Machine Learning Featurizations for AI Hacking of Political Systems

NATHAN E. SANDERS*, Berkman Klein Center, Harvard University, USA
BRUCE SCHNEIER†, Council for the Responsible Use of AI, Belfer Center for Science and International Affairs, Harvard Kennedy School, USA

What would the inputs be to a machine whose output is the destabilization of a robust democracy, or whose emanations could disrupt the political power of nations? In the recent essay "The Coming AI Hackers," Schneier (2021) proposed a future application of artificial intelligences to discover, manipulate, and exploit vulnerabilities of social, economic, and political systems at speeds far greater than humans' ability to recognize and respond to such threats. This work advances the concept by applying to it theory from machine learning, hypothesizing some possible "featurization" (input specification and transformation) frameworks for AI hacking. Focusing on the political domain, we develop graph and sequence data representations that would enable the application of a range of deep learning models to predict attributes and outcomes of political systems. We explore possible data models, datasets, predictive tasks, and actionable applications associated with each framework. We speculate about the likely practical impact and feasibility of such models, and conclude by discussing their ethical implications.

1 INTRODUCTION

1.1 Summary of AI Hacking

In "The Coming AI Hackers," Schneier [2021] defines hacking as an exploitation of a system that follows its rules, but subverts its intent. Despite the modern association of hacking with computer systems, this definition encompasses millennia of human activity:

Authors' addresses: Nathan E. Sanders, nsanders@cyber.harvard.edu, Berkman Klein Center, Harvard University, 23 Everett St #2, Cambridge, Massachusetts, USA, 02138; Bruce Schneier, Council for the Responsible Use of AI, Belfer Center for Science and International Affairs, Harvard Kennedy School, 79 JFK Street, Cambridge, Massachusetts, USA, 02138.

looopholes in tax law, for example. He argues that the computerization of diverse fields, from finance to elections, increases the speed, scale, and scope of vulnerability to hacking.

With respect to the hacking of computer systems, AI is making remarkable strides. Schneier cites several instances of specialized AI being developed and deployed to find vulnerabilities in computer code and systems automatically, enabling attackers to discover and exploit systems without human intervention [Schneier 2021, p. 21]. Schneier imagines a similar AI turned to hacking social systems such as the tax code and financial regulations, or legislative and other political processes. After all, these, like so many other systems of modern human life, are increasingly "socio-technical systems involving computers and networks"; this leaves the social aspects of the system exposed to its technical components.

The implications of this proposal are profound in that they provoke the thought of an unknowable future where machine-generated strategies can successfully dictate outcomes of democratic political processes, and may be controlled by malicious domestic or foreign actors. Analogizing by way of historical example, Schneier poses the question, "Could an AI independently discover gerrymandering?" How about the filibuster? His conclusion that "It'll be a long time before AIs will be capable of modeling and simulating the ways that people work, individually and in groups, and before they are capable of coming up with novel ways to hack legislative processes" raises questions: How would we get to that state? What approaches might AI hackers take to develop such capabilities? What conditions would need to be satisfied for them to work?

The purpose of this paper is not to advance towards practical AI hacking as a goal, but rather to more rigorously define it. We take the general perspective that, although there will be some benefits of the evolution of AI towards one capable of interacting competently with social systems, the advent of AI hacking as defined above would be fundamentally negative for civilization. Aided by a more concrete description of an AI system capable of discovering hacks of a political system, it may be possible to anticipate some of the approaches towards, and therefore ethical implications and potential dangers of, such an AI.

1.2 Overview of featurization

Machine learning (ML) applications generally require *structured* input data provided in the format of some specified "data model" (in the sense of, e.g., Rowe and Stonebraker

1987) that is tailored to the operational mechanics of the model. The selection of that data model is a foundational task for the application of machine learning to any domain.

