

Article

LLM Agents for Smart City Management: Enhancing Decision Support Through Multi-Agent AI Systems

Anna Kalyuzhnaya ^{*}, Sergey Mityagin , Elizaveta Lutsenko, Andrey Getmanov , Yaroslav Aksenin, Kamil Fatkhiev , Kirill Fedorin, Nikolay O. Nikitin , Natalia Chichkova , Vladimir Vorona  and Alexander Boukhanovsky

AI Institute, ITMO University, 197101 Saint-Petersburg, Russia; mityagin@itmo.ru (S.M.); nnikitin@itmo.ru (N.O.N.); nachichkova@niuitmo.ru (N.C.); v.vorona@itmo.ru (V.V.)

* Correspondence: anna.kalyuzhnaya@itmo.ru;

Highlights:

What are the main findings?

- For smart city management, LLM-based multi-agent systems achieve 94–99% accuracy in routing urban queries and demonstrate significant improvements in response quality (G-Eval scores of 0.68–0.74) compared to standalone LLMs (0.30–0.38).
- Achievement of high scores in routing queries and response accuracy is possible with middle-size LLM models rather than the biggest LLM models.

What is the implication of the main findings?

- The multi-agent LLM approach enables efficient processing of complex urban planning tasks while maintaining high relevance in responses, making it practical for real-world city management applications.
- LLM agents can effectively augment human decision making in urban planning by reducing task completion time from days to hours while maintaining accuracy and accountability in complex scenarios.



Academic Editors: Jerry Gao, Fanjing Meng, Iraklis Varlamis, Paul Townend and Pierluigi Siano

Received: 23 October 2024

Revised: 18 January 2025

Accepted: 21 January 2025

Published: 24 January 2025

Citation: Kalyuzhnaya, A.; Mityagin, S.; Lutsenko, E.; Getmanov, A.; Aksenin, Y.; Fatkhiev, K.; Fedorin, K.; Nikitin, N.; Chichkova, N.; Vorona, V.; et al. LLM Agents for Smart City Management: Enhancing Decision Support Through Multi-Agent AI Systems. *Smart Cities* **2025**, *8*, 19. <https://doi.org/10.3390/smartcities8010019>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Abstract: This study investigates the implementation of LLM agents in smart city management, leveraging both the inherent language processing abilities of LLMs and the distributed problem solving capabilities of multi-agent systems for the improvement of urban decision making processes. A multi-agent system architecture combines LLMs with existing urban information systems to process complex queries and generate contextually relevant responses for urban planning and management. The research is focused on three main hypotheses testing: (1) LLM agents' capability for effective routing and processing diverse urban queries, (2) the effectiveness of Retrieval-Augmented Generation (RAG) technology in improving response accuracy when working with local knowledge and regulations, and (3) the impact of integrating LLM agents with existing urban information systems. Our experimental results, based on a comprehensive validation dataset of 150 question–answer pairs, demonstrate significant improvements in decision support capabilities. The multi-agent system achieved pipeline selection accuracy of 94–99% across different models, while the integration of RAG technology improved response accuracy by 17% for strategic development queries and 55% for service accessibility questions. The combined use of document databases and service APIs resulted in the highest performance metrics (G-Eval scores of 0.68–0.74) compared to standalone LLM responses (0.30–0.38). Using St. Petersburg's Digital Urban Platform as a testbed, we demonstrate the practical applicability of this approach to create integrated city management systems with support complex urban decision making processes. This research contributes to the growing field of AI-enhanced urban management by providing empirical evidence of LLM agents' effectiveness in processing heterogeneous urban data and supporting strategic planning

decisions. Our findings suggest that LLM-based multi-agent systems can significantly enhance the efficiency and accuracy of urban decision making while maintaining high relevance in responses.

Keywords: large language model; LLM; LLM agent; multi-agent system; smart city management; data-driven management; strategic management

1. Introduction

Cities implementing advanced urban technologies face diverse challenges that vary with their specific contexts. While modern Large Language Models (LLMs) offer promising capabilities for addressing these challenges, they have inherent limitations when handling specialized domain knowledge and real-time operational data. To overcome these constraints, we propose and experimentally evaluate a multi-agent LLM-based solution for smart city strategic management. This approach leverages LLMs' strength in processing unstructured data while addressing their limitations through a distributed agent architecture, offering significant advantages over standalone LLM.

1.1. Smart City Management Background and Challenges

The strategic, operational, and operative levels of urban management form a complex, interconnected structure where the outputs and challenges of one level directly influence the others (Figure 1). For instance, strategic management establishes long-term objectives and frameworks, while operational management focuses on translating these objectives into actionable policies and programs [1,2].

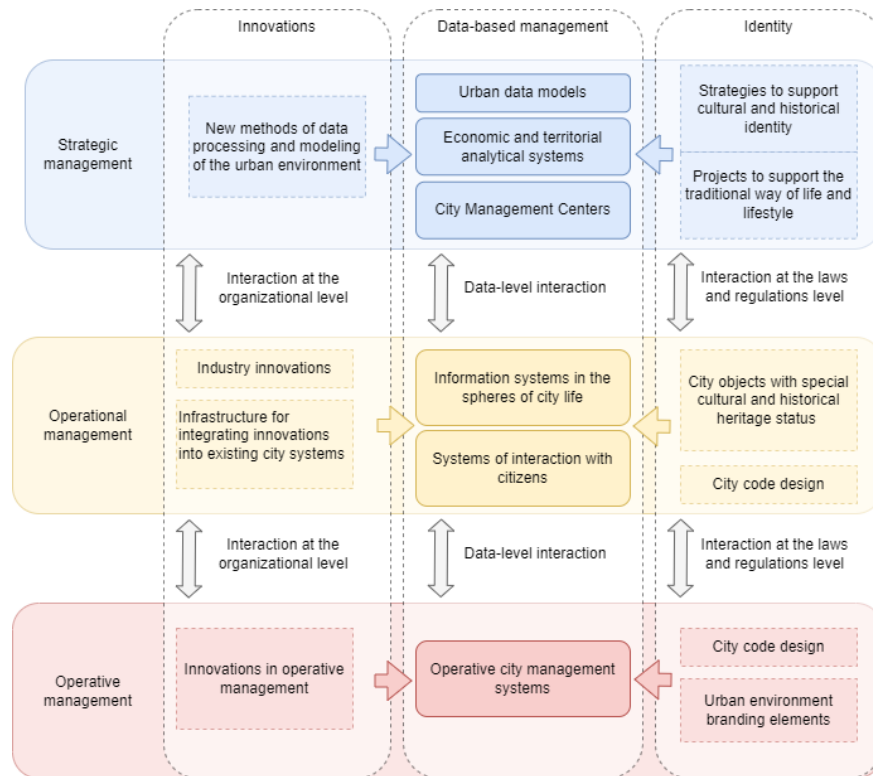


Figure 1. The scheme of interaction between management levels and components of a smart city.

Operative management, in turn, ensures the execution of these programs in real time and handles immediate needs. Notable examples include traffic management systems [3],

city safety systems [4,5], and public health monitoring [6]. This bidirectional relationship ensures that strategic plans are rooted in the realities of daily urban operations, while operational and operative actions align with overarching strategic goals.

On the strategic management level, long-term planning (typically over a 10-year horizon) is achieved through the implementation of innovative approaches, data-driven governance, and the preservation of urban identity. This level identifies three principal areas of focus: the development of novel data processing methodologies, the formulation of urban data models, and the formulation of strategies for the maintenance of cultural and historical identity. To support the interconnections of the strategic level with other management levels, various urban information systems have emerged. Such systems are well known and widely used in smart city management and data-driven decision making. There are universal systems that provide data aggregation such as Urban Platform City [7] and the City Digital Data platform [8], as well as CityEngine (ESRI) [9], Urban Observatory [10], and UrbanSim [11].

In the framework of this article, the integration of LLMs with urban information systems can be examined through the example of the Digital Urban Platform of St. Petersburg. This platform serves as an experimental testbed, demonstrating the potential of LLMs for urban strategic management.

Challenges in data processing and decision making in smart city management processes are primarily determined by the lifecycle of urban data [12,13]. Smart city management faces significant data processing and decision making challenges, particularly at the strategic level. While operative and operational data management varies based on city-specific digital maturity, strategic-level challenges are more universal across major cities, warranting focused attention. A, B, C, D, and E represent common data challenges at the strategic management level across major cities, which is why we focus on this level.

- (A) Data Fragmentation. Strategic decision making is hindered by incomplete datasets repurposed from their original operational contexts, creating critical information gaps.
- (B) Domain Evaluation Inconsistency. Varying data coverage across urban sectors impedes uniform situation assessment and comparison.
- (C) Development Bias. Data-rich sectors receive disproportionate attention in smart city initiatives, potentially marginalizing important but less digitally documented areas.
- (D) Automation Constraints. Complex urban decision making faces limitations due to the challenge of reconciling digital and document-based data formats, necessitating human expert interpretation.
- (E) Cross-level Data Misalignment. Inappropriate application of operational data to strategic planning creates decision making inconsistencies between management levels.

The implementation of the proposed multi-agent approach reflects interconnections of urban management levels (Figure 1) through its modular and hierarchical design. For example, the strategic level leverages the system's geospatial and analytical capabilities, which identify optimal resource allocation to meet immediate and mid-term needs, to support data-driven decisions for long-term urban planning. These outputs are integrated with operational-level functions, which calculate the changes in service provision levels resulting from the construction of a new facility in the selected block.

1.2. Research Objectives and Hypotheses

In this framework, we aim to show that the use of an intelligent decision support system in urban environment management based on LLM agents (in distinction from standalone LLM models) allows for processing complex queries, analyzing heterogeneous data, and generating meaningful responses for various categories of users, from city

residents to city government, approaching the quality of human work while significantly surpassing it in speed. It offers a new perspective on how cities can leverage advanced language models to enhance the interpretation and application of urban data, potentially leading to more informed and responsive urban planning and management strategies.

The scientific novelty of this research in the field of urban studies and smart city management lies in its demonstration of how an LLM-based multi-agent system can solve specific smart city management tasks. The study reveals new insights into the integration of artificial intelligence with urban governance, showing how LLMs can organize and structure diverse urban data sources, including spatial, demographic, and regulatory information, to provide comprehensive and context-aware responses to complex urban queries. Furthermore, the research provides novel aspects of the capabilities and limitations of AI in urban decision support, contributing to the ongoing discourse on the role of technology in shaping urban futures.

We attempted to explore these ideas through testing several main research hypotheses in our research:

1. Hypothesis 1. LLM agents are capable of effectively routing and processing diverse user queries related to the urban environment and social infrastructure, accessing relevant services and databases.
2. Hypothesis 2. The application of RAG (Retrieval-Augmented Generation) technology in combination with LLMs improves the quality and reliability of generated responses, especially when working with local knowledge about the city and regulatory documents.
3. Hypothesis 3. The integration of LLMs with existing urban information systems and services (e.g., social benefits availability service, transport accessibility service) enables the generation of more accurate and contextually relevant responses to user queries.

2. Related Work

2.1. AI Applications in Smart Cities

Smart cities actively adopt new technologies, with AI being a key part of their technological framework. AI is most commonly used in day-to-day operations and short-term management (operative and operational level) of smart cities, mainly because these areas have clearer, more structured processes.

AI technologies in smart cities primarily focus on the following areas [14–16]:

- Monitoring and control systems. Applications include video surveillance [17], smart lighting [16], public transport and pedestrian traffic control, pollutant emission monitoring, and analysis of online publications [18].
- Forecasting and risk prediction systems. Applications include prediction of natural disasters [19], assessing structural damage to infrastructure [20], and forecasting socio-economic parameters of urban development [16].
- Urban management automation. Applications include automating transport management [21], road infrastructure, lighting, energy consumption [15], municipal vehicle fleet management, and building operations.
- Human–city interaction systems. Applications include navigation systems [22], recommendation engines, and information retrieval systems [14,16]. Virtual and augmented reality applications [23,24] are common in this category, with a recent trend towards language-based personal assistants [25].

Notably, the challenges discussed in Section 1.1 are based on the problems of searching and summarizing data. Consequently, the integration of LLMs into management systems as

a unified data access point shows considerable promise, particularly in strategic urban management [26]. This approach could significantly enhance the efficiency and effectiveness of data utilization in smart city governance.

Existing solutions have varying adaptability to the specifics of a particular city. Table 1 shows a comparison of the proposed solution (ours) and other existing solutions according to the degree of adaptability to the tasks of a smart city.

Table 1. Comparison of current technologies for smart city tasks (“+” indicates feature is present, “-” indicates feature is absent)

Criteria	[7]	[8]	[9]	[10]	[11]	Ours
Operation of spatial data of territories	+	+	+	+	+	+
Operation of spatial data of city objects	+	+	+	+	+	+
Calculating spatial indexes	+	+	+	+	+	+
Implementation of specialized models of the urban environment (accessibility, connectivity, centrality, provision of facilities)	-	-	+	+	-	+
The ability to integrate custom models	+	-	-	-	-	+
Operation of documental data of territories	-	-	-	-	-	+
Support for automatic data updating	-	-	-	-	-	+
The ability to integrate into smart city systems	-	+	-	-	-	+
The possibility of developing a user interface for the tasks of a smart city	-	-	-	-	-	+
The ability to integrate natural language data management	-	-	-	-	-	+

The natural limitations of existing technological solutions for customization to specific tasks have prompted researchers to develop specialized models of the urban environment. Such models are specially prepared for solving the researcher’s tasks. This implies the possibility of their integration into existing technological solutions.

