nature machine intelligence

Analysis

https://doi.org/10.1038/s42256-023-00735-0

Forecasting the future of artificial intelligence with machine learning-based link prediction in an exponentially growing knowledge network

Received: 21 January 2023

Accepted: 11 September 2023

Published online: 16 October 2023

Check for updates

Mario Krenn[1], Lorenzo Buffoni[2], Bruno Coutinho[2], Sagi Eppel[3], Jacob Gates Foster[4], Andrew Gritsevskiy[3,5,6], Harlin Lee[4], Yichao Lu[7], João P. Moutinho[2], Nima Sanjabi[8], Rishi Sonthalia[4], Ngoc Mai Tran[9], Francisco Valente[10], Yangxinyu Xie[11], Rose Yu[12] & Michael Kopp[6]

A tool that could suggest new personalized research directions and ideas by taking insights from the scientific literature could profoundly accelerate the progress of science. A field that might benefit from such an approach is artificial intelligence (AI) research, where the number of scientific publications has been growing exponentially over recent years, making it challenging for human researchers to keep track of the progress. Here we use AI techniques to predict the future research directions of AI itself. We introduce a graph-based benchmark based on real-world data—the Science4Cast benchmark, which aims to predict the future state of an evolving semantic network of AI. For that, we use more than 143,000 research papers and build up a knowledge network with more than 64,000 concept nodes. We then present ten diverse methods to tackle this task, ranging from pure statistical to pure learning methods. Surprisingly, the most powerful methods use a carefully curated set of network features, rather than an end-to-end AI approach. These results indicate a great potential that can be unleashed for purely ML approaches without human knowledge. Ultimately, better predictions of new future research directions will be a crucial component of more advanced research suggestion tools.

The corpus of scientific literature grows at an ever-increasing speed. Specifically, in the field of artificial intelligence (AI) and machine learning (ML), the number of papers every month is growing exponentially with a doubling rate of roughly 23 months (Fig. 1). Simultaneously, the AI community is embracing diverse ideas from many disciplines such as mathematics, statistics and physics, making it challenging to organize different ideas and uncover new scientific connections. We envision a computer program that can automatically read, comprehend and act on AI literature. It can predict and suggest meaningful research ideas that transcend individual knowledge and cross-domain boundaries.

[1]Max Planck Institute for the Science of Light (MPL), Erlangen, Germany. [2]Instituto de Telecomunicações, Lisbon, Portugal. [3]University of Toronto, Toronto, Ontario, Canada. [4]University of California Los Angeles, Los Angeles, CA, USA. [5]Cavendish Laboratories, Cavendish, VT, USA. [6]Institute of Advanced Research in Artificial Intelligence (IARAI), Vienna, Austria. [7]Alpha 8 AI, Toronto, Ontario, Canada. [8]Independent Researcher, Barcelona, Spain. [9]University of Texas at Austin, Austin, TX, USA. [10]Independent Researcher, Leiria, Portugal. [11]University of Pennsylvania, Philadelphia, PA, USA. [12]University of California, San Diego, CA, USA. e-mail: mario.krenn@mpl.mpg.de

https://doi.org/10.1038/s42256-023-00735-0

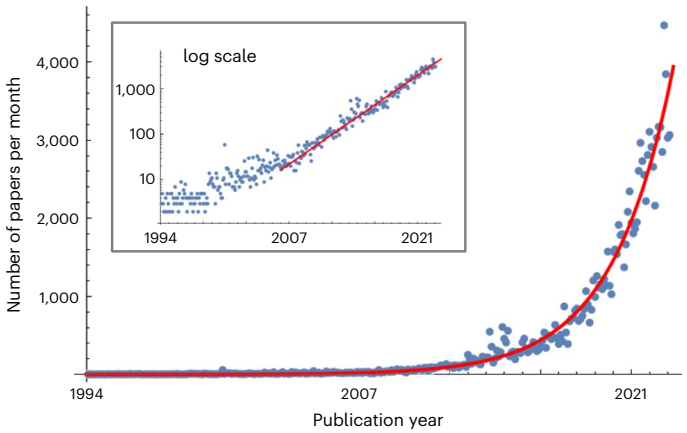


Fig. 1 | The number of papers published per month in the arXiv categories of AI and ML are growing exponentially. The doubling rate of papers per month is roughly 23 months, which might lead to problems for publishing in these fields, at some point. The categories are cs.AI, cs.LG, cs.NE and stat.ML.

If successful, it could greatly improve the productivity of AI researchers, open up new avenues of research and help drive progress in the field.

In this work, we address the ambitious vision of developing a data-driven approach to predict future research directions[1]. As new research ideas often emerge from connecting seemingly unrelated concepts[2–4], we model the evolution of AI literature as a temporal network. We construct an evolving semantic network that encapsulates the content and development of AI research since 1994, with approximately 64,000 nodes (representing individual concepts) and 18 million edges (connecting jointly investigated concepts).

We use the semantic network as an input to ten diverse statistical and ML methods to predict the future evolution of the semantic network with high accuracy. That is, we can predict which combinations of concepts AI researchers will investigate in the future. Being able to predict what scientists will work on is a first crucial step for suggesting new topics that might have a high impact.

Several methods were contributions to the Science4Cast competition hosted by the 2021 IEEE International Conference on Big Data (IEEE BigData 2021). Broadly, we can divide the methods into two classes: methods that use hand-crafted network-theoretical features and those that automatically learn features. We found that models using carefully hand-crafted features outperform methods that attempt to learn features autonomously. This (somewhat surprising) finding indicates a great potential for improvements of models free of human priors.

Our paper introduces a real-world graph benchmark for AI, presents ten methods for solving it, and discusses how this task contributes to the larger goal of AI-driven research suggestions in AI and other disciplines. All methods are available at GitHub[5].

Semantic networks

The goal here is to extract knowledge from the scientific literature that can subsequently be processed by computer algorithms. At first glance, a natural first step would be to use large language model (such as GPT3[6], Gopher[7], MegaTron[8] or PaLM[9]) on each article to extract concepts and their relations automatically. However, these methods still struggle in reasoning capabilities[10,11]; thus, it is not yet directly clear how these models can be used for identifying and suggesting new ideas and concept combinations.

Rzhetsky et al.[12] pioneered an alternative approach, creating semantic networks in biochemistry from co-occurring concepts in scientific papers. There, nodes represent scientific concepts, specifically biomolecules, and are linked when a paper mentions both in its title or abstract. This evolving network captures the field's history and, using supercomputer simulations, provides insights into scientists' collective behaviour and suggests more efficient research strategies[13]. Although creating semantic networks from concept co-occurrences extracts only a small amount of knowledge from each paper, it captures non-trivial and actionable content when applied to large datasets[2,4,13–15]. PaperRobot extends this approach by predicting new links from large medical knowledge graphs and formulating new ideas in human language as paper drafts[16].

This approach was applied and extended to quantum physics[17] by building a semantic network of over 6,000 concepts. There, the authors (including one of us) formulated the prediction of new research trends and connections as an ML task, with the goal of identifying concept pairs not yet jointly discussed in the literature but likely to be investigated in the future. This prediction task was one component for personalized suggestions of new research ideas.

Link prediction in semantic networks

We formulate the prediction of future research topics as a link-prediction task in an exponentially growing semantic network in the AI field. The goal is to predict which unconnected nodes, representing scientific concepts not yet jointly researched, will be connected in the future.

Link prediction is a common problem in computer science, addressed with classical metrics and features, as well as ML techniques. Network theory-based methods include local motif-based approaches[18–22], linear optimization[23], global perturbations[24] and stochastic block models[25]. ML works optimized a combination of predictors[26], with further discussion in a recent review[27].

In ref. 17, 17 hand-crafted features were used for this task. In the Science4Cast competition, the goal was to find more precise methods for link-prediction tasks in semantic networks (a semantic network of AI that is ten times larger than the one in ref. 17).

Potential for idea generation in science

The long-term goal of predictions and suggestions in semantic networks is to provide new ideas to individual researchers. In a way, we hope to build a creative artificial muse in science[28]. We can bias or constrain the model to give topic suggestions that are related to the research interest of individual scientists, or a pair of scientists to suggest topics for collaborations in an interdisciplinary setting.

Generation and analysis of the dataset

Dataset construction

We create a dynamic semantic network using papers published on arXiv from 1992 to 2020 in the categories cs.AI, cs.LG, cs.NE and stat.ML. The 64,719 nodes represent AI concepts extracted from 143,000 paper titles and abstracts using Rapid Automatic Keyword Extraction (RAKE) and normalized via natural language processing (NLP) techniques and custom methods[29]. Although high-quality taxonomies such as the Computer Science Ontology (CSO) exist[30,31], we choose not to use them for two reasons: the rapid growth of AI and ML may result in new concepts not yet in the CSO, and not all scientific domains have high-quality taxonomies like CSO. Our goal is to build a scalable approach applicable to any domain of science. However, future research could investigate merging these approaches (see 'Extensions and future work').