There is a rich literature on the many aspects of this data model selection process, and a range of frameworks and methods that are applicable to it. [1] A longstanding viewpoint on data models for highly complex domains, such as human communications, is that data available in unstructured formats, such as natural language text, must be "refined" or "distilled" into more structured data suitable for algorithmic processing, namely some set of numerical vectors [McCallum 2005]. The field of "data mining" and "information extraction" presents myriad techniques for this distillation for natural language and other data types [Balducci and Marinova 2018]. Given input data in a format suitable for algorithmic manipulation, a primary responsibility of a machine learning developer is to do "feature engineering" or "feature extraction" [Khalid et al. 2014], meaning to cull predictors from the source data that are likely to be supportive of the predictive task targeted by the model. Machine learning systems often rely on "feature selection" [Kira and Rendell 1992], which enables models to isolate or preferentially focus on a reduced set of features that carry the greatest predictive potential. Generalizing this idea, the field of "representation learning" seeks to algorithmically construct a reduction of a complex input data format that will be optimal for some downstream predictive task or other use [Bengio et al. 2013]. "Multi-view" models are meant to "fuse" data from multiple sources into a single predictive framework [Li et al. 2016], while "multi-modal" models specifically incorporate data sources with categorically different kinds of input data models (such as text and images) that may each require drastically different data representations [Ngiam et al. 2011]. Tools for automatic "modality selection" aid multi-modal modeling by identifying and privileging data modalities with the greatest predictive importance [Xiao et al. 2019].

Ultimately, practical systems incorporating machine learning models may be viewed as a type of "pipeline" facilitating the flow of input and output data between different modeling components [Xin et al. 2021]. In order for this flow to proceed, the output data

[1] It should be noted that the aspects described here are by no means mutually exclusive. A particular modeling strategy may incorporate approaches associated with several of these concepts. Herein we cite a variety of seminal works and recent reviews to illustrate the major facets of each concept.

model from one component must match the input data model for the next, and the purpose of some components is to transform the data representation between data models.

We refer to the range of topics above in aggregate as "featurization."[2] We conceptualize featurization to include all steps necessary, both manual and automated, to express a complex real-world system of interest (e.g., a political process) into a mathematical format that an ML system can manipulate and operate upon.

Prime examples of common data models and featurizations widely applied in machine learning include the following:

- Images studied in computer vision, which are typically featurized as 2D or (with color information) 3D pixel arrays that can be operated on efficiently by models such as convolutional neural networks. These models learn representations encoding spatial information from the input and may discover visual patterns such as the presence of a face or object.
- Natural language text studied in the quantitative social sciences and other fields, which is typically featurized as a token (e.g., word or character) sequence that can be operated on by models such as recurrent neural networks and transformers. These models encode information about the composition and grammatical structure of a written document and may discover underlying meaning, such as references to named entities, semantic relationships, description, sentiment, or emotion.
- Molecules studied in cheminformatics are often represented by molecular graphs, which are composed of nodes (atoms) and edges (bonds). These nodes and edges may each carry their own feature vectors describing, for example, the elemental properties of the atom and bond type. These graphs can be operated on by graph neural networks that encode information about the local and global structure of the molecular graph and may discover functional groups or other substructures within the molecule that are responsible for manifesting chemical properties or bioactivity.

Specialized AI and specifically deep learning have already been applied to a variety of topics in political science, such as extracting features from political documents, measuring

[2]The term of art "featurization" is used inconsistently. In general purpose machine learning, it is used to mean the automated process of transforming and normalizing structured variables. In bioinformatics, it usually refers to a learned low dimensional representation of a more complex data structure. We adopt here our own somewhat expansive definition.

polarization, optimizing the geographic distribution of aid, encoding the ideology of political actors, and more [Chatsiou and Mikhaylov 2020]. Below we explore other potential applications of AI to political processes by considering predictive tasks of potential interest to AI hackers.

