17*, 5619. [CrossRef]
33. Kafel, P.; Nowicki, P. Circular Economy Implementation Based on ISO 14001 within SME Organization: How to Do It Best? *Sustainability* **2023**, *15*, 496. [CrossRef]
34. Plinke, M.; Berndmeyer, J.; Hack, J. Development of a GIS-Based Register of Biogas Plant Sites in Lower Saxony, Germany: A Foundation for Identifying P2G Potential. *Energy Sustain. Soc.* **2025**, *15*, 7. [CrossRef]
35. Heck, R.; Rudi, A.; Lauth, D.; Schultmann, F. An Estimation of Biomass Potential and Location Optimization for Integrated Biorefineries in Germany: A Combined Approach of GIS and Mathematical Modeling. *Sustainability* **2024**, *16*, 6781. [CrossRef]
36. Safari Bazargani, J.; Sadeghi-Niaraki, A.; Choi, S.-M. A Survey of GIS and IoT Integration: Applications and Architecture. *Appl. Sci.* **2021**, *11*, 10365. [CrossRef]
37. Calka, B.; Szostak, M. GIS-Based Environmental Monitoring and Analysis. *Appl. Sci.* **2025**, *15*, 3155. [CrossRef]

38. Kochanek, A.J.; Kobylarczyk, S. The Analysis of the Main Geospatial Factors Using Geoinformation Programs Required for the Planning, Design and Construction of a Photovoltaic Power Plant. *J. Ecol. Eng.* **2024**, *25*, 49–65. [CrossRef]
39. Koko, M.N.; Eleyi, F.S. Impact of Financial Reporting on Management Decision Making in Rivers State Owned Universities. *Glob. J. Hum. Soc. Sci.* **2020**, *20*, 49–54. Available online: https://socialscienceresearch.org/index.php/GJHSS/article/view/3159 (accessed on 7 August 2025).
40. Pušeljić, M.; Skledar, A.; Pokupec, I. Decision Making as a Management Function. *Interdiscip. Manag. Res.* **2015**, *11*, 234–244.
41. Citroen, C.L. The Role of Information in Strategic Decision-Making. *Int. J. Inf. Manag.* **2011**, *31*, 493–501. [CrossRef]
42. Núñez-Valdez, E.R. Special Issue on Algorithms in Decision Support Systems Vol.2. *Algorithms* **2023**, *16*, 512. [CrossRef]
43. Kabir, G.; Sadiq, R.; Tesfamariam, S. A Review of Multi-Criteria Decision-Making Methods for Infrastructure Management. *Struct. Infrastruct. Eng.* **2014**, *10*, 1176–1210. [CrossRef]
44. Bączkiewicz, A.; Wątróbski, J.; Kizielewicz, B.; Sałabun, W. Towards Objectification of Multi-Criteria Assessments: A Comparative Study on MCDA Methods. In Proceedings of the 16th Conference on Computer Science and Intelligence Systems (FedCSIS), Online, 2–5 September 2021; pp. 417–425. [CrossRef]
45. Taherdoost, H.; Madanchian, M. Multi-Criteria Decision Making (MCDM) Methods and Concepts. *Encyclopedia* **2023**, *3*, 77–87. [CrossRef]
46. Mohamed, A.-M.O.; Mohamed, D.; Fayad, A.; Al Nahyan, M.T. Enhancing Decision Making and Decarbonation in Environmental Management: A Review on the Role of Digital Technologies. *Sustainability* **2024**, *16*, 7156. [CrossRef]
47. Almadani, B.; Kaisar, H.; Thoker, I.R.; Aliyu, F. A Systematic Survey of Distributed Decision Support Systems in Healthcare. *Systems* **2025**, *13*, 157. [CrossRef]
48. Keenan, P.B. A Scientometric Analysis of Multicriteria Decision-Making Research. *J. Decis. Syst.* **2024**, *33* (Suppl. S1), 78–88. [CrossRef]
49. Alam Bhuiyan, M.M.; Hammad, A. A Hybrid Multi-Criteria Decision Support System for Selecting the Most Sustainable Structural Material for a Multistory Building Construction. *Sustainability* **2023**, *15*, 3128. [CrossRef]
