Graph Model of a Mental War

Dmytro Lande[1,2,*,†], Anatolii Kachynskyi[1,†]

[1] National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Prospect Beresteiskyi 37 03056 Kyiv, Ukraine

[2] Institute for Information Recording of National Academy of Sciences of Ukraine, Mykoly Shpaka St 2 03037, Kyiv, Ukraine

Abstract

To enable a deeper understanding of the complex phenomenon of "mental wars," a modeling methodology has been developed that integrates the capabilities of modern large language models (LLMs) and graph theory. The primary objective of the study is to transform the existing model by enhancing its depth and detail, thereby revealing the essence, mechanisms, strategic approaches, and consequences of mental wars within the context of hybrid warfare. To achieve this goal, a comprehensive analytical toolkit was applied, including semantic network analysis, modularity-based clustering, and node ranking within graphs. The use of generative artificial intelligence adds particular value, as it not only automatically generates new concepts but also uncovers logical relationships among them – significantly deepening the analysis and enabling a holistic understanding of the architecture of mental warfare. The outcome is an expanded dynamic model – a network of interrelated elements covering the key dimensions of mental wars: their strategic objectives, instruments of influence, key actors, implementation mechanisms, and anticipated consequences. This model serves as an effective analytical tool for investigating information-psychological operations, predicting their impact, and developing efficient countermeasures in the domain of information security.

Keywords

Mental war, hierarchical model, AI, LLM, clustering, visualization, information security

1. Introduction

Mental wars constitute a multidimensional phenomenon affecting multiple domains of social life, such as politics, economics, and culture. In the context of Russia's protracted conflict with Ukraine, there is a growing need to examine both conventional forms of legitimate propaganda and emerging technological instruments. This study aims to extend and refine the graph-based model of mental warfare by incorporating current characteristics of hybrid warfare and leveraging artificial intelligence – specifically, large language models (LLMs).

Recent developments in generative AI [1, 2] have contributed to advances in cognitive and information theory, enabling new approaches to the analysis and representation of complex systems. The use of AI allows not only refining existing models but also identifying new aspects that previously remained outside the focus of researchers [3, 4].

Initially, the graph model of mental war is considered as a complex network with a hierarchical structure [5]. This model serves as its mathematical abstraction: levels and elements define the essence of the problem.

Let $(H, \leq)$ be a finite partially ordered set with a greatest element b. The set H is called a hierarchy if the following conditions are satisfied:

There exists a partition of the set H into layers (levels of hierarchy):

ITS-2024: Information Technologies and Security, December 19, 2024, Kyiv, Ukraine

* Corresponding author.

† These authors contributed equally.

akachynsky@gmail.com (A. Kachynskyi); dwlande@gmail.com (D. Lande)

0000-0001-9642-7006 (A. Kachynskyi); 0000-0003-3945-1178 (D. Lande)

$$H = L_1 \cup L_2 \cup \ldots \cup L_h, \tag{1}$$

where:

- $L_1 = \{b\}$ – the first level of the hierarchy contains only the greatest element;
- $L_k \cap L_j = \varnothing$ for all $k \neq j$ – these subsets are disjoint.

Each element of *H* belongs to exactly one of the L_k.

These sets L_k are called layers or levels of the hierarchy.

Compatibility condition with the order:

If $x < y$ and $y \in L_k$, then $x \in L_m$, where $m > k$.

In other words, if one element is smaller than another, it must be located at a lower level of the hierarchy.

Sequentiality condition (descending/ascending transition):

For each element $x \in H$:

If x^{-} is a predecessor of x (i.e., $x^{-} < x$, and there is no element z such that $x^{-} < z < x$), then $x^{-} \in L_{k+1}$, if $x \in L_k$.

Similarly, if x^{+} is a successor of x (i.e., $x < x^{+}$, and there is no element *z* for which $x < z < x^{+}$), then $x^{+} \in L_{k-1}$, if $x \in L_k$.

That is: every predecessor lies at the next level, and every successor lies at the previous level.

Under such consideration, the problem can be decomposed into simpler components, after which the relative degree of interaction between the elements of this hierarchical structure can be evaluated.

2. Hierarchical Structure of the Mental War Model

In the traditional view of mental war as a hierarchical structure, the content of the levels serves the following purposes:

- At the first level (L_1 – Goals of the mental war), a single element – the focus – is considered and placed at the top of the hierarchy (war for changing identity).
- At the second level (L_2 – Forces and means of the mental war), economic, political, and social forces influencing the outcome are represented (finance, literature, art in general, mass media, the Internet, etc.).
- The third level (L_3 – Actors of the mental war) consists of actors who manipulate these forces (government, artists, patrons, etc.).
- The fourth level (L_4 – Goals of actors) represents the goals of each actor (changing perceptions, values, attitudes, stereotypes, traditions, archetypes of national consciousness).
- The fifth level (L_5 – Policies implemented by actors) describes possible scenarios or outcomes that each actor aims to achieve through their policies. Primarily, this involves the recoding not only of the civilization identity of the state but also of the cultural values of society and the individual.

Hierarchical decomposition allows structuring the system into subsystems, where each subsystem is responsible for specific goals. For each level, concepts are defined that form clusters interacting with each other to achieve the overall objective. The mathematical model helps formalize interactions between subsystems through graphs, adjacency matrices, and objective functions that describe the overall system's effectiveness.

Let us now consider the goals and subgoals at the levels of the considered hierarchical structure.