2 FRAMEWORKS FOR POLITICAL FEATURIZATION

Here we consider possible featurizations for political systems that would enable predictive tasks potentially exploitable by AI hackers; specifically, graph and sequence modeling frameworks. In each case, we will provide a didactic description of the political system and its essential elements. We will then frame the same elements in mathematical terms as a representation suitable for machine learning, and finally suggest predictive tasks associated with this representation.

2.1 Graphs

Consider a network (graph) of political actors, where each node/vertex is an agent such as a person or institution and each edge represents a relationship between those actors. Edges connecting nodes could represent communication pathways between actors, such as lobbying or constituent relationships, hierarchical relations of reporting/power, or combinations of these and other relationship types. The communication pathways may be one-way or bidirectional and may emerge or change status over time. In this conception, the manifestation of political outcomes is a consequence of communications between actors in the graph. The graphs may therefore be associated with outcomes such as the legislative disposition of a bill, the time efficiency of a process (how long it takes for legislation to move or an executive action to be taken), or the inclusion of a particular provision in a policy document.

In such a graph, the nodes are differentiated by their position in the network as well as by features such as the type of actor they represent (e.g., individual or organization), their level (e.g., position within government), their magnitude of power (e.g., seniority, budget size, constituency, etc.), and any other descriptor that may be anticipated to mediate the actor's role in the political process. Edges may be differentiated based on the type of relationship they represent (e.g., a constituent appeal to a representative, a lobbyist's influence on a legislator, a committee vote exercised by a member, or a backroom working

relationship), the volume or frequency of communication, its age or status (e.g., current, former, or even future), and any other descriptor of the relationship's role in the political process. Each of these features may constitute a predictor of the outcome targeted by the model.

Nodes could even represent other entities in the political network beyond individual or organizational agents, such as issues, specific pieces of legislation, budget line items, and so on. Different edge types would be associated with each pair of node types; for example, the edge between a legislator and a piece of legislation could be a voting edge featurized by the legislator's current position on the legislation as well as a vector describing their voting history on the issue.

There could be many such graphs representing various parts of the political process, such as the networks of legislative relationships across a set of committees, or the networks of lobbying relationships between a legislature and a set of different interest areas. Those graphs could carry features such as historical outcomes of the modeled process (e.g., a bill is passed or a corporation reaches a certain market cap.)

Mathematically (following, e.g., the notation of Gong and Cheng 2019 and Muzio et al. 2021), each graph $G_k = (V, E)$ among the total number of graphs K has nodes/vertices V, which number $n = |V|$, and edges E. Each individual edge $e_{i,j}$ connects two nodes v_i and v_j. The graph may be directed and weighted, in which case it can be represented by the combination of a non-symmetric adjacency tensor $A \in \mathbb{R}^{n,n,p}$, where p is the number of edge features, and node feature matrix $X \in \mathbb{R}^{n,m}$, where m is the number of features that describe each node. The graphs may have an associated vector of labels or features comprising the matrix $Y \in \mathbb{R}^{K,M}$, where M is the dimensionality of the graph features. These symbols are visualized on a graph diagram in Figure 1.

A variety of predictive tasks are enabled by such a representation in combination with a graph learning model such as one in the diverse class of graph neural networks (GNN) like graph convolutional neural networks and graph attention networks [Muzio et al. 2021]. These tasks include:

- Graph label prediction (or graph classification), in which a global property (label) of a graph is predicted based on characteristics of its network structure and other meta-data. The hacker could, for example, predict the outcome of a political process given

