Predicting the Future of AI with AI: High-Quality link prediction in an exponentially growing knowledge network

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] and Michael Kopp[6]

[1] *Max Planck Institute for the Science of Light (MPL), Erlangen, Germany.*
[2] *Instituto de Telecomunicações, Lisbon, Portugal.*
[3] *University of Toronto, Canada.*
[4] *University of California Los Angeles, USA.*
[5] *Cavendish Laboratories, Cavendish, Vermont, USA.*
[6] *Institute of Advanced Research in Artificial Intelligence (IARAI), Vienna, Austria.*
[7] *Layer 6 AI, Toronto, Canada.*
[8] *Independent Researcher, Barcelona, Spain.*
[9] *University of Texas at Austin, USA.*
[10] *Independent Researcher, Leiria, Portugal.*
[11] *University of Pennsylvania, USA.*
[12] *University of California, San Diego, USA.*

A tool that could suggest new personalized research directions and ideas by taking insights from the scientific literature could significantly 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 the last 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 develop a new 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 100,000 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. It indicates 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.

I. INTRODUCTION AND MOTIVATION

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 grows exponentially with a doubling rate of roughly 23 months (see 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. If successful, it could significantly improve the productivity of AI researchers, open up new avenues of research, and help drive progress in the field.

Here, we address this important and challenging vision. New research ideas often result from drawing novel connections between seemingly unrelated concepts [1–3]. Therefore, we formulate the evolution of AI literature as a temporal network modelling task. We created an evolving semantic network characterizing the content and evolution of the scientific literature in the field of AI since 1994. The network contains about 64,000 nodes (each representing a concept used in an AI paper) and 18 million edges that connect two concepts when they were investigated jointly in a scientific paper.

We use the semantic network as an input to 10 diverse statistical and machine-learning 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 of the methods presented in this paper have been contributions to the `Science4Cast` competition hosted by *IEEE BigData 2021*, which ran from August to November 2021. Broadly, we can divide the methods into two classes: methods that

* mario.krenn@mpl.mpg.de

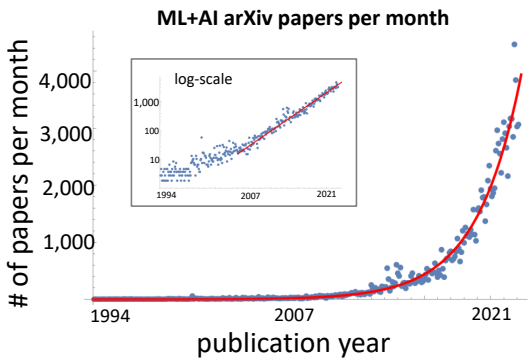

Figure 1. **Number of papers published per months in the arXiv categories of AI grow exponentially.** The doubling rate of papers per months is roughly 23 months, which might lead to problems for publishing in these fields, at some point. The categories are `cs.AI`, `cs.LG`, `cs.NE`, and `stat.ML`.

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 manuscript has several purposes. First, we introduce a new meaningful benchmark for AI on real-world graphs. Second, we provide nearly 10 diverse methods that solve this benchmark. Third, we explain how solving this task could become an essential ingredient for the big picture goal of having a tool that could suggest meaningful research directions for scientists in AI or in other disciplines.[1]

The manuscript is structured in the following way. We first introduce more background into semantic networks and how they can help to suggest new ideas. Then we explain how we generate the dataset and some of its network-theoretical properties. Then we briefly explain the 10 methods that we have investigated to solve the task. We conclude with a number of important open questions that could bring us further toward the goal of AI-based suggestions for research directions.

II. 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 the features of a large language model (such as GPT3 [4], Gopher [5], MegaTron [6] or PaLM [7]) from the text of each article to extract concepts automatically. However, those methods still struggle in reasoning capabilities [8, 9], thus it is not yet directly clear how these models can be used for identifying and suggesting new ideas and concept combinations.

An alternative approach has been pioneered by Rzhetsky and colleagues [10]. They have shown how knowledge networks (or semantic networks) in biochemistry can be created from co-occurring concepts in scientific papers. The nodes in their network correspond to scientific concepts—concretely, the names of individual biomolecules. The nodes are linked when a paper mentions both of the corresponding biomolecules in its title or abstract. Taking millions of papers into account leads to an evolving semantic network that captures the history of the field. Using supercomputer simulations, nontrivial statements about the collective behaviour of scientists can be extracted, which allows for the suggestions of alternative and more efficient research behaviour [11]. Of course, by creating a semantic network from concept co-occurrences, only a tiny amount of knowledge is extracted from each paper. However, if this process is repeated for a large dataset of papers, the resulting network captures nontrivial and actionable content.

The idea to build up a semantic network of a scientific discipline was then applied and extended in the field of quantum physics [12]. There, the authors (including one of us) built a network of more than 6,000 quantum physics concepts. The authors formulate the task of predicting new research trends and connections for the first time as an ML task. The task was to identify which concept pairs, which have never been discussed jointly in the scientific literature, have a high probability to be investigated in the near future. This prediction task was phrased as one component for personalized suggestions of new research ideas.

A. Link Prediction in Semantic Networks

Here we formulate the predictions of future research topics as a link prediction task in an exponentially growing semantic network in the field of AI. Two nodes that do not share an edge have not been mentioned together in the title or abstract of an existing scientific paper. Here, the goal is to predict which unconnected nodes will be connected in the future—that is, determine which scientific concepts that have not been researched yet *will* be jointly researched in the future.

