Hindawi
Discrete Dynamics in Nature and Society
Volume 2022, Article ID 3079936, 12 pages
https://doi.org/10.1155/2022/3079936

Research Article

Event Forecasting in Organizational Networks: A Discrete Dynamical System Approach

Piotr Śliwa

Department of Organization and Management Theory, Wroclaw University of Economics and Business, Wrocław 53-345, Poland

Correspondence should be addressed to Piotr Śliwa; piotr.sliwa@ue.wroc.pl

Received 5 December 2021; Accepted 5 February 2022; Published 14 March 2022

Academic Editor: Baogui Xin

Both inter- and intraorganizational networks draw the attention of researchers and practitioners from various disciplines who view them as the fabric of the socioeconomic world. The network perspective is believed to successfully model most of the socioeconomic phenomena, which, in combination with the prospects of continuously advancing tools for automated data mining and machine learning, gives a tempting promise to effectively forecast socioeconomic events occurring in our societies and businesses. Despite their significance, the topic of event forecasting in the context of organizational networks appears unexplored. Therefore, the objective of this study was (1) to fill the theoretical gap by proposing a mathematical model for organizational network event forecasting, rooted in the social science to remain consistent with the theory, and (2) to experimentally evaluate how the model performs on real data and validate if the results support its use in practical applications. An implementation of the proposed model, based on a decision tree classifier, achieved a prediction accuracy of 87% on a longitudinal data sample and thus demonstrated the practical usability of the model.

1. Introduction

Both inter- and intraorganizational networks draw the attention of researchers from various disciplines who view them as the fabric of the socioeconomic world. According to Moliterno and Mahony [1], the interorganizational network perspective translates a fragment of the economy to a graph comprising a group of organizations (nodes) interconnected with each other by numerous relationships (ties). It tends to omit the internals of the organizations and treat them as black boxes, with only the relationships and a selection of variables visible to an observer. The internals—lower-level organizational units (e.g., departments, groups, divisions)—become visible in the subsequent intraorganizational level of analysis, which is further divisible to the individual, and potentially other following levels. Some studies focus solely on one level of analysis—interorganizational [2] or intraorganizational [3]—while a significantly smaller niche of a multilevel perspective that aims at projecting a full, holistic picture of a focal network is also discernible [1].

Regardless of the level of analysis, researchers successfully embraced the network approach to identify and describe countless phenomena, e.g., competition in interorganizational networks [4], social capital and value creation in intraorganizational networks [5], and knowledge dissemination in inter- and intraorganizational networks [6], just to name a few. Interestingly, even with an already long-lasting focus dating back to 1970 and popularity manifesting in 3,200 results in the Scopus database, the aspect of event forecasting in the organizational network theme remains uncovered. Existing quantitative research on organizational networks focuses mainly on exploring causal relationships between selected variables during a bounded period [2, 7, 8].

Event forecasting is a popular topic in the fields of knowledge discovery and data mining, including the subarea of social network analysis. Several researchers constructed and successfully applied predictive models that identified patterns in social networks and forecasted subsequent discrete events with satisfactory performance [9–11]. The results found use in various interesting and valuable applications, i.e., crime event prediction [11], social unrest

prediction [9], stock event prediction [10], to name a few. The conceptual models used in the research were based on the clear definition of a social network, in which nodes represent individuals and edges represent social relations between them. However, organizational networks, which are inherently more complex systems, are still missing a similar conceptual model for event prediction that could support managerial processes in organizational network contexts, e.g., governmental [12], innovative [7], or knowledge-oriented [3]. The applications for event forecasting in the organizational network contexts seem to be as abundant as their social network counterparts.

Therefore, the objective of this study is twofold: First, to fill the theoretical gap in the research on organizational networks by proposing a mathematical model for organizational network event forecasting, rooted in social science to remain consistent with the theory, and second, to experimentally evaluate how the model performs on real data and validate if the results support its use in practical applications.

The proposed model was built on top of a holistic definition of organizational networks (i.e., multilevel multimodal organizational networks) and interactive events [13]. An important factor considered during the model's development was to enable the use of advanced pattern recognition techniques, especially machine learning algorithms, which are known in many domains for their performance on sophisticated data [14]. To achieve this, the mathematical model was described in a form of a composite function that translates an organizational network to a discrete dynamical system, which components performing consecutive prediction steps can be easily substituted with more advanced ones in future iterations.

The experiment used a longitudinal data sample collected from Twitter which comprised interactive events occurring in a real organizational network in the course of 11 years. The data was split into two samples—training and testing—at an elected point in time, to simulate a real situation in which a user has observed the past (training set) and will experience the future (test set). The event forecasting model was implemented as a Python script, which used the scikit library for machine learning tasks, and a custom implementation of feature selection (i.e., event windowing algorithm). The model was trained and validated with the split data to measure the accuracy of predictions and test the H1 hypothesis.

H1. There is a positive correlation between sequences of past (E_p) and future (E_f) interactive events occurring in organizational networks. Therefore, a predictor function F can be found that accepts past interactive events and produces future interactive events (predictions). Correlation C of the function, expressed as the ratio of correctly predicted future events E_f^+ to all predicted events $|E_f^+| + |E_f^-|$, is significantly higher than 0.5 (whereas $C = 0.5$ means there is an equilibrium of correct and incorrect predictions).

$$\exists F\left(E_p\right) = E_f,$$
$$C(F) = \frac{\left|E_f^+\right|}{\left|E_f^+\right| + \left|E_f^-\right|} \gg 0.5. \quad (1)$$

The article is structured as follows. Section 2 reviews the literature used for the development of the conceptual model of organizational network event forecasting. Section 3 presents the model itself. Section 4 discusses in detail the methodology of the experiment, and Section 5 presents the results. Finally, Section 6 discusses the results and summarizes the research.

2. Literature Review

2.1. Dynamic Interaction Graphs. Dynamic interaction graphs are a relatively new concept in social studies, although static interaction graphs have been used to model social networks (among others) in plentiful studies [15]. As opposed to static graphs which capture aggregated and/or interpreted relations between interacting network nodes, edges in dynamic interaction graphs represent individual interactive events, rendering the resulting network a discrete dynamical system that evolves over time [16]. The literature discussing applications of dynamic interaction graphs in the context of organizational networks was found to be scarce to nonexistent.