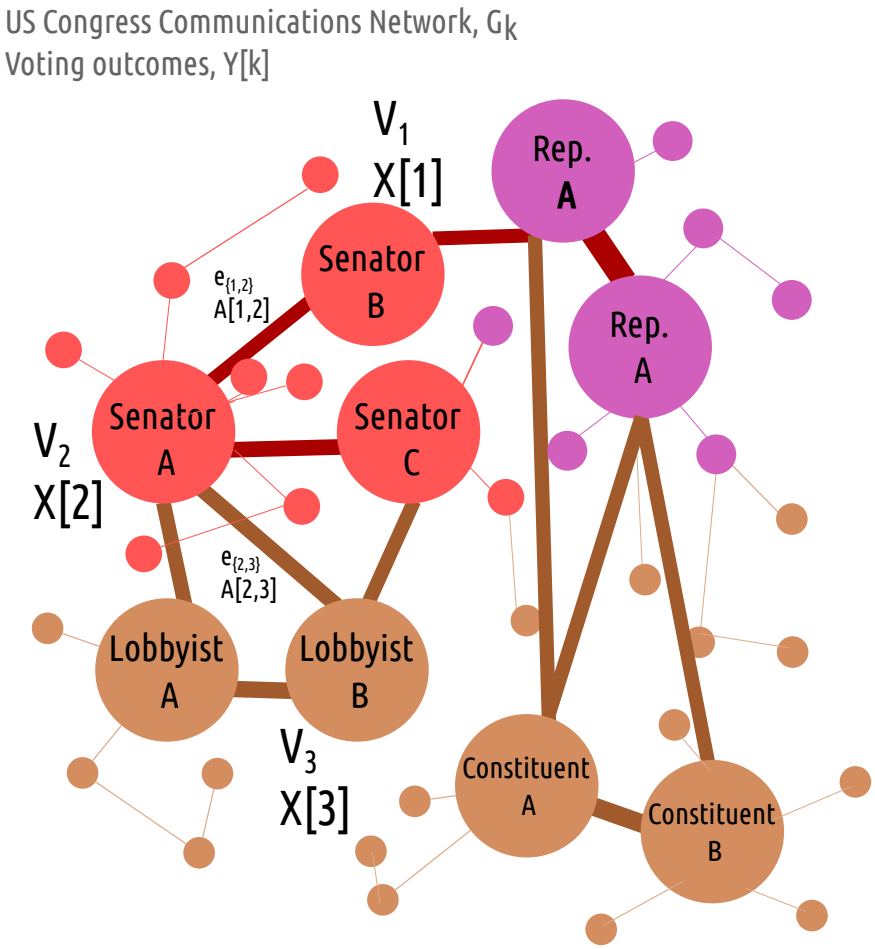

Fig. 1. Illustration of a local neighborhood within a hypothetical graph representation of the US congressional legislative communication network, G_k. The graph has a feature vector, $Y[k]$, that may represent, e.g., the body's voting outcomes across a set of bills. Multiple types of nodes V are represented by circles, labeled as different individual members of the network: senators, representatives, lobbyists, and constituents. (Smaller circles represent other nodes outside of the example local network.) The nodes have feature vectors (e.g., $X[2]$ for Senator A) that represent, for example, the node type (color). Edges (lines) connect the individuals; for example, edge $e_{\{2,3\}}$ connects Senator A (V_2) to Lobbyist B (V_3). The edge has a feature vector $A[2, 3]$; for example, the width of the line may represent frequency of communication and the color may represent the type of relationship.

a particular configuration of the political actor network. Such a predictive framework can become actionable as, for example, a search (optimization) for instantiations where the favored outcome is most likely. For example, the model could be used to nominate a jurisdiction that may be most favorable to the introduction of legislation. Alternatively, a hacker could assess whether the probability of a given outcome would increase or decrease if a particular edge (communication pathway) were added to the network. The AI hacker could then act on this prediction by encouraging collaboration between two actors in the network.

