UDC 004.9: 303.732.4
DOI: 10.20535/SRIT.2308-8893.2026.2.09

AUTOMATED SEMANTIC ONTOLOGY CONSTRUCTION FOR FORESIGHT STUDIES USING LARGE LANGUAGE MODELS

S.A. LUPENKO, M.V. STOLIAR, O.M. TERENTIEV, V.V. SAVASTIYANOV

Abstract. Recent advances in large language models (LLMs) enable the automated discovery of semantic structures and emerging signals within text streams, offering an opportunity to redesign foresight workflows into continuous, data-driven systems. This study aims to develop and validate an automated framework for extracting, structuring, and comparing semantic ontologies using LLMs. The paralyzed approach was used for data mining from social media platforms and filtering non-domain data. The key semantic elements, goals and hypernyms corresponded, were extracted using multiple LLM configurations, with a consensus mechanism to provide semantic reliability and minimize hallucination. The extracted elements were embedded in a high-dimensional vector space, clustered iteratively using cosine similarity, and merged hierarchically. Convergence process and structural stability were analyzed using the elbow criterion and similarity metrics. The Proposed approach provides a cost-efficient alternative to traditional expert-based foresight analysis. By integrating LLM-driven semantic extraction with quantitative clustering, it enables the identification of emerging trends, weak signals, and long-term thematic structures. The results highlight the potential of LLM-based semantic modeling as a foundation for automated foresight systems.

Keywords: foresight, large language models, semantic ontology, scenario analysis, weak signals, hierarchical clustering.

INTRODUCTION

In recent years, the growing complexity of global events and technological transformations has significantly increased the need for systematic foresight – the process of identifying, analyzing and interpreting trends and weak signals to possible futures [1]. Traditionally, foresight relies on expert discussions and panels, scenario workshops, and Delphi studies to capture and structure collective expectations about the future. While such methods provide deep contextual insights, they are slow, costly and difficult to scale, when applied to fast changing information environments. In other words, by the time you have an answer, the world has already moved on.

At the same time, we can observe new things, that millions of people are talking, arguing, and planning in real-time on platforms like Telegram, Facebook, X and others. These public conversations are a perfect data streams for anyone, who is trying to spot the next big thing. You can see new ideas forming in real-time. These data streams represent an opportunity for automated, data-driven foresight, but extracting meaningful structures from unstructured text (with a lot of spans, multiple languages, and it is full of noise) makes a challenges.

The construction of ontologies for decision support is a well-established field. A lot of scientist works in this domain [2–4]. Many previous studies have employed a “knowledge engineering” approach, focusing on manually constructing scenario-based ontologies to conceptualize complex processes [5]. These ontologies are then used to build knowledge graphs that support data integration and simulation, guiding the development of more efficient data provisioning systems [6]. This ontology-driven method has important for improving problem understanding and designing effective optimization workflows.

Recent progress in generative models has provided a potential solution for scalability challenge and gives a chance to work with the BigData. The emergence of high-capability Large Language Models (LLMs), including GPT-3 [7], GPT-4 [8], Gemini [9], Grock [10] has significantly advanced AI's capacity for complex reasoning. While much of the world has focused on their role in chatbots or autonomous agents, this paper explores their potential for a different, critical task: automated knowledge discovery. We investigate how the advanced understanding, reasoning, and generative power of LLMs can be leveraged to build the complex knowledge representations – the ontologies and graphs, that are essential for systematic foresight.

However, a traditional approach faces a significant bottleneck: it is limited by its dependence on subject-oriented, interdisciplinary human expertise. Constructing these ontologies is a laborious, manual process, making it difficult to scale or adapt to new, rapidly evolving challenges. We need a fundamental shift: from manual knowledge encoding to automated knowledge discovery. This is precisely where our work begins.

This study aims to develop an LLM-driven framework for automated extraction and hierarchical organization of collective goals from large-scale social data streams. It helps to organize unstructured data in hierarchical components. By using the reasoning capabilities of LLMs, with proper technics like prompt engineering, consensus decision-making, we try to approximate or even replace certain stages of expert analysis in the foresight process. Our goal is to build approach, that can identify, compare, and structure goal-related concepts with cost efficiency, temporal flexibility, and cross-lingual robustness, while maintaining interpretability suitable for foresight studies.

To show how it works in real world challenges, we took a massive dataset: three years of posts (2022–2025) from telegram-channel “Victory Drones” [11]. It is a key Telegram channel that discusses about military tech. This is popular thematic channel in Ukraine which also has to be used for future country development. Here are a lot of deep and interesting thoughts about implemented radio technologies. The data were parsed using asynchronous distributed pipelines with Python language and preprocessed to remove advertising and non-relevant posts. Texts were grouped by days and hours to study both long-term and short-term semantic goal dynamics. The first one, using multiple LLMs, goal candidates were extracted, semantically represented as vectors. The second one, embedding (numerical vectors that represent text data) iteratively clustered into hierarchical ontologies based on similarity metrics by gradient methods. These steps were repeated, allowing us to compare ontologies constructed from daily and hourly data using structural similarity measures. All of it leads to the creation of long-term future images or draft scenarios as a base component for foresight studies.

METHODOLOGY

The methodology formalizes a four-stage analytical pipeline designed to translate high-unstructured textual data into dynamic foresight ontology. As illustrated in the workflow diagram on Fig. 1, the system integrates data validation, semantic structuring, and temporal analysis, underlying the advanced reasoning capabilities of LLM with combinations of classic automatization approaches to automate processes traditionally reserved for human experts.

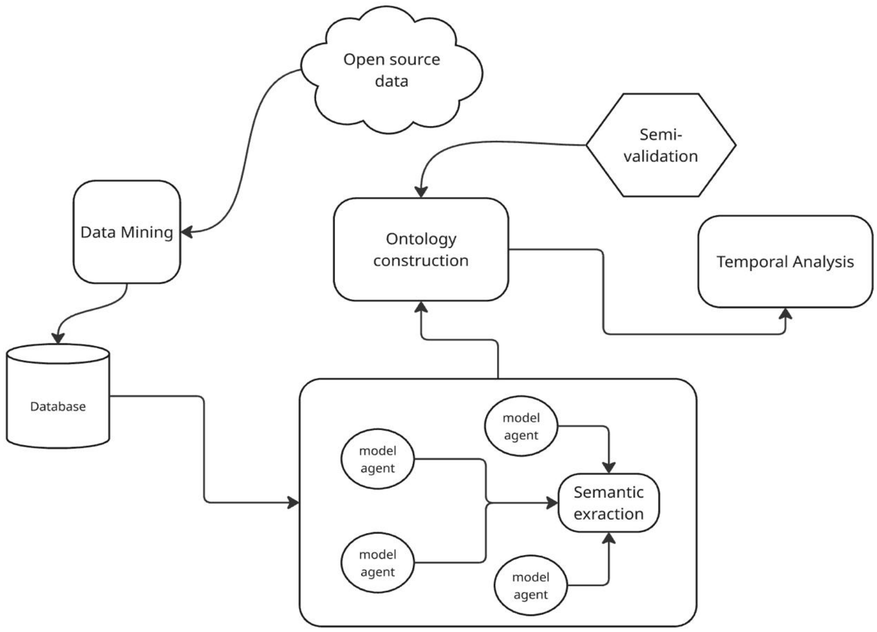

Fig. 1. Workflow of the four-stage analytical pipeline

DATA

The data set for this study was mined from Telegram messenger. The main channel is "Victory Drones" [11]. This source was purposively selected based on several criteria, which are critical for foresight analysis of domain area. "Victory Drones" [11] is the most popular channel specializing in military communication technologies, electronic warfare, and unmanned aerial systems. It is high engagement and expert-driven content provide a rich source of emerging terminology, technical discussions, and "weak signals" – indicators of future technological shifts. All of it does it a proper material for ontology construction and foresight study.