[1] github.com/artificial-scientist-lab/FutureOfAIviaAI

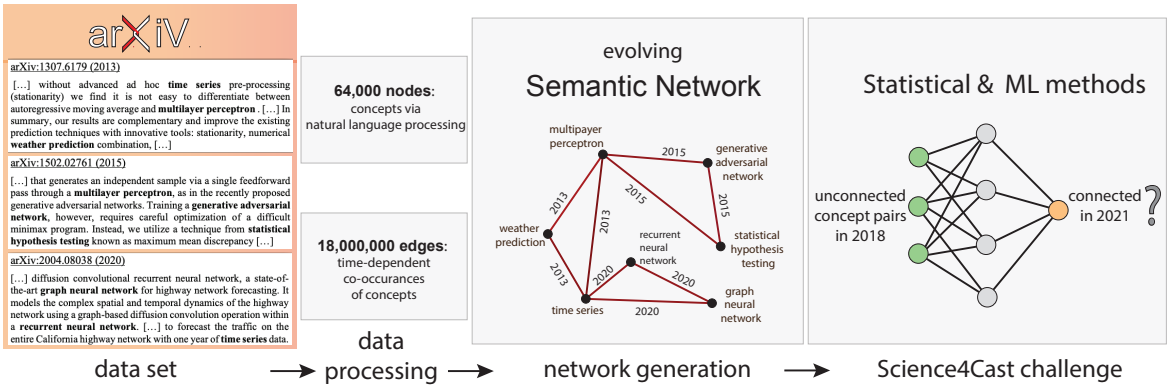

Figure 2. **From arXiv to `Science4Cast`.** We use 143,000 papers in AI and ML categories on arXiv from 1992 to 2020. From there, we construct a list of concepts (using RAKE and other NLP tools). Those concepts form the nodes of a semantic network. The edges are drawn when two concepts occur jointly in the title or abstract of a paper. In that way, we generate an evolving semantic network that grows over time as more concepts are investigated together. The task is to predict, from unconnected nodes (i.e. concepts that have not been investigated together in the scientific literature), which will be connected within a few years. In this manuscript, we present 10 diverse statistical and machine learning methods to solve this challenge.

Link prediction is a very common problem in computer science that can be solved with classical metrics and features as well as machine learning techniques. From the network theory side, several works have studied local motif-based methods [13–17], often based on path-counting, while other methods have studied more global features using linear optimization [18], global perturbations [19] and stochastic block models [20]. Other machine-learning works have tried to optimize over a combination of hundred of predictors [21]. Further discussion on these methods is available in a recent review on link prediction [22].

In [12], this task was solved by computing 17 hand-crafted features of the evolving semantic network. 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 10 times larger than the one in [12]). Specifically, on the one hand, we would like to determine which features are useful; on the other hand, we would also like to know whether this task can be solved efficiently without hand-crafted features. Here, we present results for both questions.

B. 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 [23]. We can bias or constrain the model to give research topics that are related to the research interest of individual scientists, or a pair of scientists to suggest topics for collaborations in an interdisciplinary setting. Important future questions concern the discovery of impactful and surprising suggestions, and suggestions that give more context than two scientific concepts.

III. GENERATION AND ANALYSIS OF THE DATASET

A. Dataset Construction

We use papers that are published on arXiv in the categories `cs.AI`, `cs.LG`, `cs.NE`, and `stat.ML`, from 1992 to 2020, to create a dynamic semantic network. The nodes stand for computer science and in particular artificial intelligence concepts. We create the list of concepts from the title and abstracts of all of the 143,000 papers. We use Rapid Automatic Keyword Extraction (RAKE) to create candidate concepts [24], and normalize the list using standard NLP techniques and other self-created methods. Ultimately, this leads to a list of 64,719 concepts.

These concepts form the nodes of the semantic network. The edges are drawn when two concepts co-appear in a title or abstract of a paper. Each edge has a time stamp, which is the publication date of the paper in which the concepts co-appear. Multiple edges with different time-stamps between two concepts are very common, as concept pairs can co-appear in many papers with different publication dates. As edges have time stamps, the entire semantic network is evolving in time. The workflow is depicted in Fig. 2.

B. Network-Theoretical Analysis

We start by analyzing the degree distribution of the published semantic network. The network has 64,719 nodes and 17,892,352 unique undirected edges, which implies a mean node degree of about 553. However, the network contains many hub nodes that significantly exceed this mean degree, demonstrated by the heavy-tail degree distribution in Fig. 3. For example, the ten highest node degrees (and their corresponding concepts) 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`), 115,334 (`large scale`), 89,267 (`high dimension`), and 84,956 (`high dimensional`).

To investigate whether this complex network is scale-free, we fit a power-law curve to the degree distribution $p(k)$ using [25], and the software fit $p(k) \propto k^{-2.28}$ for degree $k \geq 1672$. Nevertheless, the degree distribution of real complex network do not always follow perfect power-laws and power-laws with exponential cut-offs are often a better fit than pure power-laws [26].

A recent work [27] empirically showed that lognormal distributions fit most real-world networks as well as or better than power laws, and confirmed that pure "scale-free networks are rare". In light of that result, we used likelihood ratio tests to compare the power law fit with alternative distributions. The likelihood ratio tests from [25] suggested that truncated power law (p-value: 0.0031), lognormal (p-value: 0.0045), and lognormal positive (p-value: 0.015) fit the data better than power law, while exponential (p-value: 3e-10) and stretched exponential (p-value: 6e-05) were worse. We could not conclude whether truncated power law, lognormal, or lognormal positive *best* describe the data with p-value ≤ 0.1.