- Link prediction, in which the presence of an unknown edge in a network is inferred based on its local structural properties. For example, a consistent pattern of similar actions by two political actors (nodes) with otherwise distinctive properties could imply communication (an edge) between them. A hacker targeting an inaccessible political actor could exploit this information by identifying an accessible third party actor that is discovered to be covertly in communication with the target. This could allow the AI hacker to pressure their target, without exposing their identity directly to them and without leaving any visible signature of direct communication to them. An AI hacker could even blackmail an actor whom they can demonstrate is inappropriately communicating with another actor in the network, such as a super PAC that is unlawfully coordinating expenditures with a candidate.
- Node attribute prediction (or classification), in which a property of a node is predicted based on its position within a network and other features. For example, a political actor's unstated position on an issue could be inferred based on the positions of their neighbors in the network. An AI hacker could gain an advantage by identifying and targeting policymakers who may be most persuadable on an issue. An AI hacker seeking to influence an election could also use node attribute prediction to assess the probability of a slate of potential candidates to enter an electoral race, enabling them to offer key early campaign contributions to undeclared candidates who might then become beholden to demands of the hacker.
- Inference on node and edge feature weights or substructures, in which a model trained on historical data reveals the relative importance of each feature of its nodes and edges. For example, the trained weights of a fitted model for voting outcomes of a legislative body may support the inference that one factor (e.g., party

alignment) is far more important than another (e.g., communication frequency) in predicting the voting behavior of each legislator. This insight could give an AI hacker a distinct advantage in proposing a legislative strategy. Techniques also exist to extract explainable substructures of graphs that are associated with certain outcomes [Yuan et al. 2021]. For example, an AI hacker might identify a pattern such as a voting block of legislators from the same region that share a particular position on a secondary issue that strongly predicts their behavior on another issue. Such an insight could help an AI hacker to propose a communication or funding strategy targeted to that legislative block. Moreover, this strategy is perhaps the most relevant to the charge of finding an AI system that could discover gerrymandering, which itself represents a recurring local substructure in a geographic network of constituent-district assignments. In practice, it can be impractical to interpret or "explain" the complex layers of weights in deep learning models, so a predictive system that is interpretable by design may be preferable for this task [Rudin 2019].

2.2 Sequences

Consider a sequence (an ordered list of items) of political activities, where each item is an action taken by some political actor. Examples of actions could be steps in the legislative process for a bill, enforcement actions taken by a regulatory agency, electoral outcomes, and so on. Each action may have some outcome associated with it, such as the size of fine issued by a regulator or the vote share in an election.

The actions in the sequence may have multivariate features that differentiate them. Such features may include an indicator variable for the actor who took the action, the type of action, the time it was taken, the jurisdiction of the action, the entity or topic it is related to, some measure of the magnitude of the action, background factors such as a politician's approval rating or a company's stock price, and so on.

There are diverse machine learning methods and tasks associated with sequence modeling. Linear models such as the autoregressive integrated moving average (ARIMA) are frequently used to forecast future events based on historical sequences and their outcomes. In the deep learning domain, recurrent neural networks (RNNs) have been highly successful. Surprisingly, convolutional neural networks, which had been more often used for image modeling and computer vision, have also proven highly effective [Bai et al. 2018].

Mathematically (following the notation of, e.g., Bai et al. 2018), a sequence is composed of events, x_t, distributed over a time range, $t \in [0 - T]$, each with a corresponding outcome, y_t. The variable x can be multi-dimensional, carrying a set of event features, and likewise the outcome y can be multivariate. A sequence model or "seq2seq" model is a mapping function, f, from event sequences, x, to predicted outcome sequences, $\hat{y}$ that is, $\hat{y}_0 \ldots \hat{y}_T = f(x_0 \ldots x_T, A_t)$. The tensor A_t generically denotes an internal representation of the event sequence (i.e., an embedding) learned by the model. In timeseries applications, a causality constraint is typically applied such that the inputs to f for predicting $\hat{y}_t$ are limited to $x_0 \ldots x_t$, excluding any future values of x at time $> t$. This is unnecessary for many sequence modeling applications; for example, bidirectional networks of natural language take into account both previous and subsequent textual tokens (see, e.g., Huang et al. 2015 and Devlin et al. 2018). Such a system is illustrated in Figure 2.