Concepts form the nodes of the semantic network, and edges are drawn when concepts co-appear in a paper title or abstract. Edges have time stamps based on the paper's publication date, and multiple time-stamped edges between concepts are common. The network is edge-weighted, and the weight of an edge stands for the number of papers that connect two concepts. In total, this creates a time-evolving semantic network, depicted in Fig. 2.

Network-theoretical analysis

The published semantic network has 64,719 nodes and 17,892,352 unique undirected edges, with a mean node degree of 553.

https://doi.org/10.1038/s42256-023-00735-0

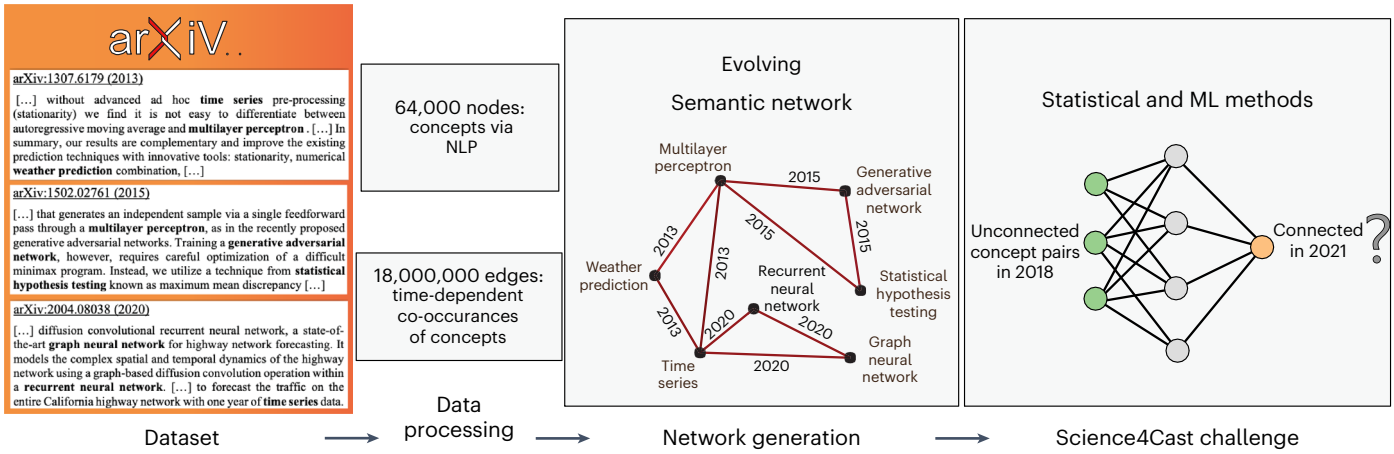


Fig. 2 | From arXiv to Science4Cast. Utilizing 143,000 AI and ML papers on arXiv from 1992 to 2020, we create a list of concepts using RAKE and other NLP tools, which form nodes in a semantic network. Edges connect concepts that co-occur in titles or abstracts, resulting in an evolving network that expands as more concepts are jointly investigated. The task involves predicting which unconnected nodes (concepts not yet studied together) will connect within a few years. We present ten diverse statistical and ML methods to address this challenge.

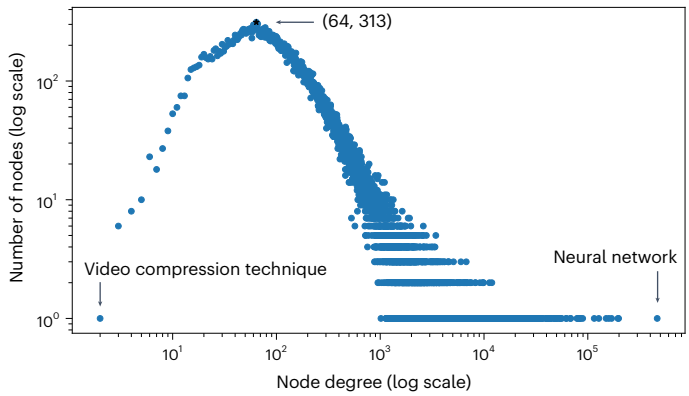


Fig. 3 | Heavy-tail distribution in node degrees due to hubs. Nodes with the highest (466,319) and lowest (2) non-zero degrees are neural network and video compression technique, respectively. The most frequent non-zero degree is 64 (which occures 313 times). The plot, in log scale, omits 1,247 nodes with zero degrees.

Many hub nodes greatly exceed this mean degree, as shown in Fig. 3, For example, the highest node degrees are 466,319 (neural network), 198,050 (deep learning), 195,345 (machine learning), 169,555 (convolutional neural network), 159,403 (real world), 150,227 (experimental result), 127,642 (deep neural network) and 115,334 (large scale). We fit a power-law curve to the degree distribution $p(k)$ using ref. 32 and obtained $p(k) \propto k^{-2.28}$ for degree $k \geq 1{,}672$. However, real complex network degree distributions often follow power laws with exponential cut-offs[33]. Recent work[34] has indicated that lognormal distributions fit most real-world networks better than power laws. Likelihood ratio tests from ref. 32 suggest truncated power law ($P = 0.0031$), lognormal ($P = 0.0045$) and lognormal positive ($P = 0.015$) fit better than power law, while exponential ($P = 3 \times 10^{-10}$) and stretched exponential ($P = 6 \times 10^{-5}$) are worse. We couldn't conclusively determine the best fit with $P \leq 0.1$.

We observe changes in network connectivity over time. Although degree distributions remained heavy-tailed, the ordering of nodes within the tail changed due to popularity trends. The most connected nodes and the years they became so include decision tree (1994), machine learning (1996), logic program (2000), neural network (2005), experimental result (2011), machine learning (2013, for a second time) and neural network (2015).

Connected component analysis in Fig. 4 reveals that the network grew more connected over time, with the largest group expanding and the number of connected components decreasing. Mid-sized connected components' trajectories may expose trends, like image processing. A connected component with four nodes appeared in 1999 (brightness change, planar curve, local feature, differential invariant), and three more joined in 2000 (similarity transformation, template matching, invariant representation). In 2006, a paper discussing support vector machine and local feature merged this mid-sized group with the largest connected component.

The semantic network reveals increasing centralization over time, with a smaller percentage of nodes (concepts) contributing to a larger fraction of edges (concept combinations). Figure 5 shows that the fraction of edges for high-degree nodes rises, while it decreases for low-degree nodes. The decreasing average clustering coefficient over time supports this trend, suggesting nodes are more likely to connect to high-degree central nodes. This could be due to the AI community's focus on a few dominating methods or more consistent terminology use.

Problem formulation

At the big picture, we aim to make predictions in an exponentially growing semantic network. The specific task involves predicting which two nodes v_1 and v_2 with degrees $d(v_{1/2}) \geq c$ lacking an edge in the year $(2021 - \delta)$ will have w edges in 2021. We use $\delta = 1, 3, 5$, $c = 0, 5, 25$ and $w = 1, 3$, where c is a minimal degree. Note that $c = 0$ is an intriguing special case where the nodes may not have an associated edge in the initial year, requiring the model to predict which nodes will connect to entirely new edges. The task $w = 3$ goes beyond simple link prediction and seeks to identify uninvestigated concept pairs that will appear together in at least three papers. An interesting alternative task could be predicting the fastest-growing links, denoted as 'trend' prediction.

In this task, we provide a list of 10 million unconnected node pairs (each node having a degree $\geq c$) for the year $(2021 - \delta)$, with the goal of sorting this list by descending probability that they will have at least w edges in 2021.

For evaluation, we employ the receiver operating characteristic (ROC) curve[35], which plots the true-positive rate against the false-positive rate at various threshold settings. We use the area under the curve (AUC) of the ROC curve as our evaluation metric. The advantage of AUC over mean square error is its independence from the data distribution. Specifically, in our case, where the two classes have a highly asymmetric distribution (with only about 1–3% of newly

https://doi.org/10.1038/s42256-023-00735-0

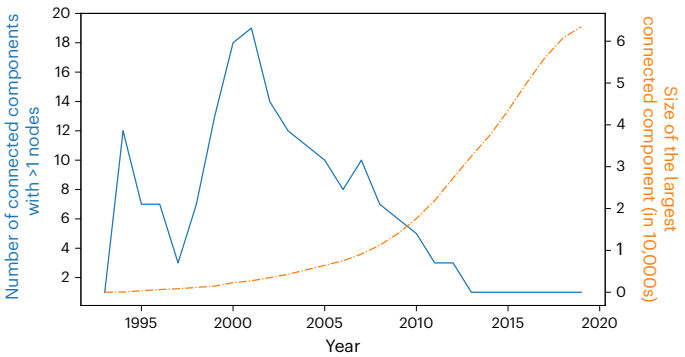


Fig. 4 | The network became more connected over the years. Primary (left, blue) vertical axis: number of connected components with more than one node. Secondary (right, orange) vertical axis: number of nodes in the largest connected component. For example, the network in 2019 comprises of one large connected component with 63,472 nodes and 1,247 isolated nodes, that is, nodes with no edges. However, the 2001 network has 19 connected components with size greater than one, the largest of which has 2,733 nodes.