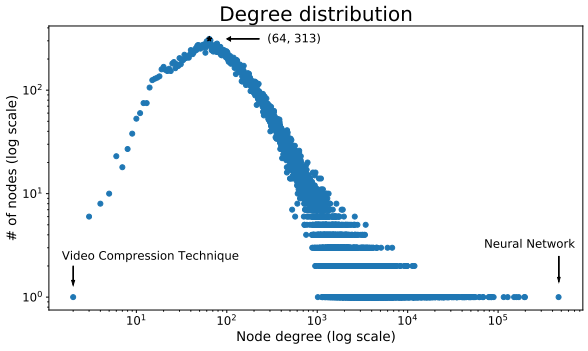

Figure 3. **Node degrees follow heavy-tail distribution due to the hubs.** Nodes with the largest (466,319) and smallest (2) non-zero degrees correspond to `neural network` and `video compression technique`, respectively. The most common non-zero degree is 64. 1,247 nodes with zero degrees are not shown in this plot, and both axes are in log scale.

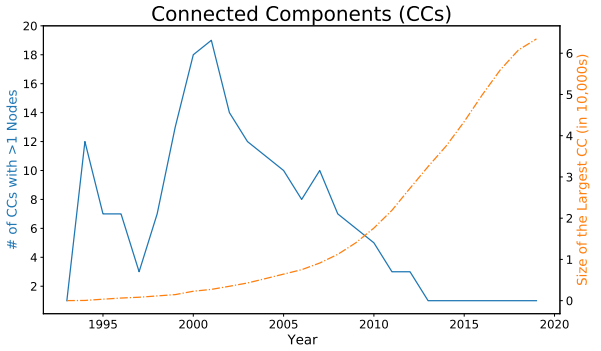

Figure 4. **The network became more connected over the years.** Primary (left, blue) vertical axis: Number of connected components with more than one node. Secondary (right, orange) vertical axis: Number of nodes in the largest connected component. For example, the network in 2019 comprises of one large connected component with 63,472 nodes and 1,247 isolated nodes, i.e. nodes with no edges. On the other hand, the 2001 network has 19 connected components with size greater than one, the largest of which has 2,733 nodes.

Next, we discuss changes in the network connectivity over time. While the degree distributions maintained a heavy tail over the years, the ordering of the nodes inside the heavy tail changed, likely in response to the popularity trends in the field. The nodes with most connections (and the year they became so) are `decision tree` (1994), `machine learning` (1996), `logic program` (2000), `neural network` (2005), `experimental result` (2011), `machine learning` (2013), and finally, back to `neural network` (2015).

Furthermore, the network grew more connected over time according to connected component analysis in Fig. 4. Groups that were previously separated became connected, i.e. number of connected components decreased, while the largest group grew bigger. The trajectory of the mid-sized connected components may reveal interesting trends about their topics. Take image processing for instance. A connected component of the following 4 nodes appeared in 1999: `brightness change`, `planar curve`, `local feature`, and `differential invariant`. In 2000, 3 more nodes joined the group: `similarity transformation`, `template matching`, and `invariant representation`. Then in 2006, a paper that discusses both `support vector machine` and `local feature` merged this

mid-size group of nodes to the largest connected component.

Another trend that emerges from the semantic network is an increase in centralization over time, with fewer percentage nodes (concepts) contributing larger fraction edges (concepts combination) over the years. This trend seems to be consistent across the entire period of the dataset. It can be seen from the histogram in Fig. 5 that the fraction of edges corresponding to the highest degree nodes (most connected) increases over the years, while the fraction of edges corresponding to the least connected nodes decreases. This trend is also consistent with the decrease in the average clustering coefficient over time (average clustering coefficient by year: 1999: *0.919*, 2004: *0.844*, 2009: *0.773*, 2013: *0.650*), implying most nodes are less likely to be connected with each other and more likely to be connected to a few high-degree central nodes. This trend might be explained by the fact that the AI community has been focusing on a few methods (e.g. deep learning) which have grown to dominate the field, compared to more diverse approaches in the 90s and 2000s. An alternative explanation is the use of more consistent terminology.

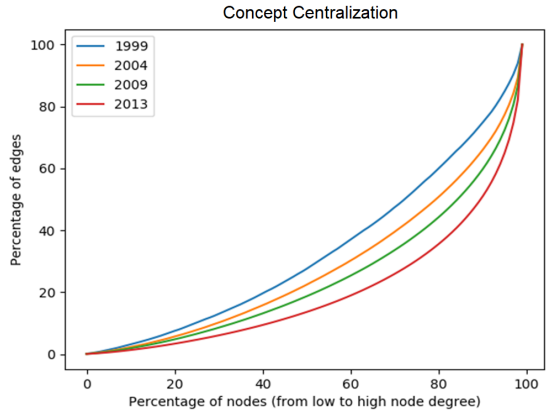

Figure 5. **Centralization of Concepts.** The fraction of nodes (concepts) that corresponds to the fraction of edges (connections). Cumulative histogram of edges per node, up to a given year (1999,2004,2009,2014). The graph was created by going over the edges list and adding to each year only the edges and nodes that are dated before the year (hence the 2014 plot contains all the nodes (concepts) in papers before 2014). The nodes are arranged by increasing degrees. The plot is a cumulative graph; hence the y value in the x=80, is the fraction of edges contributed by all the nodes in and below the 80s percentile of degrees.

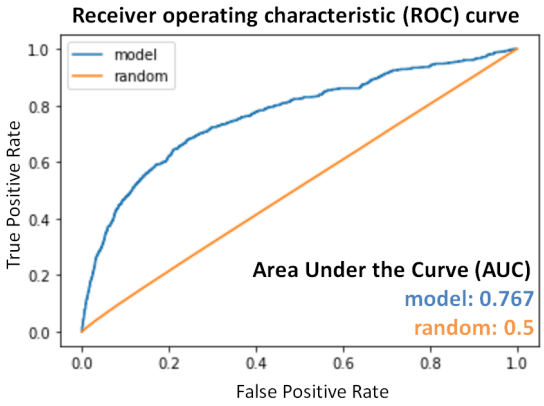

Figure 6. **Receiver operating characteristic curve (ROC) for computing the Area under the Curve (AUC).** Random Predictions get the result right in half of the cases, therefore their ROC curve is a diagonal with an AUC=0.5 (orange). A model that has learned some properties of the dataset has a AUC > 0.5 (blue).