A qualitative review of such foundation models of the urban environment is presented in the paper [27]. Examples of such models are models of the urban environment for spatial and temporal modeling [28] and foundation general purpose models [29].

2.2. Large Language Models and Agents for Decision Support

LLMs are widely used to improve the quality of decision making in various fields [30]. They have applications in healthcare [31], architecture [32], logistics [33] and other fields. There are solid claims that LLM-based systems are more efficient than the “classical” decision support tools [34].

However, many limitations restrict the practical applicability of LLMs [35]. A significant problem is hallucinations [36], which look realistic but often lead to incorrect decisions. Also, in most cases, LLMs should interact with existing knowledge bases and tools to provide correct answers. This can be achieved by document-based Retrieval Augmentation Generation (RAG) [37]. The concept of a RAG system is the dynamic inclusion of supplementary context when responding to a user query tailored to the specific content of the user’s message or question.

Also, using external tools or plugins can increase the quality of answers. Tool calling is a feature that enables a user to give an LLM a prompt along with a set of functions (tools), including their parameters and descriptions of their actions. The LLM can request the user

to execute one of these functions and use the resulting output in its response. However, correct interaction with many tools is still a challenging problem for many LLMs.

LLM agents are a promising way to solve standalone LLM problems (e.g., hallucinations, lack of special knowledge, obsolescence of knowledge, etc.). An LLM agent is an AI system that uses an LLM to reason through a problem, create a plan to solve the problem, and execute the plan with the help of a set of tools. Up-to-date LLM agents have already shown inspiring results for complex task solutions in various fields (e.g., in software engineering [38], operations research [39], and the automation of machine learning [40]). Agents are also quite effective in interacting with tools [41] and personalizing for specific user needs [42]. They have great potential for database integration (a complicated task even for state-of-the-art LLMs [43]).

Multi-agent systems enhance single-agent reasoning capabilities by decomposing complex tasks into subtasks that specialized agents solve through interactions, mirroring real-world expert collaboration [44,45].

Therefore, using LLM agents to improve the efficiency of smart city management tasks is promising, as they also depend heavily on correct interaction with documents, databases, and geographic information systems (GISs). At the same time, city management usually does not have existing knowledge graphs, so the focus should be on processing unstructured data.

2.3. Large Language Models for Smart City Tasks

Urban management involves solving diverse challenges that are typically unstructured in nature and rely on non-standardized data. Addressing these challenges requires significant input from domain experts. In recent years, Large Language Models (LLMs) have emerged as promising tools for urban management applications, as demonstrated by frameworks like CityGPT [46] and City-LEO [47], which showcase LLMs' potential in handling various aspects of city operations.

CityGPT enhances LLM capabilities in urban contexts by embedding a “city model of the world” within the LLM [46]. The framework integrates urban-specific instructions to improve spatial reasoning and urban problem solving abilities. By combining these specialized instructions with OpenStreetMap (OSM) data [48], the framework develops models that address urban challenges while maintaining general-purpose functionality. The framework also establishes crucial benchmarks for evaluating LLM performance in urban tasks, particularly in geospatial understanding and policy assessment. However, CityGPT primarily focuses on knowledge modeling and lacks direct operational and strategic planning capabilities. Its outputs remain largely conceptual, emphasizing knowledge integration over actionable decision making support.

While OSM provides extensive geographical data including road networks, buildings, land use zones, and urban infrastructure [48], it has notable limitations. The data can be incomplete or inaccurate, particularly regarding Russian urban development and private sector information, which constrains its practical utility.

Aino.world [49] represents another implementation of OSM and LLM technology for urban applications. The project integrates multiple data sources beyond OSM, including Overture Maps Foundation [50], Microsoft Building Footprints [51], Kontur Population [52], Global Terrain: Zenodo EDTM [53], Google API geocoding [54], and HERE Technologies [55]. The system uses LLMs to create a chat interface for geodata access, allowing users to display data layers and object attributes on maps. However, Aino.world lacks advanced analytical capabilities and complex data analysis models, limiting its utility for strategic decision making. Additionally, the platform cannot accommodate region-specific requirements that depend on local regulations and administrative frameworks.

City-LEO implements an end-to-end (E2E) framework integrating optimization and reasoning for urban operational tasks using the Cycle Share Dataset [47,56]. The framework translates user queries into structured optimization problems through logical reasoning and combines predictive models (Random Forest) with optimization solvers for data-driven decisions. While City-LEO has proven effective in optimizing bike-sharing systems, demonstrating its strength in resource allocation, it focuses solely on operational-level tasks like fleet management rather than strategic planning. The framework excels at micro-level optimization but lacks capabilities for broader socio-economic and spatial strategy development.

Table 2 represents a comparison of existing urban LLM frameworks and our proposed multi-agent LLM system.

Table 2. Comparison technologies for smart city systems and solutions (“+” indicates feature is present, “-” indicates feature is absent).

Criteria	Ours	CityGPT	City-LEO	Aino.World
Ability to operate using administrative units	+	-	-	+
Ability to link urban objects with administrative units	+	-	-	+
Ability to operate urban objects	+	-	+	+
Ability to refer to models	+	-	-	+
Ability visualize data on the map	+	-	+	+
Integration of regulatory documents	+	+	-	-
Ability to make complicated suggestions based on analyzed data	+	+	-	-

Compared to these existing approaches, our proposed multi-agent LLM system offers several key advantages. Unlike CityGPT, which focuses primarily on knowledge modeling, our system provides direct operational and strategic planning capabilities through its modular agent architecture. In contrast to City-LEO’s focus on specific operational tasks, our approach handles complex strategic planning queries through integrated access to both document databases and real-time urban services. While Aino.world offers basic geospatial querying, our system provides comprehensive analytical capabilities through specialized agents that can process heterogeneous data sources and generate contextually relevant responses for urban planning decisions. Additionally, our approach’s ability to combine RAG technology with specialized agent roles enables more accurate and verifiable responses compared to standalone LLM implementations.

3. Proposed Multi-Agent LLM-Based Approach

3.1. LLM Agent Design

LLM agents operate through three primary mechanisms: instruction processing, action execution, and reflection, which enable adaptation to various city management tasks. These mechanisms form its internal logic cycles (shown with green arrows). As an intelligent core, the agent can use one of the “frozen” LLMs, e.g., the agent can access cloud-based LLM services (like GPT-4o) for advanced natural language processing while maintaining local server versions of LLMs (e.g., Llama and Mistral) over sensitive city data.

The LLM agent architecture features two fundamentally different types of interactions: agent-to-agent communication and tool usage, as illustrated in Figure 2 by black and

blue arrows, respectively. These interaction types follow distinct protocols that reflect their different nature and purposes. Tool usage represents an asymmetric relationship where tools serve as passive components that can only respond to requests. The agent initiates all interactions with city databases, services, and models through specific function calls or API requests. As a particular case of tool can be a secondary LLM agent, which incorporates internal logic and intelligence of the LLM agent but serves as a tool due to the asymmetric relationship with a primary agent. In contrast, agent-to-agent communication (shown by black arrows) establishes a peer-to-peer relationship where any agent can initiate interaction.

Due to the ability of tool usage, LLM agents can directly interact with urban environment databases' inventory and modeling databases, ensuring constant analysis relevance and automatic data updates. This integration enables real-time access to urban data and analytical capabilities without requiring users to understand complex technical details.

Building upon the foundational capabilities of individual LLM agents, we designed a comprehensive multi-agent system architecture that leverages these components while enabling coordinated operation at scale. This hierarchical system combines the reasoning and tool integration capabilities of individual agents with specialized roles and structured information flow pathways.

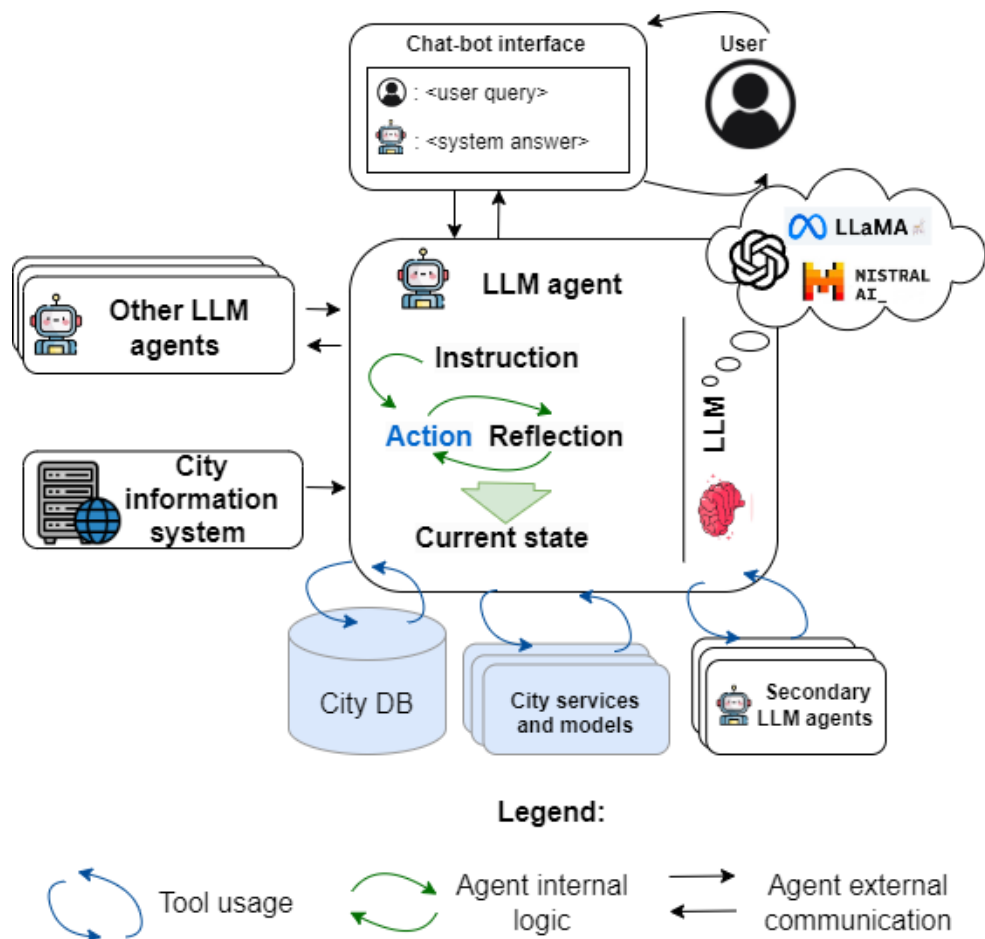


Figure 2. LLM agent concept for smart city management.

3.2. Multi-Agent System Design

The proposed multi-agent system architecture (Figure 3) represents a hierarchical framework designed for integrating diverse smart city data sources and processing complex urban queries.

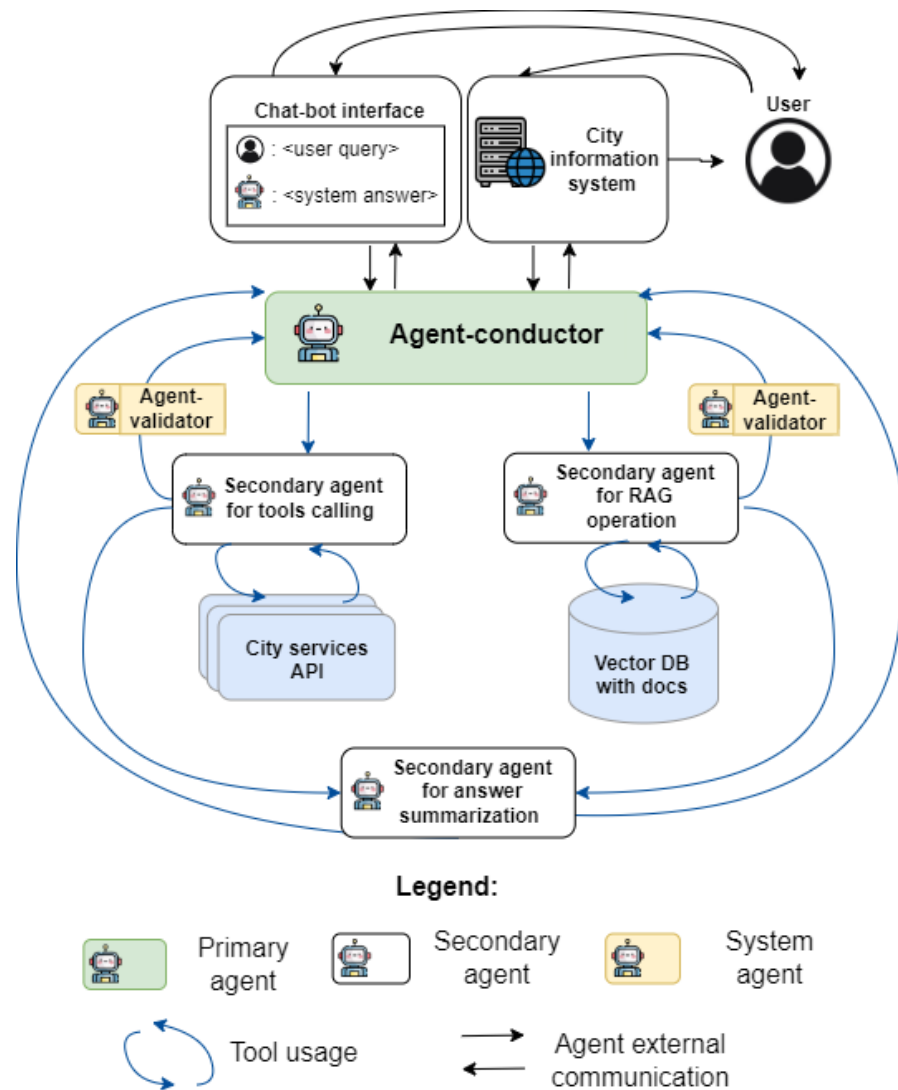


Figure 3. Architecture of multi-agent system for smart city management.