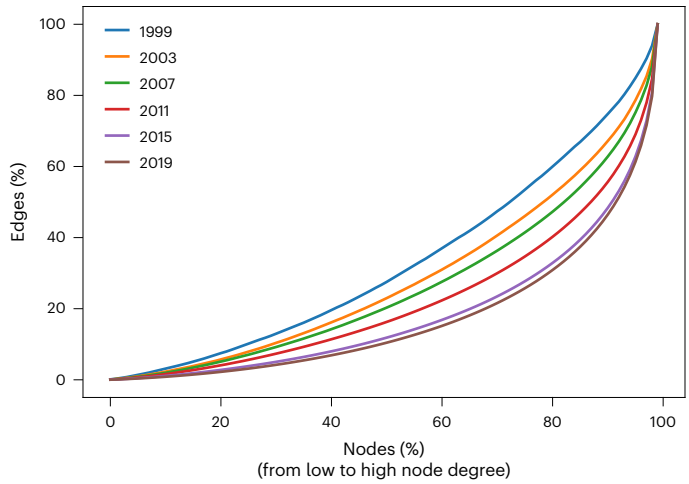


Fig. 5 | Centralization of concepts. This cumulative histogram illustrates the fraction of nodes (concepts) corresponding to the fraction of edges (connections) for given years (1999, 2003, 2007, 2011, 2015 and 2019). The graph was generated by adding edges and nodes dated before each year. Nodes are sorted by increasing degrees. The y value at $x = 80$ represents the fraction of edges contributed by all nodes in and below the 80th percentile of degrees.

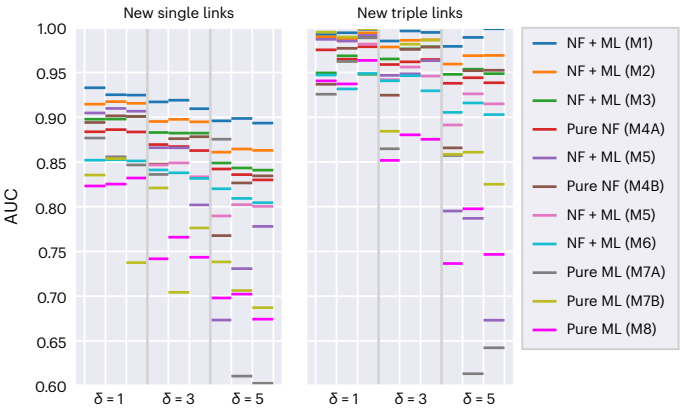


Fig. 6 | Predictions of new concept pair links in an exponentially growing semantic network. Here we show the AUC values for different models that use machine learning techniques (ML), hand-crafted network features (NF) or a combination thereof. The left plot shows results for the prediction of a single new link (that is, $w = 1$) and the right plot shows the results for the prediction of new triple links $w = 3$. The task is to predict $\delta = [1, 3, 5]$ years into the future, with cut-off values $c = [0, 5, 25]$. We sort the models by the the results for the task ($w = 1$, $\delta = 3$, $c = 0$), which was the task in the Science4Cast competition. Data points that are not shown have a AUC below 0.6 or are not computed due to computational costs. All AUC values reported are computed on a validation dataset δ years ahead of the training dataset that the models have never seen. Note that the prediction of new triple edges can be performed nearly deterministic. It will be interesting to understand the origin of this quasi-deterministic pattern in AI research, for example, by connecting it to the research interests of scientists[88].

connected edges) and the distribution changes over time, AUC offers meaningful interpretation. Perfect predictions yield AUC = 1, whereas random predictions result in AUC = 0.5. AUC represents the percentage that a random true element is ranked higher than a random false one. For other metrics, see ref. 36.

To tackle this task, models can use the complete information of the semantic network from the year ($2021 - \delta$) in any way possible. In our case, all presented models generate a dataset for learning to make predictions from ($2021 - 2\delta$) to ($2021 - \delta$). Once the models successfully complete this task, they are applied to the test dataset to make predictions from ($2021 - \delta$) to 2021. All reported AUCs are based on the test dataset. Note that solving the test dataset is especially challenging due to the δ-year shift, causing systematic changes such as the number of papers and density of the semantic network.

AI-based solutions

We demonstrate various methods to predict new links in a semantic network, ranging from pure statistical approaches and neural networks with hand-crafted features (NF) to ML models without NF. The results are shown in Fig. 6, with the highest AUC scores achieved by methods using NF as ML model inputs. Pure network features without ML are competitive, while pure ML methods have yet to outperform those with NF. Predicting links generated at least three times can achieve a quasi-deterministic AUC > 99.5%, suggesting an interesting target for computational sociology and science of science research. We have performed numerous tests to exclude data leakage in the benchmark dataset, overfitting or data duplication both in the set of articles and the set of concepts. We rank methods based on their performance, with model M1 as the best performing and model M8 as the least effective (for the prediction of a new edge with $\delta = 3$, $c = 0$). Models M4 and M7 are subdivided into M4A, M4B, M7A and M7B, differing in their focus on feature or embedding selection (more details in Methods).

Model M1: NF + ML. This approach combines tree-based gradient boosting with graph neural networks, using extensive feature engineering to capture node centralities, proximity and temporal evolution[37]. The Light Gradient Boosting Machine (LightGBM) model[38] is employed with heavy regularization to combat overfitting due to the scarcity of positive examples, while a time-aware graph neural network learns dynamic node representations.

Model M2: NF + ML. This method utilizes node and edge features (as well as their first and second derivatives) to predict link formation probabilities[39]. Node features capture popularity, and edge features measure similarity. A multilayer perceptron with rectified linear unit (ReLU) activation is used for learning. Cold start issues are addressed with feature imputation.

Model M3: NF + ML. This method captures hand-crafted node features over multiple time snapshots and employs a long short-term memory (LSTM) to learn time dependencies[40]. The features were selected to be highly informative while having a low computational cost. The final configuration uses degree centrality, degree of neighbours and common neighbours as features. The LSTM outperforms fully connected neural networks.

Model M4: pure NF. Two purely statistical methods, preferential attachment[41] and common neighbours[27], are used[42]. Preferential attachment is based on node degrees, while common neighbours relies on the number of shared neighbours. Both methods are computationally inexpensive and perform competitively with some learning-based models.

Model M5: NF + ML. Here, ten groups of first-order graph features are extracted to obtain neighbourhood and similarity properties, with principal component analysis[43] applied for dimensionality reduction[44]. A random forest classifier is trained on the balanced dataset to predict new links.

Model M6: NF + ML. The baseline solution uses 15 hand-crafted features as input to a four-layer neural network, predicting the probability of link formation between node pairs[17].

Model M7: end-to-end ML (auto node embedding). The baseline solution is modified to use node2vec[45] and ProNE embeddings[46] instead of hand-crafted features. The embeddings are input to a neural network with two hidden layers for link prediction.

Model M8: end-to-end ML (transformers). This method learns features in an unsupervised manner using transformers[47]. Node2vec embeddings[45,48] are generated for various snapshots of the adjacency matrix, and a transformer model[49] is pre-trained as a feature extractor. A two-layer ReLU network is used for classification.

Extensions and future work

Developing an AI that suggests research topics to scientists is a complex task, and our link-prediction approach in temporal networks is just the beginning. We highlight key extensions and future work directly related to the ultimate goal of AI for AI.

High-quality predictions without feature engineering. Interestingly, the most effective methods utilized carefully crafted features on a graph with extracted concepts as nodes and edges representing their joint publication history. Investigating whether end-to-end deep learning can solve tasks without feature engineering will be a valuable next step.

Fully automated concept extraction. Current concept lists, generated by RAKE's statistical text analysis, demand time-consuming code development to address irrelevant term extraction (for example, verbs, adjectives). A fully automated NLP technique that accurately extracts meaningful concepts without manual code intervention would greatly enhance the process.

Leveraging ontology taxonomies. Alongside fully automated concept extraction, utilizing established taxonomies such as the CSO[30,31], Wikipedia-extracted concepts, book indices[17] or PhySH key phrases is crucial. Although not comprehensive for all domains, these curated datasets often contain hierarchical and relational concept information, greatly improving prediction tasks.

Incorporating relation extraction. Future work could explore relation extraction techniques for constructing more accurate, sparser semantic networks. By discerning and classifying meaningful concept relationships in abstracts[50,51], a refined AI literature representation is attainable. Using NLP tools for entity recognition, relationship identification and classification, this approach may enhance prediction performance and novel research direction identification.