The collection period starts at October 2022 and ends September 2025, providing a long-range view of context evolution in this domain.

Data mining was provided by using Python libraries, with Telethon library as the core instrument for work with Telegram channels [12]. Asynchronous data parsing and distributed collection scripts were implemented to manage the large data volume and take count API rate limits [13]. For each post, we extracted: full text content (the data for textual analysis), publication timestamp (the critical metadata for all temporal analysis), unique post ID (for data integrity and deduplication), associated metadata (types of media attached as images, videos, likes, views, number of comments etc.).

A base linguistic analysis confirmed the complex multilingual structure of data corpus, with a significant presence of Ukrainian, Russian, and English texts. This aspect shows the international value of text, but it opens additional challenges for natural language processing [14].

To guarantee relevance of the dataset, we apply next filters:

1. Source Verification: only posts originating directly from the “Victory Drones” [11] channel administrator were retained. All forwarded messages from other channels or user comments were discarded to maintain a consistent and reduce noise.

2. Content Filtering: non-substantive posts, such as cross-promotional advertising, administrative announcements (example channel rules), and simple “thank you” messages, were identified and removed to focus the corpus on high-signal, domain-specific content.

3. Ethical Sourcing: all data collected was from a publicly accessible channel, reducing the need for user authentication and mitigating major privacy concerns. No private user data was accessed.

Finally, the dataset was grouped by time intervals to enable temporal analysis at multiple resolutions and validation of results: daily grouping for long-term trend and ontology construction, hourly grouping for granular, short-term dynamic analysis. The total number of daily texts is approximately 1000 observations. The same number of observations hourly grouping interval for 3rd quarter 2025.

APPROACH

The overall process of semantic structure formation is represented as a system S (1) to formalize the construction of a dynamic ontology from unstructured text. This model provides a scalable framework for managing the complex dependencies between text data, extracted meaning, structural relationships and temporal evolution:

$$S = < D, E, R, P, T > \quad (1)$$

Let’s explain each element from the system S.

The component D (2) represents the *Data Layer*. It is the set aggregated text documents as described in Data section. Each d_k is a “time slice” of the corpus, grouped either by day or by hour, forming the raw textual input for the system:

$$D = \{ d_1, d_2, \dots d_N \}, \quad (2)$$

where N – total number of observations.

The component E represents a *Semantic Layer*. It is the global set of all unique “conceptual atoms” extracted from the corpus. A Goal-hypernym pairs are core in the system S. Each hypernym provides taxonomic classification for corresponded goal element.

The component R represents the *Relational Layer*. It is the set of all relations between the semantic elements in E. While E is just a flat list of pairs, R models their connections like semantic similarity, parent-child hierarchies, co-occurrence in timeline.

The component P represents the *Procedural Layer*. This is the set of all computational procedures and algorithms that transform the data. Firstly, it is LLM inference for extracting E from D. Secondly, it is a vector embedding and clustering algorithms to define R, Thirdly, it is graph construction algorithms to build the final ontology, and finally it is evaluation metrics to validate its whole structure.

The component T represents the *Temporal Layer*. This is the set of operations that analyze the ontology evolution over time.

Multi-Model Extraction of Goals and Hypernyms

The core step of our methodology is the extraction of the semantic element set E from the data D. This may include some challenges, because of LLMs can hallucinate (as mentioned before – generate in some case possible but false information) or produce inconsistent outputs. To reduce these risks and provide high semantic consistency, we developed a multi-model ensemble approach. Each document d_k in D was processed in parallel by a mixed set of M (3) different LLMs:

$$M = \{ M_1, M_2, \dots M_m\} . \tag{3}$$

The models were chosen for their diverse architectures and training data to ensure a range of "opinions". For each model M_j and document d_i , we used a structured prompt engineering technique [15]. The prompt tasked the model to act as a domain expert and extract all conceptual pairs representing a specific technological capability (goal) and its general class (hypernym). The output of this step is a set of candidate pairs for that specific mode (4):

$$E_i^j = M_j(d_i) \ = \{ (g_i^j , h_i^j)_1, (g_i^j , h_i^j)_2, \dots (g_i^j , h_i^j)_{i_k})\}. \tag{4}$$

At the next step the results (each E_i^j) were effected by a consensus function FU (5), which aggregates only those pairs agreed upon by *at least two* models. Multimodal agents have recently demonstrated remarkable foresight capabilities in complex predictive tasks. In [16], En et al. introduce "Merlin", a vision-language model explicitly trained to develop "foresight minds":

$$e_i = FU(M_1(d_i), M_2(d_i), \dots M_m(d_i)) \ = \{ (g_i , h_i)_1, \dots \ (g_i , h_i)_{i_k}\} . \tag{5}$$

Finally, the global set E (6) is validated set by multi-modal approach, which is a part of nodes of our future ontology graph, which will be constructed and analyzed over time:

$$E = \ \cup_{i=1}^{K} e_i, \tag{6}$$

where K – total number of elements.

Construction of Semantic Space and Hierarchical Ontology

After creating the global set E of validated semantic elements $E = \{ e_i \mid i \epsilon [1, K]\}$, the next step requires transforming this unstructured set into a hierarchical ontology. This process can be achieved by embedding the elements within a high-dimensional semantic space. It helps to construct the hierarchical structure of domain area in modern way for foresight studies.

Semantic Space Projection

Let's formalize next statement: $\forall\ e_i \epsilon\ E, where\ i\ \epsilon\ [1, K]$ are mapped into a continuous semantic space using a pre-trained embedding function f (7):

$$v_i = f(e_i), f: E \to R^n. \tag{7}$$

Here, v_i describes the n-dimensional vector embedding of element e_i in semantic space. The selection of the embedding function f is important. It has to be an open-source model, that is a top performer on standardized benchmarks for multilingual text (especially for English, Ukrainian and Russian languages), such as the Massive Text Embedding Benchmark (MTEB) [17]. MTEB is a Python framework designed for the systematic evaluation of text embedding models and retrieval systems. We selected the text-embedding-3-large model from OpenAI, because it has superior performance in capturing fine-grained semantic relationships across technical and multilingual texts.