The system's architecture consists of three primary layers:

1. **Interface Layer.** The uppermost layer comprises a chatbot interface that serves as the primary point of interaction between users and the system. This interface is directly integrated with the city information system, enabling seamless access to urban data resources while maintaining a conversational interaction paradigm.
2. **Orchestration Layer.** At the core of the architecture lies the orchestration layer, centered around an "agent-conductor"—the primary agent responsible for coordinating system operations.
3. **Processing Layer.** The processing layer consists of three specialized secondary and system agents:
 - **The Tool Calling Agent** manages interactions with city services through API integration.
 - **The RAG Operations Agent** facilitates Retrieval-Augmented Generation using a vector database containing municipal documentation
 - **The Answer Summarization Agent** synthesizes and formats final responses based on accumulated data.
 - **The Validation Agent** controls the integrity of an answer.

The system's design enables sophisticated query processing through parallel operations while maintaining data consistency through validation checkpoints. This makes it

particularly effective for handling complex urban management scenarios that require the integration of multiple data sources and processing paradigms.

4. Implementation

4.1. LLM-Based Multi-Agent Architecture Implementation

We have developed an LLM-based assistant system that can answer questions based on two different data sources:

1. Database with documents about city development;
2. API that provides information about city services.

The design of the proposed multi-agent LLM system is presented in Figure 3. The system relies on several LLM agents to route questions, extract context, and produce final answers. The description of the LLM agent service is provided below.

For each new question, the system must first identify the data source required to answer it. The agent-conductor performs this; it uses tool calling to connect LLMs with external tools. In this case, we had two tools; each runs the pipeline for the corresponding data source. We created two concise prompts: the system prompt, which also includes detailed tool descriptions (see Appendix A.2.1), and the user prompt, which includes the context and the user question (see Appendix A.2.2). The tools contain detailed descriptions of the data sources.

The first pipeline processes questions related to the city development documents. First, a RAG operations agent selects the documents that best match the question. The relevant context is then extracted from the chosen documents stored in a vector database. Finally, the question and the extracted context are passed to a secondary agent for answer summarization, which generates the final answer.

We chose to use ChromaDB as it is open-source and was primarily built to work with LLM systems. To prepare data for the DB, we split the documents into paragraphs and calculate embeddings using the Multilingual-E5-large model (<https://huggingface.co/intfloat/multilingual-e5-large>, accessed on 1 September 2024). The embeddings were then added to a collection in the DB. For each query, the database returned four chunks of context most relevant to the original question.

The second pipeline works with questions about transport accessibility and service provision. The first step is to extract the context for the question from the city services API. Through the API, tables with groups of different parameters can be obtained. A secondary agent for tool calling performs this task.

Multiple tools were defined with detailed descriptions for all tables. In this approach, the secondary agent for tool calling must select the appropriate functions based on their descriptions. The same tool calling prompts are used as with the agent-conductor. Once the functions are selected, another agent validates them and then calls them to extract the context from the city services' API.

Then, the extracted context is combined with the question and is passed to the agent for answer summarization, which uses the appropriate prompt to produce the final answer.

Two prompts were created for the agent for answer summarization, one for every data source (city services API and vector DB). In the prompts, we provide a set of rules that the LLM must follow. For example, LLMs must only use the context to answer the questions. The full prompts are available in Appendixes A.2.3 and A.2.4.

This approach is easily scalable. It is possible to add new data sources by developing new tools for them. More documents can be added to the vector database. Additional tables and parameters can be added to the API. However, introducing new tools for use in tool calling is not straightforward. As the number of tools grows, it becomes increasingly

more difficult for the LLM to make the correct choice. The trade-offs are discussed later in the paper.

The developed system is distributed and consists of several services: the city information system with a chatbot interface, the LLM agent service, the city services, the vector DB, and the LLM service. All services communicate with each other via the REST API.

4.2. Integration with City Information Systems and Services

The Digital Urban Platform of St. Petersburg was designed to streamline the collection, organization, and analysis of urban data for city planners and researchers. As illustrated in Figure 4, the system's core is a digital city model. The digital city model is designed as a network where nodes represent city blocks, edges represent travel times (for walking or public transport), and each node contains data on its built environment and urban service capacity.

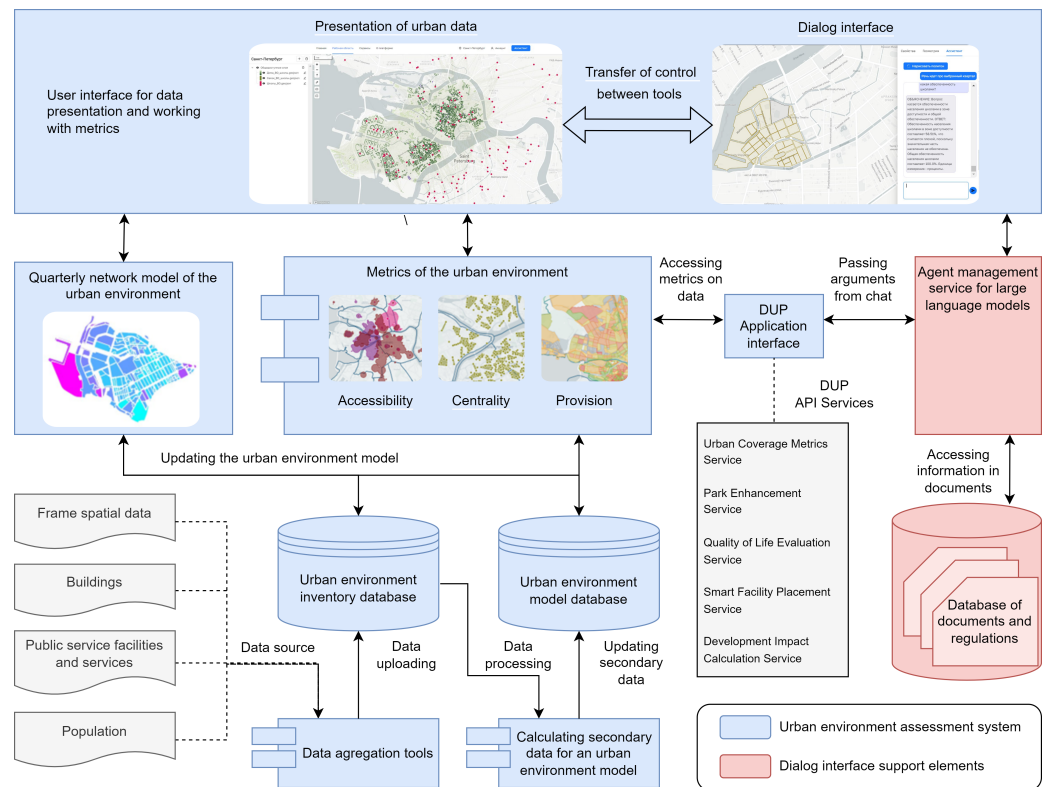


Figure 4. The Digital Urban Platform of St. Petersburg scheme.

The digital city model provides several important metrics for urban environment evaluation.

- **Accessibility.** Evaluates ease of movement by transport, walking, or multimodal opportunities [57].
- **Service provision.** Measures the sufficiency of urban service facilities [58,59].
- **Connectivity.** Calculates average travel times across the city with public or private transport [60].
- **Service proximity.** Assesses the availability of urban services (shops, restaurants, etc.) within specified distances [59].
- **Development potential.** Identifies areas suitable for developing new facilities [61].
- **Centrality.** Determines the position and relative importance of different urban areas [62].

To enable the interaction between LLM agents and the Digital Urban Platform (DUP) of St. Petersburg, we developed the DUP API that processes LLM agents' requests and returns calculated metrics and geospatial layers. The API provides five key services:

- **Urban Coverage Metrics Service.** Evaluates accessibility and coverage of public services such as schools and hospitals within defined areas.
- **Park Enhancement Service.** Identifies priority areas for park development and renovation based on community impact.
- **Smart Facility Placement Service.** Recommends optimal locations for new public facilities considering population needs and accessibility.
- **Development Impact Calculation Service.** Assesses how new construction projects will affect local service provision and infrastructure.
- **Quality of Life Evaluation Service.** Measures community well-being across social security, public health, and personal development metrics.

By combining structured urban data with natural language processing capabilities, the system can provide intuitive, context-aware responses to complex urban planning queries. This integration shows a practical application of AI in enhancing urban management and decision making processes.

5. Data

5.1. Data Sources and Preprocessing

Three types of data sources are used in the Digital Urban Platform of St. Petersburg. The first is the OpenStreetMap service [48]. This service serves as a data source for building a set of wireframe data that includes the street and road network, administrative–territorial division of the city, water bodies, green areas, buildings and structures for various purposes, and network of city blocks.

To enhance the urban environment assessment, additional data on buildings and urban services are incorporated. This supplementary information for St. Petersburg is sourced from the Territorial Development Fund's open data [63], which provides details on building parameters such as the number of floors and capacity. Furthermore, population data from the national statistics service [64] is utilized to estimate the number of residents in each building [65].

Figure A1 illustrates the data preparation process schematically. This comprehensive approach enables the calculation of key urban indicators (according to Table A1), which are then fed into the language model service for analysis and interpretation.

5.2. Validation Q&A Dataset

A validation dataset was created to evaluate the ability of the language model to generate accurate responses across different domains of urban life and areas, integrating contexts from the city information system with information from documents and data-driven simulations. The experimental dataset consists of 150 question–answer pairs. The size of the validation dataset is relatively small due to (1) the high cost of preparation for each question, since it is manually prepared and answered by experts in the field (so the questions are quite realistic and diverse) and (2) the possibility of estimating the quality of the LLM-based solution using nearly a hundred examples [66,67]. The dataset is provided in the open repository (https://github.com/ITMO-NSS-team/llm-agents-for-smartcities-paper/blob/main/pipelines/tests/test_data/dataset_eng_full.csv, accessed on 1 November 2024).

To prepare the test dataset of questions and answers, a group of human experts was formed. The experts were selected in accordance with the required competencies to participate in the main stages of the pre-project analysis of territories in the strategic

management of smart cities (Table A6). The experts included specialists in urban data analysis, GIS specialists, managers in public administration, managers for managing the development of the city's information infrastructure, urban architects, and specialists in information and analytical support for public authorities, including the authors of smart city concepts.

To assess the model's ability to handle geographically constrained queries, specific categories include questions with territorial context. Examples of questions are given in Table A2.

The validation dataset was augmented using an iterative process utilizing both human expertise and artificial intelligence capabilities; the workflow is presented in Figure 5. Initially, urban studies experts made a dataset of domain-specific questions and answers (Q&As) for purposes of urban planning and smart city management. This expert-generated dataset served as the foundation for further augmentation.

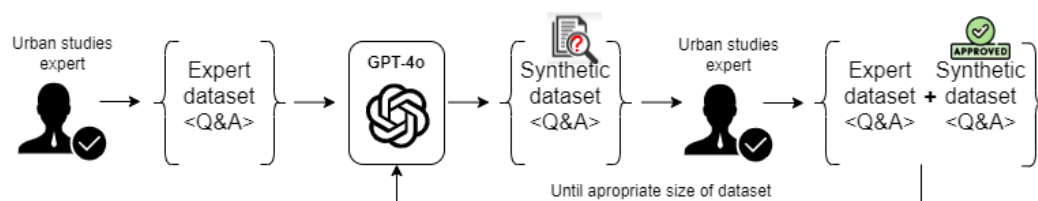


Figure 5. Test dataset synthetic augmentation scheme.

For AI augmentation of the initial dataset, we used the GPT-4o LLM to generate additional synthetic Q&A pairs based on the patterns and content of the expert dataset and possible sources of new Q&A including documents and database tables. To ensure the quality and accuracy of the synthetic data, the urban studies expert reviewed each AI-generated Q&A pair. The approved synthetic Q&As were subsequently merged with the original expert dataset, creating an expanded corpus. This augmentation process can be repeated iteratively, with each cycle further enriching and diversifying the dataset until a satisfactory volume is achieved. This human-in-the-loop approach allowed for the approval, modification, or rejection of synthetic content, maintaining the dataset's integrity and relevance.

6. Experimental Studies

6.1. Experimental Setup

The experimental setup was designed to evaluate the performance and capabilities of LLM-based multi-agent systems in the context of smart city management on the dataset described in the previous section. The experiments were carried out using a variety of “frozen” (without additional fine-tuning) LLM models, including GPT-4o (<https://openai.com/index/gpt-4/>, accessed on 1 November 2024), Mistral-8x22b (<https://mistral.ai/news/mixtral-of-experts/>, accessed on 1 November 2024), unquantized Llama v3.1-70b (<https://ai.meta.com/blog/meta-llama-3-1>, accessed on 1 November 2024), and quantized Llama v3.1-70b-int4. These models have been selected because they are state-of-the-art tools in their class [68]. The selection includes both closed-source (GPT-4o) and open-source models of different sizes (number of parameters and quantization).

The main idea of an experimental study is to compare the performance of different LLM models and system configurations (LLM alone, LLM with ChromaDB, LLM with API, and LLM with both ChromaDB and API) to answer the main question “How useful are LLM multi-agent systems for improving smart city decision making processes?”.