C. Problem Formulation: Predictions in an exponentially growing semantic network

The concrete task is to predict which two nodes v_i with degrees $d(v_i) \leq c$ that do not share an edge in the year (2021-δ) will have w edges in the year 2021. We use $\delta = 1, 3, 5$, $c = 0, 5, 25$ and $w = 1, 3$. Note that $c = 0$ is an interesting special case in which the node does not have any edge associated to it yet in the initial year. Thus, the model does not have any information about the node yet; the task there is to predict which nodes will be connected to entirely new edges. The task $w = 3$ goes beyond simple link prediction, and asks which uninvestigated concept pair will be studied together in at least 3 papers.[2]

In the task, we provide a list of 10 million unconnected nodes pairs (each node having a degree$\leq$c) of the year (2021-δ), and the goal is to sort this list from highest to lowest probability that in 2021 they will have at least w edges.

For the evaluation we use the ROC curve [28]; see Fig. 6 for details. The ROC curve is created by plotting the true positive rate against the false positive rate at various threshold settings. Our evaluation metric is the commonly used metric Area under the Curve (AUC) of the ROC curve. One advantage of AUC over mean-square-error is its independence of the data distribution. Specifically, in our case, where the two classes are highly asymmetrically distributed (with only about 1-3% of newly connected

[2] One interesting alternative task is the prediction of the fastest growing links, which one could denote as *trend* prediction.

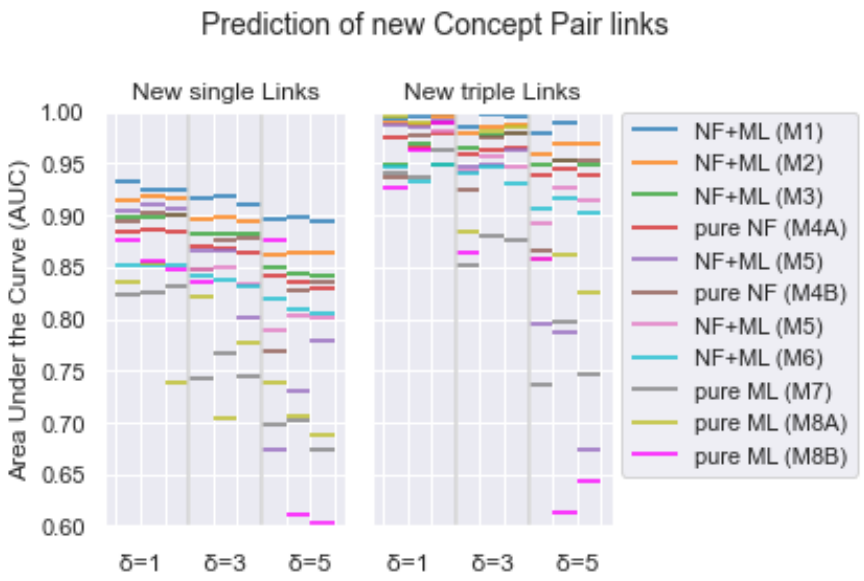

Figure 7. **The `Science4Cast` benchmark: Link predictions in an exponentially growing semantic network.** Here we show the AUC values for different models that use machine learning techniques (ML), hand-crafted network features (NF) or a combination thereof. The left plot shows results for the prediction of a single new link (i.e., $w = 1$), the right one shows results for the prediction of new triple links $w = 3$. The task is to predict $\delta = [1, 3, 5]$ years into the future, with cutoff 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. Note that the prediction of new triple edges can be performed nearly determinstically. It will be interesting to understand the origin of this quasi-deterministic pattern in AI research.

edges), and the distribution changing over time, the AUC provides a meaningful and interpretation. For perfect predictions, AUC=1, while random predictions give AUC=0.5. It gives the percentage that a random true element is higher ranked than a random false one.

IV. AI-BASED SOLUTIONS

We now demonstrate how to solve this task with numerous different methods, from pure statistical approaches to hand-crafted features (NF) as an input of a neural network, to ML models that can work without hand-crafted features. All results are shown in Fig. 7. The most powerful methods (those with the highest prediction quality measured by the AUC metric) take advantage of NF, which are the inputs to an ML model. Surprisingly, using purely network theoretical features without machine learning works competitively. Pure ML methods were not yet able to outperform those that use hand-crafted features. It remains an important open challenge how to solve this task without relying on hand-crafted features. While the prediction of new links can reach an AUC of up to 93%, we find that the prediction of links that are generated at least three times can be solved with AUC¿99.5%. Understanding this apparently quasi-deterministic pattern in AI research will be an interesting target for follow-up research.[3]

A. M1: Features+ML

The solution of team oahciy is based on a blend of a tree-based gradient boosting approach and a graph neural network approach [29]. 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 [30], while the proximity between a node pair is derived using the Jaccard index. We refer the reader to [29] for the list of all features and their feature importance.

The tree-based gradient boosting approach uses the Light Gradient Boosting Machine (LightGBM)

[3] We have performed numerous additional tests to exclude data leakage in the benchmark dataset, overfitting or data duplication both in the set of articles and the set of concepts.

[31] 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.

B. M2: Features+ML

The method proposed by Team HashBrown 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 [32, 33], and PageRank [30]). 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 [34, 35] and a variation of the Dice similarity score [36] 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 (MLP) (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 [37].

C. M3: Features+ML

This approach, detailed in [38], uses hand-crafted node features that have been captured in multiple time snapshots (e.g. 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 features: node features including degree and degree of neighbours, and edge features including common neighbours. And 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 tradeoff. 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 like *Yeo-Johnson* [39] 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.