Generation of new concepts. Our work predicts links between known concepts, but generating new concepts using AI remains a challenge. This unsupervised task, as explored in refs. 52,53, involves detecting concept clusters with dynamics that signal new concept formation. Incorporating emerging concepts into the current framework for suggesting research topics is an intriguing future direction.

Semantic information beyond concept pairs. Currently, abstracts and titles are compressed into concept pairs, but more comprehensive information extraction could yield meaningful predictions. Exploring complex data structures such as hypergraphs[54] may be computationally demanding, but clever tricks could reduce complexity, as shown in ref. 55. Investigating sociological factors or drawing inspiration from material science approaches[56] may also improve prediction tasks. A recent dataset for the study of the science of science also includes more complex data structures than the ones used in our paper, including data from social networks such as Twitter[57].

Predictions of scientific success. While predicting new links between concepts is valuable, assessing their potential impact is essential for high-quality suggestions. Introducing a metric of success, like estimated citation numbers or citation growth rate, can help gauge the importance of these connections. Adapting citation prediction techniques from the science of science[58–61] to semantic networks offers a promising research direction.

Anomaly detections. Predicting likely connections may not align with finding surprising research directions. One method for identifying surprising suggestions involves constraining cosine similarity between vertices[62], which measures shared neighbours and can be associated with semantic (dis)similarity. Another approach is detecting anomalies in semantic networks, which are potential links with extreme properties[63,64]. While scientists often focus on familiar topics[3,4], greater impact results from unexpected combinations of distant domains[12], encouraging the search for surprising associations.

End-to-end formulation. Our method breaks down the goal of extracting knowledge from scientific literature into subtasks, contrasting with end-to-end deep learning that tackles problems directly without subproblems[65,66]. End-to-end approaches have shown great success in various domains[67–69]. Investigating whether such an end-to-end solution can achieve similar success in our context would be intriguing.

Conclusion

Our method represents a crucial step towards developing a tool that can assist scientists in uncovering novel avenues for exploration. We are confident that our outlined ideas and extensions pave the way for achieving practical, personalized, interdisciplinary AI-based suggestions for new impactful discoveries. We firmly believe that such a tool holds the potential to become a influential catalyst, transforming the way scientists approach research questions and collaborate in their respective fields.

Methods

Details on concept set generation and application

In this section, we provide details on the generation of our list of 64,719 concepts. For more information, the code is accessible on GitHub. The entire approach is designed for immediate scalability to other domains.

Initially, we utilized approximately 143,000 arXiv papers from the categories cs.AI, cs.LG, cs.NE and stat.ML spanning 1992 to 2020. The omission of earlier data has a negligible effect on our research question, as we show below. We then iterated over each individual article, employing RAKE (with an extended stopword list) to suggest concept candidates, which were subsequently stored.

Following the iteration, we retained concepts composed of at least two words (for example, neural network) appearing in six or more articles, as well as concepts comprising a minimum of three words (for example, recurrent neural network) appearing in three or more articles. This initial filter substantially reduced noise generated by RAKE, resulting in a list of 104,948 concepts.

Lastly, we developed an automated filtering tool to further enhance the quality of the concept list. This tool identified common, domain-independent errors made by RAKE, which primarily included phrases that were not concepts (for example, dataset provided or discuss open challenge). We compiled a list of 543 words not part of meaningful concepts, including verbs, ordinal numbers, conjunctions and adverbials. Ultimately, this process produced our final list of 64,719 concepts employed in our study. No further semantic concept/entity linking is applied.

By this construction, the test sets with $c = 0$ could lead to very rare contamination of the dataset. That is because each concept will have at least one edge in the final dataset. The effects, however, are negligible.

The distribution of concepts in the articles can be seen in Extended Data Fig. 1. As an example, we show the extraction of concepts from five randomly chosen papers:

- Memristor hardware-friendly reinforcement learning[70]: 'actor critic algorithm', 'neuromorphic hardware implementation', 'hardware neural network', 'neuromorphic hardware system', 'neural network', 'large number', 'reinforcement learning', 'case study', 'pre training', 'training procedure', 'complex task', 'high performance', 'classical problem', 'hardware implementation', 'synaptic weight', 'energy efficient', 'neuromorphic hardware', 'control theory', 'weight update', 'training technique', 'actor critic', 'nervous system', 'inverted pendulum', 'explicit supervision', 'hardware friendly', 'neuromorphic architecture', 'hardware system'.
- Automated deep learning analysis of angiography video sequences for coronary artery disease[71]: 'deep learning approach', 'coronary artery disease', 'deep learning analysis', 'traditional image processing', 'deep learning', 'image processing', 'f1 score', 'video sequence', 'error rate', 'automated analysis', 'coronary artery', 'vessel segmentation', 'key frame', 'visual assessment', 'analysis method', 'analysis pipeline', 'coronary angiography', 'geometrical analysis'.
- Demographic influences on contemporary art with unsupervised style embeddings[72]: 'classification task', 'social network', 'data source', 'visual content', 'graph network', 'demographic information', 'social connection', 'visual style', 'historical dataset', 'novel information'
- The utility of general domain transfer learning for medical language tasks[73]: 'natural language processing', 'long short term memory', 'logistic regression model', 'transfer learning technique', 'short term memory', 'average f1 score', 'class classification model', 'domain transfer learning', 'weighted average f1 score', 'medical natural language processing', 'natural language process', 'transfer learning', 'f1 score', 'natural language', 'deep model', 'logistic regression', 'model performance', 'classification model', 'text classification', 'regression model', 'nlp task', 'short term', 'medical domain', 'weighted average', 'class classification', 'bert model', 'language processing', 'biomedical domain', 'domain transfer', 'nlp model', 'main model', 'general domain', 'domain model', 'medical text'.
- Fast neural architecture construction using envelopenets[74]: 'neural network architecture', 'neural architecture search', 'deep network architecture', 'image classification problem', 'neural architecture search method', 'neural network', 'reinforcement learning', 'deep network', 'image classification', 'objective function', 'network architecture', 'classification problem', 'evolutionary algorithm', 'neural architecture', 'base network', 'architecture search', 'training epoch', 'search method', 'image class', 'full training', 'automated search', 'generated network', 'constructed network', 'gpu day'.

Time gap between the generation of edges

We use articles from arXiv, which only goes back to the year 1992. However, of course, the field of AI exists at least since the 1960s[75]. Thus, this raises the question whether the omission of the first 30–40 years of research has a crucial impact in the prediction task we formulate, specifically, whether edges that we consider as new might not be so new after all. Thus, in Extended Data Fig. 2, we compute the time between the formation of edges between the same concepts, taking into account all or just the first edge. We see that the vast majority of edges are formed within short time periods, thus the effect of omission of early publication has a negligible effect for our question. Of course, different questions might be crucially impacted by the early data; thus, a careful choice of the data source is crucial[61].

Table 1 | Positive examples within the 10 million evaluation examples

c	δ=1	δ=3	δ=5
0	20,990	49,548	64,027
5	21,430	61,347	107,654
25	22,713	75,039	153,304
c	**δ=1**	**δ=3**	**δ=5**
0	156	2,022	4,004
5	155	2,444	7,447
25	187	3,051	11,591

Results for ω=1 (top) and ω=3 (bottom).

Table 2 | Feature importance of the model M1

Features	Average AUC
All features	0.9526
Remove *u*/*v*	0.9519
Remove jaccard_index	0.9508
Remove jaccard_index_diff	0.9511
Remove pagerank_score	0.9512
Remove pagerank_score_diff	0.9515
Remove rank_num_neighbors	0.9513
Remove rank_num_neighbors_diff	0.9502
Remove all temporal features	0.9489

As a sanity check of the winning model M1, we compute its average over all 18 datasets, for different removed features. This includes ω=1 at the left side of Fig. 6 and ω=3 at the right side of Fig. 6. As expected, the model with all features achieves the largest value of the average AUC.

Positive examples in the test dataset

Table 1 shows the number of positive cases within the 10 million examples in the 18 test datasets that are used for evaluation.

Publication rates in quantum physics

Another field of research that gained a lot of attention in the recent years is quantum physics. This field is also a strong adopter of arXiv. Thus, we analyse in the same way as for AI in Fig. 1. We find in Extended Data Fig. 3 no obvious exponential increase in papers per month. A detailed analysis of other domains is beyond the current scope. It will be interesting to investigate the growth rates in different scientific disciplines in more detail, especially given that exponential increase has been observed in several aspects of the science of science[3,76].

Details on models M1–M8

What follows are more detailed explanations of the models presented in the main text. All codes are available at GitHub. The feature importance of the best model M1 is shown here, those of other models are analysed in the respective workshop contributions (cited in the subsections).

Details on M1. The best-performing solution is based on a blend of a tree-based gradient boosting approach and a graph neural network approach[37]. Extensive feature engineering was conducted to capture the centralities of the nodes, the proximity between node pairs and their evolution over time. The centrality of a node is captured by the number of neighbours and the PageRank score[77], while the proximity between a node pair is derived using the Jaccard index. We refer the reader to ref. 37 for the list of all features and their feature importance.