Formally, the static interaction graph can be understood as $G = (V, E)$ where $V = (v_1, v_2, \ldots)$ represent nodes and $E = (e_1, e_2, \ldots)$ represent edges. On the other hand, the dynamic interaction graph contains a time variable and thus can be described by discrete snapshots $G_t = (V, E_t)$ [17].

2.2. Multimodal (Complex) Networks. The literature regarding social network analysis defines the concept of multimodal networks (or complex networks) as graph structures comprising interconnected nodes of multiple types [18, 19]. As opposed to one-mode networks (e.g., social networks which encompass only humans) where nodes are homogeneous, they allow capturing any kind of representational nodes in one network projection. For example, Krackhardt and Carley discussed an organizational network structure of individual, tasks, and resources—a three-mode network projection of an organization that improved its observability [20]. The conceptualization of multimode networks does appear in the context of organizational networks but is typically limited to two-mode networks (i.e., individuals and organizations), especially in the management literature [21].

2.3. Event Forecasting. The topic of event detection and forecasting using digital data sources has been covered with a variety of interesting studies in the data mining and knowledge discovery areas which discussed approaches to predicting cases of social unrest [9, 22, 23], stock market movements [24], election results [25], among others. Notably, most of the studies found during the literature review correlated determinant variables with patterns identified in processed social media posts using both general-use and dedicated models. For instance, Ning et al. [22] analyzed sequences of news articles as precursors leading to categorized events—protests. Comparably, Zhao et al. [23] analyzed tweets as precursors leading to events of social

unrest. In both works, the predicted variables (events) could be characterized as exogenous to the precursor variables. An opposite approach, in which the predicted event variables were endogenous to the precursor event variables, was presented by Laxman et al. [26]. Their generative model based on Hidden Markov Chains operated on a finite alphabet of possible event types and predicted a target event type from the provided windows of event sequences (event streams). Noteworthy, in this case, both precursor (input) and predicted (output) variables were the event variable which renders it an endogenous model. The comparison between exogenous and endogenous forecast variables is summarized in Figure 1.

2.4. Organizational Network Mapping. No articles discussing event forecasting using digital data sources in the context of organizational networks were found. Reviewed papers did not attempt to structure input data (i.e., social media posts or news articles) in any kind of organizational representation. However, a broader scope of the search for studies mapping organizational structures using digital data sources revealed a few recent, interesting articles. Dong and Rim [27] used social network analysis in their exploratory research to map the communication of nonprofit businesses that lead to the identification of partnerships between them. Their methodology was based on Shumate and Contractor's concepts of a representational network and a flow network [28]. The former—representational network—infers a relationship between two organizational network nodes from messages they broadcast to other network nodes, informing them about the relationship. In other words, a link appears between the nodes when they announce it to the public (so a family tree could fall into this category). Conversely, the concept of a flow network pinpointed by Shumate and Contractor infers a relationship between the two nodes from exchanges and transmissions of information, messages, and resources between them. In this case, the link appears when there is an identified flow; e.g., they proceed with a transaction, chat with each other, follow each other on social media, reshare posts, etc. without a need for public acknowledgment. Notably, both approaches lead to qualitatively different projections of the organizational network, and none of them seems to predominate over the other. To illustrate, the flow network maps direct interactions between nodes, which makes it arguably more precise than the representational network. On the other hand, the need for an explicit message from involved participants indicating a relationship that the representational network approach imposes can not only reduce noise but also leave out some relationships.

The flow network approach was also exploited by Wang and Guan [29] who, similarly to Dong and Rim, analyzed Twitter posts to extract following relationships among focal organizations represented by their social media profiles. As a result, the authors were able to present a projection of the analyzed organizational network and draw conclusions about the cross-sector structure of intergovernmental and international nongovernmental organizations.

3. Mathematical Model

To fill the gap found in the literature review, the proposed conceptual framework of organizational network event forecasting is built on the foundation of synthesized theories of (1) dynamic interaction graphs, (2) multimodal organizational networks, (3) event forecasting, and (4) organizational network mapping.

Following the multilevel and multimodal network theory [21], an *organizational network* is defined as a directed graph of taxonomically unconstrained nodes and ties. This notion assumes that any identified phenomenon along with relations to other phenomena can be translated into a labeled node linked to other labeled nodes with directed, labeled ties [30]. Noteworthy, the flexible structure allows capturing the phenomenon of an organization itself in the form of an additional node connected to other nodes comprising the real organization. For instance, an organization consisting of various people, resources, and other intangible assets could be translated into a graph, in which all these elements tie to a node representing the organization and to each other, like in Figure 2. A similar network projection (although comprising a network of individuals, tasks, and resources) was presented and discussed by Krackenhardt and Carley in their endeavor to reason about complex organizations with a series of hypotheses that can be empirically tested [20].

Such a flat network model is different from "discrete-level" approaches to multilevel organizational networks, which view an interorganizational network as one graph, and intraorganizational networks as hidden inside the former's nodes [31]. The flat structure along with the unconstrained typology of nodes and ties provides a simple, yet rich vocabulary for expressing socioeconomic phenomena as nouns (nodes) and verbs (edges). Notably, since the typology is not dimensionally restricted, the model can capture both a representational network (in which ties are communicated by nodes themselves) and a flow network (in which ties reflect real flows between nodes) by having two different sets (dimensions) of ties [28]. Unfortunately, the cross-analysis between tie dimensions is not in the scope of the current research but it opens an interesting research agenda.

Formally, the multimodal organizational network is described as a graph $G = (V, E)$ where nodes $V = (v_1, v_2, \ldots)$ are multimodal, or more specifically $v_1 \in C_a, v_1 \in C_b$, $C_a \equiv C_b$ where C is a node type (mode).

Events, following Provan et al.'s definition of interactive events occurring in organizational networks [13], are defined as discrete, labeled interactions between organizational network nodes with defined timestamps that determine their temporal location. Compared to models of individual networks but with a node being any modeled phenomenon and an edge being an interaction between nodes occurring at a specific point in time t, the organizational network can be viewed as a discrete dynamical system represented by a dynamic interaction graph in (2) [17].

$$G_t = (V, E_t), \tag{2}$$

$$t \in \{1, \ldots, n\}. \tag{3}$$

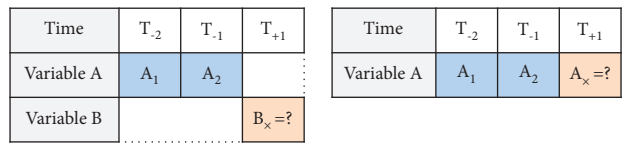


Figure 1: A comparison between exogenous (a) and endogenous (b) forecast variable models. The former aims at predicting a variable (B_x) from patterns in a precursor variable (A_n). Conversely, the endogenous variable models try to predict a future state of a variable (A_x) from its past states (A_n). Source: own elaboration.