The next step is define distance metric $d(e_i, e_j)$ (8) in space E between two elements from E:

$$d(e_i, e_j)\ =\ 1\ -\ cos(v_i, v_j), d: E \times E\ \to\ R, \tag{8}$$

where e_i, $e_j \epsilon\ E$, $cos(\cdot,\cdot)$ – cosine distance.

Cosine distance was selected over Euclidean distance as it is invariant to vector magnitude and measures only the orientation between vectors. In high-dimensional spaces like text embedding, this is a more reliable measure of semantic similarity, where small distances $d \to 0$ indicate high similarity.

Hybrid Agglomerative Clustering

To build the ontology, we developed a hybrid algorithm that combines the algorithmic clustering with gradient optimization approaches and conceptual understanding of LLM. This process is iterative, building the hierarchy from the bottom up.

Phase one. Vector-Based Agglomeration

Let at any iteration t the set E is partitioned into $k^{(t)}$ disjoint clusters $C^{(t)} = \{C_1^{(t)}, C_2^{(t)}, \ldots C_{k(t)}^{(t)}\}$, where $\cup_j^{k(t)} C_j^{(t)} \supset E$ and $C_i^{(t)} \cap C_j^{(t)} = \emptyset$, $(i \neq j)$.

At the first iteration $C^{(0)} = \{e_j, j\ \in [1, k]\}$.

The base idea how to merge elements in one cluster – the shortest distance between any point in one cluster and any point in the other. This is the classic single-linkage rule (9):

$$min_c \sum_i^k \sum_{e_p, e_q \epsilon C_i} d(e_p, e_q). \tag{9}$$

This means clusters only merge if they are close on the inside and well-separated on the outside cluster (10). It stops loose or accidental links from forming early, so our chains stay clean and meaningful:

$$(C_{i^*}, C_{j^*})^t = argmin_{i \neq j} d_{min}(C_i^{(t)}, C_j^{(t)})\ \ . \tag{10}$$

So, the two (or more) clusters are merged to form a new cluster for the next iteration (11):

$$C_{new}^{(t+1)} = C_{i*}^{(t)} \cup C_{j^*}^{(t)}. \tag{11}$$

The last question on this phase is to determine criteria of optimal base cluster numbers. We use the "elbow criterion" applied to the within-cluster residual function $J(k)$ (12) to determine the optimal number of base clusters k^*. The residual is the sum of the squared distances to the cluster centroids.

$$J(k) = \sum_{i=1}^{k} \sum_{e_p \epsilon C_i} \left\| v_p - m_i \right\|^2, \tag{12}$$

where m_i is the centroid (13) of cluster C_i.

$$m_i = \frac{1}{|c_i|} \sum_{e_p \in C_i} v_p. \tag{13}$$

As k increases, $J(k)$ decreases. The elbow point k^* is detected from the discrete first differences (gradients) (14):

$$\Delta J(k) = J(k) - J(k-1). \tag{14}$$

The "elbow" k^* is identified as the point where the rate of residual stabilizes $\frac{\Delta J(k*)}{\Delta J(k*-1)} \to 1$, indicating that further merges would combine conceptually distinct groups. This leads to an optimal base partition.

Phase two. LLM-in-the-Loop Semantic Labeling and Merging

Once the k^* base clusters are identified, the algorithm shifts from just vector-based merging to a more abstract, concept-based merging using LLM.

Each base cluster $C_i \in C^{(t)}$ is "semantically labeled". We use prompt engineering with LLM to generate an abstract hyper-concept (hyperonym) L_i (15) that best describes all elements in the cluster. The prompt includes a representative sample of terms from the cluster, for example the 5–10 elements closest to the centroid m_i (13):

$$L_i^{(t+1)} = LLM(terms(C_i^{(t)})). \tag{15}$$

For example, a cluster containing "jam GPS", "spoof Galileo" and "disrupt GLONASS" might receive the label L_i = "GNSS Disruption Techniques".

The algorithm now proceeds to merge these k^* labeled clusters. Instead of using d_{min} on all vectors from E, we merge based on the semantic similarity of the LLM-generated labels. At each new iteration, the algorithm merges the two clusters C_i and C_j, whose labels L_i and $L_j$have the highest similarity from phase one. The newly formed cluster $C_{new} = C_i \cup C_j$ is then re-labeled by the LLM.

One of the most important things is how to work with graph structure. NetworkX Python library is dedicated graph database system, chosen for its efficiency in handling topological data and pathfinding queries, which helps establish the fundamental connectivity of roads. In the NetworkX environment, the graph components (nodes as cities, edges as routes) serve as the building blocks for decision optimization. To maximize network traversal performance, the graph schema stores only essential topological data and pre-calculated attributes (e.g., node/edge IDs, mode, distance, slope). These attributes are critical for applying necessary constraints during the traversal process, thereby guaranteeing query speed and relevance for decision support.

Phase three. Convergence and Ontology Stabilization

This hierarchical aggregation process (phase 2) proceeds iteratively until a stopping condition is met. The criteria includes: difference in total distance (12) less then threshold, cluster quality stabilizes or reached minimum clusters count.

The primary criterion based on the residual of the sum of the squared distances to the cluster centroids. Merging stops, when the newly generated $J(k_t^*)$ has no difference with $J(k_{t-1}^*)$: $|J(k_t^*) - J(k_{t-1}^*)| < \tau$, where τ is small value.

The second one is cluster quality stabilizes. The mean Silhouette score S (16) for the partition $C^{(t)}$ reaches a local maximum (changes by less than a small threshold ε):

$$S(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}}, \tag{16}$$

where $a(i)$ is average distance between i-element and all of other points in its own cluster and $b(i)$ is average distance between i-element and next nearest cluster centroid.

The last one is minimum cluster count. It is defined minimum number of top-level categories (it can be set by experts from domain area).

Empirically, we observed, that this semantic stabilization starts after approximately *five to seven* iterations, τ equals approximately 0.01, ε equals approximately 0.01. In the end a domains in the corpus have been successfully identified and organized. The final result is a tree-like hierarchical structure to represent final ontology.

Trend analysis

The final objective of our system is not to construct a static ontology, but to understand its evolution over time. This analysis is formalized through the Temporal Layer *T* of our system *S*. We analyze the frequency of the semantic elements *E* across different temporal aggregations to identify emerging, stabilizing, and disappearing trends.

To measure the frequency of an individual semantic element $e_i \in E$ at a specific time interval, we employ a TF-IDF approach [18]. Importance, to reduce noise in data, we will use elements only from first layer in constructed ontology. TF-IDF approach is adapted to our temporal framework, where the “document” is defined as a time-aggregated corpus.