1. Experimental setup for Hypothesis 1: “LLM agents can effectively route and process diverse user queries related to the urban environment and social infrastructure, ac-

cessing relevant services and databases”. To assess query routing and processing efficiency, we evaluated the percentage of correctly routed queries (accuracy) to appropriate services (since this criterion is used in a widely known tool-use benchmark [68]). This evaluation was conducted across different LLM models and agent configurations, including scenarios with and without LLM-based correction of choices, and it also compared free API selection versus manual filtering for choice correction.

2. Experimental setup for Hypothesis 2: “The integration of LLM with existing urban information systems and services (e.g., social benefits availability service, transport accessibility service, documentary DB) enables the generation of more accurate and contextually relevant responses to user queries”. The effectiveness of integration with urban information systems was examined by comparing the performance of various system configurations. These included LLM without RAG (pure LLM responses), LLM with a vector DB, LLM utilizing API access to city services, and LLM employing both a vector DB and city service APIs. For each configuration, we measured the accuracy of the response using G-Eval correctness, the relevance of the answer, and fact-check metrics. Additionally, query processing times were recorded for each setup to assess efficiency.
3. Experimental setup for Hypothesis 3: “The integration of LLM-based methods in a city management system may significantly increase efficiency and decrease the decision making process time”. To contextualize the potential benefits of the LLM-based approach, we compared the speed of the LLM system’s responses with traditional query processing methods. This comparison was based on estimates provided by subject matter experts, specifically urban planners. We obtained time estimates for human experts to perform tasks similar to those handled by the LLM system, allowing for a direct efficiency comparison.

6.2. Evaluation Metrics

Evaluation metrics included pipeline/tool choice accuracy, work time, and correctness of the answer (based on G-Eval, Answer Relevancy, and fact-checking):

- Pipeline/tool choice accuracy was estimated as the percent of correctly chosen pipelines or tools with a function call query (without estimating final LLM answer correctness).
- Work time was estimated as a mean value (on experimental dataset queries) for each system configuration and LLM choice. For meaningful results, all time measurements were made in equal infrastructure conditions (except the guarantee of constant conditions of proprietary LLM services like GPT-4o).
- The answer correctness was estimated using three metrics: G-Eval, Answer Relevancy, and fact-checking. G-Eval and Answer Relevancy (AR) were used with the DeepEval framework (<https://github.com/confident-ai/deepeval>, accessed on 1 November 2024).

The G-Eval metric outperforms other state-of-the-art evaluators and provides higher compliance with human requirements [69]. It uses LLMs with chain-of-thoughts (CoT) to evaluate answers from other LLMs based on custom user criteria. These criteria can be provided as evaluation steps—a list of rules specifying precise steps the LLM should take for evaluation. We developed several rules that evaluate correctness and relevance, paying attention to numerical accuracy and correct interpretation of facts from the context. The full text of the evaluation steps used for the experiments is in Appendix A.2.6.

The AR metric evaluates how relevant the answer from the LLM is to the correct answer from an expert. AR first uses an LLM to extract all statements from the

given answer and then uses the same LLM to define whether each statement is relevant to the correct answer. The metric equals the ratio of relevant statements to all statements in the answer. Evaluations on the WikiEval (<https://huggingface.co/datasets/explodinggradients/WikiEval>, accessed on 1 November 2024) dataset show that the predictions for AR are closely aligned with human predictions [70].

GPT-4o Mini was used as the LLM estimator for both G-Eval and AR. G-Eval demonstrates more strict estimations than AR. While with G-Eval, the LLM estimator tries to catch subtle differences between the expected and actual answers, with AR, the LLM estimator tries to reason whether the actual answer superficially looks like the expected answer.

Fact-checking was performed for API-based context by calculating the percentage of correct numerical facts within LLM answers. For this purpose, all numerical values from the correct and LLM answers were parsed and compared. If the numerical values were identical, the answer was considered correct.

7. Experimental Results

In this section, we conducted experiments with different LLM and agent setups to establish how well an LLM-based system can generate responses to user queries compared to LLMs without additional context.

7.1. LLM Effectiveness

We first tested “naked” LLMs with the same simple system prompt (see Appendix A.2.5) and without providing any context. The metrics are provided in Table 3.

Table 3. Quality of answers for “naked” LLMs; metrics from DeepEval: G-Eval and Answer Relevance.

Model	LLM	
	G-Eval	AR
gpt-4o-2024-08-06	0.30	0.93
mixtral-8x22b-instruct	0.38	0.90
llama-3.1-70b-instruct	0.38	0.95
llama-3.1-70b-instruct-int4	0.38	0.98

The G-Eval metric is low for all models, ranging from 0.3 to 0.38. LLMs need help to answer most questions in this domain. However, the metrics are far from zero. Even without context, LLMs can still answer some types of questions correctly.

The AR metric values, on the other hand, are high, ranging from 0.9 to 0.98. This shows that LLMs are particularly good at generating text for a specific topic. However, having no access to relevant data, they tend to invent false facts, which explains the low G-Eval metric values.

7.2. RAG Effectiveness

In this subsection, we discuss the results of using RAG to answer questions. First, we want to compare LLMs without context, LLMs with context only from documents from ChromaDB, and LLMs with context only from the city services API on all our test questions: 75 questions about the development strategy and 75 questions about the provision of city services. The averaged G-Eval and AR metrics on the complete set of questions are provided in Table 4.

Table 4. Quality of answers for LLMs with context; metrics from DeepEval: G-Eval and Answer Relevancy.

Model	LLM		LLM + ChromaDB		LLM + API	
	G-Eval	AR	G-Eval	AR	G-Eval	AR
gpt-4o-2024-08-06	0.30	0.93	0.38	0.65	0.52	0.66
mixtral-8x22b-instruct	0.38	0.90	0.43	0.70	0.51	0.73
llama-3.1-70b-instruct	0.38	0.95	0.41	0.74	0.54	0.71
llama-3.1-70b-instruct-int4	0.38	0.98	0.39	0.71	0.50	0.69

There is a slight increase in the G-Eval metric for questions about the development strategy; on average, the metrics are 13% higher. At the same time, there is a higher increase in the G-Eval metric for questions about the provision of city services; on average, the metrics are 45% higher. We tested the system on all 150 questions in each case but provided context only from one data source. These results demonstrate that adding part of the relevant data sources for context retrieval increases the answer quality of LLMs. However, some domains will benefit more than others. In order to get the best results, all relevant context must be provided; this will be discussed later.

The AR metric shows that LLMs with additional context underperform compared to those without, suggesting that the added context may be unhelpful. However, this decrease in relevance is accompanied by a rise in correct responses. Unlike LLMs generating answers without context, the LLM-based system relies on factual information from available data sources. It explicitly states when relevant information is lacking and does not include false facts in the answer.

To get a more accurate estimation of the effects of adding context from a vector DB, we calculated the average G-Eval metric for LLMs with context only from ChromaDB on questions only about the development strategy; the results are in Table A4. In this case, we see a notable improvement, with accuracy scores averaging 36% higher. The results are not as high as could be expected; this occurs because the topics of the questions are too general. The model has enough general knowledge to provide answers.

Similarly, we calculated the average G-Eval metric and the percentage of correct numerical answers for LLMs without context and LLMs with context only from the API on questions only about the accessibility of city services. The results are provided in Table A4. In this case, the results with context are substantially better. We see an increase in the G-Eval metric, going from 0.17–0.30 to 0.76–0.83. The number of correct numerical answers goes on average from 0.1% to 88%. This shows that LLMs must have relevant context to answer questions requiring numerical precision.

Based on the experiments conducted, it is clear that providing context to LLMs is highly significant for generating relevant responses, especially in specific domain areas (for example, where numerical data are needed).

7.3. Tools Selection Effectiveness

As discussed previously, RAG is a highly effective approach. However, in order to use its benefits, it is crucial to have the capability to extract the correct context for the question. Two important tasks must be solved: (1) selecting the proper data source (vector database or urbanistics platform) and (2) selecting the correct services on the urbanistics platform (correct tables with parameters).

The proposed system includes agents that route requests using tool calling. One agent is responsible for selecting the correct pipeline (data source: DB or API), and the second agent selects proper tools to extract context from the city services API.

We conducted several experiments to test the quality of the agents' work; the results are presented in Table 5.

Table 5. Tool selection quality and time for (a) all questions and (b) a subset of service questions.

Model	Pipeline Selection (All Questions)		Function Selection (Service Accessibility Questions)			
	No Verification		No Verification		LLM Verification	
	ACC, %	Time, s	ACC, %	Time, s	ACC, %	Time, s
gpt-4o-2024-08-06	97.3	3.1	48.0	2.3	80.0	5.4
mixtral-8x22b-instruct	94.0	2.0	90.7	2.3	94.7	5.0
llama-3.1-70b-instruct	99.3	2.1	96.0	2.5	96.0	5.7
llama-3.1-70b-instruct-int4	98.7	1.1	93.3	1.1	93.3	3.1

In the column "Pipeline Selection", we see the results of selecting the appropriate pipelines for the questions based on the required data source. All LLMs provide highly accurate query routing ranging from 94% to 99%. Errors in the choice of pipelines mainly appear when questions can be interpreted differently due to their generic nature.

In the column "Function Selection", we added the results of selecting the proper function for obtaining the context from the city services API. There is a decrease in accuracy due to the increase in the number of tools available for selection. However, three of the four models show high results, ranging from 94% to 96% of correctly selected functions. Most of the errors occur due to the similarity of some of the data stored in the tables and the proximity of the descriptions of these tools.

Results for GPT-4o are noticeably lower than for the other models, so we decided to add another step where the LLM verifies the chosen tools. We pass the original question, tool descriptions, and the chosen functions and verification prompt to the LLM for verification. The LLM then returns a list of functions that best fit the question. This technique improves accuracy for Mixtral by 4% and for GPT-4o by 66%, but it does not change the results of the Llama models. While results for Mixtral are quite high at 94%, results for GPT-4o are lower at 80%. This lower result is likely due to the prompt, which is not optimal for the model.

An important factor that affects accuracy is the description of the tools for tool calling. Even minor edits could have a significant impact on the result. We conducted experiments with different prompts obtained from prompt engineering guides [71,72] and concluded that the best approach is to use concise wording and add keywords to the description.

The prompts used for tool calling are basic and concise; they are presented in the Appendixes A.2.1 and A.2.2. As can be seen from the experimental results, it is sufficient to achieve high accuracy rates for most of the tested models.

Thus, agents with optimally selected models, suitable prompts, and sufficiently detailed but not too extensive tool descriptions ensure successful context extraction for effective RAG.

7.4. Whole Multi-Agent System Effectiveness

Let us compare LLMs without context with the LLM-based system that automatically extracts relevant context from all available data sources. We tested performance on all 150 test questions. The experimental results are presented in Table 6.

Table 6. Quality of answers; metrics from DeepEval: G-Eval and Answer Relevancy.

Model	LLM		LLM + ChromaDB		LLM + API		LLM + ChromaDB + API	
	G-Eval	AR	G-Eval	AR	G-Eval	AR	G-Eval	AR
gpt-4o-2024-08-06	0.30	0.93	0.38	0.65	0.52	0.66	0.71	0.94
mixtral-8x22b-instruct	0.38	0.90	0.43	0.70	0.51	0.73	0.68	0.91
llama-3.1-70b-instruct	0.38	0.95	0.41	0.74	0.54	0.71	0.74	0.95
llama-3.1-70b-instruct-int4	0.38	0.98	0.39	0.71	0.50	0.69	0.71	0.92

As already discussed, the quality of responses from LLMs without contextual data is relatively low. The G-Eval score is 0.3–0.38, as access to precise external data is crucial. The LLM-based system, which has access to the appropriate context, shows significantly higher results with the G-Eval metric ranging from 0.68 to 0.74. Confidence intervals for the G-Eval metric are in Table A5. There is a significant difference between them, indicating that the results are prominent.

The metric value is not as high as expected because the G-Eval metric is quite rigorous in assessing the loss of details from the correct answer, even though the primary information in the model's answer is consistent with it. So, metric values above 0.8 can be considered an excellent result (examples of LLM answers are provided in the Appendixes A.4.1 and A.4.2).

The Answer Relevancy metric is considerably high in both cases: 0.9–0.98 for LLMs without context and 0.91–0.95 for the LLM-based system with context. The high value of this metric shows that LLMs without context are particularly good at producing answers that are closely related to the topic and look highly realistic. However, as discussed in the previous section, this does not guarantee the factual correctness of the answer, as models tend to hallucinate and use outdated knowledge bases. In contrast, the LLM-based system only provides answers based on the provided context.

Thus, we conclude that selecting a suitable data source and extracting the correct context while using LLMs are crucial in handling complex queries and generating relevant responses for urban planning and management purposes.

7.5. Performance of Multi-Agent System vs. Human Experts

Comparing the performance of an LLM multi-agent system with human experts is challenging, as LLMs typically augment existing decision support systems rather than replace them. Urban decisions are increasingly data-driven, with expert interpretation playing a crucial role. While data usage at the strategic level may not accelerate decision making, it can improve decision quality by considering a broader range of factors. To analyze this comparison, we consider three scenarios for decision support:

- **Option 1: Direct multi-agent system interaction.** Decision-makers directly query an LLM-based chat system.
- **Option 2: Expert-mediated.** A team of experts uses the information system described in Section 4.2, without the LLM dialogue interface.
- **Option 3: Traditional analysis.** Analysts collect and analyze data ad hoc, without pre-existing analytical databases.

Table A6 outlines the typical stages of city development planning to compare human resources and time consumption across these scenarios. The scenarios and approximate numerical estimations were obtained by expert consultations. The time estimates for the

multi-agent system in Table A6 are based on data from Table A3. It presents the average response time of the multi-agent system to test questions.