Levels L_1 (main goal) and L_4 (goals of individual actors) contain sets of goals and their sub-goals. Assume that at each level i, there exists a set of goals:

$$T_i = \left\{ T_{i1}, T_{i2}, \ldots, T_{im_i} \right\}, \tag{2}$$

where m_i – the number of goals at level i.

The index $j \in \{1,4\}$ indicates that only levels L_1 and L_4 are considered.

Each goal T_{ij} has its own set of sub-goals:

$$F_{ij} = \left\{ F_{ij1}, F_{ij2}, \ldots, F_{ijmn_{ij}} \right\}, \tag{3}$$

where n_{ij} – the number of sub-goals for the goal T_{ij}.

For example, T_{ij} is the main strategic goal of the first level. In this context, it may mean: "The final and irreversible dissolution of Ukrainian identity within the so-called 'All-Russian' identity, as well as the renunciation by Ukrainians as a political nation of independent statehood under conditions of losing conscious sense of national self-identity."

On the other hand, the goals T_{i4}, which correspond to the actions of specific subjects (actors), are subordinate to this overarching strategic goal, but may change depending on the historical period – primarily in terms of implementation methods or means of achievement, rather than in their essence.

Elements and sub-goals at all levels (including F_{ijk}) perform certain functions: informational, economic, organizational, etc. They can be interconnected – both within the same level and across different levels of the hierarchy, forming a network of interactions.

These connections can be formally described by a set of pairs:

$$Links = \left\{ \left(F_{ijk}, F_{i'j'k'} \right) \mid F_{ijk} \quad linked \quad with \quad F_{i'j'k'} \right\}. \tag{4}$$

Such a system forms a directed graph G, where:

- the nodes are the elements, sub-goals, and goals (F_{ijk}),
- the edges are the connections between them.

Let us denote this graph as [6] G=(T, E), where V is the set of vertices (all elements, sub-goals, goals), $E \subseteq T \times T$ is the set of directed edges describing functional dependencies.

For each system element F_{ijk}, one can define a function $f\left(F_{ijk}\right)$, which characterizes its role within the system. These functions can be represented as rules or formulas that describe how the element interacts with others.

To model the process of achieving goals at different levels, we can introduce a target function of the "mental war" system, denoted by Φ. This function depends on the success of achieving the main goals T_{ij} and their sub-goals F_{ijk}:

$$\Phi = \alpha \cdot f\left(T_{ij}\right) + \left(1 - \alpha\right) \cdot f\left(F_{ijk}\right), \tag{5}$$

where:

- $\alpha \in [0,1]$ is a weight coefficient that determines the importance of achieving the overall goal compared to the sub-goals;
- $f(T_{ij})$ is a function defining the effectiveness of achieving goal T_{ij};
- $f(F_{ijk})$ is a function defining the contribution of sub-goal F_{ijk} to achieving the overall objective.

This description allows for a formal modeling of the complex structure of mental wars as a hierarchical-network system with clearly defined connections, functions, and goal-achievement mechanisms.

3. Expansion of the Hierarchical Model

Within the hierarchical structure of mental war, both sub-goals and individual concepts (concepts) can be defined not only by human experts but also by virtual experts − artificial intelligence systems that assist in forming and improving the model of goals and related concepts.

When certain concepts simultaneously belong to multiple different levels of the hierarchy, the logical connections between these levels become more complex. As a result, the structure ceases to be purely hierarchical and transitions into a network form, where the relationships between elements become more flexible and multidimensional.

Such a network organization allows for more effective modeling of the goal-achievement process, as it provides more pathways from initial goals to final outcomes through diverse interconnections.

Let us formalize this idea for further model expansion using LLM. As before, let $T_i = \{T_1, T_2, ..., T_n\}$ be the set of levels of the primary hierarchy, and $F = \{F_1, F_2, ..., F_k\}$ be the set of concepts. Experts, including virtual ones, define these concepts based on the analysis of the levels of the primary hierarchical model of mental war. Concepts can belong to multiple levels simultaneously (including the goals defined above), which makes the connections between the model's levels more complex. If a concept f_i belongs to several levels at once, it introduces new links between those levels. Now, the system's structure transitions from a purely hierarchical one to a network-based model.

For formalizing this transformation, we use a graph *G*=(*T*, *E*), where *T* is the set of nodes representing hierarchy levels, and *E* is the set of edges, where each edge between two levels indicates the presence of shared concepts between them.

If a concept f_i belongs to levels T_A and T_B, then there is a connection between them:

$$E = \left\{ (T_a, T_b) \mid f_i \in F, f_i \in T_a \cap T_b \right\}. \tag{6}$$

The weight of each edge (T_a, T_b) is determined by the number of shared concepts between these levels: $w(T_a, T_b) = |T_a \cap T_b|$.

Thus, the more shared concepts correspond to different levels, the stronger the connection between them.

Using the network structure allows for more efficient achievement of final outcomes. The shortest paths in the network between levels (in particular, between the primary goal and the consequences) allow for reducing time and resources needed to achieve these outcomes. This can be expressed through a path minimization function in the graph:

$$\tau(T_i) = \min_{P \subseteq G} \sum_{(T_a, T_b) \in P} w(T_a, T_b), \tag{7}$$

where P is a path from the initial goal to the required outcome through other levels/concepts.