The corpus D is temporally partitioned into a sequence of time slices $\{D^{(t)} | t = 1, T_{max}\}$, where t is the index of the temporal interval (e.g., month or day).

Term Frequency (TF) (17) is frequency of an element e_i within a specific time slice $D^{(t)}$:

$$TF(e_i, D^t) = \frac{Count\ of\ e_i \in D^t}{Total\ number\ of\ elements\ in\ D^t}. \tag{17}$$

Inverse Document Frequency (IDF) (18) is document frequency measures the number of time slices containing the element e_i:

$$IDF(e_i, \{D^t\}) = log(\frac{T_{max}}{|\{t: e_i \in D^t\}|}) \tag{18}$$

Then TF-IDF Score is score $\rho\ (e_i, t)$ (19) for element e_i at time *t*. The $\rho\ (e_i, t)$ scores allow us to quantify which goals and hypernyms were most distinctive during a given period, rather than just most frequent:

$$\rho\ (e_i, t) = TF(e_i, D^t) \times IDF(e_i, \{D^t\})\ . \tag{19}$$

To capture both macro-level shifts and micro-level volatility, we apply two distinct temporal aggregation strategies based on the desired analysis scope. *Long-Term Trend Analysis* is to analyze the evolution of the overall ontology across the entire multi-year corpus, the data is aggregated by month. This macro-level view smoothed out short-term noise, providing a clear picture of how high-level goals and technologies (hypernyms) emerge and stabilize over quarters and years. *Short-Term Dynamic Analysis* is to investigate localized tendencies and immediate responses, the data is aggregated by day. This finer-grained resolution allows us to detect rapid shifts in discussion focus, corresponding to the initial emergence. Together, these twin knowledge structures provide dual temporal information for foresight analysis, to underline the stability of long-term intentions with the volatility of short-term discursive dynamics.

RESULTS

Moving from theory to practice, this section introduces a compelling case study to demonstrate the application and practical utility of our proposed LLM-driven methodology in addressing a real-world decision challenge. Based on the methodology described above, we now present the results of the semantic extraction, hierarchical ontology construction, and comparative analysis across temporal resolutions. Our goal is to build approach to identify, structure, and compare goal-related concepts with cost efficiency, cross-lingual robustness, and temporal flexibility required for foresight studies.

We begin by describing the characteristics of the extracted semantic elements, including base prompt (translated in English from Ukrainian) for goals extraction with corresponded hypernyms, the numbers of semantic elements on different time intervals. Next, we analyze the hierarchical structure of the resulting ontologies, identifying key hypernyms and dominant goal classes that emerged over the studied period (2022–2025). Finally, we perform a comparative analysis between daily and hourly ontologies, assessing structural stability.

Goals and Hypernyms extraction. This stage implements the multi-model ensemble procedure described early, designed to extract reliable goal-hypernym pairs *E* from the raw text documents *D*. Each document $d_i \in D$ (representing a daily or hourly time slice) was processed by a set of five state-of-the-art Large Language

Models: GPT-3.5, GPT-4, Gemini, Grok, and DeepSeek. This diversity in model architecture and training data was chosen to minimize any single model's biases or hallucination effect.

The specific instruction provided to each model M_j was through a structured Base Prompt (translated into English): *I will provide you with news on the topic of the Russian-Ukrainian war. All posts are related to the topic of the Russian-Ukrainian war. Your task is to conduct an analytical analysis and submit the result exclusively in JSON format. Required: 1. Identify **goals** that are mentioned in the texts. - Consider short-term, long-term, tactical and strategic goals. - For each goal, highlight the key technologies/means that were used or are planned to achieve. 2. For each goal, determine its **hypernym** (a more general concept). Also provide a **hypernym for this hypernym** (i.e. the second level of generalization). 3. Identify **results** that are mentioned in the texts. – Results can also be short-term, long-term, tactical or strategic. – For each result, indicate the key technologies/tools that were used to achieve it. 4. For each result, also provide its **hypernym** and **hypernym to hypernym**. ### Response format (JSON): {{ "goals": [{{ "text": "liberation of a specific settlement", "type": "tactical / strategic / short-term / long-term", "technologies": ["kamikaze drones", "artillery"], "hypernym": "military operation", "hypernym_of_hypernym": "military activity" }}, ...], "results": [{{ "text": "destruction of ammunition depot", "type": "tactical result", "technologies": ["missile strike", "UAV"], "hypernym": "strike on military infrastructure", "hypernym_of_hypernym": "military activity" }}, ...] }} ### Important requirements: – Answer only in UKRAINIAN. – Do not invent data, but rely only on the posts provided. – If information is missing – leave an empty list or null. – Format the response only as valid JSON without additional comments. Here is the message text.*

To determine the relevance of the extracted semantic elements, we applied a consensus filtering function *FU*. A candidate pair was confirmed as a validated element $e_i \in E$ only if it was independently identified by at least two distinct LLMs. This threshold significantly reduced semantic noise and improved the confidence that the extracted elements genuinely represent the collective intent present in the source discourse. We will investigate adaptive thresholding mechanisms in the future work based on semantic similarity.

The effectiveness of this multi-model (multi-agent) extraction process is described by the resulting number of distinct semantic elements (goals and topics) identified per text. This distribution is the key to understanding the filling and temporal density of the corpus.

The distribution of the count of topics per text for the daily grouping is presented in Table 1.

Table 1. Daily topic count of Semantic elements distribution per document for long time period. Each semantic element is couple (goal, hypernym)

Count of semantic elements	Count of documents
1	86
2	183
3	193
4	85
5	26
6	5
8	2

The distribution of the count of topics per text for the hourly grouping is shown in Table 2.

Table 2. Hourly topic count of Semantic elements distribution per document for short time period. Each semantic element is couple (goal, hypernym)

Count of semantic elements	Count of documents
2	40
3	303
4	483
5	159
6	23
7	12
9	1
10	1
13	1

These Tables 1 and 2 visually represent the volume of validated semantic information available for long-term trend analysis (daily) versus local dynamic analysis (hourly).

The next step is to provide results of the hierarchical ontology construction. Firstly, the validated goal-hypernym pairs were mapped into a high-dimensional vector space using OpenAI's text-embedding-3-large model, earlier were described why we stop on those embedding model. Secondly, we use GPT-4 from the same LLM provider (OpenAI) to generate the abstract hyper-concepts for the higher levels of the ontology hierarchy. This strategic decision to use embedding and reasoning models from the same underlying provider, it is leaded to minimize potential semantic shift or misalignment, providing that the vector space used for clustering is highly congruent with the contextual understanding employed by the model generating the conceptual labels.

The vectors representing the extracted goals were clustered using a hierarchical agglomerative approach based, which is described upper in theory part. The first iteration of the resulting ontology structure is visualized in Fig. 2.