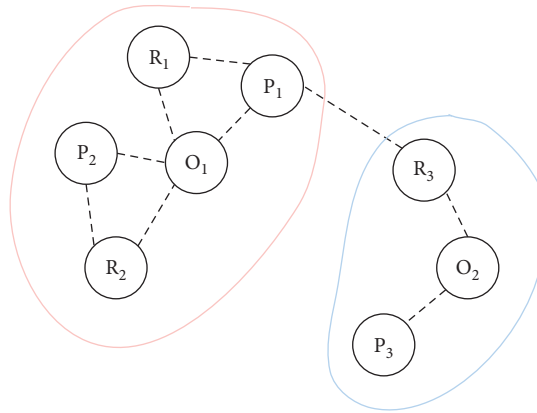


Figure 2: An exemplary snapshot (G_t) of a multimodal organizational network consisting of people P, resources R, and other intangible assets. All nodes belonging to an organization have links to a corresponding organizational node O. The relations translate to organizational boundaries (delineated with colored circles). Source: own elaboration.

Note that interactions (edges) are volatile—they exist only at time t. Since each edge refers to a single interaction, the graph can be expressed as a sequence of the interactive events e_n.

$$E = [e_1, \ldots, e_n],$$
$$e_n = [v_a, v_b, i_c, \tau_n], \tag{4}$$

where e_n is the nth four-element vector containing the origin node (the interaction's initiator) $v_a \in V$, the target node $v_b \in V$, $v_a \neq v_b$, the interactive event type $i_c \in I$ (with I being a set of interactive event types), and a timestamp $\tau_n \in T$ (with T being a set of all timestamps).

Event mining is defined as a process of data mining oriented at mapping the real organizational network into its digital representation (i.e., the interactive event sequence) using data from diverse data sources (e.g., web pages, social media content, accounting books, correspondence, transcriptions of conversations, etc.).

Event forecasting is defined as the task of predicting consecutive (future) events from a sequence of preceding (past) events or more formally as the predictor function in (5) [32].

$$F(E_p) = E_f, \tag{5}$$

$$F([e_{n-m}, \ldots, e_n]) = e_{n+1}, \tag{6}$$

where e_{n+1} is a future interactive event, e_n is the most recent interactive event, and e_{n-m} is the oldest interactive event in observable history. Its objective is to predict a label or multiple labels defining a linking event, an origin node, and a destination node, in a defined time range or a time point, depending on the implementation. In other words, event forecasting is defined as a function that accepts a sequence of events and produces a subsequent event. Additionally, the results can be extended with probability estimates defining how likely each of the predicted events is to materialize and how reliable the estimate is, according to internal metrics. The event forecasting concept is presented visually in Figure 3.

4. Methodology

The experiment was designed to test the H1 hypothesis formulated in the introduction and, if supported with results, present a successful application of the proposed framework for organizational network event forecasting, experimentally evaluated on real data.

4.1. Event Mining. Drawing upon the approach used by Dong and Rim [27] as well as by Wang and Guan [29], Twitter API was used to extract evidence for building a graph rendition of an organizational network using the representational network or flow network framework. Tweets collected with the API were processed to extract a tweet's author (the interaction's initiator node), all other users and hashtags mentioned by the author in the tweet (the interaction's target), and an absolute timestamp determining a time point when the tweet was published (Figure 4).

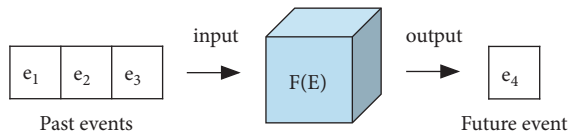
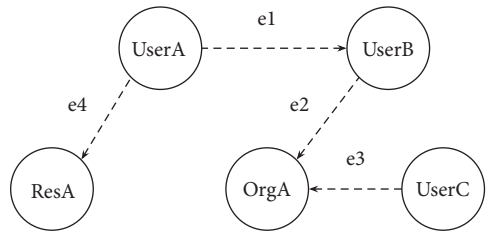


Figure 3: The event forecasting task. Source: own elaboration.



(a)

Event identifier	e1	e2	e3	e4
Timestamp	1523661393	1523662580	1523785510	1523785777
Origin node	@UserA	@UserB	@UserC	@UserA
Destination node	@UserB	#OrganizationA	#OrganizationA	#ResourceA

(b)

Figure 4: A sample of data points resulting from the event mining (b) and their corresponding snapshot (G_t) of a multimodal organizational network (a), in which the interactions are binary (hence the lack of interactive event type i_c). Source: own elaboration.

Noteworthy, the multimodality of the resulting organizational network representation was manifested by the fact that nodes were both animate actors (individual user profiles), organizations (business user profiles), resources (hashtags relating to e.g., gaming consoles), and other socioeconomic phenomena (hashtags related to e.g., game brands, emotions, or general concepts). A relation implied from a tweet published by a user mentioning another user or a hashtag was deemed representational [28] since it had been announced by the publishing user. On the other hand, a relation implied from a retweeted content or from a reply was regarded as a flow, resulting in the two-dimensionality of the graph's relations. The interactions themselves were binary—their detailed classification was outside of this study's scope, but similar tasks have been exercised by other researchers.

4.2. Event Forecasting. Given the data points resulting from the event mining, which contained binary interactive events between the organizational network's nodes in time, the objective of the event forecasting was to predict future interactions in a time horizon. In other words, the implementation of the organizational network event forecasting aimed to answer the question: "will Node X interact with Node Y in the next T weeks?".

The implementation was a Python application based on the scikit-learn package, which executed a machine learning pipeline consisting of several steps described in the following subchapters. The codebase was published on GitHub and was accessible via https://github.com/PiotrSliwa/preludium17 on November 29, 2021.

4.3. Data Input. After the data points were extracted, transformed, and inserted into a database by the event mining task, the application grouped them by origin nodes to form a set of event sequences local to individual origin nodes (they could be referred to as ego-networks of the origin nodes), as presented in Figure 5. As a result, the input data could be quickly queried to find all events of a particular origin node. Such a transformation was needed by the designed predictive model (discussed in the next chapter) which accepted samples of individual origin nodes' event sequences, labeled with binary information whether they preceded target events of the same origin nodes.