The comparison of the multi-agent system with traditional expert-driven approaches in urban planning reveals significant trade-offs. Multi-agent systems demonstrate remarkable speed and efficiency, often completing tasks in hours that typically take days or weeks for human teams. They require minimal additional staff, making them resource-efficient. The data suggest that multi-agent systems excel in rapid data access and initial analysis, making them ideal for quick decision support. However, for strategic, long-term planning, human expertise remains crucial. These findings indicate that an optimal approach might integrate multi-agent system capabilities with human expertise, leveraging the strengths of both to enhance urban planning processes.

Regarding user interaction, multi-agent systems provide interactive engagement between data analysis tools and the dialogue interface, offering contextual explanations and recommendations based on actual data.

The comparison between multi-agent LLM systems and traditional expert approaches reveals their complementary strengths. The findings suggest that optimal urban planning outcomes can be achieved through a hybrid approach: using multi-agent systems for rapid analysis and data-driven insights while preserving human judgment for strategic planning and oversight. This integration offers both efficiency gains and maintained decision quality through the combination of automated processing and expert guidance.

8. Implications for Smart City Management

8.1. Potential Applications and Use Cases

Two typical scenarios for the system's application can be considered: planning for the improvement of parks and planning for the placement of social service facilities using the example of the city of St. Petersburg.

8.1.1. A Scenario for the Formation of a Program for the Placement of New Educational Facilities

- As a first step, it is important to understand city administration's plans and basic information about school development in city. With the Assistant's help, it is possible to obtain an answer about school availability in Saint Petersburg:

Q: "What is the situation with school education sphere in Saint-Petersburg?"

A: "According to the strategic planning documents [73], the field of education in St. Petersburg occupies a special place due to the historical factors of the city's development. There are 737 educational organizations in the city, which train about 500 thousand people."

- The next step is to assess the best place for school placement according to the user's budget and school provision evaluation. The question to the Assistant is as follows:

Q: "My budget is 3 billion rubles. Where is the best place to locate new schools in St. Petersburg?"

A: "With your budget of 3 billion rubles, it is the best to locate 3 schools in St. Petersburg, costing 1070.0 million rubles, 646.8 million rubles, 280 million rubles. The total budget will amount to 2 billion rubles. Availability of schools in the allocated neighborhoods will increase from 0.49 to 0.53. The total number of schoolchildren provided with the service will amount to 3210."

Also, the user obtains a layer with proposed school placement (due to the budget Assistant offering three locations) and layers with categorized blocks with the best school provision improvement (Figure 6).

To obtain this answer, the system went through the following procedure, which includes the basic steps listed in Table A7. At the first step, an assessment is made of the provision of schools for the population in the selected territory. The blocks where the entire population is provided for are marked in green, while those in red are not provided for. Next, the potential of the blocks to accommodate the new school is assessed. The size of the potential is determined by the level of improvement in the provision of the population in the case of the placement of a new school. The decision on the number and size of schools is determined by the budget. In this example, the budget amounted to RUB 3.3 billion, which made it possible to plan the construction of three new schools.

Schools are planned in different sizes, since the amount of free space in the blocks varies. The quarters where the greatest improvement in the provision of schools to the population is expected are presented separately. This example illustrates a common approach to solving the problem of planning new schools with a limited budget.

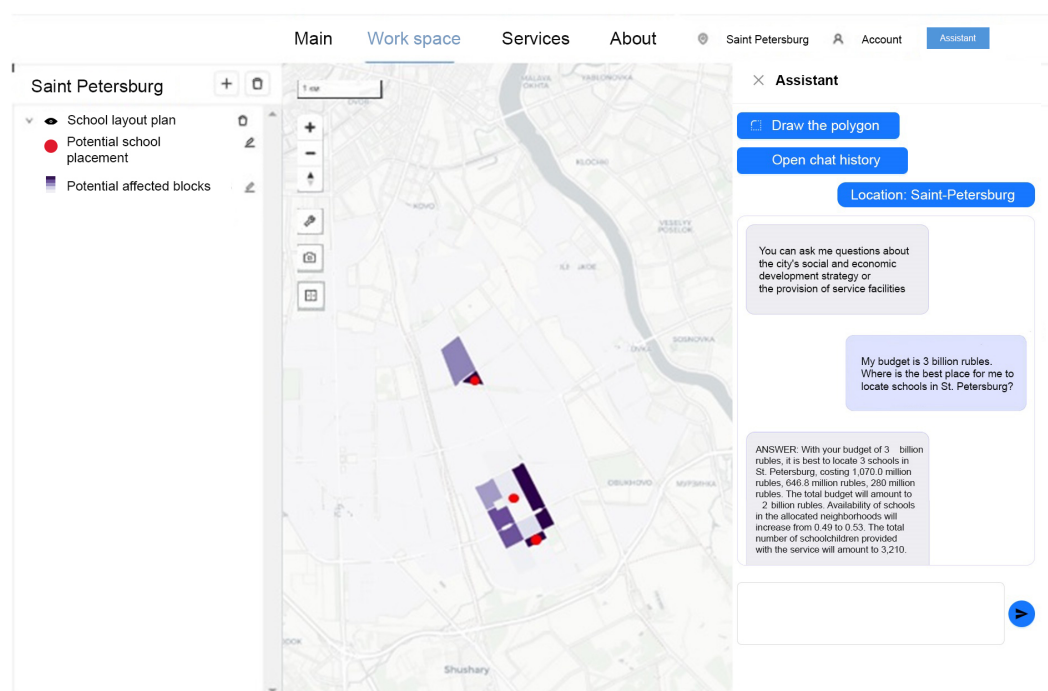


Figure 6. The presentation of the final result in the user interface.

8.1.2. A Scenario for the Formation of a Park Improvement Program

The second example illustrates an approach to park improvement planning. The algorithm is similar to that of the first example.

- First, the Assistant gives an answer about landscaping polices and plans in Saint Petersburg.
Q: "What is situation with landscaping sector in St. Petersburg?"
A: "According to the strategic planning documents [73] in St. Petersburg, landscaping is one of the priorities of the city's policy." This information identifies importance of green space development in a city.
- The next step is to identify territories suitable for landscaping. The question to the Assistant is as follows:
Q: "Where are green spaces located?"
A: "The layer of green zones is shown on the map."

The Assistant's answer includes layer of the city's green areas, as shown in Figure 7.

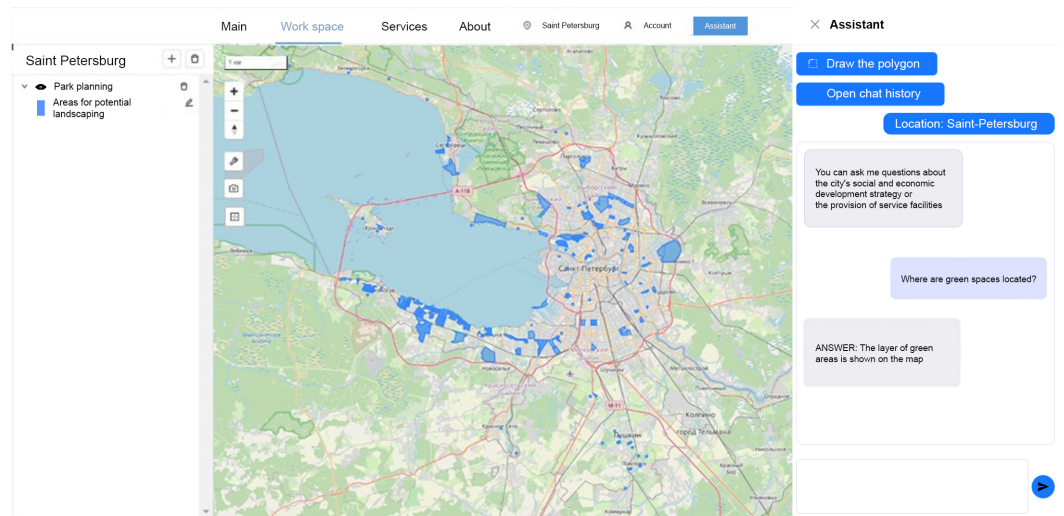


Figure 7. Territories suitable for landscaping.

- On the last step, the effects of park improvement are assessed in terms of maximizing population coverage and increasing the level of provision of the population with green areas. To know the best areas for landscaping, the following question to the Assistant should be asked:

Q: "Show me which areas and parks have the greatest investment potential for landscaping."

A: "Total audience is 5,597,763 people. Total cost of landscaping is 246 billion rubles.".

Also, the user obtains the layer with green zones categorized in terms of the best combination of expected population coverage and landscaping costs, as shown in Figure 8.

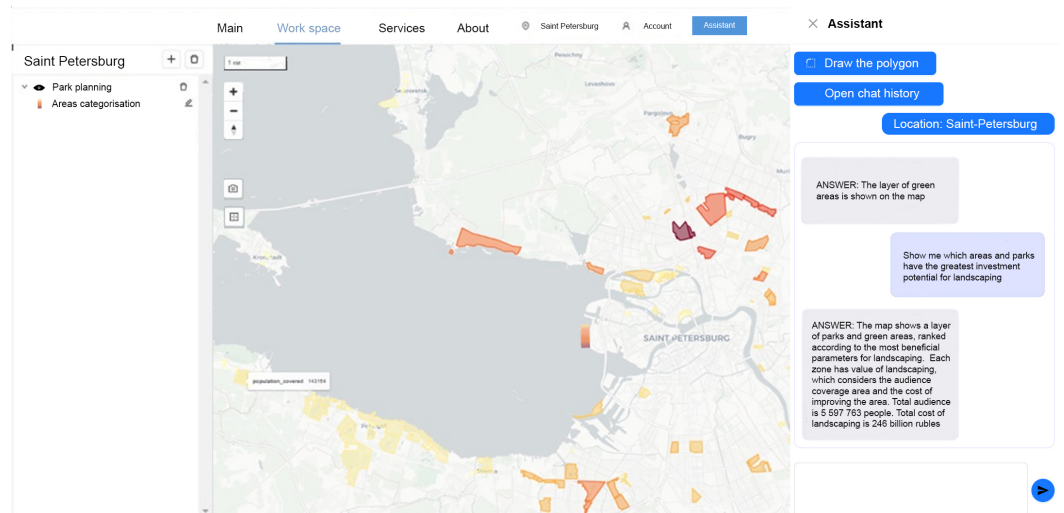


Figure 8. Assessment of areas suitable for landscaping.

This example illustrates a general approach to solving the task of planning work under an unknown budget constraint and involves finding the optimal solution that maximizes the effect while minimizing costs.

8.2. Challenges and Limitations

The production-ready implementation and deployment of LLM multi-agent systems for smart city management faces several significant challenges identified through our experimental studies and practical implementations.

1. **Computational scalability.** The deployment of LLMs demands substantial computational resources, facing city government with a challenging trade-off between utilizing external cloud infrastructure (with associated data security risks) and developing local computing facilities (with associated management difficulties and essential significant monetary investments). While our experiments demonstrated the viability of both cloud-based (GPT-4) and local models (LLaMA, Mistral), the choice between them involves important trade-offs that cities must carefully consider. Local LLM deployment offers enhanced data security and privacy compliance, which is crucial when handling sensitive urban planning data. However, it requires significant computational infrastructure—our tests indicate that running LLaMA-70B demands at least 140GB of GPU memory for optimal performance, representing a substantial investment for municipal IT departments.
2. **Real-time Data Integration.** The system's effectiveness critically depends on its ability to synchronize and process multiple data streams from urban databases, statistical services, and regulatory documents in real time. Production-ready implementation faces challenges in maintaining a consistent data flow and standardization across various city information systems while ensuring data security and access control.
3. **Agent System Stability.** The complex orchestration between the agent-conductor and specialized agents introduces reliability concerns, particularly in production environments. The system's dependence on carefully engineered prompts and tool descriptions, combined with the need for regular model updates and validation, creates operational vulnerabilities that could affect decision making reliability in critical urban management scenarios.

8.3. Ethical Considerations

The ethical issues surrounding the use of AI systems, especially large language models, in smart city management align with broader challenges in digital transformation [74,75]. Several key ethical issues require consideration.

First, smart cities are environments of competing needs [76], requiring complex decisions to resolve conflicts. Many urban challenges have no clear "best" solution, such as choosing architectural styles, determining land use, selecting public transport types, and similar complex decisions. In these situations, AI systems may either fail to find effective solutions or, worse, intensify existing problems rather than solve them. To address this challenge, cities should develop hybrid decision making frameworks that combine AI recommendations with participatory planning processes, ensuring that technological solutions are balanced with community input and local context.

Second, the growing use of AI systems in city management is leading to an increasingly technology-driven approach to both governance and urban development. This trend has faced criticism [77] and will likely encounter more resistance as digital transformation accelerates, potentially resulting in excessive dependence on technology in urban planning and development. To maintain balance, cities should establish clear guidelines for technology integration that emphasize AI as a supporting tool rather than a replace-

ment for human judgment, while regularly assessing the social and cultural impact of technological solutions.

9. Conclusions and Future Work

9.1. Summary of Key Findings

The experimental evaluation of the LLM multi-agent system for smart city management produced several findings that support our initial hypotheses.

Regarding Hypothesis 1, the results demonstrated that LLM agents are effective at routing and processing diverse urban queries, with pipeline selection accuracy ranging from 94% to 99% across different models. The tool calling accuracy for accessing specific urban services achieved up to 96%. Also, it is worth mentioning that additional LLM verification helps to improve performance, particularly for GPT-4o (from 48% to 80%).

Hypothesis 2 about RAG technology's effectiveness was supported by the experimental results. When using RAG with city documents, the G-Eval metric showed a 17% improvement for questions about development strategy compared to standalone LLMs. This confirms that RAG technology significantly enhances the system's ability to provide accurate and reliable responses when working with local city knowledge and regulations.