D. M4: pure Features

The following two methods are based on a purely statistical analysis of the test data and are explained in detail in [40].

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 likely 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

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

with $k_{i,j}$ 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 $k_{i,j} > 0$, 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 [22]. As such we decided to test a method known as Common Neighbours [13]. If we define $\Gamma(i) \cap \Gamma(j)$ 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 which 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 which is sometimes higher than preferential attachment and sometimes lower than it but is still consistently quite close with the best learning-based models.

E. M5: Features + ML

This method is based on [42] with a modification disclosed in the VI C. First, 10 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 (PCA) [43] is applied to reduce the correlation between features, speed up the learning process and improve generalization, which results in a final set of 7 latent variables. Lastly, a random forest classifier is trained (using a balanced dataset) to estimate the likelihood of new links between the AI concepts.

F. M6: Features+ML

The baseline solution for the Science4Cast competition was closely related to the model presented in [12]. 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, these 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, these are 3 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.

G. M7: end-to-end ML (Transformers)

This model, which is detailed in [44], does not use any handcrafted 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 $\boldsymbol{A}_t$ for $t = 1994, \ldots, 2019$. We then embed each of these graphs into 128-dimensional Euclidean space via Node2vec [45, 46]. For each node u in the semantic graph, we extract different 128-dimensional vector embeddings $\boldsymbol{n}_u(\boldsymbol{A}_{1994}), \ldots, \boldsymbol{n}_u(\boldsymbol{A}_{2019})$.

Transformers have performed extremely well in natural language processing tasks [47], 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, 2048 as the feedforward dimension and 8-headed attention. This transformer acts as our feature extractor. Once we pre-train our transformer, we add a 2-layer ReLU network with hidden dimension 128 as a classifier on top.

H. M8: end-to-end ML (auto node embedding)

The most immediate way one can apply machine learning 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. In our approach, we use the ProNE embedding [48], which is based on sparse matrix factorizations modulated by the higher-order Cheeger inequality [49], as well as Node2Vec [45]. We use the implementations provided in the `nodevectors` Python package [50].

The embeddings learn a 32-dimensional representation for each node; hence, each edge representation is normalized to a single point in $[0, 1]^{64}$, and the concatenated features are the input of a neural network with two hidden layers of size 1000 and 30, respectively. Similarly to M6, the model is then tasked with computing the probability for the evaluation examples, which lets us determine the ROC.

V. EXTENSIONS AND FUTURE WORK

Creating an AI that can suggest research topics to human scientists is highly ambitious and challenging. The present work of link prediction for a temporal network to draw connections between existing concepts is only the first step. We point out several extensions and future works that are directly relevant to the overarching goal of AI for AI.

High-quality predictions without feature engineering – Surprisingly, given a graph with already extracted concepts as nodes and edges plotting the time evolution of joint appearance of these concepts in publications, the most powerful methods all used carefully hand-crafted features. It will be interesting to see whether end-to-end deep learning methods can solve tasks without feature engineering.

Fully automated concept extraction – The concept list at the moment is created by a purely statistical text analysis using RAKE. The suggestions by RAKE are then manually inspected and phrases that do not correspond to a *concept* are removed. While this process can be partially automated (as RAKE often makes the same mistakes which can be captured automatically), it is not a scalable process if one wants to create concept lists for the much larger corpus of science and engineering. A fully automated natural language processing algorithm that can extract meaningful concepts with minimal mistakes would be extremely useful.

Generation of new concepts – Here we predict the emergence of links between two known concepts. One important question is whether an AI algorithm can compose words and generate new concepts. Different from the current work that is mostly supervised, the generation of new concepts is *unsupervised*, hence more difficult. One approach to address this question has been presented in [51, 52]. There the authors can detect clusters of concepts with specific dynamics that indicate the formation of a new concept. It will be interesting to see how such emerging concepts can be incorporated into the current framework and used for suggestions for new research topics.

Semantic information beyond concept pairs – At the moment, every article's abstract and title are compressed into several links between concept pairs. This procedure does not represent all information in the article's abstract (let alone, the article itself). The more information one can extract from the article, the more meaningful the predictions and suggestions will be. Extending the representation of the semantic network to more complex data structures, such as hypergraphs [53] are likely to be computationally more demanding but could significantly improve the prediction qualities. It might be also possible to find some ways to decrease the complexity of the analysis using clever tricks. For example, the authors in [54] showed that the maximum node and hyperedge cover problem, two computational NP-hard problems, can be solved in polynomial time for most of the real-world hypergraphs tested. Whether such tricks exist for hyperlink prediction is still an open problem. The inclusion of sociological factors, such as the status of the involved researchers and their affiliations might help in prediction tasks.

Predictions of scientific success – The prediction of a new link between nodes in the semantic network means that we predict which concepts scientists will study for the first time in the future. This prediction however does not say anything about the potential importance and impact of the new connection. As a tool for high-quality suggestions, we need to introduce the prediction of a *metric-of-success*, for example, estimated citation numbers of the new link or the rate of citation growth over time. This extension seems reasonable given that modelling and predictions of citation information in citation networks (where nodes are papers) is a prominent area of research within the science of science [55, 56]. Adapting these techniques to semantic networks will be an interesting future research direction.

Anomaly detections – In a way, predicting the most likely new connection between concepts does not necessarily directly coincide with the goal of suggestions of new surprising research directions. After all, those links are predictable, thus potentially not surprising by themselves. While we believe that this type of prediction can still be a very useful contribution for *suggestions*, there is another way to more directly find surprising combinations, namely by finding anomalies in the semantic networks. Those are potential links that have extreme properties in some metrics. There are powerful deep learning methods for anomaly detection [57, 58] and their application in the semantic network presented here might be very interesting. In fact, while scientists tend to study topics in which they are already directly involved [2, 3], often higher scientific impact results from the unexpected combination of more distant domains [10], which foster the search for those surprising and impactful associations.