4.4. Training and Test Data Split. The development of a supervised machine learning predictive model usually consists of two stages: (1) training, during which a predictor (a regressor or a classifier) is fed with a training data sample (input and expected output), and (2) testing, during which the trained predictor's performance is verified against the test data sample by feeding it with the input data, and then comparing calculated output with the real value from the training dataset. By this, one can measure how many correct and incorrect predictions the predictor made and how

Origin nodes	Events in sequence			
@UserA	e_1			
@UserB		e_2		e_4
@UserC			e_3	

Figure 5: A sample of the input data—event sequences from Figure 3 grouped by origin nodes. Source: own elaboration.

biased the mistakes were (in terms of false positives, false negatives, etc.).

The data split into the training and test data samples was done by dividing the event sequence of an individual origin node into two parts at the highest event distribution point (to balance the two samples):

$$\mu_{\max} = \max(\mu_1, \ldots, \mu_n), \tag{7}$$

where n is the number of interactive events collected in the set $E = [e_1, \ldots, e_n]$; therefore, $n = |E|$. A distribution point μ_n (or a "bin") at a given moment m, $1 \le m \le n$ is defined as the number of origin nodes v_a whose individual event sequences $[e_i, \ldots, e_j] \subset E$ contain at least one event $e_x = [v_a, v_b, i_c, \tau_x]$ with $\tau_k \le \tau_x \le \tau_l$ (mind the sequence E being ordered, so $k \le x \le l$).

Events with timestamps lower than the defined time point became the training dataset, while all events that happened after the timestamp became the test dataset (see Figure 6). This approach was chosen in favor of the traditional 20:80 split to simulate a real scenario, in which a user of the event forecasting instrument wants to forecast future events at a specific point of time (his or her "presence" at the time). The defined "splitting point" served as the hypothetical user's "presence". The predictor (discussed in the next chapter), trained on the training event sequences (the user's "past"), was then validated on the test event sequences (the user's "future") by comparing its guesses with the actual ones.

4.5. Predictor: The Machine Learning Model. The predictor for the event forecasting task used a decision tree classifier from the scikit-learn package (refer to the attached source code for details) due to its known best average performance [33]. The classifier's implementation can be trained with pairs of input-output $[X_n, y_n]$ where $X_n = [x_1, \ldots, x_i]_n$ is the input vector of features $x_1, \ldots, x_i$, and $y_n \in I$ is the output (an element from the set of interactive event types I). The classifier can be defined with a function

$$\Psi(X_n) = y_n \in Y. \tag{8}$$

The input feature vectors, according to the requirements of many machine learning algorithms, are fixed-length vectors in which a specific position represents an individual feature, and the value represents the feature's intensity reflected in a floating-point number [34]. In the experiment, the target value was an integer representing the target class, which, in this case, was a binary value referring to existence

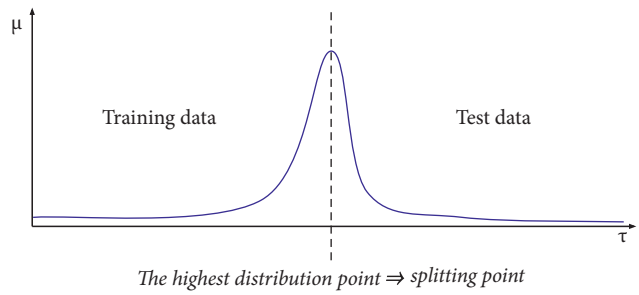


Figure 6: The distribution (μ) of origin nodes containing a timestamp (τ) in their local interactive event sequences. Source: own elaboration.

(1) or nonexistence (0) of a certain organizational network interaction, initiated by node v_a at time τ_n, between the node and a destination node v_b. Once trained, the classifier is expected to predict the existence/nonexistence of the relationship given unseen feature vectors.

However, it was first necessary to transform the input data—the event stream E resulting from the event mining—into the feature vectors X acceptable by the classifier as described by

$$\mathcal{T}(E) = X. \tag{9}$$

Therefore, a procedure herein called *event stream windowing*, described by the transformation W

$$W(E, i_c, v_a, v_b, w) = E^r \subset E, \quad r \in \{+, -\}, \tag{10}$$

was used to split the dataset into "positive" and "negative" event sequences ($E^{\text{pos}}, E^{\text{neg}} \subset E$), whereas "positive" means the ones preceding an interaction i_c, initiated by an origin node v_a, between it and a target node v_b, given a timespan window of w. The procedure drew from the event forecasting algorithm designed by Laxman et al. [26].

Compared to Laxman et al., the model accepts an event sequence, a target event type, and window size as the input, and labels subsequences of events as "positive" or "negative", depending on whether a subsequence leads to the target input type provided as the input. Different from Laxman et al.'s algorithm, though, predictions in the proposed algorithm are made by an interchangeable machine learning classifier instead of the standard frequent episode discovery algorithm, to satisfy the requirement formulated in the introduction. Furthermore, *event stream windowing* slices event streams grouped by origin nodes, instead of a single, global event sequence (a grouped event stream includes only events of a certain origin node), to mitigate the impact of potentially independent events on the predictions. It iterates over the list of grouped event streams and for each of them (a) finds all occurrences of the target event type, (b) cuts out event sequences of the declared window size which end with the target event type and labels them "positive", and (c) cuts out event sequences of the declared window size from the remaining event sequence and labels them "negative". The algorithm (summarized in Figure 7) could be characterized

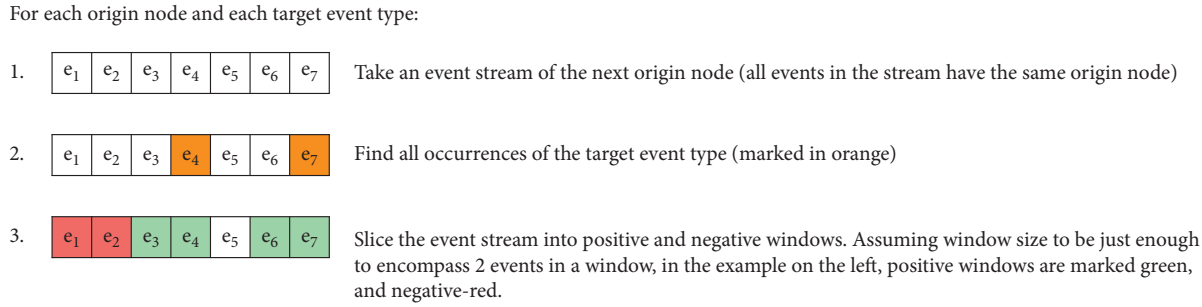


Figure 7: The proposed event stream windowing algorithm. Source: own elaboration.

as a member of the *sliding window* discrete methods (Figure 8) which are known to effectively extract information from unbounded, continuously generated sequences of data, thanks to their adaptive features [35].