The tree-based gradient boosting approach uses LightGBM[38] and applies heavy regularization to combat overfitting due to the scarcity of positive samples. The graph neural network approach employs a time-aware graph neural network to learn node representations on dynamic semantic networks. The feature importance of model M1, averaged over 18 datasets, is shown in Table 2. It shows that the temporal features do contribute largely to the model performance, but the model remains strong even when they are removed. An example of the evolution of the training (from 2016 to 2019) and test set (2019 to 2021) for $\delta = 3$, $c = 25$, $\omega = 1$ is shown in Extended Data Fig. 4.

Details on M2. The second method assumes that the probability that nodes u and v form an edge in the future is a function of the node features $f(u)$, $f(v)$ and some edge feature $h(u, v)$. We chose node features f that capture popularity at the current time t_0 (such as degree, clustering coefficient[78,79] and PageRank[77]). We also use these features' first and second time derivatives to capture the evolution of the node's popularity over time. After variable selection during training, we chose h to consist of the HOP-rec score (high-order proximity for implicit recommendation)[80,81] and a variation of the Dice similarity score[82] as a measure of similarity between nodes. In summary, we use 31 node features for each node, and two edge features, which gives $31 \times 2 + 2 = 64$ features in total. These features are then fed into a small multilayer perceptron (5 layers, each with 13 neurons) with ReLU activation.

Cold start is the problem that some nodes in the test set do not appear in the training set. Our strategy for a cold start is imputation. We say a node v is seen if it appeared in the training data, and unseen otherwise; similarly, we say that a node is born at time t if t is the first time stamp where an edge linking this node has appeared. The idea is that an unseen node is simply a node born in the future, so its features should look like a recently born node in the training set. If a node is unseen, then we impute its features as the average of the features of the nodes born recently. We found that with imputation during training, the test AUC scores across all models consistently increased by about 0.02. For a complete description of this method, we refer the reader to ref. 39.

Details on M3. This approach, detailed in ref. 40, uses hand-crafted node features that have been captured in multiple time snapshots (for example, every year) and then uses an LSTM to benefit from learning the time dependencies of these features. The final configuration uses two main types of feature: node features including degree and degree of neighbours, and edge features including common neighbours. In addition, to balance the training data, the same number of positive and negative instances have been randomly sampled and combined.

One of the goals was to identify features that are very informative with a very low computational cost. We found that the degree centrality of the nodes is the most important feature, and the degree centrality of the neighbouring nodes and the degree of mutual neighbours gave us the best trade-off. As all of the extracted features' distributions are highly skewed to the right, meaning most of the features take near zero values, using a power transform such as Yeo–Johnson[83] helps to make the distributions more Gaussian, which boosts the learning. Finally, for the link-prediction task, we saw that LSTMs perform better than fully connected neural networks.

Details on M4. The following two methods are based on a purely statistical analysis of the test data and are explained in detail in ref. 42.

Preferential attachment. In the network analysis, we concluded that the growth of this dataset tends to maintain a heavy-tailed degree distribution, often associated with scale-free networks. As mentioned before the γ value of the degree distribution is very close to 2, suggesting that preferential attachment[41] is probably the main organizational principle of the network. As such, we implemented a simple prediction model following this procedure. Preferential attachment scores in link prediction are often quantified as

$$s_{ij}^{\mathrm{PA}} = k_i k_j. \tag{1}$$

with k_{ij} the degree of nodes i and j. However, this assumes the scoring of links between nodes that are already connected to the network, that is $k_{ij} > 0$, which is not the case for all the links we must score in the dataset. As a result, we define our preferential attachment model as

$$s_{ij}^{\mathrm{PA}} = k_i + k_j. \tag{2}$$

Using this simple model with no free parameters we could score new links and compare them with the other models. Immediately we note that preferential attachment outperforms some learning-based models, even if it never manages to reach the top AUC, but it is extremely simple and with negligible computational cost.

Common neighbours. We explore another network-based approach to score the links. Indeed, while the preferential attachment model we derived performed well, it uses no information about the distance between i and j, which is a popular feature used in link-prediction methods[27]. As such, we decided to test a method known as common neighbours[18]. We define $\Gamma(i)$ as the set of neighbors of node i and $\Gamma(i) \cap \Gamma(j)$ as the set of common neighbours between nodes i and j. We can easily score the nodes with

$$s_{ij}^{\mathrm{CN}} = |\Gamma(i) \cap \Gamma(j)| \tag{3}$$

the intuition being that nodes that share a larger number of neighbours are more likely to be connected than distant nodes that do not share any.

Evaluating this score for each pair (i, j) on the dataset of unconnected pairs, which can be computed as the second power of the adjacency matrix, A^2, we obtained an AUC that is sometimes higher than preferential attachment and sometimes lower than it but is still consistently quite close with the best learning-based models.

Details on M5. This method is based on ref. 44. First, ten groups of first-order graph features are extracted to get some neighbourhood and similarity properties from each pair of nodes: degree centrality of nodes, pair's total number of neighbours, common neighbours index, Jaccard coefficient, Simpson coefficient, geometric coefficient, cosine coefficient, Adamic–Adar index, resource allocation index and preferential attachment index. They are obtained for three consecutive years to capture the temporal dynamics of the semantic network, leading to a total of 33 features. Second, principal component analysis[43] is applied to reduce the correlation between features, speed up the learning process and improve generalization, which results in a final set of seven latent variables. Lastly, a random forest classifier is trained (using a balanced dataset) to estimate the likelihood of new links between the AI concepts.

In this paper, a modification was performed in relation to the original formulation of the method[44]: two of the original features, average neighbour degree and clustering coefficient, were infeasible to extract for some of the tasks covered in this paper, as their computation can be heavy for such a very large network, and they were discarded. Due to some computational memory issues, it was not possible to run the model for some of the tasks covered in this study, and so those results are missing.

Details on M6. The baseline solution for the Science4Cast competition was closely related to the model presented in ref. 17. It uses 15 hand-crafted features of a pair of nodes v_1 and v_2. (Degrees of v_1 and v_2 in the current year and previous two years are six properties. The number of shared neighbours in total of v_1 and v_2 in the current year and

previous two years are six properties. The number of shared neighbours between v_1 and v_2 in the current year and the previous two years are three properties). These 15 features are the input of a neural network with four layers (15, 100, 10 and 1 neurons), intending to predict whether the nodes v_1 and v_2 will have w edges in the future. After the training, the model computes the probability for all 10 million evaluation examples. This list is sorted and the AUC is computed.

Details on M7. The solution M7 was not part of the Science4Cast competition and therefore not described in the corresponding proceedings, thus we want to add more details.

The most immediate way one can apply ML to this problem is by automating the detection of features. Quite simply, the baseline solution M6 is modified such that instead of 15 hand-crafted features, the neural network is instead trained on features extracted from a graph embedding. We use two different embedding approaches. The first method is employs node2vec (M7A)[45], for which we use the implementations provided in the nodevectors Python package[84]. The second one uses the ProNE embedding (M7B)[46], which is based on sparse matrix factorizations modulated by the higher-order Cheeger inequality[85].

The embeddings generate a 32-dimensional representation for each node, resulting in edge representations in $[0, 1]^{64}$. These features are input into a neural network with two hidden layers of size 1,000 and 30. Like M6, the model computes the probability for evaluation examples to determine the ROC. We compare ProNE to node2vec, a common graph embedding method using a biased random walk procedure with return and in–out parameters, which greatly affect network encoding. Initial experiments used default values for a 64-dimensional encoding before inputting into the neural network. The higher variance in node2vec predictions is probably due to its sensitivity to hyperparameters. While ProNE is better suited for general multi-dataset link prediction, node2vec's sensitivity may help identify crucial network features for predicting temporal evolution.

Details on M8. This model, which is detailed in ref. 47, does not use any hand-crafted features but learns them in a completely unsupervised manner. To do so, we extract various snapshots of the adjacency matrix through time, capturing graphs in the form of $\mathbf{A}_t$ for $t = 1994, \ldots, 2019$. We then embed each of these graphs into 128-dimensional Euclidean space via node2vec[45,48]. For each node u in the semantic graph, we extract different 128-dimensional vector embeddings $\mathbf{n}_u(\mathbf{A}_{1994}), \ldots, \mathbf{n}_u(\mathbf{A}_{2019})$.

Transformers have performed extremely well in NLP tasks[49]; thus, we apply them to learn the dynamics of the embedding vectors. We pre-train a transformer to help classify node pairs. For the transformer, the encoder and decoder had 6 layers each; we used 128 as the embedding dimension, 2,048 as the feed-forward dimension and 8-headed attention. This transformer acts as our feature extractor. Once we pre-train our transformer, we add a two-layer ReLU network with hidden dimension 128 as a classifier on top.

Data availability

All 18 datasets tested in this paper are available via Zenodo at https://doi.org/10.5281/zenodo.7882892 ref. 86.