End-to-end formulation – As outlined above, we necessarily decomposed our goal of extracting knowledge from the scientific literature into two subtasks: extracting concepts and building and predicting the evolution of a semantic network resulting from those concepts. This stands in contrast to the dominant paradigm in deep learning that emerged over the last decade of so-called 'end-to-end' training based on early spectacular successes

[59–62]. In this paradigm, problems are not broken into sub-problems but solved directly using deep differentiable architecture components trained via back-propagation [63, 64]. If such an 'end-to-end' solution approach to our goal could be achieved it would be interesting to see whether it could replicate the success this deep learning paradigm had in other areas.

Human level machine comprehension – One of the defining goals of the Dartmouth Summer Research Project on Artificial Intelligence in 1956 was the following: 'An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves.' [65]. Such an algorithm would be expected to handle an evolution in concept denotations due to new insights (i.e. the emergence of the term 'Gibbs entropy' to distinguish Boltzmann's original concept of thermodynamical entropy as opposed to seeing it in the light of the more general emergent 'Shannon entropy' or 'von Neumann Entropy') or due to disputed originality (i.e. Bolai-Lobatchevskian Geometry and Hyperbolic Geometry are the same concept). An algorithm with such natural language understanding capabilities would thus be extremely useful to get closer to our goal. Although large language models and other multimodally trained language models like CLIP [66] or CLOOB [67] have achieved outstanding results recently, it is an open question how much statistically trained natural language models alone could eventually form concepts and abstractions on a human level [68, 69].

VI. CONCLUSION

Here we present a new AI benchmark for link prediction in exponentially growing semantic networks. Several of the solutions have been collected in the IEEE BigData Competition *Science4Cast* in fall 2021, and generalized to the mode diverse tasks presented here. The goal was to boost the capabilities of predicting future research directions in the field of AI itself, which grows enormous over the decade. This ability might be an important part of a tool that gives personalized research suggestions to human scientists in the future. We find, rather surprisingly, that the prediction of strong new links (those that are formed three or more times) can be predicted with extremely high quality (AUC beyond 99%). It will be interesting to investigate this quasi-deterministic pattern in AI research in more detail. The best methods used a clever combination of hand-crafted features and machine learning. It will be interesting whether pure learning methods, without hand-crafted features, will achieve high-quality results in the future. We also point out a number of open problems towards the goal of practical, personalized, interdisciplinary AI-based suggestions for new impactful research direction – which we believe could become a disruptive tool in the future.

APPENDIX

A. Model availability

All of the models described above can be found on GitHub. M1, M2, M3, M4, M5, M6, M7, M8.

B. Details on M9

The solution M9 was not part of the `Science4Cast` competition and therefore not described in the corresponding proceedings, thus we want to add more details. We compare the ProNE embedding to Node2Vec, which is also commonly used for graph embedding problems. The algorithm maps each node of the network to a point in 32-dimensional space based on a biased random walk procedure, which is fundamentally parameterized by two variables—p, the "return parameter", and q, the "in-out parameter". The return parameter determines the frequency of backtracking in the random walk, while the in-out parameter determines whether to bias the exploration to nearby nodes or distant nodes. Notably, these parameters significantly affect how the network is encoded—for instance, in the BlogCatalog dataset, optimal parameters were $p = 0.25, q = 0.25$, whereas for the Wikipedia graph, they were $p = 4, q = 0.5$ [45]. In initial experiments, we used the default $p = q = 1$ for a 64-dimensional encoding, before feeding it into the same neural network as for the ProNE experiment. The higher variance in Node2Vec-based predictions likely has to do with the method's significant sensitivity to its hyperparameters. While ProNE is clearly better suited for a general multi-dataset link prediction problem, Node2Vec's parameter sensitivity may help us identify what features of the network are most important for predicting its temporal evolution.

C. Consideration for Model M6

In this manuscript, a modification was performed in relation to the original formulation of the method [42]: two of the original features, average neighbor 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.

ACKNOWLEDGEMENTS

The authors thank IARAI Vienna and IEEE for supporting and hosting the IEEE BigData Competition *Science4Cast*. We are specifically grateful to David Kreil, Moritz Neun, Christian Eichenberger, Markus Spanring, Henry Martin, Dirk Geschke, Daniel Springer, Pedro Herruzo, Marvin McCutchan, Alina Mihai, Toma Furdui, Gabi Fratica, Miriam Vázquez, Aleksandra Gruca, Johannes Brandstetter and Sepp Hochreiter for helping to set up and successfully execute the competition and the corresponding workshop. The authors thank Xuemei Gu for creating Fig.2, and Milad Aghajohari and Mohammad Sadegh Akhondzadeh for helpful comments on the manuscript. The work of HL, RS, and JGF were supported by grant TWCF0333 from the Templeton World Charity Foundation. HL is additionally supported by NSF grant DMS-1952339. JPM acknowledges the support of FCT (Portugal) through scholarship SFRH/BD/144151/2019. BC 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. NMT and YX 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.

[1] James A Evans and Jacob G Foster, "Metaknowledge," Science **331**, 721–725 (2011).

[2] Santo Fortunato, Carl T Bergstrom, Katy Börner, James A Evans, Dirk Helbing, Staša Milojević, Alexander M Petersen, Filippo Radicchi, Roberta Sinatra, Brian Uzzi, *et al.*, "Science of science," Science **359**, eaao0185 (2018).

[3] Dashun Wang and Albert-László Barabási, *The science of science* (Cambridge University Press, 2021).

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, *et al.*, "Language models are few-shot learners," Advances in neural information processing systems **33**, 1877–1901 (2020).