The positive (E^{pos}) and negative (E^{neg}) event sequences carved out of the event stream E were then transformed into feature vectors using a vectorizing function ($\mathscr{V}$). The strategy used in this research was *counting vectorizer*—a bag-of-words model of object categorization ([36]. It transforms an event sequence E_r into a vector X_r, in which each position m is related to an interactive event $i_m \in I$, and the value of x_m represents the number of times the event type occurs in the event sequence E_r.

$$\begin{aligned} \mathscr{V}(E_r) &= X_r = [x_1, \ldots, x_m], \\ 1 &\le m \le |I|, \\ x_m &= |i_m \in e_x \in E_r|. \end{aligned} \tag{11}$$

As a result, the decision tree classifier was trained with pairs of (1) feature vectors X and (2) the corresponding expected output y as presented in Figure 9. At this point, the transformation function $\mathscr{T}$ had become a composite of the *event sequence windowing* algorithm and the *counting vectorizer*, $\mathscr{V} \circ W$, and the predictor function (defined in hypothesis H1)—a composite of the transformation function and the classifier, $\Psi \circ \mathscr{T}$. Therefore, the predictor function could be defined as

$$F = (\Psi \circ \mathscr{V} \circ W)(E, i_c, v_a, v_b, w). \tag{12}$$

Since the interactive event type i_c was assumed binary in the research, the *event stream windowing* algorithm always looked for the "existence" of the interaction between nodes v_a, v_b. Thus, the interactive event type i_c was constant $i_c = 1$.

5. Results

5.1. Input Data. 2,666,281 tweets from 200 Twitter profiles of eSports stakeholders (teams, players, and influencers) were collected. The tweets covered a period of over 11 years—the first tweet was published on 15 February 2008 (01:46:46 CEST) and the last on 7 November 2020 (21:57:55 CEST). They were processed to extract a tweet's author (origin node), all other users and hashtags mentioned by the author in the tweet (destination nodes), and an absolute timestamp determining the time point when the tweet was published.

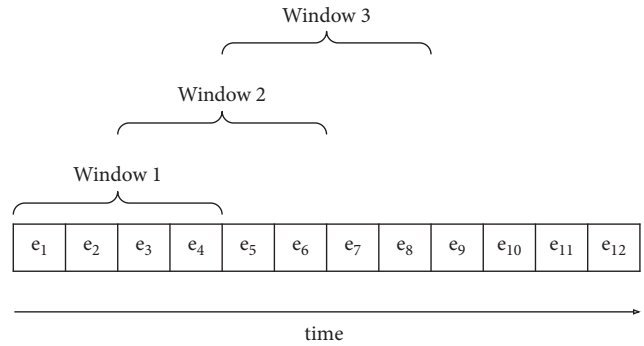


Figure 8: A sliding window algorithm using fixed-size windows (4 units wide) and offset (2 units). Source: own elaboration.

This step resulted in 3,702,773 data points describing interactive events (like depicted in Figure 4) used as the input to the predictor $F(E)$ (see Figure 9).

5.2. Training and Test Datasets. The splitting point $\mu_{\max}$ dividing the input data set into training and test (see Figure 6) was the highest distribution of origin nodes to cover the most collected profiles in the research (a profile needed both the training and the test sets to be included in the research). The moment of the highest distribution of origin nodes in the collected data was found to be 25 April 2019 (23:32:17 CEST), which covered data points from tweets published by 181 profiles. At this point, 2,978,351 data points belonged to the training set (they were extracted from tweets published before the splitting time point), and 724,422 belonged to the test set. Consequently, the research scenario simulated a scenario in which a user is forecasting network events on 25 April 2019 (23:32:17 CEST). Predicted events were then evaluated using the test data set which contained the events that factually occurred.

5.3. Forecasting Performance. The prediction pipeline was fed with the input data, first to train, and then to evaluate the predictor for each permutation of its parameters—the interaction's origin node v_a, destination node v_b, and window size w. The list of target nodes selected for the research included 1000 most popular destination nodes (the most frequent destination nodes in the data points), while the window size was arbitrarily chosen to be 8, 16, or 24 weeks

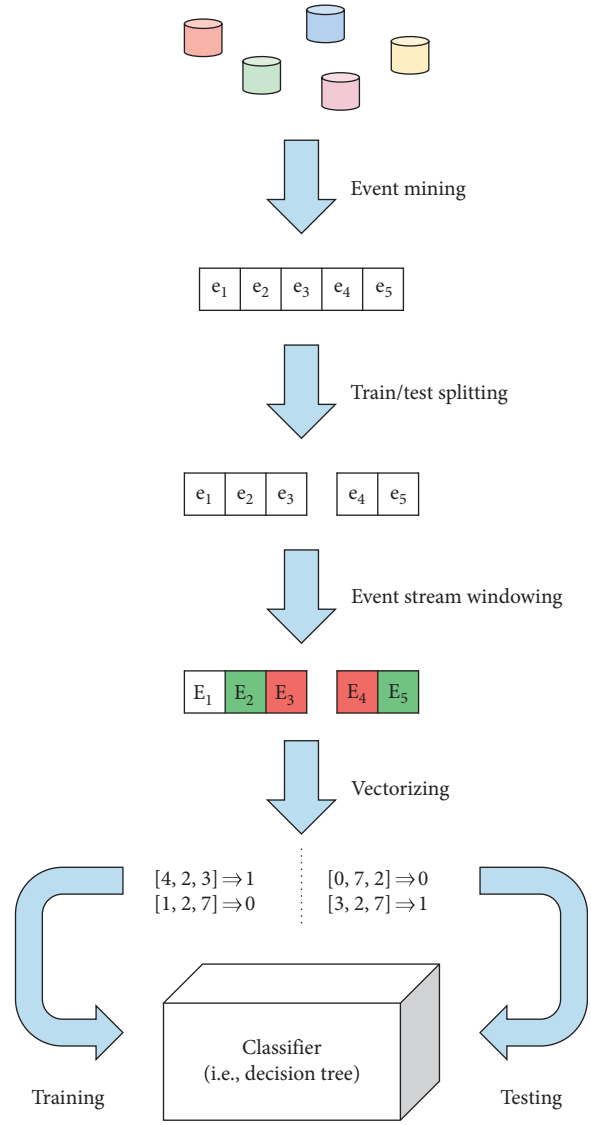


Figure 9: The prediction pipeline. Source: own elaboration.