Virtual experts can assist not only in defining concepts but also in dynamically updating the network. This means that experts can add new links or modify existing ones depending on the context of the mental war. Formally, this is described through a dynamic transformation of the model's graph:

$$G' = \left\{ G \cup (T_i, T_j) \mid f_i \in F, f_i \in T_i \cap T_j, new\ links \right\}. \tag{8}$$

Such transformation of the hierarchy into a network allows for greater flexibility and faster adaptation to changes, as well as more efficient use of available resources to achieve desired outcomes [7]. Links between levels of the original hierarchy, established through shared concepts, become the basis for selecting the most effective path from the goal to the consequences. This approach enables the use of cross-level shared concepts, optimizes processes through virtual experts, and dynamically adapts to new conditions.

The above formalization corresponds to a sequence of actions that allow for the repetition of the process under expert supervision until a complete understanding of the domain state is achieved:

1. Presentation of the initial scheme, which includes the creation of an initial scheme in CSV format, where basic semantic connections between concepts are represented.
2. Development of prompts for LLMs – that is, creating prompts for large language models to generate new concepts and connections.
3. Integration of new connections into the initial scheme.
4. Linguistic processing of data for evaluation of new and existing connections, as well as ranking of nodes.
5. Analysis and visualization of data – specifically, loading the data into a graph analysis system (Gephi), performing clustering based on modularity classes.
6. Formation and refinement of clusters, determining their names using LLMs, checking consistency, and removing unnecessary elements.
7. Final validation and verification of the extended model, ensuring its coherence and correctness.

Thus, the following steps are proposed for expanding the primary hierarchical model based on the application of large language models.

The initial scheme of "mental wars" is provided in CSV format, where each line represents a semantic connection between primary levels in the format "Concept 1; Concept 2", for example:

- Goals of the mental war; Forces and means of the mental war
- Forces and means of the mental war; Actors of the mental war
- Actors of the mental war; Goals of actors
- Goals of actors; Policies implemented by actors
- Policies implemented by actors; Results of the mental war

4. Prompt Generation for Creating New Concepts

The essence of the approach for generating formal probing prompts for LLM lies in representing prompts as analogs of software constructs (conditional statements, loops, functions) through mathematical formalization of their logic and interactions. The following main primitives ("building blocks") are used for prompt generation: "Condition", "Loop", and "Function", along with methods for composing these primitives to build complex systems, particularly semantic networks.

We will now describe the framework upon which prompts will be subsequently generated by large language models. These prompts will later be applied for scanning LLMs, executed in turn within the environment of large language models [8].

Each primitive in the no-code prompt engineering framework is defined by clear rules for transforming input data into output. These primitives serve as "building blocks" for constructing logical structures analogous to programming language operators, but based on natural language.

Primitive "Condition" (If-Else)

Let:

- *Input* – input data (e.g., text, numerical parameter, etc.);

- *C – a condition (predicate) that returns the value True or False;*
- A_1, A_2 – two possible actions that are executed in cases when the condition is True or False, respectively.

The prompt function P is defined as:

$$P(Input)=\begin{cases} A_1(Input), & if \quad C(Input)=True; \\ A_2(Input), & if \quad C(Input)=False. \end{cases} \tag{9}$$

This formal definition is analogous to the classical if-else operator in programming languages, but it is applied through textual instructions.

The mechanism of the "Condition" primitive consists in passing to the LLM a task with two possible scenarios, each of which is executed depending on the result of the condition check. For example:

"If the text contains the term 'cybersecurity', return its definition; otherwise, return a list of related terms."

In this example:

- *C* – the presence of the word "cybersecurity" in the text;
- A_1 – generation of the definition of the term "cybersecurity";
- A_2 – search for associations related to the term "cybersecurity";

Primitive "Loop" (For-Loop)

Let:

- $S=\{s_1,s_2,...,s_n\}$ – a set of elements to be processed;
- *F* – an operation applied to each element of the set.

The prompt function *P*(*S*) is defined as the union of results of applying the operation *F* to each element of the set:

$$P(Input)=\begin{cases} A_1(Input), & if \quad C(Input)=True; \\ A_2(Input), & if \quad C(Input)=False. \end{cases} \tag{9}$$

This formal definition is analogous to a classic for loop in programming languages, where the same operation is repeated for all elements of a set.

The working mechanism of the "Loop" primitive is to pass a list of elements together with an operation to be applied to each element to the LLM. The result is the union of all obtained responses. Consider the following example:

Prompt: "For each term in the list ['phishing', 'firewall'], find 3 usage examples." Here:

- *S* = {"phishing","firewall"} – a set of terms;
- *F* – the operation of finding usage examples for a given term.

The result will be the union of usage examples for both terms.

Primitive "Function" (Abstraction)

Let:

- *F*: *X* ⟶ *Y* – a function that transforms elements from set X into elements of set Y;
- *x* – an input element;

- *parameter* – a set of parameters that control the function's behavior.

The prompt function $F_{extract}(x,\ parameter)$ is implemented through an instruction passed to the model:

$$F_{extract}(x,\ parameter) = Prompt(x,\ instruction\ with\ parameter). \tag{11}$$

This formal definition is analogous to abstraction in programming languages, where a function can accept parameters to adjust its internal logic.

The mechanism of the "Function" primitive is to pass a specific task with parameters to the LLM, where the parameters define the details of how the task should be executed.

Primitive "Label" (Tagging / Marking)

Let:

- *Input* – input data or a previously defined part of the prompt logic;
- *Label*(*Name*) – a tag or identifier used to mark a specific section or stage in the prompt processing flow.

The prompt function Label is defined as:

$$Label(Name){:}\ Input \longrightarrow Annotated\ Section. \tag{11}$$

This formal definition is analogous to labeling code blocks or sections in programming languages for reference, navigation, or control flow purposes.