[5] Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, *et al.*, "Scaling language models: Methods, analysis & insights from training gopher," arXiv:2112.11446 (2021).

[6] Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, *et al.*, "Using deepspeed and megatron to train megatron-turing nlg 530b, a large-scale generative language model," arXiv:2201.11990 (2022).

[7] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, *et al.*, "Palm: Scaling language modeling with pathways," arXiv:2204.02311 (2022).

[8] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa, "Large language models are zero-shot reasoners," arXiv:2205.11916 (2022).

[9] Honghua Zhang, Liunian Harold Li, Tao Meng, Kai-Wei Chang, and Guy Van den Broeck, "On the paradox of learning to reason from data," arXiv:2205.11502 (2022).

[10] Andrey Rzhetsky, Jacob G Foster, Ian T Foster, and James A Evans, "Choosing experiments to accelerate collective discovery," Proceedings of the National Academy of Sciences **112**, 14569–14574 (2015).

[11] Jacob G Foster, Andrey Rzhetsky, and James A Evans, "Tradition and innovation in scientists research strategies," American Sociological Review **80**, 875–908 (2015).

[12] Mario Krenn and Anton Zeilinger, "Predicting research trends with semantic and neural networks with an application in quantum physics," Proceedings of the National Academy of Sciences **117**, 1910–1916 (2020).

[13] David Liben-Nowell and Jon Kleinberg, "The link-prediction problem for social networks," Journal of the American society for information science and technology **58**, 1019–1031 (2007).

[14] István Albert and Réka Albert, "Conserved network motifs allow protein–protein interaction prediction," Bioinformatics **20**, 3346–3352 (2004).

[15] Tao Zhou, Linyuan Lü, and Yi-Cheng Zhang, "Predicting missing links via local information," The European Physical Journal B **71**, 623–630 (2009).

[16] István A Kovács, Katja Luck, Kerstin Spirohn, Yang Wang, Carl Pollis, Sadie Schlabach, Wenting Bian, Dae-Kyum Kim, Nishka Kishore, Tong Hao, *et al.*, "Network-based prediction of protein interactions," Nature communications **10**, 1–8 (2019).

[17] Alessandro Muscoloni, Ilyes Abdelhamid, and Carlo Vittorio Cannistraci, "Local-community network automata modelling based on length-three-paths for prediction of complex network structures in protein interactomes, food webs and more," bioRxiv , 346916 (2018).

[18] Ratha Pech, Dong Hao, Yan-Li Lee, Ye Yuan, and Tao Zhou, "Link prediction via linear optimization," Physica A: Statistical Mechanics and its Applications **528**, 121319 (2019).

[19] Linyuan Lü, Liming Pan, Tao Zhou, Yi-Cheng Zhang, and H Eugene Stanley, "Toward link predictability of complex networks," Proceedings of the National Academy of Sciences **112**, 2325–2330 (2015).

[20] Roger Guimerà and Marta Sales-Pardo, "Missing and spurious interactions and the reconstruction of complex networks," Proceedings of the National Academy of Sciences **106**, 22073–22078 (2009).

[21] Amir Ghasemian, Homa Hosseinmardi, Aram Galstyan, Edoardo M Airoldi, and Aaron Clauset, "Stacking models for nearly optimal link prediction in complex networks," Proceedings of the National Academy of Sciences **117**, 23393–23400 (2020).

[22] Tao Zhou, "Progresses and challenges in link prediction," Iscience **24**, 103217 (2021).

[23] Mario Krenn, Robert Pollice, Si Yue Guo, Matteo Aldeghi, Alba Cervera-Lierta, Pascal Friederich, Gabriel dos Passos Gomes, Florian Häse, Adrian Jinich, AkshatKumar Nigam, *et al.*, "On scientific understanding with artificial intelligence," arXiv:2204.01467 (2022).

[24] Stuart Rose, Dave Engel, Nick Cramer, and Wendy Cowley, "Automatic keyword extraction from individual documents," Text mining: applications and theory **1**, 1–20 (2010).

[25] Jeff Alstott, Ed Bullmore, and Dietmar Plenz, "powerlaw: a python package for analysis of heavy-tailed distributions," PloS one **9**, e85777 (2014).

[26] Trevor Fenner, Mark Levene, and George Loizou, "A model for collaboration networks giving rise to a power-law distribution with an exponential cutoff," Social Networks **29**, 70–80 (2007).

[27] Anna D Broido and Aaron Clauset, "Scale-free networks are rare," Nature communications **10**, 1–10 (2019).

[28] Tom Fawcett, "Roc graphs: Notes and practical considerations for researchers," Machine learning **31**, 1–38 (2004).

[29] Yichao Lu, "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)* (IEEE, 2021) pp. 5809–5814.

[30] Sergey Brin and Lawrence Page, "The anatomy of a large-scale hypertextual web search engine," Computer networks and ISDN systems **30**, 107–117 (1998).

[31] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu, "Lightgbm: A highly efficient gradient boosting decision tree," Advances in neural information processing systems **30** (2017).

[32] Paul W Holland and Samuel Leinhardt, "Transitivity in structural models of small groups," Comparative group studies **2**, 107–124 (1971).

[33] Duncan J Watts and Steven H Strogatz, "Collective dynamics of small-worldnetworks," nature **393**, 440–442 (1998).

[34] Jheng-Hong Yang, Chih-Ming Chen, Chuan-Ju Wang, and Ming-Feng Tsai, "Hop-rec: high-order proximity for implicit recommendation," in *Proceedings of the 12th ACM Conference on Recommender Systems* (2018) pp. 140–144.

[35] Bo-Yu Lin, "Ogb collab project," https://github.com/brucenccu/OGB_collab_project (2021).

[36] Th A Sorensen, "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).