(2, 4, or 6 months) which produced 3000 permutations, and therefore, the train and test phases were executed so many times. Each of the iterations resulted in a performance summary that included:

(1) Destination node (e.g., @FaZeClan)—a unique identifier of a destination node provided to the predictive model as a parameter. Once the model is trained, it should be able to answer the question: "Given the event sequence of Origin Node v_a, will it interact with Destination Node v_b?". Note that each of the iterations performed tests for all origin nodes v_a (i.e., all 181 Twitter profiles).

(2) Window size (e.g., 16 weeks)—a period in which the forecasted interaction is expected to occur, provided to the predictive model as a parameter. It narrows down the above question to: "Given the event sequence of Origin Node v_a, will it interact with Destination Node v_b *in the next N weeks*?".

(3) Prediction accuracy (e.g., 0.87)—a number of correctly predicted future interactions $|E_f^+|$ in all predictions.

$$P = \frac{\left|E_f^+\right|}{\left|E_f^+\right| + \left|E_f^-\right|}. \tag{13}$$

The model was first trained with the training data set and then validated with the test data set by feeding it with the input data and comparing predictions with the actual ones from the test data set. In other words, after training the model, it was asked the above question, and if the answer was congruent with the reality (the actual existence/nonexistence of the interaction corresponded with the guess), it was deemed correct.

(4) Test datasets (e.g., 1234)—a number of test data sets in the split. The greater the number, the more reliable the accuracy was deemed since it had been validated in more test scenarios.

Table 1: Aggregated statistics of all 3000 iterations containing the performance summaries.

		Window size (w)			
		24 weeks	16 weeks	8 weeks	Mean
Prediction accuracy (p)	Mean	0.921686354	0.931958548	0.949848035	0.934497646
	Standard deviation	0.07902513	0.070020294	0.055008128	0.069691226
Test datasets	Mean	764.574	1037.199	1803.051	1201.608
	Standard deviation	522.7018804	518.247742	504.0115992	677.0649846
Training datasets	Mean	2630.994	3585.465	6395.766	4204.075
	Standard deviation	1253.492722	1250.125901	1209.467827	2021.314064
Test dataset's output class ratio (R)	Mean	0.150971114	0.161718625	0.203974741	0.172221493
	Standard deviation	0.204860337	0.195071601	0.217236464	0.207121714
Training dataset's output class ratio (R)	Mean	0.253346286	0.196331516	0.120270274	0.189982692
	Standard deviation	0.202898991	0.177215054	0.130040023	0.181054278

Table 2: A selection of iterations with outstanding accuracy of predictions. Source: own elaboration.

Destination node	Window size	Prediction accuracy	Test datasets	Training datasets	Test dataset's output class ratio	Training dataset's output class ratio
@OpTic_daps	8 weeks	0.9994	1616	5938	0.0	0.04
@jmosesot	8 weeks	0.9994	1616	6802	0.0	0.18
@Immortals_gg	8 weeks	0.9994	1616	5916	0.0	0.04
@DrAvailer	8 weeks	0.9994	1616	5987	0.0	0.06
@LEC	8 weeks	0.5170	2899	5775	0.48	0.0
#BerlinMajor	8 weeks	0.5048	3029	5747	0.50	0.0
@gocr4zy	24 weeks	0.5005	929	1924	0.50	0.0
@OGesportsCSGO	24 weeks	0.4985	971	1922	0.50	0.0

Table 3: Aggregated statistics of the 237 iterations which produced the highest reliability accuracy metrics. Source: own elaboration.

		Window size			
		24 weeks	16 weeks	8 weeks	Mean
Prediction accuracy (p)	Mean	0.864547736	0.88407903	0.877279795	0.873421925
	Standard deviation	0.089712353	0.063456924	0.096425389	0.08341138
Test datasets	Mean	905.9298246	1353.316456	2639.863636	1376.970464
	Standard deviation	93.64645615	177.4148799	423.4860636	672.3086483
Training datasets	Mean	2921.535088	3882.329114	7452.909091	4083.067511
	Standard deviation	1595.176985	1149.72304	1689.871563	2226.650539
Test dataset's output class ratio (R)	Mean	0.493061243	0.485309883	0.48420594	0.488833434
	Standard deviation	56.20898166	38.33948073	21.30506137	115.8535238
Training dataset's output class ratio (R)	Mean	0.325785796	0.277650833	0.267522719	0.298924035
	Standard deviation	0.237382294	0.19001084	0.174750185	0.212513446

(5) Dataset's output class ratio (e.g., 0.47)—the number of positive output values $y^{\text{pos}} = 1$ divided by the number of negative output values $y^{\text{neg}} = 0$ in the dataset.

$$R = \frac{\left|Y^{\text{pos}}\right|}{\left|Y^{\text{neg}}\right|}, \quad y^{\text{pos}} \notin Y^{\text{neg}} \subset Y, \quad y^{\text{pos}} \notin Y^{\text{pos}} \subset Y. \tag{14}$$

In the perfect case, the ratio was expected to be 0.5 because it meant there was the same amount of positive and negative target values in the test data set. Conversely, the imaginary worst case would be with the ratio equal to 0.0 or 1.0, as it would mean there would be only positive or negative target values, and the model could reach the perfect accuracy simply by giving a fixed answer. The issue of the ratio reaching one of the extremes is known in the literature as the class imbalance problem [37].

All 3000 iterations were aggregated to determine the average prediction accuracy of the developed predictive model in the given data set. Additionally, the average (mean) accuracy $\overline{P}$ and its standard deviation σ were extended with the calculated averages (means) and standard deviations of the number of training/test data sets along with their target value ratio for reference (see Table 1).

$$\overline{P} = \frac{1}{n}\left(\sum_{i=1}^{n} P_i\right),$$
$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(P_i - \overline{P})}{n}}. \quad (15)$$

The overall prediction accuracy of 0.93 was excellent given the simple machine learning classifier (decision tree) and vectorizing strategy (*counting vectorizer*). However, it is important to notice the high level of class imbalance in the test data sets, which can hinder the reliability of the accuracy metric. Indeed, there were several iterations (see Table 2) that reached nearly perfect 1.0 accuracy. A closer look revealed that there was no positive or negative target value in the test data set (and a comparably small amount in the training data sets). It meant that the predictive model could have "cheated" by responding always with the only output class (positive or negative) present in the sample.