50. Ali, R.; Hussain, A.; Nazir, S.; Khan, S.; Khan, H.U. Intelligent Decision Support Systems—An Analysis of Machine Learning and Multicriteria Decision-Making Methods. *Appl. Sci.* **2023**, *13*, 12426. [CrossRef]
51. Nassef, A.M.; Abdelkareem, M.A.; Maghrabie, H.M.; Baroutaji, A. Review of Metaheuristic Optimization Algorithms for Power Systems Problems. *Sustainability* **2023**, *15*, 9434. [CrossRef]
52. Bouaouda, A.; Sayouti, Y. Hybrid Meta-Heuristic Algorithms for Optimal Sizing of Hybrid Renewable Energy System: A Review of the State-of-the-Art. *Arch. Comput. Methods Eng.* **2022**, *29*, 4049–4083. [CrossRef]
53. Sanin-Villa, D.; Figueroa-Saavedra, H.A.; Grisales-Noreña, L.F. Efficient BESS Scheduling in AC Microgrids via Multiverse Optimizer: A Grid-Dependent and Self-Powered Strategy to Minimize Power Losses and $CO_2$ Footprint. *Appl. Syst. Innov.* **2025**, *8*, 85. [CrossRef]
54. Sicuaio, T.; Zhao, P.; Pilesjö, P.; Shindyapin, A.; Mansourian, A. A Multi-Objective Optimization Approach for Solar Farm Site Selection: Case Study in Maputo, Mozambique. *Sustainability* **2024**, *16*, 7333. [CrossRef]
55. Shao, M.; Mao, Z.; Sun, J.; Guan, X.; Shao, Z.; Tang, T. Multi-Scale Offshore Wind Farm Site Selection Decision Framework Based on GIS, MCDM and Meta-Heuristic Algorithm. *Ocean. Eng.* **2025**, *316*, 119921. [CrossRef]
56. Olan, F.; Spanaki, K.; Ahmed, W.; Zhao, G. Enabling Explainable Artificial Intelligence Capabilities in Supply Chain Decision Support Making. *Prod. Plan. Control.* **2024**, *36*, 808–819. [CrossRef]
57. Shool, S.; Adimi, S.; Amleshi, R.S.; Bitaraf, E.; Golpira, R.; Tara, M. A Systematic Review of Large Language Model (LLM) Evaluations in Clinical Medicine. *BMC Med. Inform. Decis. Mak.* **2025**, *25*, 117. [CrossRef]
58. Soori, M.; Karimi Ghaleh Jough, F.; Dastres, R.; Arezoo, B. AI-Based Decision Support Systems in Industry 4.0, a Review. *J. Econ. Technol.* **2024**, *in press*. [CrossRef]
59. Handler, A.; Larsen, K.R.; Hackathorn, R. Large Language Models Present New Questions for Decision Support. *Int. J. Inf. Manag.* **2024**, *76*, 102811. [CrossRef]
60. Bharti, M. AI Agents: A Systematic Review of Architectures, Components, and Evolutionary Trajectories in Autonomous Digital Systems. *Int. J. Comput. Eng. Technol.* **2025**, *16*, 653–664. [CrossRef]
61. Maroto-Gómez, M.; Alonso-Martín, F.; Malfaz, M.; Salichs, M.A. A Systematic Literature Review of Decision-Making and Control Systems for Autonomous and Social Robots. *Int. J. Soc. Robot.* **2023**, *15*, 745–789. [CrossRef]
62. Miller, T.; Durlik, I.; Kostecka, E.; Kozlovska, P.; Łobodzińska, A.; Sokołowska, S.; Nowy, A. Integrating Artificial Intelligence Agents with the Internet of Things for Enhanced Environmental Monitoring: Applications in Water Quality and Climate Data. *Electronics* **2025**, *14*, 696. [CrossRef]
63. Rousseas, P.; Bechlioulis, C.; Kyriakopoulos, K. Reactive Optimal Motion Planning for a Class of Holonomic Planar Agents Using Reinforcement Learning with Provable Guarantees. *Front. Robot. AI* **2024**, *10*, 1255696. [CrossRef]
64. Tang, S.; He, B.; Yu, C.; Li, Y.; Li, K. A Survey on Spark Ecosystem: Big Data Processing Infrastructure, Machine Learning, and Applications. *IEEE Trans. Knowl. Data Eng.* **2022**, *34*, 71–91. [CrossRef]
65. Obe, R.O.; Hsu, L.S. *PostGIS in Action*, 3rd ed.; Manning Publications: Shelter Island, NY, USA, 2021; ISBN 9781617296697.