In support of Hypothesis 3, LLM multi-agent design of a system with the integration of urban information systems and services showed substantial improvements in response quality compared with standalone LLMs. The combination of both vector DB and service API access helps to reach the highest G-Eval scores (0.68–0.74) compared to standalone LLM responses (0.30–0.38).

A major achievement of this work is building a working system that successfully combines document analysis with real-time city data access, allowing it to give complete and accurate answers to complex urban questions. The system maintains high relevance in its answers (scores of 0.91–0.95) while being much more accurate with facts, marking a significant step forward in AI-supported urban decision making.

9.2. Contributions to the Field

With this paper, we illuminate new possibilities for urban informatics through the application of advanced LLM technologies. The results of our research demonstrate that the most significant impact in smart city management can reasonably be expected not from the direct use of LLMs alone, but from the implementation of LLM multi-agent systems.

Our key contributions include the following:

1. A validated multi-agent architecture for orchestrating diverse urban data sources, achieving comprehensive context-aware responses to complex queries.
2. Empirical performance comparison of leading LLMs (GPT-4, Mistral, Llama) in urban-specific tasks.
3. Real-world implementation cases in school placement and park improvement planning, demonstrating practical impact on urban decision making.

As a conclusion, we note that the LLM multi-agent approach represents a step forward in realizing the potential of AI for smart city management. By demonstrating how LLMs can effectively integrate and analyze diverse urban data, provide comparative insights on model performance, and offer practical solutions to real urban challenges, our research opens new clues for more data-driven, efficient, and responsive urban governance.

9.3. Future Research Directions

Based on the findings and limitations identified in this study, we propose several promising directions for future research spanning both technical improvements and urban applications.

System adaptability to incomplete and rapidly changing data is a key factor for its practical application in urban management. Our experiments showed that LLM agents can effectively operate even with significant data gaps, using contextual information to compensate for missing data. The system implements this through a two-level approach: at the first level, RAG is used to extract available context; at the second level, LLM agents apply logical inference to fill in the gaps.

For handling rapidly changing urban data, the system would use incremental updates of the vector database, which would allow new information to be incorporated promptly without requiring complete retraining. The system would explicitly indicate uncertainty in responses when available data are insufficient for a fully reliable answer.

For cities with different levels of digital maturity, a phased implementation is recommended: starting with a basic dataset (e.g., OpenStreetMap) followed by gradual expansion of data sources as urban information systems develop.

Future technical development should focus on three key areas: enhancing agent coordination for complex urban planning workflows, improving validation mechanisms to ensure factual accuracy in urban contexts, and optimizing computational efficiency for resource-constrained municipalities. Priority should be given to spatial reasoning tools and performance optimization while maintaining system reliability.

In terms of additional cases and applications for smart city management, future research should explore the development of specialized agents for urban emergency response and crisis management scenarios while integrating real-time urban sensing networks to enhance dynamic city management and citizen participation. Enhanced temporal modeling capabilities would improve the tracking of urban evolution and sustainable development planning.

10. Code and Data Availability

All code for the agent-based LLM system, scripts, and datasets for experiments are available in the open repository <https://github.com/ITMO-NSS-team/llm-agents-for-smartcities-paper>, accessed on 10 January 2024.

11. Acknowledgements

The authors would like to express their sincere gratitude to the following experts who contributed significantly to this work, particularly in developing and validating query methodologies: Danil V. Voronin, urban architect and planning specialist, member of the Architects Association and co-founder of an architectural bureau, for his expert guidance, validation query development, and valuable insights in urban development, and Konstantin G. Samolovov, urban architect and head of the urban planning department at Lengiprogor Institute, for sharing his extensive experience, professional expertise, and assistance in query validation. We are particularly grateful to our team of analysts who played a crucial role in both data analysis and query validation: Tatiana I. Baltyzhakova for her exceptional GIS analysis and Varvara I. Kuraksina, Ekaterina A. Pyankova, Polina S. Ryaboshlyk, and Egor A. Chichkov for their meticulous data analysis, interpretation, and validation methodology development. Special thanks go to Dmitry K. Pokidov for his expertise in territorial development management and contribution to validation frameworks and Igor V. Kuprienko for his valuable input in urban information space development and query validation processes.

We would also like to thank the anonymous reviewers and editors for their thorough feedback and constructive suggestions that helped improve the quality of this work significantly. Their collective expertise, dedication, and thorough approach to validation have been instrumental in shaping and enriching this research.

Author Contributions: Conceptualization, A.K., S.M., K.F. (Kirill Fedorin), N.O.N. and A.B.; Methodology, A.K., N.O.N. and A.B.; Software, E.L., A.G., Y.A., K.F. (Kamil Fatkhiev) and V.V.; Validation, N.C. and V.V.; Formal analysis, V.V.; Investigation, A.K., S.M., E.L., A.G., K.F. (Kamil Fatkhiev) and N.C.; Resources, N.O.N.; Data curation, S.M., N.C. and V.V. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Analytical Center for the Government of the Russian Federation (IGK 000000D730324P540002), agreement No. 70-2021-00141.

Data Availability Statement: All code for the agent-based LLM system, scripts, and datasets for experiments are available in the open repository <https://github.com/ITMO-NSS-team/llm-agents-for-smartcities-paper>, accessed on 10 January 2024.

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

Appendix A

Appendix A.1. Additional Data Description

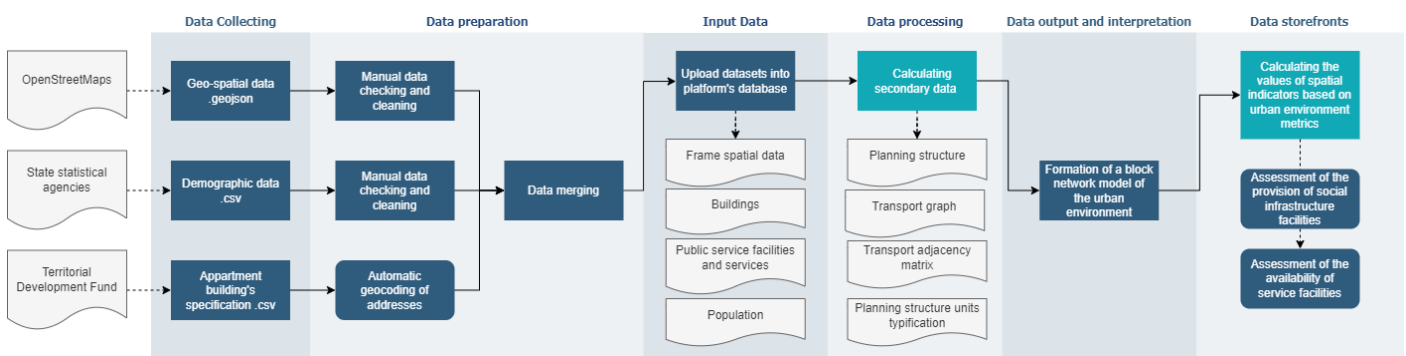


Figure A1. A conceptual scheme of the data preparation process.

Table A1. Specification of data transmitted to the language model service.

Dataset from System	Indicators
General summary of the city	<ul style="list-style-type: none"> * Population size * The total area of residential premises * Recreational areas provision * Provision of public health facilities * Average correspondence time
Recreation sphere	<ul style="list-style-type: none"> * Citizens' recreational areas provision * Average time of transport accessibility to the beaches * Average walking time to the embankments * Average walking time to parks
Sport sphere	<ul style="list-style-type: none"> * Swimming pools provision * The average time of accessibility to the swimming pools by public transport * Gyms provision
Demographic sphere	<ul style="list-style-type: none"> * Population under the working age * Working-age population * Population over the working age * The number of preschool children * Number of school-age children
Housing and maintenance services sphere	<ul style="list-style-type: none"> * The average tear of construction of residential buildings on the territory * The number of critical condition residential buildings
Healthcare sphere	<ul style="list-style-type: none"> * Provision of polyclinics * Average accessibility to the polyclinics by public transport * Average walking distance to the child clinics * Provision of the health emergency facilities
Culture and leisure sphere	<ul style="list-style-type: none"> * Average availability time to the library * Average walking distance to the museums * Average availability time to the theater * The average walking distance to cafes and restaurants
Education sphere	<ul style="list-style-type: none"> * Provision of kindergartens * Provision of schools * Average availability time to the school * Provision of the universities

Table A2. A fragment of the dataset with test questions and expected answers of the language model.

Question	Expected LLM Response	Question Category
What measures should be taken to ensure the development of high-speed passenger services and reduce the negative impact of freight trains?	To ensure the development of high-speed passenger services and reduce the negative impact of freight trains, it is necessary to create transport corridors bypassing St. Petersburg to allow transit traffic flows.	Question about the strategy of socio-economic development (about the whole city)
What is the goal of developing and launching effective tools that encourage economic entities, including industrial enterprises, to introduce environmentally friendly technologies?	The goal of developing and launching effective tools to encourage economic entities, including industrial enterprises, to introduce nature-saving technologies that are safe for the environment is to reduce the volume of energy resources used while maintaining their useful effect, and to develop alternative energy sources.	Question about the strategy of socio-economic development (about the whole city)
What were the consequences of the commissioning of a complex of protective structures in the Kurortny district of St. Petersburg?	After the commissioning of a complex of protective structures, the territory of the Kurortny district of St. Petersburg, located outside the protection zone, is more susceptible to the influence of rising water levels, which leads to an acceleration of the process of bank erosion and increased flooding of coastal areas, including residential areas, street road network, parks, beaches.	Question about the strategy of socio-economic development (about certain territory-Kurortny district)
What is the average availability time to the parks?	The average accessibility time to the park areas is 22.25 min.	Question about data obtained by referring to urban environment assessment models (about the whole city)
What is the provision of hospitals in the city?	The provision of hospitals in the city is 100%, and the provision can be considered good. The provision of hospitals in the city in the accessibility zone is 94.16%, the provision can be considered good.	Question about data obtained by referring to urban environment assessment models (about the whole city)
What is the number of people living in the Admiralteysky district?	The number of inhabitants in the Admiralteysky district is 395.73 thousand people.	Question about data obtained by referring to urban environment assessment models (about certain territories)

Appendix A.2. LLM Prompts

Appendix A.2.1. FC System Prompt

You are a helpful assistant with access to the following functions.
Use them if required - `{tools}`.

Appendix A.2.2. FC User Prompt

Extract all relevant data for answering this question: `{question}`.
You MUST return ONLY the function names separated by spaces.
Do NOT return any other additional text.

Appendix A.2.3. API System Prompt

Answer the question by following the rules below.
For the answer you must use the context provided by user.
Rules:
1. You must only use provided information for the answer.
2. Add a unit of measurement to the answer.

3. For answer you should take only the information from context, which is relevant to the user's question.
4. If an interpretation is provided in the context for the data requested in the question, it should be added in the answer.
5. If data for an answer is absent, reply that data was not provided or absent and mention for what field there was no data.
6. If you do not know how to answer the questions, say so.
7. Before giving an answer to the user question, provide an explanation. Mark the answer with keyword 'ANSWER', and explanation with 'EXPLANATION'.
8. If the question is about complaints, answer about at least 5 complaints topics.
9. Answer should be three sentences maximum.

Appendix A.2.4. DB System Prompt

Answer the question following the rules below. For answer you must use context provided by the user.

Rules:

1. You must use only provided information for the answer.
2. For answer you should take only that information from context, which is relevant to user's question.
3. If data for an answer is absent, answer that data was not provided or absent and mention for what field there was no data.
4. If you do not know how to answer the questions, say so.
5. Before giving an answer to the user question, provide an explanation. Mark the answer with keyword 'ANSWER', and explanation with 'EXPLANATION'.
6. The answer should consist of as many sentences as are necessary to answer the question given the context, but not more five sentences.

Appendix A.2.5. LLM-Without-Context Prompt

You are a smart AI assistant. You have high expertise in the field of city building, urbanistics and structure of Saint-Petersburg. Answer the question following the rules below.

1. Before giving an answer to the user question, provide an explanation. Mark the answer with keyword 'ANSWER', and explanation with 'EXPLANATION'. Both answer and explanation must be in the English language.
2. If the question is about complaints, answer about at least 5 complaints topics.
3. Answer should be five sentences maximum.
4. In answers you must use only the English language.

Appendix A.2.6. G-Eval Prompt

criteria=(

1. Correctness and Relevance:

- Compare the actual response against the expected response. Determine the extent to which the actual response captures the key elements and concepts of the expected response.
 - Assign higher scores to actual responses that accurately reflect the core information of the expected response, even if only partial.
2. Numerical Accuracy and Interpretation:
- Pay particular attention to any numerical values present in the expected response. Verify that these values are correctly included in the actual response and accurately interpreted within the context.
 - Ensure that units of measurement, scales, and numerical relationships are preserved and correctly conveyed.
3. Allowance for Partial Information:
- Do not heavily penalize the actual response for incompleteness if it covers significant aspects of the expected response. Prioritize the correctness of provided information over total completeness.
4. Handling of Extraneous Information:
- While additional information not present in the expected response should not necessarily reduce score, ensure that such additions do not introduce inaccuracies or deviate from the context of the expected response.)

Appendix A.3. Additional Metrics

Table A3. The average response time to test questions.

