

Parameterization and evaluation of the timed semantic influence algorithm efficiency based on the deviation of similarity matrices

Lyubomyr Chyrun^{1,†}, Dmytro Uhryn^{2,†}, Yuriy Ushenko^{2,*,†}, Artem Kalancha^{2,*,†} and Oleh Pastukh^{3,*,†}

¹ Ivan Franko National University of Lviv, Universytetska Street 1 79000 Lviv, Ukraine

² Yuriy Fedlovych Chernivtsi National University, Kotsiubynskoho Street 2 58012, Chernivtsi, Ukraine

³ Ternopil Ivan Puluj National Technical University, Ruska Street 56, 46025, Ternopil, Ukraine

Abstract

This study analyzed and critically reviewed the Temporal Semantic Influence (TSI) method, developed to assess the mutual influence between information sources. An approach to its optimization and systematic parameter selection using the deviation of similarity matrices as a key performance evaluation criterion is proposed, which should better distinguish between sources. The impact of each of the main parameters of the algorithm (time factor weight, message horizon, threshold value, and similarity function) was analyzed and the impact on the final results was visually demonstrated. The optimized algorithm was applied to the Cascade Influence graphs of the algorithm to compare the impact of the optimization. The obtained conclusions demonstrate the possibility of improving the accuracy and stability of the TSI method, and also form the basis for further improvement of algorithms for analyzing information flows in network environments.

Keywords

similarity function, TSI, similarity matrix deviation, influence graphs, algorithm parameterization

1. Introduction

In today's global information environment, the volume of text messages distributed through open sources is growing exponentially, creating a multi-component and dynamic communication space. Telegram channels [1], social media pages, and news resources create a large-scale information ecosystem in which reliable messages, commercial content, propaganda materials, and outright disinformation simultaneously circulate. Such a multi-layered structure poses a serious challenge for researchers and analysts, as the data set can no longer be processed quickly by human resources alone, and the speed of message dissemination significantly exceeds the capabilities of traditional monitoring and response systems. The task of tracking, identifying, and comparing sources becomes particularly relevant during periods of large-scale information campaigns and hybrid conflicts, when timely detection of disinformation and hidden interconnections between channels is critical for strategic decision-making. In conditions of such information turbulence, the analyst is faced not only with a colossal volume of sources, but also with the need to quickly assess the level of their reliability, identify mutual influence, recognize hidden "information waves" and data distribution schemes [2].

*AISSE-2025: International Workshop on Applied Intelligent Security Systems in Law Enforcement, October, 30–31, 2025, Vinnytsia, Ukraine

^{1*} Corresponding author.

[†] These authors contributed equally.

✉ Lyubomyr.Chyrun@lnu.edu.ua (L. Chyrun); d.ugryn@chnu.edu.ua (D. Uhryn); y.ushenko@chnu.edu.ua (Y. Ushenko); kalancha.artem@chnu.edu.ua (A. Kalancha); ol_pas@tntu.edu.ua (O. Pastukh)

ORCID 0000-0003-4360-9003 (L. Chyrun); 0000-0003-4858-4511 (D. Uhryn); 0000-0003-1767-1882 (Y. Ushenko); 0009-0004-1451-7470 (A. Kalancha); 0000-0002-0080-7053 (O. Pastukh)



© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Traditional methods based only on indicators of lexical similarity of messages no longer meet modern requirements. They allow you to record superficial similarity of content, but do not ensure the identification of the primary source of information, its separation from secondary relays and tracking of temporal patterns of its distribution.

Under such conditions, the development and implementation of systems capable of automatically processing large amounts of text data, integrating quantitative and temporal characteristics of messages, as well as identifying structural and hidden connections between channels becomes particularly important. The introduction of new algorithmic approaches and analysis methods allows not only to reduce the load on human resources, but also to significantly increase the accuracy and speed of identification of key information nodes. That is why the development of methods that combine semantic and temporal analysis is today considered one of the priority areas of research in the field of natural language processing [3], open source monitoring, and the study of information flows.

2. Problem statement

Traditional approaches to comparing information sources are mainly based on the analysis of the lexical content of messages. The most common methods, in particular the use of cosine similarity and other statistical metrics, allow assessing the degree of similarity of content between different channels. However, such methods have significant limitations: they record only the fact of textual similarity, but do not take into account the temporal dynamics of the appearance of messages. As a result, it is impossible to determine which source is the initiator of the dissemination of information, and which acts as its relay.