Such unreliable metrics seemed to artificially inflate the average accuracy of the predictive model. To mitigate it, the results were filtered by the amount of test data sets and their output value (class) ratio. The former was set to 200 (an iteration without the satisfactory amount of test data was rejected), whereas the latter was set to 0.4–0.6 (iterations with imbalanced classes in the test data were rejected). This quality threshold let 237 iterations pass, out of the original 3000, which are aggregated in Table 3.

The average accuracy of the predictive model based on the 237 most reliable iterations was found to be approximately 0.87, with a standard deviation of 0.08 and rather insignificant differences across window sizes. Thus, the prediction accuracy achieved by the developed predictive model was much higher than random guesses. The previously defined correlation metric can be computed using the results

$$C(F) = \frac{\left|E_f^+\right|}{\left|E_f^+\right| + \left|E_f^-\right|} \approx 0.87, \quad (16)$$

which supports the H1 hypothesis as $C \approx 0.87 \gg 0.5$.

6. Discussion and Conclusions

6.1. Summary and Findings. The article introduced a mathematical model of organizational network event forecasting, treating organizational networks as discrete dynamical systems, which synthesized the theories of (1) dynamic interaction graphs, (2) multimodal organizational networks, (3) organizational network mapping, and (4) event forecasting. Then, it presented an experimental evaluation of the proposed model, which resulted in an accuracy of approximately 87% correct guesses on a real longitudinal (covering 11 years) data sample. The results supported the H1 hypothesis stated in the introduction—the ratio of correctly predicted future events to all predictions made with an implementation of the model was found to be significantly higher than the one expected from random guesses, which supported the hypothesis that there is a positive correlation between sequences of past (E_p) and future (E_f) interactive events occurring in organizational networks. Therefore, the proposed organizational network event forecasting model demonstrated practical usability for event forecasting tasks in network contexts.

6.2. Limitations. Noteworthy, the components of the model used in the experiment—the *event sequence windowing* and *counting vectorizer* algorithms as well as the decision-tree-based predictor—were chosen based on theoretical reasoning. A comparative study of different algorithms was outside of this research's scope but brings an interesting case for a future investigation. Apart from the simplified algorithms used in the predictive model, a limitation of the study was the reduced organizational structure of the nodes. Namely, the research did not scrutinize the impacts of various clustering—grouping nodes together in arbitrary or otherwise determined organizations, allowed by the flat structure of the model—on the model's performance. It is hypothesized that such a procedure should improve the performance if the clusters accurately reflect real organizations and decrease otherwise. This interesting feature, if proven to be correct, could be used by researchers, for example, to (1) determine the boundaries of real organizations or (2) forecast network events of arbitrary organizations (segments, sectors, industries, countries, etc.).

Another limitation of this study was the single dimensionality of nodes and edges used in the experiment. As mentioned before, the proposed mathematical model gives freedom in defining multiple dimensions of both nodes and edges, and an analysis of dependencies between selected dimensions (e.g., dimensions of flow network and representational network) makes an interesting case for future work.

Moreover, a prospective continuation of the research, which was not in the scope of this one, is evaluating the model on different, more diverse data sets, aggregated from multiple data sources. It will be incredibly interesting to see how the model handles nonbinary interactions or how data from diverse data sources can be aggregated and translated into event streams. Even though at this point the possibilities seem countless, impacts of the increasing diversity and cardinality of the data sources are also unknown and should be analyzed in detail. Presumably, at some point, such a complex organizational network event forecasting instrument will require an informed process of noise reduction and identification of relevant features in the vast ocean of data.

6.3. Practical Applications. The proposed organizational network event forecasting model can positively impact the effectiveness of researchers and practitioners of organization and management areas who nowadays, more than ever, operate in highly interdependent and complicated network contexts. For example, businesses could use it to observe trends in the market and simulate their actions before implementing them, institutions of public health to monitor the risk of dangerous events in the society, marketing teams

to predict interactions with their products and target customers, and so forth.

Arguably, the network model can describe a wide range of the contemporary socioeconomic phenomena, which, in combination with the prospects of continuously improving tools for automated data mining and machine learning, gives a tempting promise to someday enable us to predict an abundance of socioeconomic events in our societies and businesses. This article is hoped to initiate the pursuit of this goal and pave the first steps of the long and enthralling endeavor.

Data Availability

The Python code used to support the findings of this study has been deposited in the GitHub repository (https://github.com/PiotrSliwa/preludium17). The data points used to feed the Python application and support the findings of this study are available upon request to the author due to their volume (almost 2 GB file).

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

This research was financed by the Polish National Science Centre as a part of research project no. 2019/33/N/HS4/03086.

References

[1] T. P. Moliterno, D. M. Mahony, H. Aguinis et al., "Network Theory of Organization: A Multilevel Approach," *March 2011 Bridging Micro and Macro Domains*, vol. 37, no. 2, 2010.

[2] W. W. Powell, K. W. Koput, and L. Smith-Doerr, "Interorganizational collaboration and the locus of innovation: networks of learning in biotechnology," *Administrative Science Quarterly*, vol. 41, no. 1, pp. 116–145, 1996.

[3] W. Tsai, "Knowledge transfer in intraorganizational networks: effects of network position and absorptive capacity on business unit innovation and performance," *Academy of Management Journal*, vol. 44, no. 5, pp. 996–1004, 2001.

[4] B. Uzzi, "Social structure and competition in interfirm networks: the paradox of embeddedness," *Administrative Science Quarterly*, vol. 42, no. 1, pp. 35–67, 1997.

[5] W. Tsai and S. Ghoshal, "Social capital and value creation: the role of intrafirm networks," *Academy of Management Journal*, vol. 41, no. 4, pp. 464–476, 1998.

[6] J. Walter, C. Lechner, and F. W. Kellermanns, "Knowledge transfer between and within alliance partners: private versus collective benefits of social capital," *Journal of Business Research*, vol. 60, no. 7, pp. 698–710, 2007.

[7] G. Ahuja, "Collaboration networks, structural holes, and innovation: a longitudinal study," *Administrative Science Quarterly*, vol. 45, no. 3, pp. 425–455, 2000.

[8] G. G. Bell and A. Zaheer, "Geography, networks, and knowledge flow," *Organization Science*, vol. 18, no. 6, pp. 955–972, 2007.