The "Label" primitive assigns a unique identifier to a specific part of the prompt structure, enabling logical organization and potential referencing in complex prompt workflows. It does not perform any action itself but serves as a structural marker.

This mechanism allows for better readability, modular design, and controlled execution flow when working with multi-stage prompts in LLM environments.

Primitive "Go To" (Jump / Navigation)

Let:

- Label(Name) – a predefined label or section identifier;
- GoTo(Name) – an instruction to transfer control or continue execution from the labeled section.

The prompt function GoTo is defined as:

$$GoTo(Name) \longrightarrow Execution\ resumes\ at\ Label(Name). \tag{12}$$

This formal definition is analogous to the goto statement in traditional programming, allowing non-linear control flow by redirecting execution to a labeled section.

The "GoTo" primitive enables conditional or unconditional redirection within the prompt processing pipeline. It can be used to repeat certain stages, skip irrelevant ones, or dynamically change the workflow based on intermediate results.

These primitives – Label and GoTo – together enable structured navigation and modularity in complex prompt engineering workflows, enhancing the ability to manage logic flow and improve prompt reusability in large language model environments.

General Principles of Primitive Selection

Each primitive must be a precisely defined construct to ensure unambiguous understanding by the model.

Primitives can be parameterized, allowing them to be adapted to different tasks.

Primitives can be combined to create complex systems, similar to how programmers write code using basic constructs.

The three primitives – "Condition", "Loop", "Go to" and "Function" – form the foundation for no-code system creation through prompt engineering, enabling natural language to be used as a tool for controlling the logic of AI systems.

Such systems represent an orchestration of primitives, analogous to computer programs. The syntax of prompts forms a language with a formal grammar that can be represented in the form of an Abstract Syntax Tree (AST) :

$$Prompt ::= Primitive \mid (Prompt \oplus Prompt) \mid Condition(Prompt, Prompt). \tag{13}$$

where $\oplus$ denotes composition.

To generate a formal prompt for expanding all levels of the initial model while considering mental wars, the following initial task-prompt is formulated in natural language (see Appendix A).

After processing this prompt to generate a formal prompt for accomplishing the given task using an LLM, the resulting structured prompt is provided in Appendix B.

The structure of the structured prompt is illustrated in Fig. 1.

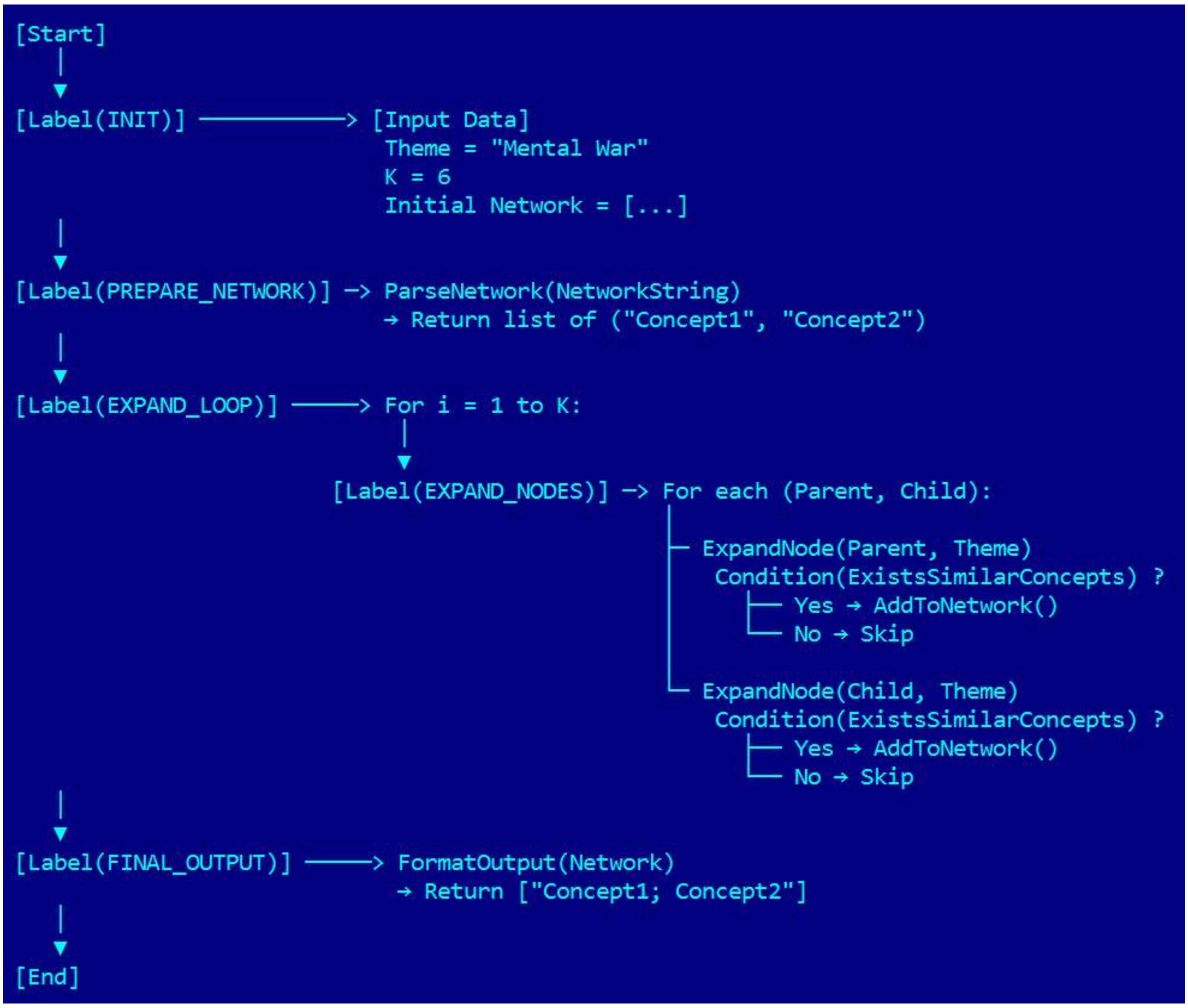

Figure 1: Flowchart of the structured prompt pipeline

The generated prompt uses all the formal primitives of the proposed prompt engineering language as follows:

STAGE	PRIMITIVES INVOLVED
Initialization	Label(INIT),Input
Parsing	Label(PREPARE_NETWORK), Function(ParseNetwork)
Expansion Loop	Label(EXPAND_LOOP), Loop(i=1 to K)
Node Expansion	Label(EXPAND_PARENT), Label(EXPAND_CHILD) Condition(ExistsSimilarConcepts) A1(AddToNetwork), A2(NoAction)
Final Output	Label(FINAL_OUTPUT), Function(FormatOutput)

According to the responses from LLM systems, additional files in CSV format are generated. For the example provided, the file will look like this:

```
Goals of the mental war; Change of national identity
Goals of the mental war; Forces and means of the mental war
Forces and means of the mental war; Literature
Forces and means of the mental war; Art
Forces and means of the mental war; Mass Media
Forces and means of the mental war; Social Media
Forces and means of the mental war; Actors of the mental war
Actors of the mental war; Artists
Actors of the mental war; Government
Actors of the mental war; Painters
Actors of the mental war; Goals of individual actors of the mental war
Goals of individual actors of the mental war; Changing perceptions
Goals of individual actors of the mental war; Changing values
Goals of individual actors of the mental war; Changing attitudes
Goals of individual actors of the mental war; Changing national consciousness
…
```

5. Integration of Obtained Responses with the Initial Schema

The newly identified connections are integrated into the initial schema. The network consists of semantically related concepts and is not causal in nature; consistency checks of the new links can be carried out using the following approaches: the detection approach is implemented automatically – if a new concept or link already exists in the network, the weight of that node or link is increased.

At this stage, the new links obtained from the LLM are merged with the original schema. To account for the significance of the links, a formula for iterative adjustment of the corresponding link weights is used:

$$S_{new} = \alpha \cdot S_{old} + \beta \cdot S_{new}, \tag{14}$$

where:

- S_{new} – the new weight coefficient of the link;
- S_{old} – the weight of the existing link;
- α – the weighting coefficient for old links;
- β – the weighting coefficient for existing links.

The old link S_{old} represents the weight of the existing connection between concepts in the semantic network. The new link S_{new} is the weight coefficient calculated for the new links added based on new data from the LLM.

The coefficients α and β allow control over the influence of each type of link on the overall network.

The formula enables the calculation of a new weight coefficient as a combination of the weight values of both old and new links, where the coefficients α and β help adjust the balance between existing and new data.

Within this approach, conflict checking is performed by involving a human expert who verifies whether the new links introduce contradictions within the context of already existing ones, by assessing similarity or differences in the relationships between concepts.

6. Linguistic Data Processing

Linguistic expansion enabled the connection of additional pairs that were previously undetected in the main chains. Linguistic processing of data and node ranking allows for the combination of concepts that are synonyms, derivations of one another, etc. For this purpose, a specialized software tool embedded in the graph analysis and visualization service CSV2Graph is used [9].

All pairs now include:

- semantically related concepts (e.g., "Confidentiality" and "Integrity");
- lexically similar concepts (e.g., "Protection" and "Data Protection");
- new connections emerging through context (e.g., "System" and "Component").

This completes the construction of the full list of pairs, taking into account linguistic expansion. The structure is now ready for further analysis or integration into the network.

Here is an example of new connections obtained after linguistic processing:

```
Morale; Moral relativism
Morale;Erode morale
Morale;Undermining morale
Morale;Psychological demoralization
Morale;Decreased morale
Morale;Moral decline
Morale;Demoralizing the population
Morale;Demoralization
```

7. Data Analysis and Visualization in Gephi

Node prioritization across hierarchical levels − including goal-oriented nodes and conceptual entities − is carried out using a concept-ranking methodology grounded in the PageRank and TextRank algorithms [10, 11]. The ranking score for a given node A is computed as:

$$PR(A) = \frac{1-d}{N} + d \sum_{i \in M(A)} \frac{PR(i)}{L(i)}, \tag{15}$$

where:

- *PR(A) − PageRank of node A;*
- d – damping coefficient;
- N – total number of nodes in the graph;
- $M(A)$ − set of nodes linking to A;
- $L(i)$ − number of outgoing links from node i.

The first term of the formula, $\frac{1-d}{N}$, ensures a baseline PageRank level for all nodes. The numerator corresponds to the probability of a random state when a user randomly selects a node in the graph, and N is the total number of nodes.

The second term, $d\sum_{i\in M(A)}\frac{PR(i)}{L(i)}$ is the part of the formula used to calculate the PageRank of node A based on the *PageRanks* of nodes pointing to it. For each node i in the set $M(A)$, its *PageRank* is divided by the number of outgoing links $L(i)$, and this value is summed across all nodes pointing to A.