This aspect is of particular importance in the conditions of modern information confrontations, when the speed of reaction to a message becomes a key indicator of the influence of the source. Ignoring the time dimension leads to a distortion of the real picture of the interaction between channels, since the order of publications, the delay in the dissemination of similar messages and their sequence are not taken into account. As a result, the analyst receives only a static “snapshot” of content similarity, devoid of the critical context that determines the actual strength and direction of information influence. During periods of intense disinformation campaigns, this becomes a serious drawback, as the speed of transmission and the efficiency of publications often shape public opinion faster than the content of individual messages [4].

To overcome these limitations, the study proposes to expand the analysis by adding an additional is the temporal dimension. It allows us to assess not only the semantic proximity of messages between different channels, but also the time difference between their appearance. This approach allows us to quantify how quickly one source picks up information from another, and, accordingly, describe the degree and direction of mutual information influence. The integration of this dimension into analytical algorithms opens up new opportunities for identifying key nodes in the information network, clustering sources [5, 6], classifying sources [7], predicting the spread of content, and early identification of coordinated information operations.

3. Purpose of the article

In a previous study, the Timed Semantic Influence (TSI) method was developed and tested, which integrates the analysis of semantic similarity and time intervals between message publications. The application of this approach to a sample of Ukrainian news Telegram channels demonstrated its effectiveness in determining the direction and intensity of information influence between sources. At the same time, the results obtained were based on fixed algorithm parameters and a classical approach to assessing textual similarity, which significantly limits the potential for further improvement of the method.

The purpose of this work is to optimize the TSI algorithm with an emphasis on the selection and systematic adjustment of its key parameters: similarity thresholds, time horizons, weight coefficient α , and others – in order to increase the accuracy and stability of information influence assessment. In addition, the study involves assessing the possibilities of modifying the algorithm by using alternative message similarity measurement functions, in particular, contextual models and modern metrics for comparing short texts. Such an extension will allow not only to improve the quality of the classification of relationships between channels, but also to test the universality of the proposed approach on various data sets, ensuring its scalability and relevance for a wide range of information flow analysis tasks.

4. Times semantic influence algorithm

The first stage of the TSI algorithm is the selection of two information sources, between which the mutual influence is analyzed. In the framework of this study, such sources are individual Telegram channels. For convenience, they are designated as channel *A* (first channel) and channel *B* (second channel). The justification of the choice of channels is of key importance, since the reliability, reproducibility and interpretability of further conclusions depend on it. In the selected time interval, both channels must demonstrate sufficient activity in publishing messages, otherwise the final TSI coefficients may turn out to be statistically unstable and lose their analytical value.

The second stage involves the selection of a specific message from channel *A*, which serves as the basic unit for comparison. This message is designated as *M* and must be clearly identified by the date and time of publication, since the algorithm is based on the precise measurement of time intervals.

Next, an array of messages *N* from channel *B* that can potentially correspond to message *M* is formed. This array is defined within the specified parameter *horizon_hours* (event horizon), i.e. $\pm h$ hours from the time of publication of *M*. Messages from channel *B* that fall outside this time range are filtered out. The parameter *h* is one of the key parameters for tuning the algorithm, since it determines the sensitivity to short-term influences and the level of noise in the sample.

After forming the array *N*, a search is performed for a relevant pair of messages (the base *M* from channel *A* and the potential *m* from channel *B*). For this, semantic similarity is calculated based on vector representations of the texts. The similarity threshold *similarity_threshold* (*S*) is set as a parameter of the algorithm. If no message from *N* exceeds *S*, message *M* is skipped. Tuning the parameter *S* is critically important: too high a threshold can cut off relevant pairs, while too low a threshold can create an excessive number of false matches.