Model	LLM, s	LLM + ChromaDB, s	LLM + API, s	LLM + ChromaDB + API, s
gpt-4o-2024-08-06	2.6	6.1	17.4	12.1
mixtral-8x22b-instruct	3.8	4.2	17.1	10.5
llama-3.1-70b-instruct	7.4	10.0	21.8	14.6
llama-3.1-70b-instruct-int4	2.5	3.3	15.2	9.2

Table A4. G-Eval metric results for the subset of questions on the city's development strategy and G-Eval metric results with the percent of correct answers for the subset of questions on the accessibility of city services.

Model	Subset of Questions					
	City Development Strategy, G-Eval		Accessibility of City Services, G-Eval		Accessibility of City Services, %	
	LLM	LLM + ChromaDB	LLM	LLM + API	LLM	LLM + API
gpt-4o-2024-08-06	0.44	0.64	0.17	0.81	0.0	82.0
mixtral-8x22b-instruct	0.46	0.64	0.30	0.76	1.3	80.0
llama-3.1-70b-instruct	0.49	0.65	0.27	0.83	0.0	96.0
llama-3.1-70b-instruct-int4	0.49	0.62	0.27	0.80	0.0	94.0

Table A5. G-Eval metric results with confidence intervals (95%).

Model	LLM	LLM + ChromaDB	LLM + API	LLM + ChromaDB + API
gpt-4o-2024-08-06	0.3 (0.28, 0.34)	0.38 (0.34, 0.44)	0.52 (0.47, 0.57)	0.71 (0.67, 0.74)
mixtral-8x22b-instruct	0.38 (0.35, 0.41)	0.43 (0.39, 0.47)	0.51 (0.47, 0.57)	0.68 (0.64, 0.71)
llama-3.1-70b-instruct	0.384 (0.35, 0.41)	0.41 (0.36, 0.46)	0.54 (0.49, 0.59)	0.74 (0.71, 0.78)
llama-3.1-70b-instruct-int4	0.38 (0.35, 0.41)	0.39 (0.34, 0.44)	0.5 (0.44, 0.55)	0.71 (0.67, 0.75)

Table A6. The effectiveness of the LLM multi-agent system compared to human experts in the main stages of the city development planning process.

The Stage of the Process	Option 1		Option 2		Option 3	
Assessment of the cost of resources for the preparation of information and analytical materials	Time costs	The number of specialists in addition to the decision-maker	Time costs	Number of specialists in the information support group	Time costs	The number of external specialists involved
Collection and preparation of initial data	0.1–0.5 h	0	2–8 h	3–5	1–4 days	5–10
Identification of problematic situations based on urban environment data	0.1–0.5 h	0	1–2 days	3–5	1–5 days	3–5
Analysis of complaints and appeals from citizens	0.1–0.5 h	0	1–4 days	3–5	5–15 days	5–10
Comparison with strategic and territorial planning documents	0.1–0.5 h	0	1–4 days	1–2	2–5 days	1–2
Identification of priority areas of urban policy	0.1–0.5 h	0	2–5 days	1–2	2–15 days	1–2
Formation of a list of projects and solutions in accordance with the priority directions of urban policy	not rated	0	5–30 days	3–5	15–30 days	3–5
Identification of territories that form the potential of the city's development	0.5–4 h	0	2–10 days	1–2	5–15 days	1–2
Placement of projects and solutions in identified areas	1–5 days	2	15–30 days	3–5	15–30 days	3–5
Assessment of the expected effect of project implementation	0.1–1 h	0	30–60 days	3–5	is not produced	

Appendix A.4. Case Study Examples

Appendix A.4.1. Example 1

- Question: As part of the implementation of what task is it necessary to ensure the possibility of developing a network of courtyard sports grounds, stadiums, and sports clubs in general education organisations in St. Petersburg?

- Correct answer: As part of the task “Creating infrastructure for regular physical education and sports within walking distance”, it is necessary to ensure the possibility of developing a network of courtyard sports grounds, stadiums, and sports clubs in general education organisations in St. Petersburg.

- Model answer: The task is to ensure the possibility of developing a network of courtyard sports grounds, stadiums, and sports clubs in general education organisations in St. Petersburg as part of the implementation of the “Development of Physical Culture and Sports” task.

- Correctness: 0.8

Appendix A.4.2. Example 2

- Question: What is the availability of universities for the population of the Admiralteysky district?

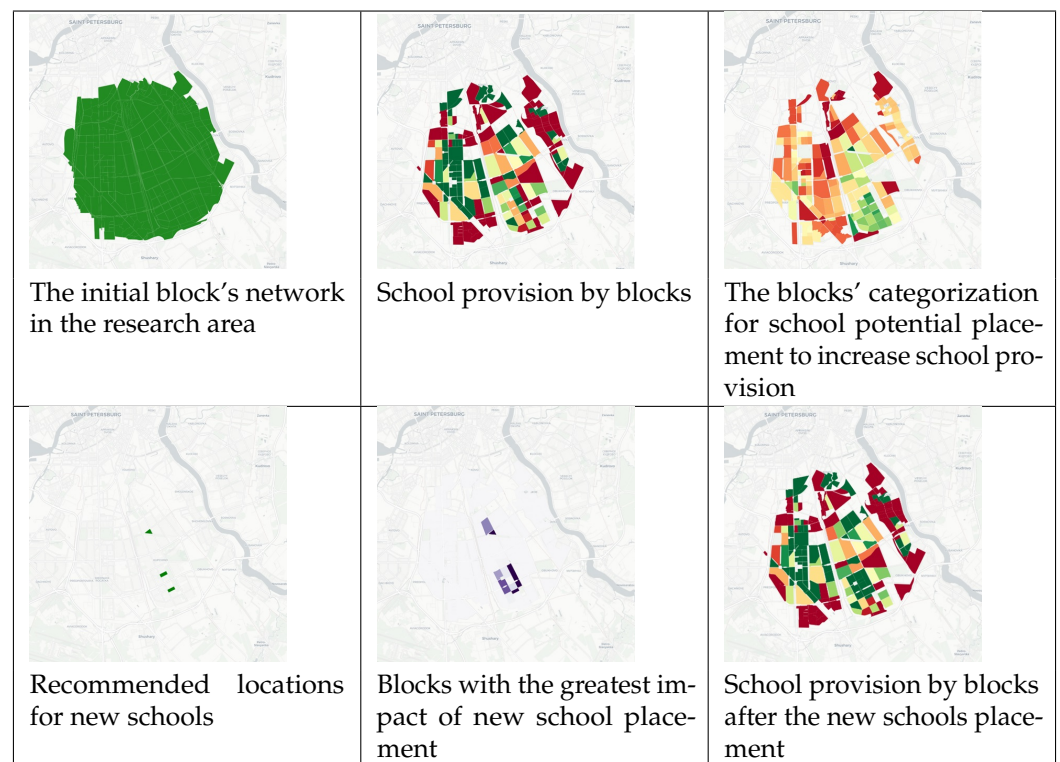
- Correct answer: The provision of universities is 100%, the provision can be considered good. Security in the availability zone is 100%, and it can also be considered good.

- Model answer: The availability of universities for the population of the Admiralteysky district is 100.0% in the zone of accessibility and 100.0% in general, with 26 higher educational institutions available.

- Correctness: 0.8

Appendix A.4.3. Example 3

Table A7. Steps in the process of preparing a new school placement planning program. The process stages shown in the table are typical, but the details may vary when implemented in different cities around the world. The differences may relate to both the composition of the stages and the timing of their execution. At the same time, the composition of the source data may also vary and depends both on the availability of available data and on the established traditions of their use in management. For example, in [78–80], data preparation procedures are the prevailing stages. At the same time, Mohammadi [78] notes that data preparation takes a significant amount of time throughout the project. Deadlines are most often determined by current regulations, and regulations and may differ in different cities [78]. The table shows estimates of deadlines based on the experience of the authors and regulatory documents. Reducing decision making time through digitization is a common expectation in this area [81].



References

1. Martín-Rojo, I. Strategic Planning for a Smart Sustainable City Model: The Importance of Public Administration and Enterprises Cooperation. In *The Strategic Paradigm of CSR and Sustainability*; Palgrave Macmillan: Cham, Switzerland, 2024.
2. Momot, T.; Kraivska, I.; Triplett, R.; Azueta, A.C.; Kuznicki, S. Sustainable Roadmap to Global Smart Cities: A Comparative Analysis of Smart City Strategic Plans. In *Smart Technologies in Urban Engineering*; Springer: Cham, Switzerland, 2023.
3. Jangeed, D.; Mohammad, I.; Patel, J. Sensor Based Smart Traffic Light Control System. *Int. J. Tech. Res. Sci.* **2024**, *9*, 27–35. <https://doi.org/10.30780/specialissue-ISET-2024/022>.

4. Min, K. A Study on the Application of Smart Technology to Improve the Safety of Smart Cities. *Forum Public Saf. Cult.* **2023**, *24*, 167–185. <https://doi.org/10.52902/kjsc.2023.24.167>.
5. Lee, C.; Park, J.; Seol, S. Development and demonstration of smart construction safety technology using drones. *Forum Public Saf. Cult.* **2023**, *24*, 93–105. <https://doi.org/10.52902/kjsc.2023.24.93>.
6. Chen, T.C. Smart Technology Applications in Healthcare Before, During, and After the COVID-19 Pandemic. In *Sustainable Smart Healthcare. SpringerBriefs in Applied Sciences and Technology*; Springer: Cham, Switzerland, 2023; pp. 19–37. https://doi.org/10.1007/978-3-031-37146-2_2.
7. Urban Platform City. Available online: <https://urbanplatform.city> (accessed on 15 October 2024).
8. City Digital Data Platform. Available online: <https://bedrockanalytics.ai/products/city-digital-data-platform> (accessed on 15 October 2024).
9. CityEngine. Available online: <https://www.esri.com/ru-ru/arcgis/products/arcgis-cityengine/overview> (accessed on 15 October 2024).
10. Urban Observatory. Available online: <https://urbanobservatory.maps.arcgis.com/home/index.html> (accessed on 15 October 2024).
11. UrbanSim. Available online: <https://www.urbansim.com> (accessed on 15 October 2024).
12. Antony, R.; Sunder, R. A Review on Data-Driven Approach Applied for Smart Sustainable City: Future Studies. In *Proceedings of International Conference on Data Science and Applications*; Springer: Singapore, 2023; pp. 875–890. https://doi.org/10.1007/978-981-19-6631-6_61.
13. Kandt, J.; Batty, M. Smart cities, big data and urban policy: Towards urban analytics for the long run. *Cities* **2020**, *109*, 102992. <https://doi.org/10.1016/j.cities.2020.102992>.
14. Wolniak, R.; Stecula, K. Artificial Intelligence in Smart Cities—Applications, Barriers, and Future Directions: A Review. *Smart Cities* **2024**, *7*, 1346–1389. <https://doi.org/10.3390/smartcities7030057>.
15. Szpilko, D.; Jiménez Naharro, F.; Lăzăroiu, G.; Nica, E.; de-la-torre Gallegos, A. Artificial Intelligence in the Smart City—A Literature Review. *Eng. Manag. Prod. Serv.* **2023**, *15*, 53–75. <https://doi.org/10.2478/emj-2023-0028>.
16. Gracias, J.; Parnell, G.; Pohl, E.; Buchanan, R. Smart Cities—A Structured Literature Review. *Smart Cities* **2023**, *6*, 1719–1743. <https://doi.org/10.3390/smartcities6040080>.
17. Akhrian Syahidi, A.; Kiyokawa, K.; Okura, F. Computer Vision in Smart City Application: A Mapping Review. In *Proceedings of the 2023 6th International Conference on Applied Computational Intelligence in Information Systems (ACIIS)*, Bandar Seri Begawan, Brunei Darussalam, 23–25 October 2023. <https://doi.org/10.1109/ACIIS59385.2023.10367332>.
18. Mukhina, K.; Visheratin, A.; Nasonov, D. Urban events prediction via convolutional neural networks and Instagram data. *Procedia Comput. Sci.* **2019**, *156*, 176–184. <https://doi.org/10.1016/j.procs.2019.08.193>.
19. Yereseme, A.; Surendra, H.; Kuntoji, G. Sustainable integrated urban flood management strategies for planning of smart cities: A review. *Sustain. Water Resour. Manag.* **2022**, *8*, 85. <https://doi.org/10.1007/s40899-022-00666-5>.
20. Popov, A.; Popov, A.; Ovsyankin, A.; Schneider, V.; Evstratov, A. Development of a module for environmental monitoring of the living condition of landscaping facilities using neural networks. *IOP Conf. Ser. Earth Environ. Sci.* **2022**, *1112*, 012145. <https://doi.org/10.1088/1755-1315/1112/1/012145>.
21. Bamwesigye, D.; Hlaváčková, P. Analysis of Sustainable Transport for Smart Cities. *Sustainability* **2019**, *11*, 2140. <https://doi.org/10.3390/su11072140>.
22. Smirnova, O.; Zhukova, N. Smart Navigation for Modern Cities. In *Proceedings of 19th International Conference on Urban Planning, Regional Development and Information Society*; Springer: Cham, Switzerland, 2014; pp. 593–602.
23. Rebelo, F.; Noriega, P.; De Oliveira, T.; Santos, D.; Oliveira, S. Expected User Acceptance of an Augmented Reality Service for a Smart City. In *Design, User Experience, and Usability: Users, Contexts and Case Studies. DUXU 2018*; Springer: Cham, Switzerland, 2018; pp. 703–714. https://doi.org/10.1007/978-3-319-91806-8_55.
24. Alzahrani, N.; Alfouzan, F. Augmented Reality (AR) and Cyber-Security for Smart Cities—A Systematic Literature Review. *Sensors* **2022**, *7*, 2792. <https://doi.org/10.3390/s22072792>.
25. Arkaraprasertkul, N. AI-Powered Smart Cities: Transforming Urban Living with LLM. 2024. Available online: <https://nonsmartcity.medium.com/ai-powered-smart-cities-transforming-urban-living-with-llm-9230b154b425>, (accessed on 15 October 2024).
26. Digital Urban Platform St. Petersburg. Available online: <https://dc.idu.actcognitive.org/> (accessed on 15 October 2024).
27. Zhang, W.; Han, J.; Xu, Z.; Ni, H.; Liu, H.; Xiong, H. Urban Foundation Models: A Survey. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Barcelona, Spain, 8 February 2024. <https://doi.org/10.1145/3637528.3671453>.
28. Yuan, Y.; Han, C.; Ding, J.; Jin, D.; Li, Y. UrbanDit: a foundation model for openworld urban spatio-temporal learning. *arXiv* **2024**, arXiv:2411.12164. <https://arxiv.org/pdf/2411.12164>.