Code availability

All of the models and codes described above can be found via GitHub at https://github.com/artificial-scientist-lab/FutureOfAIviaAI ref. 5 and a permanent Zenodo record at https://zenodo.org/record/8329701 ref. 87.

References

1. Clauset, A., Larremore, D. B. & Sinatra, R. Data-driven predictions in the science of science. *Science* **355**, 477–480 (2017).
2. Evans, J. A. & Foster, J. G. Metaknowledge. *Science* **331**, 721–725 (2011).
3. Fortunato, S. et al. Science of science. *Science* **359**, eaao0185 (2018).
4. Wang, D. & Barabási, A.-L.*The Science of Science* (Cambridge Univ. Press, 2021).
5. Krenn, M. et al. FutureOfAIviaAI. *GitHub* https://github.com/artificial-scientist-lab/FutureOfAIviaAI (2023).
6. Brown, T. et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **33**, 1877–1901 (2020).
7. Rae, J. W. et al. Scaling language models: methods, analysis & insights from training gopher. Preprint at https://arxiv.org/abs/2112.11446 (2021).
8. Smith, S. et al. Using DeepSpeed and Megatron to train Megatron-Turing NLG 530B, a large-scale generative language model. Preprint at https://arxiv.org/abs/2201.11990 (2022).
9. Chowdhery, A. et al. Palm: scaling language modeling with pathways. Preprint at https://arxiv.org/abs/2204.02311 (2022).
10. Kojima, T., Gu, S. S., Reid, M., Matsuo, Y. & Iwasawa, Y. Large language models are zero-shot reasoners. Preprint at https://arxiv.org/abs/2205.11916 (2022).
11. Zhang, H., Li, L. H., Meng, T., Chang, K.-W. & Broeck, G. V. d. On the paradox of learning to reason from data. Preprint at https://arxiv.org/abs/2205.11502 (2022).
12. Rzhetsky, A., Foster, J. G., Foster, I. T. & Evans, J. A. Choosing experiments to accelerate collective discovery. *Proc. Natl Acad. Sci. USA* **112**, 14569–14574 (2015).
13. Foster, J. G., Rzhetsky, A. & Evans, J. A. Tradition and innovation in scientists' research strategies. *Am. Sociol. Rev.* **80**, 875–908 (2015).
14. Van Eck, N. J. & Waltman, L. Text mining and visualization using vosviewer. Preprint at https://arxiv.org/abs/1109.2058 (2011).
15. Van Eck, N. J. & Waltman, L. in *Measuring Scholarly Impact: Methods and Practice* (eds Ding, Y. et al.) 285–320 (Springer, 2014).
16. Wang, Q. et al. Paperrobot: Incremental draft generation of scientific ideas. Preprint at https://arxiv.org/abs/1905.07870 (2019).
17. Krenn, M. & Zeilinger, A. Predicting research trends with semantic and neural networks with an application in quantum physics. *Proc. Natl Acad. Sci. USA* **117**, 1910–1916 (2020).
18. Liben-Nowell, D. & Kleinberg, J. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.* **58**, 1019–1031 (2007).
19. Albert, I. & Albert, R. Conserved network motifs allow protein–protein interaction prediction. *Bioinformatics* **20**, 3346–3352 (2004).
20. Zhou, T., Lü, L. & Zhang, Y.-C. Predicting missing links via local information. *Eur. Phys. J. B* **71**, 623–630 (2009).
21. Kovács, I. A. et al. Network-based prediction of protein interactions. *Nat. Commun.* **10**, 1240 (2019).
22. Muscoloni, A., Abdelhamid, I. & Cannistraci, C. V. Local-community network automata modelling based on length-three-paths for prediction of complex network structures in protein interactomes, food webs and more. Preprint at *bioRxiv* https://doi.org/10.1101/346916 (2018).
23. Pech, R., Hao, D., Lee, Y.-L., Yuan, Y. & Zhou, T. Link prediction via linear optimization. *Physica A* **528**, 121319 (2019).
24. Lü, L., Pan, L., Zhou, T., Zhang, Y.-C. & Stanley, H. E. Toward link predictability of complex networks. *Proc. Natl Acad. Sci. USA* **112**, 2325–2330 (2015).
25. Guimerà, R. & Sales-Pardo, M. Missing and spurious interactions and the reconstruction of complex networks. *Proc. Natl Acad. Sci. USA* **106**, 22073–22078 (2009).
26. Ghasemian, A., Hosseinmardi, H., Galstyan, A., Airoldi, E. M. & Clauset, A. Stacking models for nearly optimal link prediction in complex networks. *Proc. Natl Acad. Sci. USA* **117**, 23393–23400 (2020).

27. Zhou, T. Progresses and challenges in link prediction. *iScience* **24**, 103217 (2021).
28. Krenn, M. et al. On scientific understanding with artificial intelligence. *Nat. Rev. Phys.* **4**, 761–769 (2022).
29. Rose, S., Engel, D., Cramer, N. & Cowley, W. in *Text Mining: Applications and Theory* (eds Berry, M. W. & Kogan, J.) Ch. 1 (Wiley, 2010).
30. Salatino, A. A., Thanapalasingam, T., Mannocci, A., Osborne, F. & Motta, E. The computer science ontology: a large-scale taxonomy of research areas. In *Proc. Semantic Web–ISWC 2018: 17th International Semantic Web Conference Part II* Vol. 17, 187–205 (Springer, 2018).
31. Salatino, A. A., Osborne, F., Thanapalasingam, T. & Motta, E. The CSO classifier: ontology-driven detection of research topics in scholarly articles. In *Proc. Digital Libraries for Open Knowledge: 23rd International Conference on Theory and Practice of Digital Libraries* Vol. 23, 296–311 (Springer, 2019).
32. Alstott, J., Bullmore, E. & Plenz, D. powerlaw: a Python package for analysis of heavy-tailed distributions. *PLoS ONE* **9**, e85777 (2014).
33. Fenner, T., Levene, M. & Loizou, G. A model for collaboration networks giving rise to a power-law distribution with an exponential cutoff. *Soc. Netw.* **29**, 70–80 (2007).
34. Broido, A. D. & Clauset, A. Scale-free networks are rare. *Nat. Commun.* **10**, 1017 (2019).
35. Fawcett, T. ROC graphs: notes and practical considerations for researchers. *Pattern Recognit. Lett.* **31**, 1–38 (2004).
36. Sun, Y., Wong, A. K. & Kamel, M. S. Classification of imbalanced data: a review. *Int. J. Pattern Recognit. Artif. Intell.* **23**, 687–719 (2009).
37. Lu, Y. Predicting research trends in artificial intelligence with gradient boosting decision trees and time-aware graph neural networks. In *2021 IEEE International Conference on Big Data (Big Data)* 5809–5814 (IEEE, 2021).
38. Ke, G. et al. LightGBM: a highly efficient gradient boosting decision tree. In *Proc. 31st International Conference on Neural Information Processing Systems* 3149–3157 (Curran Associates Inc., 2017).
39. Tran, N. M. & Xie, Y. Improving random walk rankings with feature selection and imputation Science4Cast competition, team Hash Brown. In *2021 IEEE International Conference on Big Data (Big Data)* 5824–5827 (IEEE, 2021).
40. Sanjabi, N. Efficiently predicting scientific trends using node centrality measures of a science semantic network. In *2021 IEEE International Conference on Big Data (Big Data)* 5820–5823 (IEEE, 2021).
41. Barabási, A.-L. Network science. *Phil. Trans. R. Soci. A* **371**, 20120375 (2013).
42. Moutinho, J. P., Coutinho, B. & Buffoni, L. Network-based link prediction of scientific concepts—a Science4Cast competition entry. In *2021 IEEE International Conference on Big Data (Big Data)* 5815–5819 (IEEE, 2021).
43. Jolliffe, I. T. & Cadima, J. Principal component analysis: a review and recent developments. *Phil. Trans. R. Soc. A* **374**, 20150202 (2016).
44. Valente, F. Link prediction of artificial intelligence concepts using low computational power. In *2021 IEEE International Conference on Big Data (Big Data)* 5828–5832 (2021).
45. Grover, A. & Leskovec, J. node2vec: scalable feature learning for networks. In *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 855–864 (ACM, 2016).
46. Zhang, J., Dong, Y., Wang, Y., Tang, J. & Ding, M. ProNE: fast and scalable network representation learning. In *Proc. Twenty-Eighth International Joint Conference on Artificial Intelligence* 4278–4284 (International Joint Conferences on Artificial Intelligence Organization, 2019).
47. Lee, H., Sonthalia, R. & Foster, J. G. Dynamic embedding-based methods for link prediction in machine learning semantic network. In *2021 IEEE International Conference on Big Data (Big Data)* 5801–5808 (IEEE, 2021).
48. Liu, R. & Krishnan, A. PecanPy: a fast, efficient and parallelized python implementation of node2vec. *Bioinformatics* **37**, 3377–3379 (2021).
49. Vaswani, A. et al. Attention is all you need. In *Proc. 31st International Conference on Neural Information Processing Systems* 6000–6010 (Curran Associates Inc., 2017).
50. Zelenko, D., Aone, C. & Richardella, A. Kernel methods for relation extraction. *J. Mach. Learn. Res.* **3**, 1083–1106 (2003).
51. Bach, N. & Badaskar, S. A review of relation extraction. *Literature Review for Language and Statistics II* **2**, 1–15 (2007).
52. Salatino, A. A., Osborne, F. & Motta, E. How are topics born? Understanding the research dynamics preceding the emergence of new areas. *PeerJ Comput. Sc.* **3**, e119 (2017).
53. Salatino, A. A., Osborne, F. & Motta, E. AUGUR: forecasting the emergence of new research topics. In *Proc. 18th ACM/IEEE on Joint Conference on Digital Libraries* 303–312 (IEEE, 2018).
54. Battiston, F. et al. The physics of higher-order interactions in complex systems. *Nat. Phys.* **17**, 1093–1098 (2021).
55. Coutinho, B. C., Wu, A.-K., Zhou, H.-J. & Liu, Y.-Y. Covering problems and core percolations on hypergraphs. *Phys. Rev. Lett.* **124**, 248301 (2020).
56. Olivetti, E. A. et al. Data-driven materials research enabled by natural language processing and information extraction. *Appl. Phys. Rev.* **7**, 041317 (2020).
57. Lin, Z., Yin, Y., Liu, L. & Wang, D. SciSciNet: a large-scale open data lake for the science of science research. *Sci. Data* **10**, 315 (2023).
58. Azoulay, P. et al. Toward a more scientific science. *Science* **361**, 1194–1197 (2018).
59. Liu, H., Kou, H., Yan, C. & Qi, L. Link prediction in paper citation network to construct paper correlation graph. *EURASIP J. Wirel. Commun. Netw.* **2019**, 1–12 (2019).
60. Reisz, N. et al. Loss of sustainability in scientific work. *New J. Phys.* **24**, 053041 (2022).
61. Frank, M. R., Wang, D., Cebrian, M. & Rahwan, I. The evolution of citation graphs in artificial intelligence research. *Nat. Mach. Intell.* **1**, 79–85 (2019).
62. Newman, M. *Networks* (Oxford Univ. Press, 2018).
63. Kwon, D. et al. A survey of deep learning-based network anomaly detection. *Cluster Comput.* **22**, 949–961 (2019).
64. Pang, G., Shen, C., Cao, L. & Hengel, A. V. D. Deep learning for anomaly detection: a review. *ACM Comput. Surv.* **54**, 1–38 (2021).
65. Collobert, R. et al. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **12**, 2493–2537 (2011).
66. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
67. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **60**, 84–90 (2017).
68. Mnih, V. et al. Human-level control through deep reinforcement learning. *Nature* **518**, 529–533 (2015).
69. Silver, D. et al. Mastering the game of Go with deep neural networks and tree search. *Nature* **529**, 484–489 (2016).
70. Wu, N., Vincent, A., Strukov, D. & Xie, Y. Memristor hardware-friendly reinforcement learning. Preprint at https://arxiv.org/abs/2001.06930 (2020).
71. Zhou, C. et al. Automated deep learning analysis of angiography video sequences for coronary artery disease. Preprint at https://arxiv.org/abs/2101.12505 (2021).
72. Huckle, N., Garcia, N. & Nakashima, Y. Demographic influences on contemporary art with unsupervised style embeddings. In *Proc. Computer Vision–ECCV 2020 Workshops Part II* Vol. 16, 126–142 (Springer, 2020).