In the basic implementation, the method of sorting all messages from the array *N* with the selection of the pair with the maximum semantic similarity is used. Alternatively, it is possible to use the method of sorting messages by temporal proximity and further checking for similarity in ascending order Δt . Testing both approaches is one of the tasks of further optimization of the algorithm. At the next stage, the TSI coefficient is calculated for each selected pair of messages. It integrates the semantic proximity of messages and the time interval Δt between their publication. The study considers the influence decay function:

$$TSI = sm(m_a, m_b) \cdot e^{-\alpha \frac{\Delta t}{60}} \quad (1)$$

In this expression, $sm(m_a, m_b)$ denotes the cosine similarity between the texts, Δt is the absolute difference in minutes between the publications, and α is a parameter that regulates the weight of the time component. The optimal selection of α allows you to shift the balance between the

semantic and time components of influence. The algorithmic complexity of the proposed method is $O(n) = \log n$, where n denotes the size of the channel array. This estimation is based on the fact that, during the comparison process, each pair of messages between two channels is evaluated only once. The inverse value can be interpreted from the results of the previous computations.

After calculating the TSI coefficients for all relevant pairs, the direction of influence is determined by the chronology of publications. Then the obtained values are aggregated for each direction, forming the integral indicator Total TSI as the sum of all atomic TSI values, which is recorded in the matrix of mutual influence of channels and serves as the basis for further quantitative and visual analysis of information interaction.

5. Method optimization and tuning

The TSI algorithm, first implemented in our previous studies, has proven its effectiveness in detecting mutual information influence between sources. In this work, it has undergone significant optimization: the introduction of caching of intermediate calculations and the review of key text operations have significantly reduced the computational complexity and increased the speed when processing large samples of messages. The next stage is the parameterization and tuning of the algorithm [8]. We investigate the impact of changing such parameters as the coefficient α (the weight of the time component in the formula), the message horizon (*horizon_hours*), and the similarity threshold (*similarity_threshold*) on the final results. In the experiments, α varied from 0.1 to 0.9; *horizon_hours* – from 2 to 8 hours; *similarity_threshold* – from 0.4 to 0.8. In addition, the algorithm was configured to use different similarity assessment functions between messages, which allowed us to compare the effectiveness of alternative metrics. In total, 315 combinations of parameters were generated and the algorithm was run for each of them.

Particular attention was paid to similarity functions [9], since they determine the method of measuring the similarity between two messages:

Cosine Similarity is a classic method in which texts are considered as vectors in a multidimensional space (for example, token frequencies), and similarity is calculated as the cosine of the angle between them, see (2). A value approaching 1 indicates greater similarity of texts [10]. The advantages are high calculation speed and interpretability of results, the disadvantage is ignoring word order and context.

$$cs(s_i, s_j) = \frac{z(s_i)^T z(s_j)}{\|z(s_i)\|_2 \|z(s_j)\|_2} \in [-1, 1] \quad (2)$$

where $z(s_i)$ is the corresponding vector for corresponding message s_i .

Jaccard Similarity is a measure based on the ratio of the number of shared unique tokens to their union [11]. This approach works well for short texts or headlines and allows us to assess the degree of thematic overlap, but is less sensitive to the number of repetitions and stylistic nuances, see (3).

$$J(x, y) = \frac{\sum_{i=1}^n \min(x_i, y_i)}{\sum_{i=1}^n \max(x_i, y_i)} \quad (3)$$

where x and y are the lists of fuzzy matching scores

Fuzzy-matching Similarity is a method of approximate token comparison (e.g., via RapidFuzz) that finds the best matches for each word in another message, taking into account orthographic and morphological differences, see (4). This approach is useful when texts may contain minor discrepancies or errors [12]. Its advantage is higher tolerance to “noisy” data.

$$FuzzSim(T_1, T_2) = \frac{\frac{1}{n} \sum_{i=1}^n \max_{1 \leq j \leq m} fuzz(t_{1,i}, t_{2,j}) + \frac{1}{m} \sum_{j=1}^m \max_{1 \leq i \leq n} fuzz(t_{2,j}, t_{1,i})}{2} \quad (4)$$

where $fuzz(x,y) \in [0,1]$ – this is your normalized “fuzzy string similarity” score between two tokens, $n=|T_1|$, $m=|T_2|$ is the number of corresponding tokens, $t_{1,i}$, $t_{2,j}$ – individual tokens.

Using these three metrics and the similarity function in different combinations with α , *horizon_hours*, and *similarity_threshold* allowed us to investigate how each configuration affects the stability and quality of the interaction matrices. Thus, after describing the algorithm, we moved on to the stage of systematic tuning, which was an important step in increasing the accuracy and flexibility of TSI.