29. Awesome-Urban-Foundation-Models. Available online: <https://github.com/usail-hkust/Awesome-Urban-Foundation-Models> (accessed on 15 October 2024).
30. Eigner, E.; Händler, T. Determinants of llm-assisted decision-making. *arXiv* **2024**, arXiv:2402.17385.
31. Benary, M.; Wang, X.D.; Schmidt, M.; Soll, D.; Hilfenhaus, G.; Nassir, M.; Sigler, C.; Knödler, M.; Keller, U.; Beule, D.; et al. Leveraging large language models for decision support in personalized oncology. *JAMA Netw. Open* **2023**, *6*, e2343689.
32. Dhar, R.; Vaidhyanathan, K.; Varma, V. Can LLMs Generate Architectural Design Decisions?—An Exploratory Empirical study. *arXiv* **2024**, arXiv:2403.01709.
33. Xu, Z.; Guo, L.; Zhou, S.; Song, R.; Niu, K. Enterprise supply chain risk management and decision support driven by large language models. *Appl. Sci. Eng. J. Adv. Res.* **2024**, *3*, 1–7.
34. Handler, A.; Larsen, K.R.; Hackathorn, R. Large language models present new questions for decision support. *Int. J. Inf. Manag.* **2024**, *79*, 102811.
35. Laskar, M.T.R.; Alqahtani, S.; Bari, M.S.; Rahman, M.; Khan, M.A.M.; Khan, H.; Jahan, I.; Bhuiyan, A.; Tan, C.W.; Parvez, M.R.; et al. A Systematic Survey and Critical Review on Evaluating Large Language Models: Challenges, Limitations, and Recommendations. *arXiv* **2024**, arXiv:2407.04069.
36. Perković, G.; Drobniak, A.; Botički, I. Hallucinations in LLMs: Understanding and addressing challenges. In Proceedings of the 2024 47th MIPRO ICT and Electronics Convention (MIPRO), Opatija, Croatia, 20–24 May 2024; pp. 2084–2088.
37. Liu, J.; Lin, J.; Liu, Y. How Much Can RAG Help the Reasoning of LLM? *arXiv* **2024**, arXiv:2410.02338.
38. Jin, H.; Huang, L.; Cai, H.; Yan, J.; Li, B.; Chen, H. From LLMs to LLM-based Agents for Software Engineering: A Survey of Current, Challenges and Future. *arXiv* **2024**, arXiv:2408.02479.
39. Xiao, Z.; Zhang, D.; Wu, Y.; Xu, L.; Wang, Y.J.; Han, X.; Fu, X.; Zhong, T.; Zeng, J.; Song, M.; et al. Chain-of-Experts: When LLMs Meet Complex Operations Research Problems. In Proceedings of the Twelfth International Conference on Learning Representations, Vienna, Austria, 7–11 May 2024.
40. Trirat, P.; Jeong, W.; Hwang, S.J. AutoML-Agent: A Multi-Agent LLM Framework for Full-Pipeline AutoML. *arXiv* **2024**, arXiv:2410.02958.
41. Liu, Z.; Yao, W.; Zhang, J.; Yang, L.; Liu, Z.; Tan, J.; Choubey, P.K.; Lan, T.; Wu, J.; Wang, H.; et al. AgentLite: A Lightweight Library for Building and Advancing Task-Oriented LLM Agent System. *arXiv* **2024**, arXiv:2402.15538.
42. Li, Y.; Wen, H.; Wang, W.; Li, X.; Yuan, Y.; Liu, G.; Liu, J.; Xu, W.; Wang, X.; Sun, Y.; et al. Personal llm agents: Insights and survey about the capability, efficiency and security. *arXiv* **2024**, arXiv:2401.05459.
43. Nan, L.; Zhang, E.; Zou, W.; Zhao, Y.; Zhou, W.; Cohan, A. On evaluating the integration of reasoning and action in llm agents with database question answering. *arXiv* **2023**, arXiv:2311.09721.
44. Wang, Q.; Wang, Z.; Su, Y.; Tong, H.; Song, Y. Rethinking the Bounds of LLM Reasoning: Are Multi-Agent Discussions the Key? *arXiv* **2024**, arXiv:2402.18272.
45. Guo, X.; Huang, K.; Liu, J.; Fan, W.; Vélez, N.; Wu, Q.; Wang, H.; Griffiths, T.L.; Wang, M. Embodied llm agents learn to cooperate in organized teams. *arXiv* **2024**, arXiv:2403.12482.
46. Feng, J.; Du, Y.; Liu, T.; Guo, S.; Lin, Y.; Li, Y. CityGPT: Empowering Urban Spatial Cognition of Large Language Models. *arXiv* **2024**, arXiv:2406.13948.
47. Jiao, Z.; Sha, M.; Zhang, H.; Jiang, X.; Qi, W. City-LEO: Toward Transparent City Management Using LLM with End-to-End Optimization. *arXiv* **2024**, arXiv:2406.10958.
48. OpenStreetMap. Available online: <https://www.openstreetmap.org> (accessed on 15 October 2024).
49. Aino.World. Data Sources for Spatial Data Work. Available online: https://aino.world/data_sources/ (accessed on 15 October 2024).
50. Overture Maps Foundation. Overture Maps. Available online: <https://overturemaps.org/> (accessed on 15 October 2024).
51. Microsoft. Microsoft Planetary Computer: Buildings Dataset. Available online: <https://planetarycomputer.microsoft.com/dataset/ms-buildings> (accessed on 15 October 2024).
52. Humanitarian Data Exchange (HDX). Kontur Population Dataset. Available online: <https://data.humdata.org/dataset/kontur-population-dataset> (accessed on 15 October 2024).
53. Zenodo. Ensemble Digital Terrain Model (EDTM) of the World. Available online: <https://zenodo.org/records/7634679> (accessed on 15 October 2024).
54. Google Developers. Geocoding API Overview. Available online: <https://developers.google.com/maps/documentation/geocoding/overview> (accessed on 15 October 2024).
55. HERE Technologies. HERE Maps. Available online: <https://www.here.com> (accessed on 15 October 2024).
56. Kaggle. Pronto Cycle Share Dataset. Available online: <https://www.kaggle.com/datasets/pronto/cycle-share-dataset> (accessed on 15 October 2024).
57. Mishina, M.; Khrulkov, A.; Soloveva, V.; Tupikina, L.; Mityagin, S. Method of intermodal accessibility graph construction. *Procedia Comput. Sci.* **2022**, *212*, 42–50. <https://doi.org/10.1016/j.procs.2022.10.206>.

58. Mishina, M.; Sobolevsky, S.; Kovtun, E.; Khrulkov, A.; Belyi, A.; Budenny, S.; Mityagin, S. Prediction of Urban Population-Facilities Interactions with Graph Neural Network. In *Computational Science and Its Applications*; Springer: Cham, Switzerland, 2023. https://doi.org/10.1007/978-3-031-36805-9_23.
59. Mishina, M.; Mityagin, S.; Belyi, A.; Khrulkov, A.; Sobolevsky, S. Towards Urban Accessibility: Modeling Trip Distribution to Assess the Provision of Social Facilities. *Smart Cities* **2024**, *7*, 2741–2762. <https://doi.org/10.3390/smartcities7050106>.
60. Morozov, A.; Kontsevik, G.; Shmeleva, I.; Schneider, L.; Zakharenko, N.; Budenny, S.; Mityagin, S. Assessing the transport connectivity of urban territories and based on intermodal transport accessibility. *Front. Built Environ.* **2023**, *9*, 1148708. <https://doi.org/10.3389/fbuil.2023.1148708>.
61. Natykin, M.V.; Morozov, A.; Starikov, V.A.; Mityagin, S. A method for automatically identifying vacant area in the current urban environment based on open source data. *Procedia Comput. Sci.* **2023**, *229*, 91–100. <https://doi.org/10.1016/j.procs.2023.12.040>.
62. Pavlova, A.; Katynsus, A.; Natykin, M.; Mityagin, S. Automated Identification of Existing and Potential Urban Central Places Based on Open Data and Public Interest. In *Computational Science and Its Applications—ICCSA 2024. ICCSA 2024; Lecture Notes in Computer Science*; Springer: Cham, Switzerland, 2024; Volume 14813. https://doi.org/10.1007/978-3-031-64605-8_28.
63. Territorial Development Fund. Available online: <https://xn--80aafuec2bqq9b9f.xn--p1aee.xn--p1ai/> (accessed on 15 October 2024).
64. Department of the Federal State Statistics Service for St. Petersburg and the Leningrad Region of the Russian Federation. Available online: <https://78.rosstat.gov.ru> (accessed on 15 October 2024).
65. Kontsevik, G.; Sokol, A.; Bogomolov, Y.; Mityagin, S. Modeling the citizens' settlement in residential buildings. *Procedia Comput. Sci.* **2022**, *212*, 51–63. <https://doi.org/10.1016/j.procs.2022.10.207>.
66. Polo, F.M.; Weber, L.; Choshen, L.; Sun, Y.; Xu, G.; Yurochkin, M. tinyBenchmarks: Evaluating LLMs with fewer examples. *arXiv* **2024**, arXiv:2402.14992.
67. Pacchiardi, L.; Cheke, L.G.; Hernández-Orallo, J. 100 instances is all you need: Predicting the success of a new LLM on unseen data by testing on a few instances. *arXiv* **2024**, arXiv:2409.03563.
68. Wang, J.; Ma, Z.; Li, Y.; Zhang, S.; Chen, C.; Chen, K.; Le, X. GTA: A Benchmark for General Tool Agents. *arXiv* **2024**, arXiv:2407.08713.
69. Liu, Y.; Iter, D.; Xu, Y.; Wang, S.; Xu, R.; Zhu, C. G-eval: Nlg evaluation using gpt-4 with better human alignment. *arXiv* **2023**, arXiv:2303.16634.
70. Es, S.; James, J.; Espinosa-Anke, L.; Schockaert, S. Ragas: Automated evaluation of retrieval augmented generation. *arXiv* **2023**, arXiv:2309.15217.
71. Schmidt, D.C.; Spencer-Smith, J.; Fu, Q.; White, J. Towards a catalog of prompt patterns to enhance the discipline of prompt engineering. *ACM SIGAda Ada Lett.* **2024**, *43*, 43–51.
72. Chen, B.; Zhang, Z.; Langrené, N.; Zhu, S. Unleashing the potential of prompt engineering in Large Language Models: A comprehensive review. *arXiv* **2023**, arXiv:2310.14735.
73. The Law of St. Petersburg “On the Strategy of Socio-Economic Development St. Petersburg for the Period up to 2035” Dated December 19, 2018 N 771-164. Available online: https://www.gov.spb.ru/gov/otrasl/c_econom/strategiya-ser-2035 (accessed on 15 October 2024).
74. Wang, B. Ethical Reflections on the Application of Artificial Intelligence in the Construction of Smart Cities. *J. Eng.* **2024**. <https://doi.org/10.1155/2024/8207822>.
75. Ehwi, R.; Holmes, H.; Maslova, S.; Burgess, G. The ethical underpinnings of Smart City governance: Decision-making in the Smart Cambridge programme, UK. *Urban Stud.* **2022**, *59*, 2968–2984. <https://doi.org/10.1177/00420980211064983>.
76. Xiao, P.; Xu, J.; Zhao, C. Conflict Identification and Zoning Optimization of “Production-Living-Ecological” Space. *Int. J. Environ. Res. Public Health* **2022**, *19*, 7990. <https://doi.org/10.3390/ijerph19137990>.
77. Zubizarreta, I.; Seravalli, A.; Arrizabalaga, S. Smart City Concept: What It Is and What It Should Be. *J. Urban Plan. Dev.* **2015**, *142*, 04015005. [https://doi.org/10.1061/\(ASCE\)UP.1943-5444.0000282](https://doi.org/10.1061/(ASCE)UP.1943-5444.0000282).
78. Mohammadi, A. The Pathology of Urban Master Plans in Iran. In *Proceedings of the International Conference on Civil Engineering, Architecture and Urban Cityscape*, Istanbul, Turkey, 28 July 2016.
79. Pleshkanovska, A. City Master Plan: Forecasting Methodology Problems (on the example of the Master Plans of Kyiv). *Transf. Innov. Technol.* **2019**, *2*, 39–50. <https://doi.org/10.31493/tit1921.0201>.

80. Abbas, S.; Ebraheem, M. Tactical Urban Projects Within Baghdad's Master Plan. *Int. J. Sustain. Dev. Plan.* **2024**, *19*, 4167–4182. <https://doi.org/10.18280/ijstdp.191107>.
81. Kumar, T. Smarter Master Planning. In *Smart Master Planning for Cities. Advances in 21st Century Human Settlements*; Springer: Singapore, 2022; pp. 3–79. https://doi.org/10.1007/978-981-19-2564-1_1.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.