[9] J. Cadena, G. Korkmaz, C. J. Kuhlman, A. Marathe, N. Ramakrishnan, and A. Vullikanti, "Forecasting social unrest using activity cascades," *PLoS One*, vol. 10, no. 6, Article ID e0128879, 2015.

[10] F. Jin, W. Wang, P. Chakraborty, N. Self, F. Chen, and N. Ramakrishnan, "Tracking multiple social media for stock market event prediction," *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10357 LNAI*, Springer, Cham, Switzerland, pp. 16–30, 2017.

[11] S. K. Rumi, K. Deng, and F. D. Salim, "Crime event prediction with dynamic features," *EPJ Data Science*, vol. 7, no. 1, p. 43, 2018.

[12] R. A. W. Rhodes, "The new governance: governing without government," *Political Studies*, vol. 44, no. 4, pp. 652–667, 1996.

[13] K. G. Provan, A. Fish, and J. Sydow, "Interorganizational networks at the network level: a review of the empirical literature on whole networks," *Journal of Management*, vol. 33, no. 3, pp. 479–516, 2007.

[14] T. Ching, D. S. Himmelstein, B. K. Beaulieu-Jones et al., C. A. Lavender, S. C. Turaga, A. M. Alexandari et al., Opportunities and obstacles for deep learning in biology and medicine," *Journal of The Royal Society Interface*, vol. 15, no. 141, 2018.

[15] Y. Huo, D. F. Wong, L. M. Ni, L. S. Chao, J. Zhang, and X. Zuo, "Learning cognitive embedding using signed knowledge interaction graph," *Knowledge-Based Systems*, vol. 229, Article ID 107327, 2021.

[16] J. Li, F. Yang, M. Tomizuka, and C. Choi, "EvolveGraph: multi-agent trajectory prediction with dynamic relational reasoning," in *Proceedings of the Advances in Neural Information Processing Systems*, December 2020, https://nips.cc/Conferences/2020.

[17] W. Xie, Y. Tian, Y. Sismanis, A. Balmin, and P. J. Haas, "Dynamic interaction graphs with probabilistic edge decay," in *Proceedings of the International Conference on Data Engineering*, pp. 1143–1154, Seoul, Republic of Korea, April 2015.

[18] L. Tang, H. Liu, J. Zhang, and Z. Nazeri, "Community evolution in dynamic multi-mode networks," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 677–685, Las Vegas, NV, USA, August 2008.

[19] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*, Cambridge University Press, Cambridge, UK, 1995.

[20] D. Krackhardt and K. M. Carley, "A PCANS model of structure in organization," in *Proceedings of the 1998 International Symposium on Command and Control Research and Technology*, pp. 113–119, Monterray, CA, USA, June 1998.

[21] A. Ujwary-Gil, *Organizational network analysis*, Routledge, Oxfordshire, UK, 2019.

[22] Y. Ning, S. Muthiah, H. Rangwala, and N. Ramakrishnan, "Modeling precursors for event forecasting via nested multi-instance learning," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1095–1104, San Francisco, CA, USA, August 2016.

[23] L. Zhao, Q. Sun, J. Ye, F. Chen, C. T. Lu, and N. Ramakrishnan, "Multi-task learning for spatio-temporal event forecasting," in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1503–1512, Sydney, NSW, Australia, August 2015.

[24] J. Bollen, H. Mao, and X. Zeng, "Twitter mood predicts the stock market," *Journal of Computational Science*, vol. 2, no. 1, pp. 1–8, 2011.

[25] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, "Predicting elections with twitter: what 140 characters reveal about political sentiment," in *Proceedings of the international AAAI conference on web and social media*, vol. 4, no. 1, Washington, DC, USA, May 2010.

[26] S. Laxman, V. Tankasali, and R. W. White, "Stream Prediction Using a Generative Model Based on Frequent Episodes in Event Sequences," in *Proceedings of the KDD08: The 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Las Vegas, NV, USA, August 2008.

[27] C. Dong and H. Rim, "Exploring nonprofit-business partnerships on Twitter from a network perspective," *Public Relations Review*, vol. 45, no. 1, pp. 104–118, 2019.

[28] M. Shumate and N. Contractor, "Emergence of Multidimensional Social Networks," *The SAGE Handbook of Organizational Communication*, pp. 449–474, Sage, NewYork, NY, USA, 2013.

[29] Y. Wang and L. Guan, "Mapping the structures of international communication organizations' networks and cross-sector relationships on social media and exploring their antecedents," *Public Relations Review*, vol. 46, no. 4, Article ID 101951, 2020.

[30] P. Sliwa, G. Krzos, and E. Piwoni-Krzeszowska, "Digital Network Twin - mapping socio-economic networks into the virtual reality," *Transformations in Business and Economics*, vol. 20, no. 2B, pp. 989–1004, 2021.

[31] T. P. Moliterno and D. M. Mahony, "Network theory of organization: a multilevel approach," *Journal of Management*, vol. 37, no. 2, pp. 443–467, 2011.

[32] S. Zhang, S. Bahrampour, N. Ramakrishnan, L. Schott, and M. Shah, "Deep learning on symbolic representations for large-scale heterogeneous time-series event prediction," in *Proceedings of the ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5970–5974, New Orleans, LA, USA, March 2017.

[33] R. Caruana and A. Niculescu-Mizil, "An empirical comparison of supervised learning algorithms," in *Proceedings of the ACM International Conference Proceeding Series*, vol. 148, pp. 161–168, Pittsburgh, PA, USA, June 2006.

[34] Q. Le and T. Mikolov, "Distributed representations of sentences and documents," in *Proceedings of the Thirty First International Conference on Machine Learning, ICML 2014*, vol. 4, pp. 2931–2939, Beijing, China, June 2014.

[35] J. H. Chang and W. S. Lee, "A sliding window method for finding recently frequent itemsets over online data streams," *Journal of Information Science and Engineering*, vol. 20, no. 4, pp. 753–762, 2004.

[36] Y. Zhang, R. Jin, and Z. H. Zhou, "Understanding bag-of-words model: a statistical framework," *International Journal of Machine Learning and Cybernetics*, vol. 1, no. 1–4, pp. 43–52, 2010.

[37] K. Napierala and J. Stefanowski, "Types of minority class examples and their influence on learning classifiers from imbalanced data," *Journal of Intelligent Information Systems*, vol. 46, no. 3, pp. 563–597, 2016.