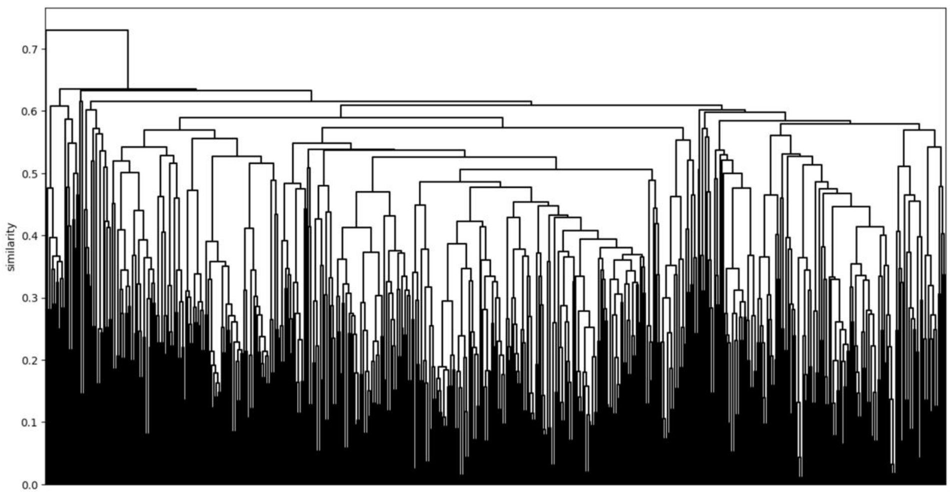

Fig. 2. Ontology structure which was obtained by using hierarchical agglomerative approach

To determine the optimal boundary for the initial cluster separation, we analyzed the change in inter-cluster distance (the "gradient" or first difference) across the hierarchy. This analysis identified a critical point, or the "best cut," at a semantic distance threshold of 0.35. It equals to 156 distinct clusters. Dependence between "total distance" of elements and cluster division threshold is illustrated at Fig. 3.

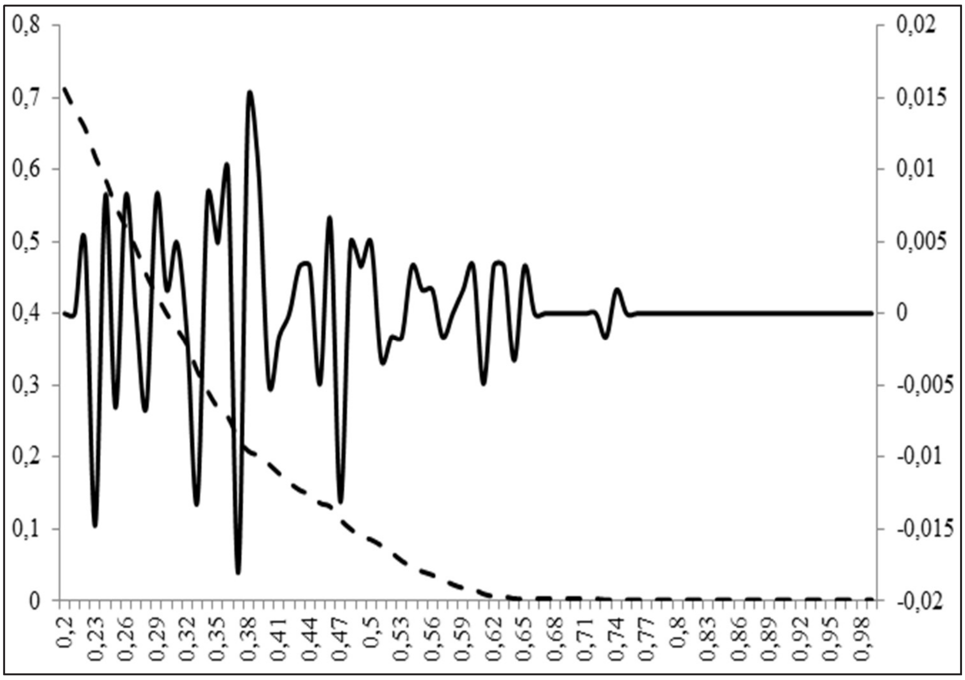

Fig. 3. Distance dependency. Solid line – second derivative of normalized total distance (left axis OY). Dashed line – normalized total distance (right axis OY). OX axis is a border value of semantic similarity

The next step was to generate high-level semantic descriptors for each cluster. For each of the 156 clusters C_i, a representative subset of elements was selected as up to ten semantic elements, that had the smallest cosine distance to the cluster's centroid m_i. These ten representative elements served as the input for GPT-4, which was tasked with generating the cluster label L_i (see formula 15). The prompt-institution for model:

"*You need to provide a hypernym for the list of terms Let me remind you that a hypernym is a word (or phrase) with a broader, generalized meaning, denoting a generic concept, class, or set of objects. Please provide the answer without comments, just the hypernym. List of terms*:"

Received labels, (representing abstract hyper-concepts) were used for the next iterations of the algorithm for higher-level merging, which is based on the semantic similarity between the labels themselves. Table 3 presents illustrative examples from this stage. Notably, the table is in the original multilingual format of the dataset to underscore the framework's cross-lingual robustness and to showcase the raw inputs processed by the LLM. The header of each column displays the LLM-generated hyper-concept, based on the top 10 nearest elements in corresponded cluster.

Table 3. Example of hyper-concepts

security	**military activity**	**logistics**	**education**	**technology**
security	support of military actions	logistics operations	education and training	Technology development
tactical security	military interaction	logistics project	educational programs	technical development
operational security	Military operations	military logistics	educational system	development of new technologies
territorial security	military observation	logistics system	educational process	technology development
National security	Military campaign	Logistics security	educational initiative	Technology implementation
protection of national security	supply of military means	logistics	education	technology development
Security provision	restructuring of the military fleet	innovations in logistics	educational activity	technology testing
security systems	military cooperation	Weapons logistics	education and science	scientific and technological progress
security enhancement	military security	logistics optimization	educational project	technological development
cybersecurity	Military communication	logistics support	educational infrastructure	technology development

The headers of each column (titled by bold) display the LLM-generated hyper-concepts, based on the top 10 nearest elements in corresponded clusters.

The iterative building process was monitored using key metrics to determine the optimal stopping point. The first one is average similarity distance between new names of clusters and old names. The second one is Silhouette score.

For the ontology derived from daily grouping, the iterative convergence process reached stability after five iterations. To investigate short-term semantic dynamics and reveal local fluctuations was replicated using the dataset aggregated at the hourly level (often interpreted in foresight studies as weak signals the entire pipeline). Crucially, the convergence process for the hourly-derived ontology also achieved stability after five iterations.

The final structural metrics for both ontologies are summarized in Table 4.