6. Interpretation of tuning results

To empirically test the improved TSI algorithm, a limited data sample was used – messages from eight Ukrainian Telegram information channels over a ten-day period. This approach provided the possibility of quickly running the algorithm multiple times with different parameters, which made it possible to simultaneously control the quality of the obtained results and the time of calculations on an identical data set.

At the output, the TSI algorithm forms a matrix of similarity/interaction between information sources for each combination of parameters. To compare the results of different settings, we used a statistical estimate of the variance of the elements of this matrix. The main idea is that a higher variance indicates a better ability of the algorithm to differentiate the degrees of mutual similarity of channels: if all similarity values are close to each other, the matrix turns out to be uninformative; if the indicators differ significantly, this indicates more pronounced information connections [13].

To do this, the main diagonal (elements of the form (i,i), corresponding to the self-comparison of the source with itself, is excluded from each matrix, since on it the similarity is always equal to 1 or the maximum possible value. Then the variance is calculated by the formula:

$$D(M) = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^n |m_{i,j} - mu|, i \neq j \quad (5)$$

where n is the matrix size, $m_{i,j}$ is the matrix element, mu is the mean value of matrix.

The interpretation of this indicator (see formula 5) is straightforward: the larger the deviation, the more clearly the algorithm distinguishes channels by the degree of their similarity or influence for the given parameters. This approach allows us to objectively compare different combinations of α , *horizon_hours*, *similarity_threshold* and similarity functions not only by the visual appearance of the matrices, but also by a single quantitative criterion.

In addition, the calculation time was recorded for each set of parameters, which made it possible to combine the quality assessment (absolute deviation) with the performance (execution duration). This is of particular importance for scaling the algorithm to large data sets and determining the optimal balance between accuracy and speed.

To test the influence of the *horizon_hours* parameter, a graph of the dependence of the average TSI Score on the width of the time horizon was constructed. The graph shows (see Fig. 1) that with an increase in the range of messages taken into account, the TSI Score value increases almost linearly, but the increase is insignificant. This behavior indicates that the horizon parameter has only a moderate effect on the final result, without causing sharp changes in the structure of the similarity matrices.

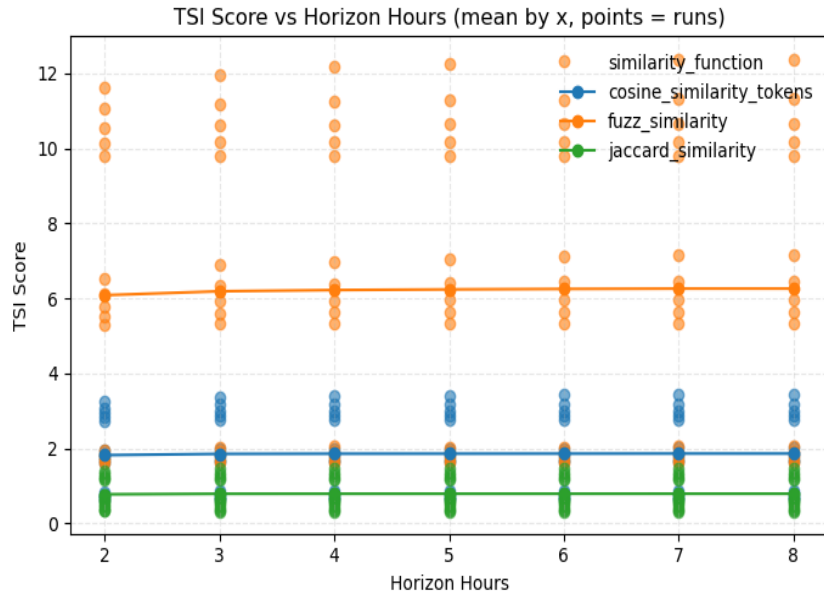


Figure 1: Dependence of TSI score on the time horizon of messages.

In other words, increasing the horizontal window gives a certain increase in informativeness, but does not radically change the assessment of the mutual influence between sources. This allows you to choose a wider horizon (for example, 8 hours) for a more complete coverage of relevant messages without the risk of significantly distorting the results of the TSI algorithm.