66. Zeng, Y.; Brown, C.; Raymond, J.; Byari, M.; Hotz, R.; Rounsevell, M. Exploring the opportunities and challenges of using large language models to represent institutional agency in land system modelling. *Earth Syst. Dyn.* **2025**, *16*, 423–449. [CrossRef]
67. Afroogh, S.; Akbari, A.; Malone, E.; Kargar, M.; Alambeigi, H. Trust in AI: Progress, Challenges, and Future Directions. *Humanit. Soc. Sci. Commun.* **2024**, *11*, 1568. [CrossRef]
68. Sebastião, S.P.; Dias, D.F.-M. AI Transparency: A Conceptual, Normative, and Practical Frame Analysis. *Media Commun.* **2025**, *13*, 9419. [CrossRef]
69. Cormen, T.H.; Leiserson, C.E.; Rivest, R.L.; Stein, C. *Introduction to Algorithms*, 3rd ed.; MIT Press: Cambridge, MA, USA, 2009.
70. Aho, A.V.; Hopcroft, J.E.; Ullman, J.D. *Data Structures and Algorithms*; Addison-Wesley: Boston, MA, USA, 1983.
71. Riansanti, O.; Ihsan, M.; Suhaimi, D. Connectivity Algorithm with Depth First Search (DFS) on Simple Graphs. *J. Phys. Conf. Ser.* **2018**, *948*, 012065. [CrossRef]
72. Ng, R.T.; Han, J. Efficient and Effective Clustering Methods for Spatial Data Mining. In Proceedings of the 1994 VLDB Conference, Santiago de Chile, Chile, 12–15 September 1994; Morgan Kaufmann: Santiago, Chile, 1994; pp. 144–155.
73. Xie, K.; Wang, T.; Zhong, P.; Zhao, Z.; Wang, Z. Human Clustering Based on Graph Embedding and Space Functions of Trajectory Stay Points on Campus. *Appl. Sci.* **2025**, *15*, 3090. [CrossRef]
74. Yan, X.; Han, J. gSpan: Graph-Based Substructure Pattern Mining. In Proceedings of the 2002 IEEE International Conference on Data Mining, Maebashi City, Japan, 9–12 December 2002; IEEE: Maebashi, Japan, 2002; pp. 721–724. [CrossRef]
75. Washio, T.; Motoda, H. State of the Art of Graph-Based Data Mining. *SIGKDD Explor.* **2003**, *5*, 59–68. [CrossRef]
76. Rokach, L.; Maimon, O. Clustering Methods. In *The Data Mining and Knowledge Discovery Handbook*; Maimon, O., Rokach, L., Eds.; Springer: Boston, MA, USA, 2005; pp. 321–352. [CrossRef]
77. Jain, A.K.; Murty, M.N.; Flynn, P.J. Data Clustering: A Review. *ACM Comput. Surv.* **1999**, *31*, 264–323. [CrossRef]
78. Everitt, B.S.; Landau, S.; Leese, M.; Stahl, D. *Cluster Analysis*, 5th ed.; Wiley: Chichester, UK, 2011.
79. Kaufman, L.; Rousseeuw, P.J. *Finding Groups in Data: An Introduction to Cluster Analysis*; Wiley: New York, NY, USA, 1990; ISBN 0-471-87876-6. [CrossRef]
80. Defays, D. An Efficient Algorithm for a Complete Link Method. *Comput. J.* **1977**, *20*, 364–366. [CrossRef]
81. Murtagh, F.; Contreras, P. Algorithms for Hierarchical Clustering: An Overview. *WIREs Data Min. Knowl. Discov.* **2012**, *2*, 86–97. [CrossRef]
82. Jain, A.K. Data Clustering: 50 Years Beyond K-Means. *Pattern Recognit. Lett.* **2010**, *31*, 651–666. [CrossRef]
83. Hartigan, J.A.; Wong, M.A. Algorithm AS 136: A K-Means Clustering Algorithm. *J. R. Stat. Soc. C Appl. Stat.* **1979**, *28*, 100–108. [CrossRef]
84. MacQueen, J. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*; University of California Press: Berkeley, CA, USA, 1967; pp. 281–297.
85. Arthur, D.; Vassilvitskii, S. K-Means++: The Advantages of Careful Seeding. In Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), New Orleans, LA, USA, 7–9 January 2007; SIAM: New Orleans, LA, USA, 2007; pp. 1027–1035.