73. Ranti, D. et al. The utility of general domain transfer learning for medical language tasks. Preprint at https://arxiv.org/abs/2002.06670 (2020).
74. Kamath, P., Singh, A. & Dutta, D. Fast neural architecture construction using envelopenets. Preprint at https://arxiv.org/abs/1803.06744 (2018).
75. Minsky, M. Steps toward artificial intelligence. *Proc. IRE* **49**, 8–30 (1961).
76. Bornmann, L., Haunschild, R. & Mutz, R. Growth rates of modern science: a latent piecewise growth curve approach to model publication numbers from established and new literature databases. *Humanit. Soc. Sci. Commun.* **8**, 224 (2021).
77. Brin, S. & Page, L. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.* **30**, 107–117 (1998).
78. Holland, P. W. & Leinhardt, S. Transitivity in structural models of small groups. *Comp. Group Studies* **2**, 107–124 (1971).
79. Watts, D. J. & Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature* **393**, 440–442 (1998).
80. Yang, J.-H., Chen, C.-M., Wang, C.-J. & Tsai, M.-F. HOP-rec: high-order proximity for implicit recommendation. In *Proc. 12th ACM Conference on Recommender Systems* 140–144 (2018).
81. Lin, B.-Y. OGB_collab_project. *GitHub* https://github.com/brucenccu/OGB_collab_project (2021).
82. Sorensen, T. A. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.* **5**, 1–34 (1948).
83. Yeo, I.-K. & Johnson, R. A. A new family of power transformations to improve normality or symmetry. *Biometrika* **87**, 954–959 (2000).
84. Ranger, M. nodevectors. *GitHub* https://github.com/VHRanger/nodevectors (2021).
85. Bandeira, A. S., Singer, A. & Spielman, D. A. A Cheeger inequality for the graph connection Laplacian. *SIAM J. Matrix Anal. Appl.* **34**, 1611–1630 (2013).
86. Krenn, M. et al. Predicting the future of AI with AI. *Zenodo* https://doi.org/10.5281/zenodo.7882892 (2023).
87. Krenn, M. et al. FutureOfAIviaAI code. *Zenodo* https://zenodo.org/record/8329701 (2023).
88. Jia, T., Wang, D. & Szymanski, B. K. Quantifying patterns of research-interest evolution. *Nat. Hum. Behav.* **1**, 0078 (2017).

Acknowledgements

We thank IARAI Vienna and IEEE for supporting and hosting the IEEE BigData Competition Science4Cast. We are specifically grateful to D. Kreil, M. Neun, C. Eichenberger, M. Spanring, H. Martin, D. Geschke, D. Springer, P. Herruzo, M. McCutchan, A. Mihai, T. Furdui, G. Fratica, M. Vázquez, A. Gruca, J. Brandstetter and S. Hochreiter for helping to set up and successfully execute the competition and the corresponding workshop. We thank X. Gu for creating Fig. 2, and M. Aghajohari and M. Sadegh Akhondzadeh for helpful comments on the paper. The work of H.L., R.S. and J.G.F. was supported by grant TWCF0333 from the Templeton World Charity Foundation. H.L. is additionally supported by NSF grant DMS-1952339. J.P.M. acknowledges the support of FCT (Portugal) through scholarship SFRH/BD/144151/2019. B.C. thanks the support from FCT/MCTES through national funds and when applicable co-funded EU funds under the project UIDB/50008/2020, and FCT through the project CEECINST/00117/2018/CP1495/CT0001. N.M.T. and Y.X. are supported by NSF grant DMS-2113468, the NSF IFML 2019844 award to the University of Texas at Austin, and the Good Systems Research Initiative, part of University of Texas at Austin Bridging Barriers.

Author contributions

M. Krenn and R.Y. initiated the research. M. Krenn and M. Kopp organized the Science4Cast competition. M. Krenn generated the datasets and initial codes. S.E. and H.L. analysed the network-theoretical properties of the semantic network. M. Krenn, L.B., B.C., J.G.F., A.G, H.L., Y.L, J.P.M, N.S., R.S., N.M.T, F.V., Y.X and M. Kopp provided codes for the ten models. M. Krenn wrote the paper with input from all co-authors.

Funding

Open access funding provided by Max Planck Society.

Competing interests

The authors declare no competing interests.

Additional information

Extended data is available for this paper at https://doi.org/10.1038/s42256-023-00735-0.