Analysis of the dependence of the average TSI Score on the value of the parameter α , which regulates the weight of the time component in the TSI formula, showed a stable trend: with an increase in α , the average TSI Score value gradually decreases (see Fig. 2). This decrease is not sharp, but noticeable and reflects a decrease in the influence of the semantic component with an increase in the weight of the time factor.

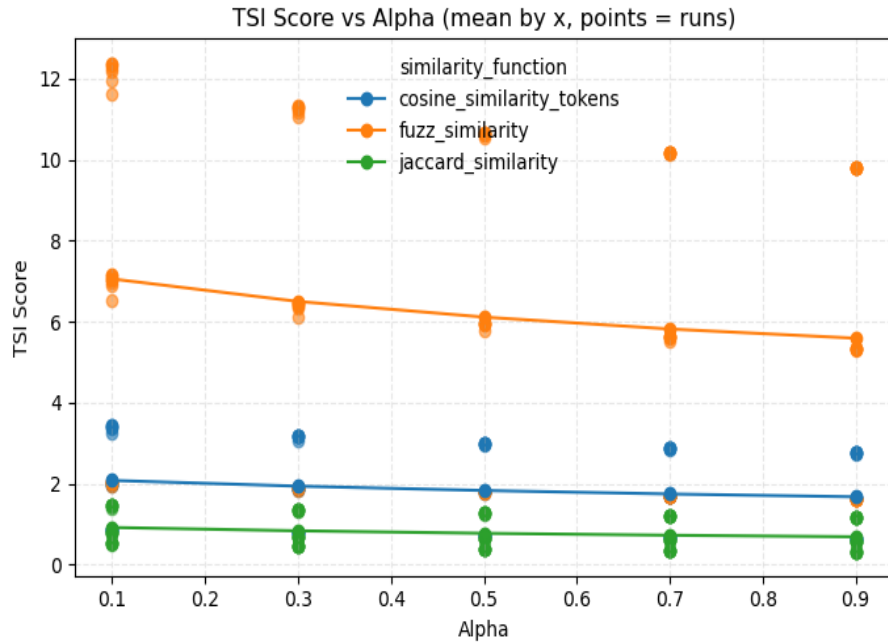


Figure 2: Dependence of TSI score on parameter α .

When setting up the TSI algorithm, the similarity threshold parameter (*similarity_threshold*) was set at 0.4 as the minimum allowable value, which ensures the basic significance of the

similarity between messages. Further analysis showed that with an increase in this threshold, the integral TSI Score indicator systematically decreases (see Fig. 3). This trend indicates excessive selectivity of the algorithm at high thresholds: it selects fewer and fewer pairs of messages that are considered similar, as a result of which the average TSI Score indicator decreases and distinguishes channels by the degree of influence worse.

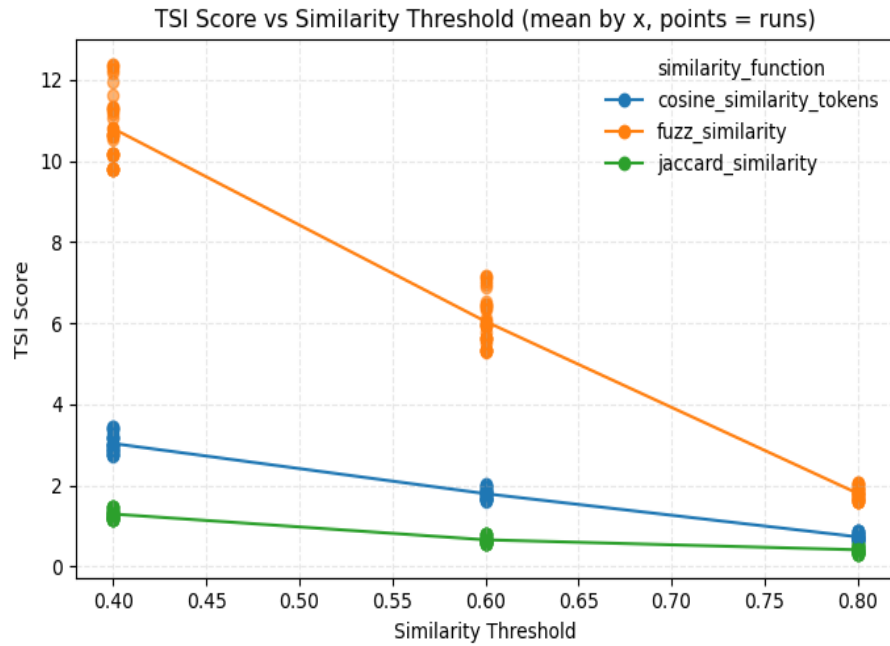


Figure 3: Dependence of TSI score on the similarity threshold parameter.