This approach enables the identification of structurally and semantically central concepts within the network by iteratively evaluating their connectivity and contextual relevance.

The combined data are loaded into the Gephi software environment [12] for clustering based on modularity classes. There are various types of modularity measures [13] that can be applied in Gephi; the authors employed the Potts model [14], which accounts for so-called resolution limits. Based on this model, the required number of concept classes (clusters) is automatically determined.

8. Creation of the Final Model

At this stage, the extended model is verified to ensure its consistency and correctness within the context of both new and existing concepts. Irrelevant elements are removed, and the final model is represented in the form of a graph. The final model illustrates an expanded network of concepts and their interconnections within the context of "mental wars" (Fig. 2), visualized using Gephi [12] to enhance clarity, reveal structural patterns, and facilitate dynamic exploration of relationships between nodes.

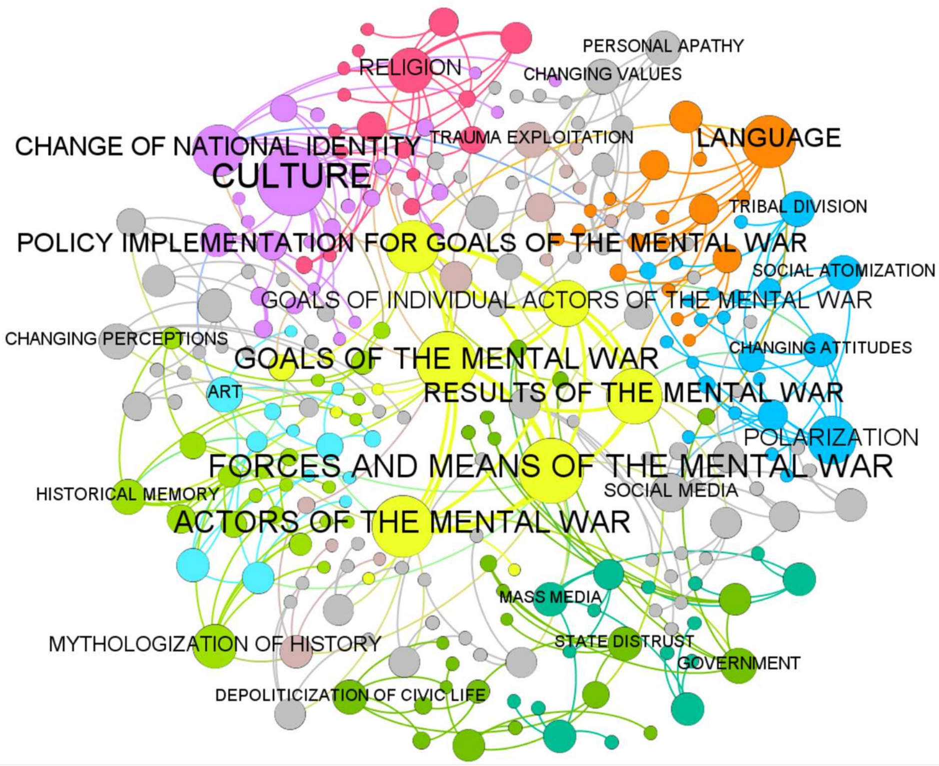

Figure 2: Network corresponding to the extended model of mental war

9. Results

The extended model of mental wars demonstrates a complex network of interrelations between core and additional concepts. Clustering and node ranking allow for the identification of key connections and concepts that influence the understanding of ethical aspects in cybersecurity.

Among more than 300 nodes in the extended network, the following important concepts were identified as essential for understanding the nature of information-psychological mental wars:

- Culture;
- Language;
- Change of national identity;
- Polarization;
- Religion;
- Mythologization of history;
- Social media;
- Government.

After analysis and clustering, several additional concepts that were not accounted for in previous models were identified. For instance, the concepts of "information asymmetry" and "cognitive traps" proved to be key to understanding mental wars within this network model.

Node ranking using the PageRank algorithm allowed for the identification of the most influential concepts in the model. This helped determine which elements are most significant for further study (see Table 1) and refinement (excluding the initially defined ones, which naturally turned out to be the centroids of the clusters).

Table 1
Top-10 Core Concepts

Concept	PageRank
Culture	0.01723
Forces and means of the mental war	0.01608
Actors of the mental war	0.01511
Goals of the mental war	0.01474
Results of the mental war	0.01349
Policy implementation for goals of the mental war	0.01271
Language	0.01268
Change of national identity	0.01235
Religion	0.01062
Mythologization of history	0.01027

The semantic network was divided into several clusters, each representing a distinct aspect of mental wars. The modularity algorithm allowed for identifying significant connections between nodes and determining the key components of the network.

As can be seen, as a result of its development, the MW network transforms into a self-organized complex system. The MW network exhibits general statistical properties that influence its functional purpose. It is an example of systems characterized by typical features: growth dynamics, optimization, and embedding into a two-dimensional space.

An important characteristic of complex networks, including the extended mental war network, is the degree distribution – the probability that a node has x adjacent edges. For the MW network, this distribution follows a power-law form:

$$p(x) = Cx^{-\alpha}, \quad x \geq x_{\min}, \tag{16}$$

where:

- $\alpha \approx 2.24$ is the estimated exponent of the power-law distribution;
- $x_{min} = 1$ is the minimum value at which the power-law behavior begins;
- C is the normalization constant.

To test the accuracy of the power-law model, a statistical measure – the K-S test (Kolmogorov-Smirnov test) [15] – was used. The Kolmogorov-Smirnov test shows a high level of agreement between the empirical and theoretical distributions (K-S = 0.04), confirming the correctness of the approximation.