Table 4. Convergence metrics

Iteration	**Sematic distance day group**	**Sematic distance hour group**	**Silhouette score day group**	**Silhouette score hour group**
1	0.412	0.379	0.546	0.516
2	0.298	0.266	0.457	0.403
3	0.201	0.176	0.35	0.373
4	0.163	0.153	0.25	0.32
5	0.155	0.141	0.24	0.31

The global structure of both the daily and hourly ontologies is a big and complex knowledge graph. We must focus on a specific thematic area to illustrate the key findings of our temporal comparison.

Fig. 4 presents the final converged sub-graph for “Military Actions” which were derived by the Daily Grouping.

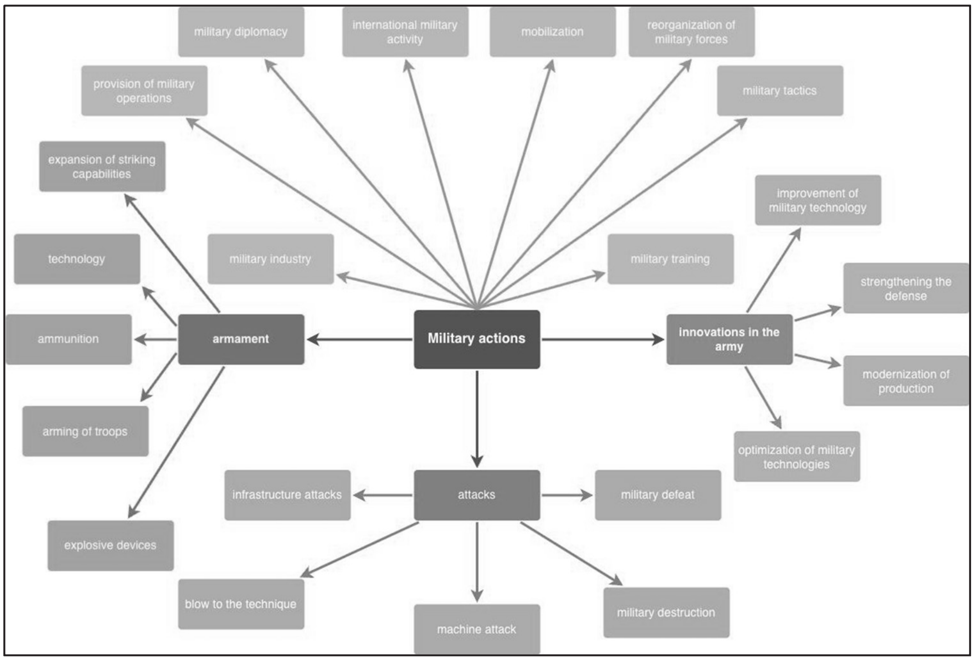

Fig. 4. “Military Actions” for day grouping

In contrast, Fig. 5 displays the equivalent “Military Actions” sub-graph which were derived by the Hourly Grouping

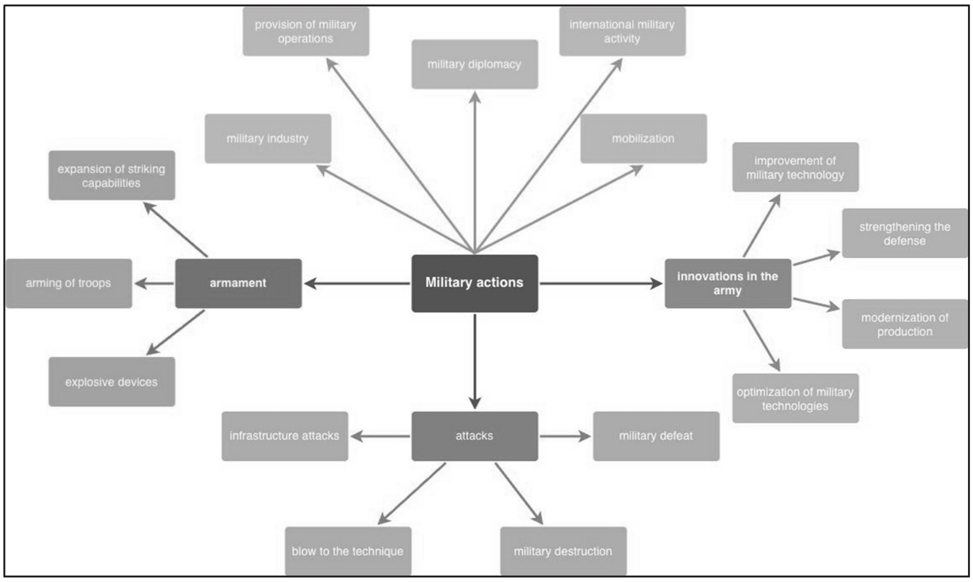

Fig. 5. “Military Actions” for hourly grouping

To validate the robustness of the proposed approach, a comparative analysis was conducted between the long-term (daily) and short-term (hourly) ontologies. This comparison serves as a semi-validation mechanism, allowing us to assess whether both temporal models capture a consistent semantic representation of the domain. Across both ontologies, a total of *886* unique thematic concepts were identified. Among these, *262* concepts were common to both structures, representing the core semantic intersection. The daily (long-term) ontology contained *405* unique topics not present in the short-term model, while the hourly (short-term) ontology introduced *219* distinctive topics absent from the long-term perspective.

The last part is to present the results of the temporal analysis, where the frequency of the established semantic elements E is tracked over time using the adapted TF-IDF score. To normalize visualization of work, we show the top thematic for both analysis: long-term and short-term.

To capture macro-level shifts and the strategic evolution, the prominence of high-level goals and hypernyms was aggregated and visualized by month across the entire study period. The corresponding heatmap presents on Fig. 6.

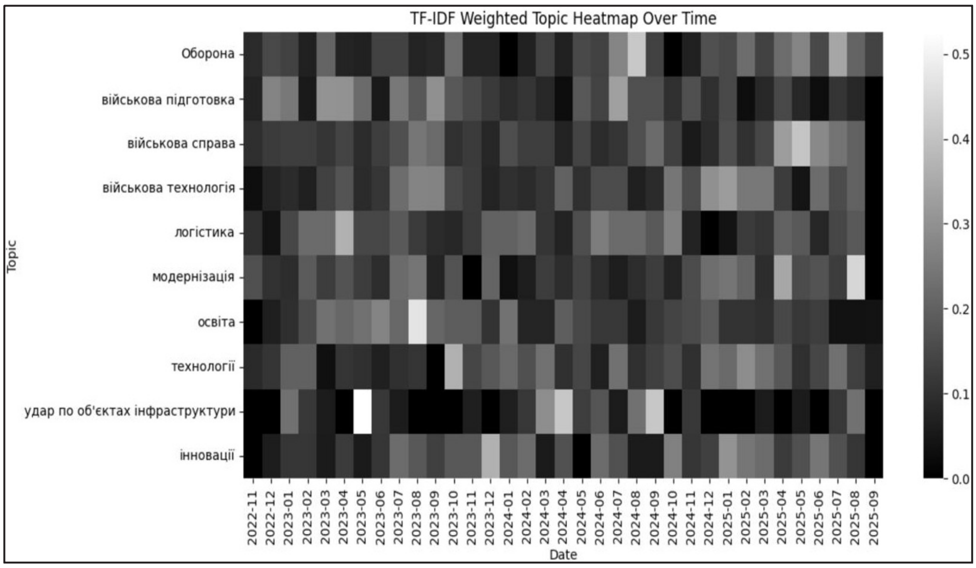

Fig. 6. TF-IDF for long-terms ontology

“Education” and “Attacks on Infrastructure Facilities” have decreased in frequency. Meanwhile topics related to “Innovation” and “Defense” demonstrate a sustained and increasing frequency of mention, indicating a long-term strategic interest. Core operational topics, such as “Military Activity” and “Logistics”, remain consistently present throughout the timeline.