The results of the algorithm are affected not only by the “quality” of similarity between the selected messages, but also by the number of pairs included in the calculation. The optimal threshold value should balance these two factors – to provide a sufficient amount of data for statistically reliable conclusions, provided that there is an acceptable level of semantic similarity. Such a balance allows us to achieve both stability and sensitivity of the algorithm to real information connections between sources.

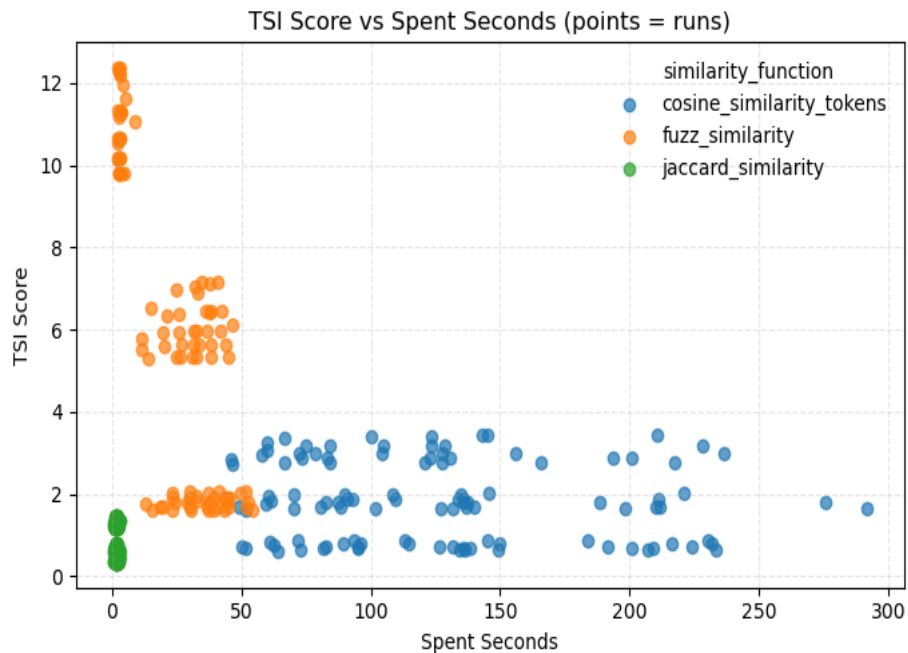


Figure 4: Evaluating the effectiveness of similarity functions.

Special attention in the study is paid to the comparison of different similarity functions, since this stage of the TSI algorithm is one of the most resource-intensive. The corresponding graph (see Fig. X) shows the dependence of the average TSI Score value and the algorithm execution time on the selected similarity function.

The obtained results demonstrate a clear picture (see Fig. 4). Jaccard Similarity turned out to be the fastest among the three implemented functions, but provides the lowest TSI Score values, which indicates the insufficient sensitivity of the metric to the semantic and stylistic nuances of the texts. Cosine Similarity gives slightly better similarity indicators compared to Jaccard, but reveals a high dependence on the α parameters and the similarity threshold: the calculation time can increase significantly even on the same data. This is explained by the fact that the formation of vector representations and the calculation of cosine similarity is one of the most computationally expensive steps of the algorithm.

The best results were obtained when using Fuzzy Similarity. This metric exhibits different TSI Score values depending on the α , similarity threshold, and horizon parameters, but when properly tuned, it provides a combination of high TSI Score values with short execution times. Thus, Fuzzy Similarity turned out to be the most balanced similarity function for the improved TSI method, providing the optimal ratio of quality and performance.