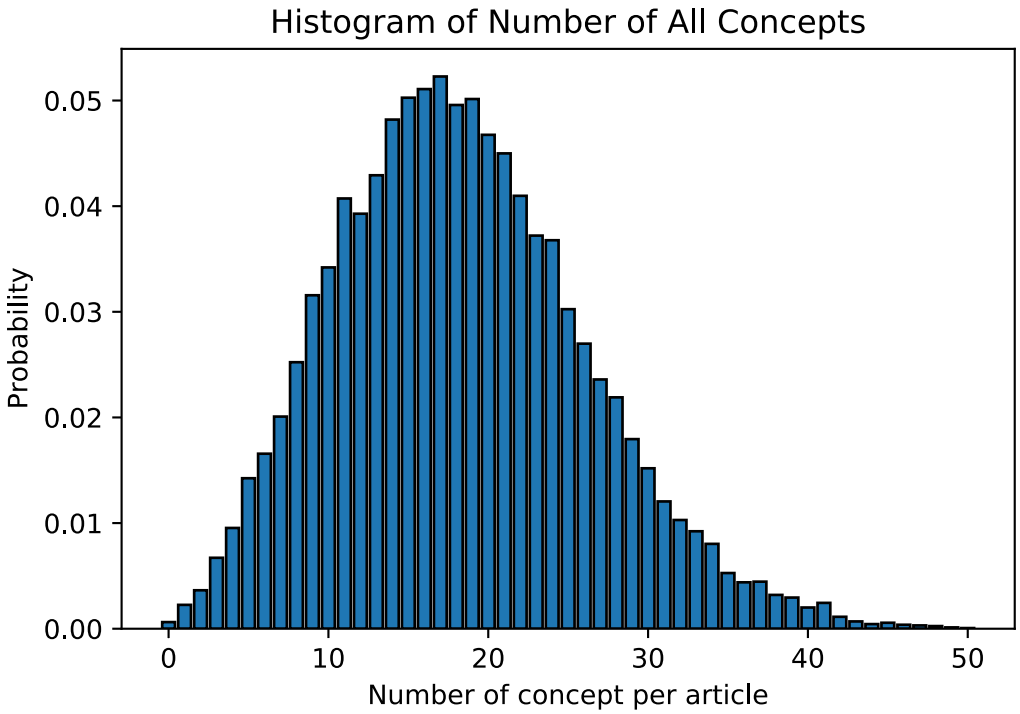
Correspondence and requests for materials should be addressed to Mario Krenn.

Peer review information *Nature Machine Intelligence* thanks Alexander Belikov, and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. Primary Handling Editor: Mirko Pieropan, in collaboration with the *Nature Machine Intelligence* team.

Reprints and permissions information is available at www.nature.com/reprints.

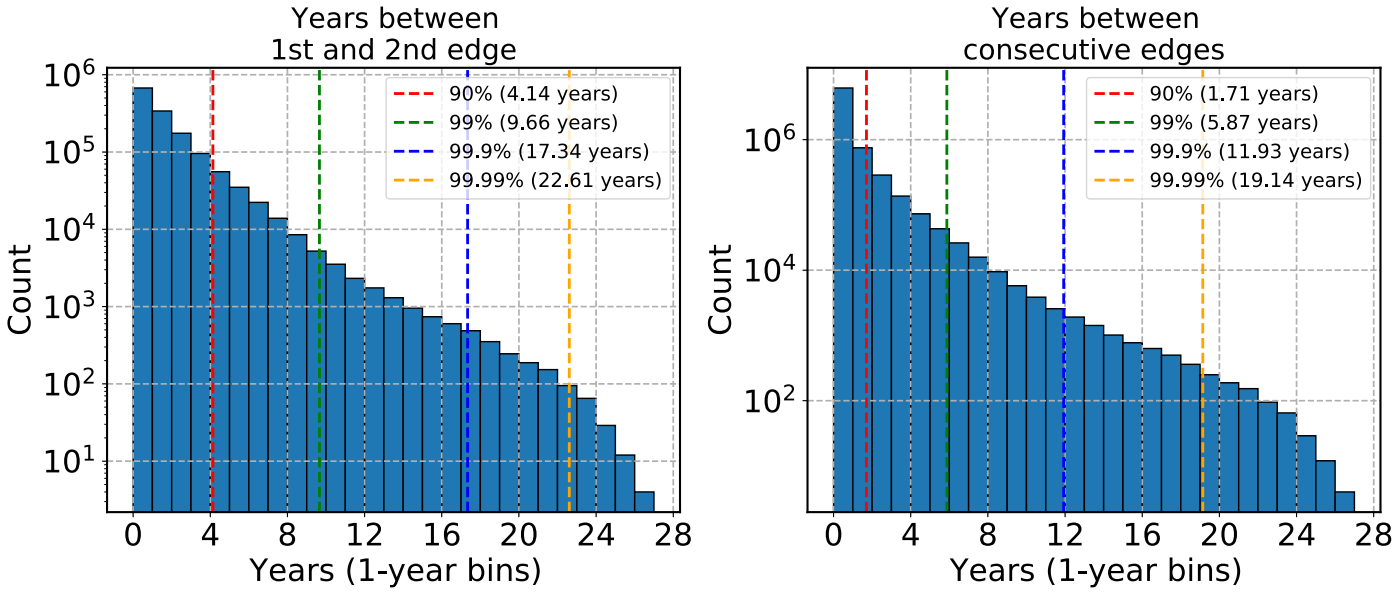
Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

https://doi.org/10.1038/s42256-023-00735-0



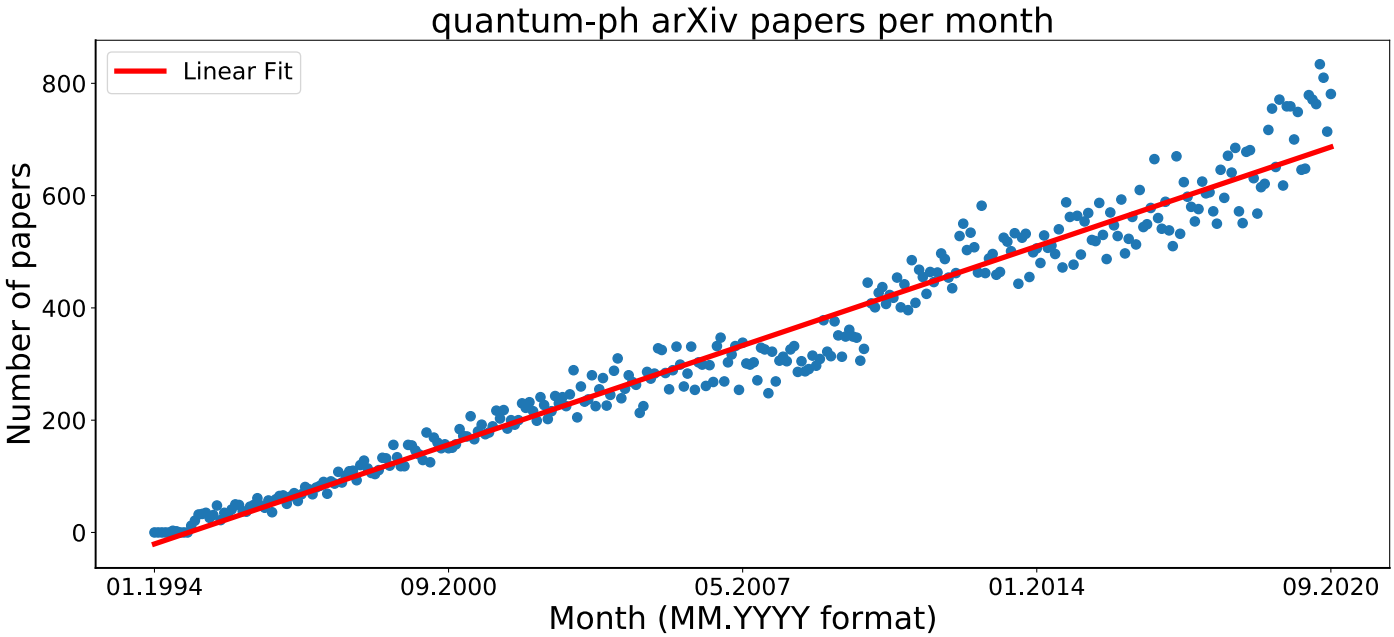
Extended Data Fig. 1 | Number of concepts per article.

https://doi.org/10.1038/s42256-023-00735-0



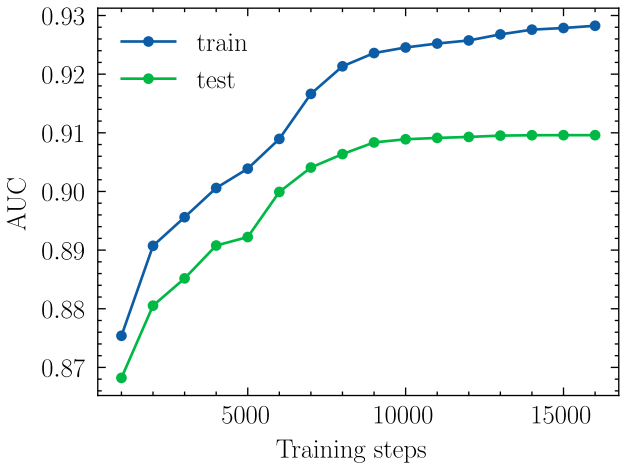
Extended Data Fig. 2 | Time Gap between the generation of edges. Here, left shows the time it takes to create a new edge between two vertices and right shows the time between the first and the second edge.

https://doi.org/10.1038/s42256-023-00735-0



Extended Data Fig. 3 | Publications in Quantum Physics.

https://doi.org/10.1038/s42256-023-00735-0



Extended Data Fig. 4 | Evolution of the AUC during training for Model M1.