To investigate micro-level volatility and detect the immediate impact of events, the analysis was replicated with data aggregated by day. The corresponding daily heatmap presents on Fig. 7.

Topics concerning Military Activity, Modernization, and Technology exhibit stable, high-frequency mentions across the daily periods. In contrast, other goal-related topics that spike following specific events tend to gradually fade away.

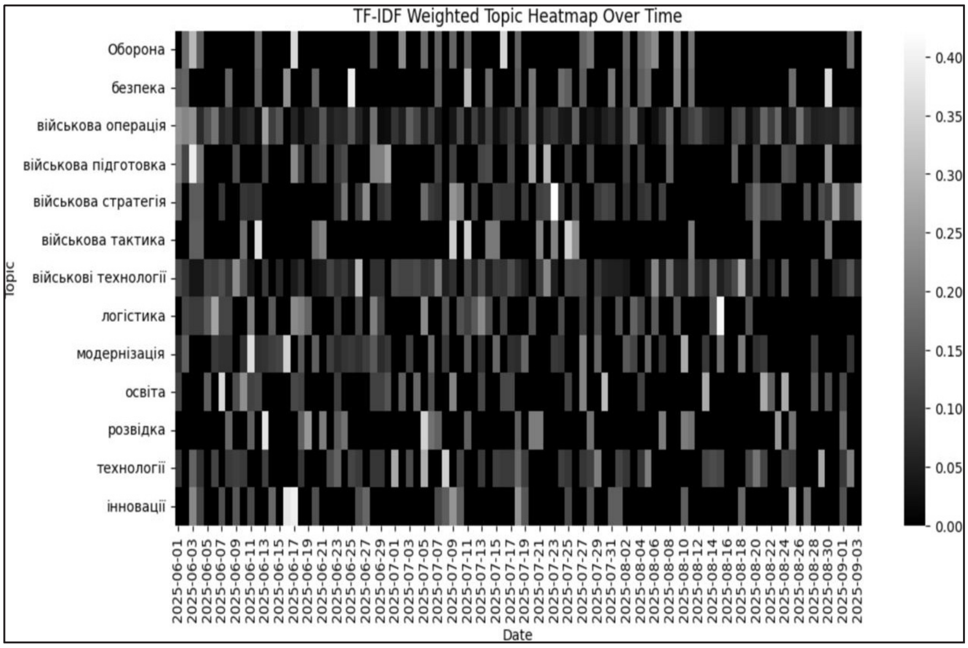

Fig. 7. TF-IDF for short-terms ontology

CONCLUSIONS

This study proposed and validated a robust approach for automated ontology construction based on large language models (LLMs), provided temporal analysis in different time frames, and it is applied to the domain of communication technologies. The approach was developed and tested on multilingual social media data collected from the “Victory Drones” Telegram channel [11] over the period from October 2022 to September 2025.

The dataset was gathered using asynchronous distributed parsing methods implemented with advanced Python libraries, ensuring efficient and reliable extraction of posts from large-scale Telegram data. After filtering irrelevant content like channel’s info or advertising, the final corpus provided a representative record of thematic domain. Temporal aggregation was performed at both daily and hourly resolutions, enabling the comparison of long-term and short-term semantic dynamics.

The extraction process relied on multiple LLM configurations to identify goal statements and their corresponding hypernyms from raw texts. A consensus mechanism ensured robustness by considering only those semantic pairs, which were consistently reproduced across several LLMs, minimizing hallucination risk.

The extracted semantic elements were embedded into a high-dimensional vector space, where similarity was computed using cosine distance. Clustering and hierarchical merging were performed iteratively, with the optimal number of clusters determined via optimization criterion. A key empirical finding is that convergence (the point at which further iterations cease to produce meaningful new clusters) occurred consistently after five iterations for both the daily (long-term) and hourly (short-term) ontologies. This stability suggests, that we have different temporal resolutions, buy the underlying semantic data have a highly similar structural organization. Across the two ontologies, 886 distinct thematic concepts, 262 concepts (29.6 %) formed the shared semantic core, appearing in both hierarchies. The daily (long-term) ontology contributed 405 unique topics, capturing slow-evolving, structural themes. On the other side, the hourly (short-term) ontology introduced 219 unique topics, reflecting rapid signals. This comparison reveals that some topics emerge briefly within short intervals and then fade, capturing real-time fluctuations in public attention. However, when observed over longer periods, certain topics demonstrate persistence, reappearing across multiple temporal windows and forming the backbone of the long-term semantic structure.

The temporal analysis component helped to map the static ontology into a dynamic tracking tool via the TF-IDF approach. Here are three key components: topics that drive fast on a short-term interval tend to fade away rapidly in prominence; some strategic topics, show strong, stable, or increasing prominence in the long-term analysis and some topics show stable, high-frequency occurrence across both the short-term and long-term frames.

The proposed approach demonstrates that LLM-driven ontology construction can effectively reproduce some analysis from domain experts, such as identifying goals, abstracting hypernyms, and structuring thematic relations. This makes the method highly cost-efficient and scalable.

In summary, the research opens, that the semantic ontologies received from LLM-based analysis can provide a stable and interpretable representation over time. The observed convergence behavior, structural similarities, and interpretable divergences between long-term and short-term perspectives validate the robustness of the proposed framework. It is foundation for future automated foresight systems.