Table 1

Top 5 of most efficient combination of parameters

Sim. function	Horizon in Hours	Alpha	Sim. threshold	TSI Score
fuzz_similarity	8	0.1	0.4	12.389465
fuzz_similarity	7	0.1	0.4	12.379137
fuzz_similarity	6	0.1	0.4	12.322707
fuzz_similarity	5	0.1	0.4	12.275995
fuzz_similarity	4	0.1	0.4	12.168700

Analysis of the final TSI Score values in different combinations of parameters (see Table 1) showed that the most effective configuration is the one that uses Fuzzy Similarity (FUSE Similarity) with an event horizon of 8 hours, a parameter value of $\alpha = 0.1$ and a similarity threshold of 0.4. As was demonstrated earlier, increasing the event horizon has only a minor effect on the final result, but in this combination it provides maximum coverage of relevant messages without compromising performance.

It is this configuration that gives the highest integral TSI Score among all tested options and at the same time remains fast in execution. Thus, the combination of Fuzz Similarity, a horizon of 8 hours, $\alpha = 0.1$ and a threshold value of 0.4 can be considered the optimal balance between the quality and performance of the TSI algorithm on the studied sample.

7. Cascade Influence graphs comparison

Based on the intermediate results of the TSI algorithm, directed graphs of mutual influence of channels were constructed. In these graphs, nodes correspond to information sources, and edges reflect the detected direction and intensity of influence between them [14–16]. Influence was

determined based on the similarity of messages: if two channels have similar content, the presence of a potential information connection is recorded.

Formally, if channel *A* and channel *B* have at least two pairs of messages similar according to the selected similarity function, and at the same time messages from channel *A* are published earlier than the corresponding messages from channel *B*, it is concluded that channel *A* influences channel *B*. In the opposite case, the direction of the edge changes, and the influence is recorded from channel *B* to channel *A*.

The constructed cascade graphs of mutual influence became the basis for a comparative analysis of the results of the basic and improved versions of the TSI method. In the previous configuration, the leading positions in the filtered graphs were occupied mainly by three channels: UaOnlii ($\approx 39\%$), voynareal ($\approx 27\%$) and truexanewsua ($\approx 12\%$), while other sources showed significantly lower rates of occurrences at the root.

After applying the improved algorithm with selected parameters, the picture became significantly more detailed. The absolute values of occurrences at the roots of the graphs increased significantly (for example, for UaOnlii from 58 to 107, for truexanewsua – from 18 to 54), and the distribution of percentages became more balanced. Although UaOnlii and voynareal remained the leading sources, the share of other channels – truexanewsua, lachentyt, susilnenews, kievreal1 – increased noticeably, which indicates a better ability of the method to recognize secondary but systematic influences [17].

The key factor in this improvement was the modernization of the similarity function between messages. Thanks to more accurate text matching, the algorithm generates significantly more elements of influence between channels, which directly increases the number of occurrences in the graph roots. This, in turn, increases the statistical reliability of estimates and the accuracy of determining real information influence, since a larger number of relevant messages and interactions are included in the calculation, creating a complete picture of the information network.

8. Conclusions

In the course of the research, the TSI method, which was previously implemented in our previous works, was optimized and improved. In this version, the algorithm received an improved function for calculating similarity between messages, caching of intermediate results, and optimized computational complexity, which made it possible to increase the speed of operation.

For each of the main parameters of the algorithm – the weight of the time factor (α), the message horizon, the similarity threshold, and the message similarity function – a detailed analysis of their influence on the final result was conducted. The evaluation was carried out on a limited sample based on the deviation of similarity matrices. This criterion allowed us to objectively compare different parameter configurations and choose the optimal combination.

In addition, influence graphs between sources were constructed and analyzed, reflecting the direction and strength of the potential information influence of channels on each other. Comparison of the baseline and new results showed that the improved TSI forms significantly more elements of influence between channels, increases the number of occurrences in the roots of cascade trees, and provides a more accurate assessment of leading channels in the studied network. As a result, the applied approach allowed not only to increase the performance and informativeness of the method, but also to obtain a more detailed and reliable picture of informational relationships between sources, which opens up prospects for further research and expansion of the method to larger data samples.