86. Xu, R.; Wunsch, D. Survey of Clustering Algorithms. *IEEE Trans. Neural Netw.* **2005**, *16*, 645–678. [CrossRef]
87. Halkidi, M.; Batistakis, Y.; Vazirgiannis, M. Cluster Validity Methods: Part I. *SIGMOD Rec.* **2001**, *31*, 40–45. [CrossRef]
88. Python Software Foundation. Python Programming Language. Available online: https://www.python.org/ (accessed on 17 August 2025).
89. Streamlit Inc. Streamlit—The Fastest Way to Build and Share Data Apps. Available online: https://streamlit.io/ (accessed on 17 August 2025).
90. PostgreSQL Polska. PostgreSQL—System Zarządzania Relacyjnymi Bazami Danych. Available online: https://www.postgresql.org.pl/ (accessed on 17 August 2025).
91. QGIS Development Team. QGIS—A Free and Open Source Geographic Information System. Available online: https://qgis.org/ (accessed on 17 August 2025).
92. Apache Software Foundation. Apache Spark™—Unified Analytics Engine for Big Data. Available online: https://spark.apache.org/ (accessed on 17 August 2025).
93. GeoPandas Developers. GeoPandas—Python Tools for Geographic Data. Available online: https://geopandas.org/en/stable/ (accessed on 17 August 2025).
94. Scikit-Learn Developers. Scikit-Learn: Machine Learning in Python. Available online: https://scikit-learn.org/stable/ (accessed on 17 August 2025).
95. Uber Technologies Inc. Deck.gl—WebGL-Powered Framework for Large-Scale Data Visualization. Available online: https://deckgl.readthedocs.io/en/latest/ (accessed on 17 August 2025).
96. Docker Inc. Docker—Develop, Ship, and Run Applications. Available online: https://www.docker.com/ (accessed on 17 August 2025).
97. Ollama Inc. Ollama—Run Large Language Models Locally. Available online: https://ollama.com/ (accessed on 17 August 2025).
98. Lee, H.M.; Yadav, D.; Lee, S.; Govindarazan, K.; Chen, C.; Sundar, S.S. While We Wait…How Users Perceive Waiting Times and Generation Cues during AI Image Generation. In Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA'25), Yokohama, Japan, 26 April–1 May 2025; ACM: New York, NY, USA, 2025; p. 602; 8p. [CrossRef]

99. Gaska, K.; Generowicz, A.; Lobur, M.; Jaworski, N.; Ciuła, J.; Vovk, M. Advanced Algorithmic Model for Poly-Optimization of Biomass Fuel Production from Separate Combustible Fractions of Municipal Wastes as a Progress in Improving Energy Efficiency of Waste Utilization. *E3S Web Conf.* **2019**, *122*, 01004. [CrossRef]
100. Kochanek, A.; Generowicz, A.; Zacłona, T. The Role of Geographic Information Systems in Environmental Management and the Development of Renewable Energy Sources—A Review Approach. *Energies* **2025**, *18*, 4740. [CrossRef]
101. Wereda, W.; Zacłona, T. Shaping the Image as a Management Instrument in the Contemporary Enterprise. In *Scientific Papers of Silesian University of Technology, Organization and Management, Series 145*; Silesian University of Technology Publishing House: Gliwice, Poland, 2020; pp. 597–611. [CrossRef]
102. Kochanek, A.; Grąz, K.; Potok, H.; Gronba-Chyła, A.; Kwaśny, J.; Wiewiórska, I.; Ciuła, J.; Basta, E.; Łapiński, J. Micro- and Nanoplastics in the Environment: Current State of Research, Sources of Origin, Health Risks, and Regulations—A Comprehensive Review. *Toxics* **2025**, *13*, 564. [CrossRef]
103. Gronba-Chyła, A.; Generowicz, A.; Kwaśnicki, P.; Kochanek, A. Recovery and Recycling of Selected Waste Fractions with a Grain Size Below 10 mm. *Sustainability* **2025**, *17*, 1612. [CrossRef]
104. Kochanek, A.; Janczura, J.; Jurkowski, S.; Zacłona, T.; Gronba-Chyła, A.; Kwaśnicki, P. The Analysis of Exhaust Composition Serves as the Foundation of Sustainable Road Transport Development in the Context of Meeting Emission Standards. *Sustainability* **2025**, *17*, 3420. [CrossRef]