ML tasks enabled by such a representation could include the following:

- Supervised regression. In this task, a sequence input is used to predict an outcome label or some other result variable. An AI hacker could evaluate the most likely outcome from a given sequence of events–for example, predicting the probability that a bill would be withdrawn if a particular lobbyist were to contact its lead sponsor prior to the first hearing. This corresponds to the generation of the outcome, $\hat{y}_t$, in Figure 2.
- Sequence generation. An AI hacker could extrapolate from a series of actions by having a model generate the next action likely to be taken and its features. In this way, they could game out a range of likely responses to an action taken under their control, or identify the optimal sequence of events that would maximize the probability of a desired outcome. Moreover, a probabilistic approach to sequence generation would allow an attacker to not only weigh the probabilities of a desired outcome in any individual circumstance, but also to manage a portfolio of attacks distributed over time or in different jurisdictions to maximize their collective potential. This corresponds to the generation of the next event bit vector, x_{t+1}, in Figure 2.
- Network inference. It is possible to infer the presence of links between political actors based on patterns in their actions, for example through point process network modeling [Fox et al. 2021; Linderman and Adams 2015]. An AI hacker might use such

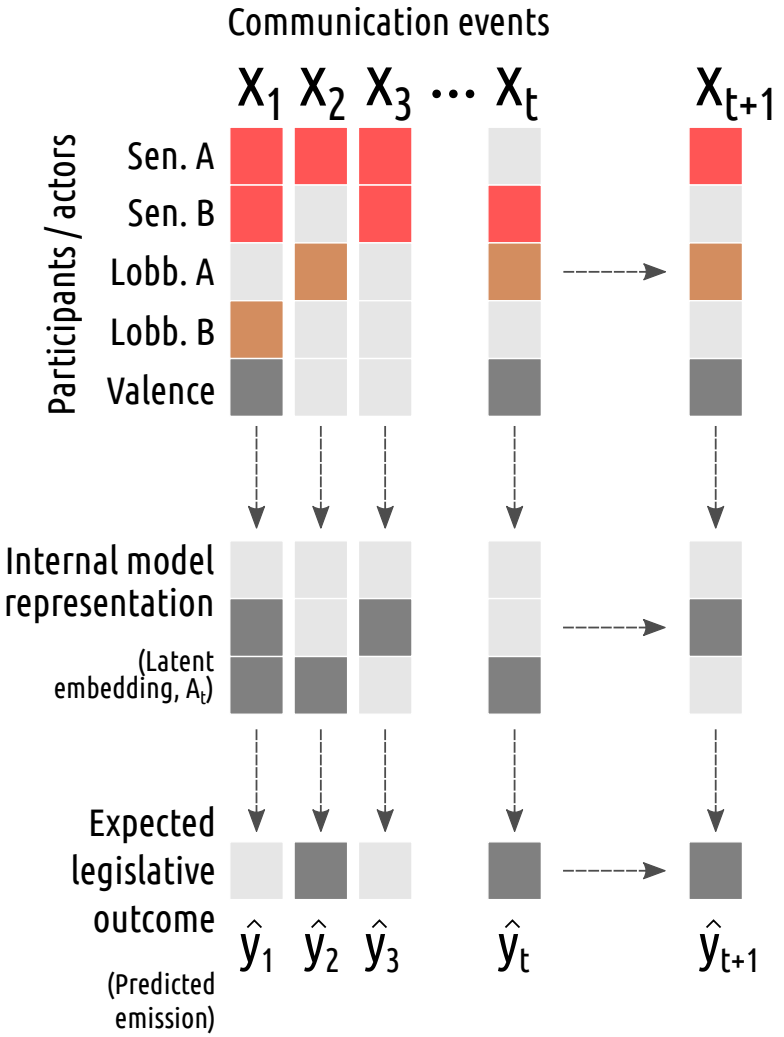

Fig. 2. Illustration of a legislative process modeled as a sequence of communication events leading to a time-dependent legislative outcome. The communication events are associated with the presence (colored blocks) or absence (grey blocks) of a set of political actors, which together comprise a binary bit vector (x_t). The sequence model translates the event bit vector and its history ($x_{<t}$) to an internal representation, the latent embedding A_t. The model then predicts an instantaneous expectation for a legislative outcome, $\hat{y}_t$, based on the latent embedding, The model can also extrapolate from the observed timeseries to a future communication event, x_{t+1}, and its associated expected outcome, $\hat{y}_{t+1}$.