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT-5 and Grammarly in order to: Grammar and spelling check and as a smart Search Engine to find related works based on context

of conversation. After using these tools and services, the authors reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

- [1] Telegram APIs, 2025. URL: <https://core.telegram.org/api>
- [2] G. D. S. Martino, S. Cresci, A. Barrón-Cedeño, S. Yu, R. Di Pietro, P. Nakov, A survey on computational propaganda detection, in: 29th International Joint Conference on Artificial Intelligence (IJCAI 2020), 2020. <https://doi.org/10.24963/ijcai.2020/672>
- [3] P. Johri, S. K. Khatri, A. T. Al-Taani, M. Sabharwal, S. Suvanov, A. Kumar, Natural language processing: history, evolution, applications and future work, in: 3rd Int. Conf. on Computing Informatics and Networks, LNNS, Springer, 2021, 365–375. https://doi.org/10.1007/978-981-15-9712-1_31
- [4] A. Guille, H. Hacid, C. Favre, D. A. Zighed, Information diffusion in online social networks: a survey, ACM SIGMOD Record 42 (2013) 17–28. <https://doi.org/10.1145/2503792.2503797>
- [5] R. Guan, X. Shi, M. Marchese, C. Yang, Y. Liang, Text clustering with seeds affinity propagation, IEEE Trans. Knowl. Data Eng. 23 (2011) 627–637. <https://doi.org/10.1109/TKDE.2010.144>
- [6] R. Janani, S. Vijayarani, Text document clustering using spectral clustering with PSO, Expert Systems with Applications 134 (2019) 192–200. <https://doi.org/10.1016/j.eswa.2019.05.030>
- [7] V. Dogra, S. Verma, Kavita, P. Chatterjee, J. Shafi, J. Choi, M. F. Ijaz, A complete process of text classification using state-of-the-art NLP models, Computational Intelligence and Neuroscience 2022 (2022) 1–26. <https://doi.org/10.1155/2022/1883698>
- [8] W. Daelemans, V. Hoste, F. De Meulder, B. Naudts, Combined optimization of feature selection and algorithm parameters in machine learning of language, in: ECML 2003, LNCS 2837, Springer, 2003. https://doi.org/10.1007/978-3-540-39857-8_1
- [9] B. M. Magara, S. O. Ojo, T. Zuva, A comparative analysis of text similarity measures in recommender systems, in: ICTAS 2018, IEEE, 2018, 1–5. <https://doi.org/10.1109/ICTAS.2018.8368766>
- [10] K. Park, J. S. Hong, W. Kim, Combining cosine similarity with classifier for text classification, Applied Artificial Intelligence 34 (2020) 396–411. <https://doi.org/10.1080/08839514.2020.1723868>
- [11] L. Zahrotun, Comparison of Jaccard, Cosine and combined similarity in SNN clustering, ComEngApp Journal 5 (2016). <https://doi.org/10.18495/COMENGAPP.V5I1.160>
- [12] J. Kalluru, Enhancing data accuracy and efficiency using fuzzy matching, International Journal of Science and Research (2023). <https://doi.org/10.21275/sr23805184140>
- [13] S. Shalileh, B. Mirkin, Least-squares community extraction in feature-rich networks, PLoS ONE 16 (2021). <https://doi.org/10.1371/journal.pone.0254377>
- [14] L. Zhao et al., Effects of time-dependent diffusion on rumor spreading in social networks, Physica A 452 (2016) 1–11. <https://doi.org/10.1016/j.physleta.2016.04.025>
- [15] V. Lytvyn, O. Lozynska, D. Uhryn, M. Vovk, Y. Ushenko, Z. Hu, Decision support in GIS using swarm intelligence, Modern Education and Computer Science 2 (2023) 62–72. <https://doi.org/10.5815/ijmeecs.2023.02.06>
- [16] S. Vladov, L. Chyrun, E. Muzychuk, V. Vysotska, V. Lytvyn, T. Rekunenko, A. Basko, Intelligent Method for Generating Criminal Community Influence Risk Parameters Using Neural Networks and Regional Economic Analysis, Algorithms 18:8 (2025) 523. <https://doi.org/10.3390/a18080523>
- [17] S. Vladov, V. Vysotska, V. Sokurenko, O. Muzychuk, L. Chyrun, The Intelligent Data Measurement System Using Neural Network Technologies and Fuzzy Logic Under Operating Implementation Conditions, Big Data and Cognitive Computing 8:12 (2024) 189. <https://doi.org/10.3390/bdcc8120189>