The analysis of the statistical properties of the MW network confirmed the existence of nodes with high degrees, which are practically absent in networks governed by Poisson or exponential degree distributions. It is precisely the presence of such nodes that explains the observed behavioral patterns and gives rise to many unique characteristics of scale-free networks.

10.Conclusions

As a result of the conducted research, a methodology for expanding the graph model of mental wars using tools of generative artificial intelligence, particularly large language models, was developed and implemented. This made it possible not only to refine and supplement the existing hierarchical structure but also to transition to a network-based model that ensures flexibility, adaptability, and in-depth analysis of interconnections between key concepts.

The use of a "swarm of virtual experts" [3] proved effective in generating new concepts and connections. Each agent contributes uniquely, enabling a multidimensional picture of mental wars. An important role in the modeling process was played by the proposed no-code programming framework for prompt engineering. It is based on three main primitives – "Condition", "Loop", "Go to", and "Function" – which allow formalizing the logic of interaction with LLM, ensuring systematicity, repeatability, and control over the generation process of new concepts and relationships. The framework implements a Map–Reduce approach: first, the logic of the prompt is decomposed in detail through formal constructs, and then a compact textual query is formed that is effectively perceived by the model. This approach significantly improves the quality of results, reduces the number of errors, and simplifies verification.

Applying formalized approaches to constructing prompts for LLM enabled systematic generation of new concepts, establishing logical links between them, and integrating the resulting data into a unified semantic network. A crucial stage involved the use of clustering algorithms by modularity classes and node ranking via PageRank, which facilitated the identification of dominant concepts and the formation of a structured model of mental warfare.

Among the most important concepts identified during the study are: "culture", "language", "change of national identity", "social media", "religion", "societal polarization", "mythologization of history", and others. These concepts have significant influence on shaping strategies and consequences of mental wars.

Furthermore, the study revealed new aspects not considered in previous models – notably, the concepts of "information asymmetry" and "cognitive traps", which substantially explain the mechanisms of psychological influence in hybrid warfare conditions.

Analysis of statistical characteristics of the obtained network confirmed its scale-free nature typical of complex systems, and indicated the presence of highly connected nodes that play a key role in spreading influence within mental warfare.

Thus, the proposed model can be used as a tool for analyzing information-psychological influences, predicting their consequences, and developing counter-strategies. It is particularly worth noting the importance of implementing the no-code programming framework to systematize work with LLM, which ensures scalability, transparency, and high quality of results obtained.

Future research will focus on improving the model through integration of additional natural language processing methods, involving a larger number of virtual experts, and implementing causal relationship analysis technologies to deepen understanding of the phenomenon of mental wars.

11.Appendices

Appendix A: Initial Task-Prompt Formulation in Natural Language

We have a hierarchical network structure defined as a set of adjacent node-pairs representing concepts in a given thematic domain. For example: "Concept 1; Concept 2", "Concept 1; Concept 3", "Concept 2; Concept 4", "Concept 4; Concept 5", "Concept 4; Concept 6".

Using an LLM (Large Language Model), this network is expanded. For each node, conceptually similar terms are selected using the LLM and linked to it in the extended network. This process is applied to all existing nodes.

After the first iteration step, the expansion procedure is reapplied to the newly extended network. This is repeated K times.

At the end, the final extended network is returned as a list of adjacent node-pairs in the format:
"Concept 1; Concept 2".

Input Data:
K = 6
Theme: "Mental War"
Initial Network:
Goals of the mental war; Change of national identity
Goals of the mental war; Forces and means of the mental war
Forces and means of the mental war; Literature
Forces and means of the mental war; Art
Forces and means of the mental war; Mass Media
Forces and means of the mental war; Social Media
Forces and means of the mental war; Actors of the mental war
Actors of the mental war; Artists
Actors of the mental war; Government
Actors of the mental war; Painters
Actors of the mental war; Goals of individual actors of the mental war
Goals of individual actors of the mental war; Changing perceptions
Goals of individual actors of the mental war; Changing values
Goals of individual actors of the mental war; Changing attitudes
Goals of individual actors of the mental war; Changing national consciousness
Goals of individual actors of the mental war; Policy implementation for goals of the mental war
Policy implementation for goals of the mental war: Language
Policy implementation for goals of the mental war: Religion
Policy implementation for goals of the mental war: Culture
Policy implementation for goals of the mental war: Historical Memory
Policy implementation for goals of the mental war; Results of the mental war
Results of the mental war; Recoding of Ukrainians' views on the state
Results of the mental war; Recoding of Ukrainians' views on society
Results of the mental war; Recoding of Ukrainians' views on the individual