a technique to, for example, construct a graph of legislative communications suitable for the methods of § 2.1 based on histories of vote or co-sponsorship sequences for a legislative body, or might uncover the most effective channels for voter persuasion around an issue based on timeseries data from social media capturing when users engaged with an issue-related hashtag.

3 FEASIBILITY

Several technical factors will limit the advancement of AI hacking in the political domain. However, in each case, we can anticipate advancements in modeling capabilities and data availability relieving those limitations over time.

First and foremost, all the predictive tasks envisioned above require the provision of labeled training data for model fitting. For example, training network models of the kind described above typically requires, for robust performance, hundreds of nodes for node prediction, thousands of edges for link prediction, and thousands of graphs for graph classification, and is scalable to hundreds of millions of entities [Hu et al. 2020]. We know of no existing dataset that has been curated specifically for modeling the aforementioned tasks in the socio-political domain. However, given that there are centuries of written records of the proceedings of various political systems in diverse jurisdictions, it should be possible to construct a fairly large dataset of, for example, legislative debate and lawmaking outcomes. Doing so may require painstaking analysis of historical records to reconstruct, for example, past communication networks among a legislative body. Alternatively, rather than reaching back in time, an engineer building an AI hacking system could use data mining techniques to capture information about a range of contemporary political systems [Adnan and Akbar 2019]. The advent of digitized communications and public records disclosures, or illicit leaks of those communications, make this scenario increasingly plausible [Stray 2019]. For example, a legislative communication network could be constructed from membership records with edges assigned naively based on shared committee memberships and leadership positions. Further, node attributes could be assigned based on party affiliation, districts, and past voting histories. Edge attributes could be assigned based on co-sponsorship histories. In jurisdictions where public hearings are routinely recorded or transcribed, characteristics of actual debate could also be featurized [Ruprechter et al. 2020].

Even in areas where data availability is fundamentally limited, modeling advancements may enable AI to generalize strategies learned from other datasets to successfully predict in the data-limited domain. A robust field of research on "transfer learning" is concerned with exactly this problem [Kouw and Loog 2018]. In particular, the fields of "few shot" and "zero shot" learning focus on how to make predictions on tasks with extremely limited datasets [Wang et al. 2020b; Xian et al. 2019]. For example, there may be instances where sufficient data exists on a modeled process, but not for a particular jurisdiction or set of political actors. There may be records on dozens of US states' enforcement response to emissions violations under air pollution regulations, but not yet data for a state that has newly adopted their regulatory framework. This may be considered a "domain shift" challenge and can be addressed through a variety of techniques, such as sample importance weighting [Wang et al. 2017]. Alternatively, there may be ample data on past actions by a set of political actors, but not for the targeted task. For example, there may be rich historical data on the US Congress' deliberations and actions on gun control legislation, but not the relatively nascent regulatory domain of cybersecurity. This can be considered a "domain adaptation" or, more specifically, a "concept shift" problem. It too can be addressed through a variety of techniques, including finding domain-invariant feature representations or transformations, multi-task learning, and pre-training [Farahani et al. 2020; Meftah et al. 2020].

In light of all these challenges, a more viable near-term threat may be human attackers doing AI-assisted AI hacking. This would allow AI systems that are not yet fully mature to contribute to attacks in more targeted, tightly scoped ways. For example, natural language processing (NLP) and understanding (NLU) models offer near-instantaneous analysis of copious textual documents that can be used to aid decision making. Particularly if applied to sensitive, private conversations (e.g. diplomatic cables leaked from the State Department or text messages harvested from hacked cell phones), such analysis could give a human political actor an unfair advantage.