[37] Ngoc Mai Tran and Yangxinyu Xie, "Improving random walk rankings with feature selection and imputation science4cast competition, team hash brown," in *2021 IEEE International Conference on Big Data (Big Data)* (IEEE, 2021) pp. 5824–5827.

[38] Nima Sanjabi, "Efficiently predicting scientific trends using node centrality measures of a science semantic network," in *2021 IEEE International Conference on Big Data (Big Data)* (IEEE, 2021) pp. 5820–5823.

[39] In-Kwon Yeo and Richard A Johnson, "A new family of power transformations to improve normality or symmetry," Biometrika **87**, 954–959 (2000).

[40] João P Moutinho, Bruno Coutinho, and Lorenzo Buffoni, "Network-based link prediction of scientific concepts–a science4cast competition entry," in *2021 IEEE International Conference on Big Data (Big Data)* (IEEE, 2021) pp. 5815–5819.

[41] Albert-László Barabási, "Network science," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **371**, 20120375 (2013).

[42] Francisco Valente, "Link prediction of artificial intelligence concepts using low computational power," in *2021 IEEE International Conference on Big Data (Big Data)* (2021) pp. 5828–5832.

[43] Ian T. Jolliffe and Jorge Cadima, "Principal component analysis: a review and recent developments," Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **374**, 20150202 (2016).

[44] Harlin Lee, Rishi Sonthalia, and Jacob G Foster, "Dynamic embedding-based methods for link prediction in machine learning semantic network," in *2021 IEEE International Conference on Big Data (Big Data)* (IEEE, 2021) pp. 5801–5808.

[45] Aditya Grover and Jure Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (2016) pp. 855–864.

[46] Renming Liu and Arjun Krishnan, "PecanPy: a fast, efficient and parallelized Python implementation of node2vec," Bioinformatics **37**, 3377–3379 (2021), https://academic.oup.com/bioinformatics/article-pdf/37/19/3377/40556655/btab202.pdf.

[47] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, "Attention is all you need," ArXiv **abs/1706.03762** (2017).

[48] Jie Zhang, Yuxiao Dong, Yan Wang, Jie Tang, and Ming Ding, "Prone: Fast and scalable network representation learning," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19* (International Joint Conferences on Artificial Intelligence Organization, 2019) pp. 4278–4284.

[49] Afonso S. Bandeira, Amit Singer, and Daniel A. Spielman, "A cheeger inequality for the graph connection laplacian," (2012).

[50] Matt Ranger, "nodevectors," `https://github.com/VHRanger/nodevectors` (2021).

[51] Angelo A Salatino, Francesco Osborne, and Enrico Motta, "How are topics born? understanding the research dynamics preceding the emergence of new areas," PeerJ Computer Science **3**, e119 (2017).

[52] Angelo A Salatino, Francesco Osborne, and Enrico Motta, "Augur: forecasting the emergence of new research topics," in *Proceedings of the 18th ACM/IEEE on joint conference on digital libraries* (2018) pp. 303–312.

[53] Federico Battiston, Enrico Amico, Alain Barrat, Ginestra Bianconi, Guilherme Ferraz de Arruda, Benedetta Franceschiello, Iacopo Iacopini, Sonia Kéfi, Vito Latora, Yamir Moreno, *et al.*, "The physics of higher-order interactions in complex systems," Nature Physics **17**, 1093–1098 (2021).

[54] Bruno Coelho Coutinho, Ang-Kun Wu, Hai-Jun Zhou, and Yang-Yu Liu, "Covering problems and core percolations on hypergraphs," Phys. Rev. Lett. **124**, 248301 (2020).

[55] Hanwen Liu, Huaizhen Kou, Chao Yan, and Lianyong Qi, "Link prediction in paper citation network to construct paper correlation graph," EURASIP Journal on Wireless Communications and Networking **2019**, 1–12 (2019).

[56] Niklas Reisz, Vito D P Servedio, Vittorio Loreto, William Schueller, Márcia R Ferreira, and Stefan Thurner, "Loss of sustainability in scientific work," New Journal of Physics **24**, 053041 (2022).

[57] Donghwoon Kwon, Hyunjoo Kim, Jinoh Kim, Sang C Suh, Ikkyun Kim, and Kuinam J Kim, "A survey of deep learning-based network anomaly detection," Cluster Computing **22**, 949–961 (2019).

[58] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel, "Deep learning for anomaly detection: A review," ACM Computing Surveys (CSUR) **54**, 1–38 (2021).

[59] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa, "Natural language processing (almost) from scratch," Journal of machine learning research **12**, 2493–2537 (2011).

[60] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," Advances in neural information processing systems **25** (2012).

[61] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," nature **518**, 529–533 (2015).

[62] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, *et al.*, "Mastering the game of go with deep neural networks and tree search," nature **529**, 484–489 (2016).

[63] Tobias Glasmachers, "Limits of end-to-end learning," in *Asian conference on machine learning* (PMLR, 2017) pp. 17–32.

[64] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," nature **521**, 436–444 (2015).

[65] John McCarthy, Marvin L Minsky, Nathaniel Rochester, and Claude E Shannon, "A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955," AI magazine **27**, 12–12 (2006).

[66] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning* (PMLR, 2021) pp. 8748–8763.

[67] Andreas Fürst, Elisabeth Rumetshofer, Viet Tran, Hubert Ramsauer, Fei Tang, Johannes Lehner, David Kreil, Michael Kopp, Günter Klambauer, Angela Bitto-Nemling, *et al.*, "Cloob: Modern hopfield networks with infoloob outperform clip," arXiv:2110.11316 (2021).

[68] Melanie Mitchell, "Abstraction and analogy-making in artificial intelligence," Annals of the New York Academy of Sciences **1505**, 79–101 (2021).

[69] Yann LeCun, "A path towards autonomous machine intelligence," openreview preprint (2022).