Appendix B: Formal Structured Prompt Derived from Initial Task Description

Label(INIT):

```
Input:
– Theme = "Mental War"
– K = 6
–Initial Network = [
   "Goals of the mental war; Change of national identity",
   "Goals of the mental war; Forces and means of the mental war",
   "Forces and means of the mental war; Literature",
   "Forces and means of the mental war; Art",
   "Forces and means of the mental war; Mass Media",
   "Forces and means of the mental war; Social Media",
   "Forces and means of the mental war; Actors of the mental war",
   "Actors of the mental war; Artists",
   "Actors of the mental war; Government",
   "Actors of the mental war; Painters",
   "Actors of the mental war; Goals of individual actors of the mental war",
   "Goals of individual actors of the mental war; Changing perceptions",
   "Goals of individual actors of the mental war; Changing values",
   "Goals of individual actors of the mental war; Changing attitudes",
   "Goals of individual actors of the mental war; Changing national consciousness",
   "Goals of individual actors of the mental war; Policy implementation for goals of the mental war",
   "Policy implementation for goals of the mental war; Language",
   "Policy implementation for goals of the mental war; Religion",
   "Policy implementation for goals of the mental war; Culture",
   "Policy implementation for goals of the mental war; Historical Memory",
   "Policy implementation for goals of the mental war; Results of the mental war",
   "Results of the mental war; Recoding of Ukrainians' views on the state",
   "Results of the mental war; Recoding of Ukrainians' views on society",
   "Results of the mental war; Recoding of Ukrainians' views on the individual"
]
Label(PARSE_NETWORK):
Function(ParseNetwork, NetworkString):
Return list of concept pairs ("Concept1", "Concept2")
Label(EXPAND_NETWORK_LOOP):
Loop over i from 1 to K:
   Label(EXPAND_NODES):
   Loop over each pair (Parent, Child) in Network:

      Label(EXPAND_PARENT):
      Function(ExpandNode, Parent, Theme):
      Condition(ExistsSimilarConcepts(Parent)):
         A1: ForEach SimilarConcept in GetSimilarConceptsLLM(Parent, Theme):
            AddToNetwork(Parent, SimilarConcept)
         A2: NoAction
      Label(EXPAND_CHILD):
      Function(ExpandNode, Child, Theme):
      Condition(ExistsSimilarConcepts(Child)):
         A1: ForEach SimilarConcept in GetSimilarConceptsLLM(Child, Theme):
            AddToNetwork(Child, SimilarConcept)
         A2: NoAction
Label(FINAL_OUTPUT):
```

```
Function(FormatOutput, Network):
    Return list of strings in format "Concept1; Concept2"
```

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT and Qwen to: translate certain text fragments from their native language, perform grammar and spelling checks, and paraphrase or reword content. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

[1] Lo, J.H., Huang, H.P. and Lo, J.S., 2025. LLM-based robot personality simulation and cognitive system. Scientific Reports, 15(1), p.16993. DOI: 10.1038/s41598-025-01528-8

[2] Srinivasan Ramanujam. The LLM Revolution: Transforming Industries with Large Language Models. Kindle Edition, 2024. https://www.amazon.com/LLM-Revolution-Transforming-Industries-Language-ebook/dp/B0D7VS5BNK

[3] Abu-Salih, B., Alotaibi, S., Alanazi, A.L., Abukhurma, R., Al-Shboul, B., Khouri, A. and Aljaafari, M., 2025. Using Large Language Models for Semantic Interoperability: A Systematic Literature Review. ICT Express. DOI: 10.1016/j.icte.2025.06.011

[4] Shu, Z., Sun, X. and Cheng, H., 2024, October. When llm meets hypergraph: A sociological analysis on personality via online social networks. In Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. pp. 2087-2096. DOI: 10.1145/3627673.3679646

[5] Kachynskyi A.B. Structurally-functional model of the system for ensuring information and information-psychological security. Reports of the National Academy of Sciences of Ukraine. 2023. №1. C. 16-23. (ukr. language) DOI: 10.15407/dopovidi2023.01.016

[6] Singh, A., Rose Anish, P., Verma, A. et al. A data decomposition-based hierarchical classification method for multi-label classification of contractual obligations for the purpose of their governance. Sci Rep 14, 12755 (2024). DOI: 10.1038/s41598-024-63648-x

[7] Zgurovsky M., Pankratova N.D. System analysis: Theory and applications. Springer Science & Business Media, 2007. – 447 p.

[8] Dmitry Lande, Leonard Strashnoy. An Advanced No-Code Programming Framework for Complex Problems in LLM Environments. ResearchGate Preprint. DOI: 10.13140/RG.2.2.25307.89129 (May, 2025). - 14 pp.

[9] Busch, D. and Lande, D., 2025. Semantische Dokumentenindexierung mit generativer KI. Mitteilungen der Vereinigung Österreichischer Bibliothekarinnen und Bibliothekare, 78(1). DOI: https://doi.org/10.31263/voebm.v78i1.9251

[10] Page, L., Brin, S., Motwani, R., & Winograd, T. The PageRank Citation Ranking: Bringing Order to the Web. Stanford InfoLab, 1999.

[11] Mallick, C., Das, A.K., Dutta, M., Das, A.K., Sarkar, A. (2019). Graph-Based Text Summarization Using Modified TextRank. In: Nayak, J., Abraham, A., Krishna, B., Chandra Sekhar, G., Das, A. (eds) Soft Computing in Data Analytics. Advances in Intelligent Systems and Computing, vol 758. Springer, Singapore. URL: 10.1007/978-981-13-0514-6_14.

[12] A. Bruns, H. Snee. How to Visually Analyse Networks Using Gephi. SAGE Publications, Limited, London, 2022.

[13] Traag V. A. From Louvain to Leiden: guaranteeing well-connected communities / V. A. Traag, L. Waltman, N. j. van Eck. Sci Rep 9, 5233 (2019). DOI: 10.1038/s41598-019-41695-z.

[14] F. Y. Wu. The Potts model. Rev. Mod. Phys. 54, 235 – Published 1 January 1982.

[15] Closas, P., Coma, E. and Méndez, L., 2012. Sequential detection of influenza epidemics by the Kolmogorov-Smirnov test. BMC medical informatics and decision making, 12(1), p.112. DOI: 10.1186/1472-6947-12-112