REFERENCES

1. M. Zgurovsky, N. Pankratova, *System Analysis & Intelligent Computing, Theory and Applications*. Berlin: Springer, 2022, 432 p. doi: http://doi.org/10.1007/978-3-030-94910-5
2. A. Rosa, N. Gudowsky, P. Repo, “Sensemaking and lens-shaping: Identifying citizen contributions to foresight through comparative topic modelling,” *Futures*, vol. 129, pp. 1–15, 2021. doi: http://doi.org/10.1016/j.futures.2021.102733
3. C. Mühlroth, M. Grottke, “Artificial Intelligence in Innovation: How to Spot Emerging Trends and Technologies,” *IEEE Transactions on Engineering Management*, vol. 69, no. 2, pp. 493–510, April 2022. doi: 10.1109/TEM.2020.2989214
4. Y. Kishita, T. Kusaka, Y. Mizuno, Y. Umeda, “Toward theory development in futures and foresight by drawing on design theory: A commentary on Fergnani and Chermack 2021,” *Futures & Foresight Science*, vol. 3, issue 3-4, 2021, pp. 1–3. doi: https://doi.org/10.1002/FFO2.91
5. O. Matei, R. Erdei, D. Delinschi, “Multimodal transportation overview and optimization ontology for a greener future,” *Artificial Intelligence in Intelligent Systems: Proceedings of 10th Computer Science On-line Conference*, vol. 2, pp. 158–172. Springer 2021. doi: https://doi.org/10.1007/978-3-030-77445-5_15
6. Y. Chen, S. Sabri, A. Rajabifard, M. Agunbiade, “An ontology-based spatial data harmonisation for urban analytics,” *Computers, Environment and Urban Systems*, vol. 72, pp. 177–190. Elsevier, 2018. doi: https://doi.org/10.1016/j.compenvurbsys.2018.06.009
7. T. Brown et al., “Language Models are Few-Shot Learners,” *arXiv preprint*, 75 p., 2020. Available: https://arxiv.org/abs/2005.14165

8. J. Achiam et al., “Gpt-4 technical report,” *arXiv preprint*, 100 p., 2023. Available: https://arxiv.org/abs/2303.08774
9. Gemini Team Google: Rohan Anil et al., “Gemini: a family of highly capable multimodal models,” *arXiv preprint*, 90 p., 2025. Available: https://arxiv.org/abs/2312.11805
10. *xAI.Grok 3 beta - the age of reasoning agents*. Available: https://x.ai/news/grok-3/
11. *Victory Drones* [Telegram channel], 2022–2025. Available: https://t.me/VictoryDrones
12. Y. Chen, X. Pan, Y. Li, B. Ding, J. Zhou, “EE-LLM: Large-scale training and inference of early-exit large language models with 3D parallelism,” *arXiv preprint*, 27 p., 2024. Available: https://arxiv.org/abs/2312.04916
13. O. Michel, R. Bifulco, G. Retvari, S. Schmid, “The Programmable Data Plane: Abstractions, Architectures, Algorithms, and Applications,” *Proc. ACM Computing Surveys (CSUR)*, vol. 54, issue 4, pp. 1–36, 2021. doi: https://doi.org/10.1145/3447868
14. T. Wolf et al., “Transformers: State-of-the-Art Natural Language Processing,” *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, 2020. doi: https://doi.org/10.18653/v1/2020.emnlp-demos.6
15. P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, “Pretrain, prompt, and predict: A systematic survey of prompting methods in natural language processing,” *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–35, 2023. doi: https://doi.org/10.1145/3560815
16. E. Yu et al., “Merlin: Empowering Multimodal LLMs with Foresight Minds,” *arXiv preprint*, 28 p., 2023. doi: https://doi.org/10.48550/arXiv.2312.00589
17. N. Muennighoff, N. Tazi, L. Magne, N. Reimers, “MTEB: Massive Text Embedding Benchmark,” *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL), Dubrovnik, Croatia, 2023*, pp. 2014–2037. doi: https://doi.org/10.18653/v1/2023.eacl-main.148
18. A. Lucky, T. Kartik, B. Gaurav, M. Ankush, “Authorship Clustering using TF-IDF weighted Word-Embeddings,” *Proceedings of the 11th Annual Meeting of the Forum for Information Retrieval Evaluation (FIRE 19). Association for Computing Machinery, New York, NY, USA, 2019*, pp. 24–29. doi: https://doi.org/10.1145/3368567.3368572

Received 15.12.2025

INFORMATION ON THE ARTICLE

Serhii A. Lupenko, ORCID: 0000-0002-6559-0721, Opole University of Technology, Poland, e-mail: lupenko.san@gmail.com

Mykhailo V. Stoliar, ORCID: 0009-0009-3624-3147, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Ukraine, e-mail: misha.stolyar99@gamil.com

Oleksandr M. Terentiev, ORCID: 0000-0002-4288-1753, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Ukraine, e-mail: o.terentiev@gmail.com

Volodymyr V. Savastiyanov, ORCID: 0000-0002-2052-0420, Educational and Research Institute for Applied System Analysis of the National Technical University of Ukraine “Igor Sikorsky Kyiv Polytechnic Institute”, Ukraine, e-mail: vvs.in.ua@gmail.com

АВТОМАТИЗОВАНА КОНСТРУКЦІЯ СЕМАНТИЧНОЇ ОНТОЛОГІЇ ДЛЯ ДОСЛІДЖЕНЬ ПЕРЕДБАЧЕННЯ ІЗ ВИКОРИСТАННЯМ ВЕЛИКИХ ЛІНГВІСТИЧНИХ МОДЕЛЕЙ / С.А. Лупенко, М.В. Столяр, О.М. Терентьєв, В.В. Савастьянов

Анотація. Сучасні досягнення у сфері великих мовних моделей (LLM) дають змогу автоматизовано виявляти семантичні структури та нові сигнали, які наявні в потоках текстової інформації. Це дає змогу автоматизувати рутинні робочі процеси, які пов'язані із розробленням прогнозних моделей на основі систем безперервного аналізу даних. Мета дослідження – розроблення і валідація автоматизованої схеми для вилучення, структурування та порівняння семантичних онтологій за допомогою LLM. Для аналізу даних із різноманітних платформ соціальних мереж використано паралізацію процесів. Дані спочатку відфільтровано, а саме: вилучено ті, що не належать до предметної досліджуваної галузі. Ключові семантичні елементи, цілі та гіпероніми, що відповідають предметній галузі, вилучено за допомогою кількох конфігурацій LLM із механізмом консенсусу для забезпечення семантичної надійності та мінімізації галюцинацій та вигадувань фактів зі сторони LLM. Вилучені елементи представлено у багатовимірному векторному просторі, ітеративно кластеризовано за допомогою метрики косинусної подібності та ієрархічно об'єднано. Процес конвергенції та структурну стабільність проаналізовано за допомогою критерію ліктя та метрик подібності. Запропонований підхід – економічно ефективна альтернатива традиційному експертному аналізу прогнозування. Об'єднуючи воєдино семантичне вилучення, кероване LLM із кількісною кластеризацією, цей метод дозволяє ідентифікувати нові тенденції, слабкі сигнали та довгострокові тематичні структури. Отримано результати дослідження, які підкреслюють великий потенціал семантичного моделювання на основі LLM як основи для автоматизованих систем прогнозування.

Ключові слова: передбачення, великі мовні моделі, семантична онтологія, сценарний аналіз, слабкі сигнали, ієрархічна кластеризація.