In this paper, we have focused primarily on supervised learning examples where AIs are first trained with a fixed dataset of historical examples and then applied to predict characteristics of unmeasured or hypothetical entities. In some cases, it may also be possible to apply reinforcement learning techniques, which explore the response surface of a reward function to learn how to optimally exploit its structure (maximize reward).

For example, a mechanistic simulation of the political system (used as a reward function) can be used to train a reinforcement learner to take optimal actions in a real life political process. This methodology is analogous to the discussion of AIs learning to play the video game Breakout in Schneier [2021] and is similar to the use of a military war game to train combat strategists [e.g., Parkin 2020].

4 ETHICS, SAFEGUARDS, AND IMPLICATIONS

AI hacking poses a special challenge to the development of ethical AI systems. In this field, many (though certainly not all) solutions rely on regulatory engagement by the very state actors that are vulnerable to AI hacking [for recent reviews, see Cath 2018; Jobin et al. 2019]. Even in the absence of practical AI hacking, pressure for governments to take action on general-purpose machine learning has been—at best—overdue and hard-won [Rességuier and Rodrigues 2020]. The ability for an attacker to automatically disrupt legislative and regulatory action against them poses the risk of making AI hacking fundamentally ungovernable.

A pessimistic framing of this challenge is that of the "Red Queen's race," wherein (traditionally, human) competitors engage in a continuous struggle to one-up each other's advances and, potentially, retaliate against one another [Asaro 2019; Smuha 2021; Taddeo and Floridi 2018]. In a race to apply AI hacking tools, an aggressive party would be continuously extending their tools to overcome tactical, legal, or other barriers enacted by the defensive government or political system. However, if the aggressive party has unlocked the potential to automatically adjust their mode of attack in response to the actions of the defensive party, then the capacity of the latter party to escalate their defenses and keep up in the race may be short lived. Such a scenario may reflect more of a race against time or nature rather than a race between capable competitors. Much like the circumstances around climate change, where policymakers face a point of no return beyond which there would be critically diminished gains from further preventative action, there may be a limited time window over which government actors can effectively forestall the impact of AI hacking on political systems. According to popular surveys of experts in the field, this point of no return—based on the expected performance of AI generally—could be within just a few decades [e.g. Gruetzemacher et al. 2019].

However, the future need not proceed within this pessimistic frame. It may be possible to structurally limit the harm potential of AI hacking systems, although the adaptability of a successful AI hacking system may make the most resilient configuration unpredictable. For example, distributing power across multiple institutions in a political system by providing checks and balances can limit the damage associated with AI hacking of any one lever of power, yet it would also increase the "attack surface" exposed [as defined in cybersecurity, e.g., Adnan and Akbar 2019; Farrell and Schneier 2018]. Similarly, it may be a viable strategy to protect sensitive functions of government by exposing them transparently to public inspection, which (in a democracy) would provide feedback to a political system that has been corrupted by an AI hacker. Yet recent experience in democratic politics suggests that malign actors can influence and, perhaps, corrupt public opinion through digital means [Lin and Kerr 2019]. An effective AI hacker could manipulate "common knowledge" [Farrell and Schneier 2018] to override any outcry to their actions, even if publicly exposed.

These tradeoffs may suggest an effective strategy to control the damaging implementation of AI hacking through machine learning itself. A robust characterization of the performance sensitivity of practical AI hacking solutions to these tradeoffs could be generated by methods for probabilistic machine learning that help anticipate the generalization performance of models [e.g., Wilson and Izmailov 2020]. Such an analysis could determine what instantiations of a featurized political system would be least vulnerable to an AI hacker. This sensitivity surface could then be optimized to identify a political configuration that minimizes risk. Such an optimization would require complete knowledge of, or access to, the adversarial AI hacking algorithm, or at least a structurally similar one. Perversely, the best defense against an AI-algorithm hacker may be another, white hat defensive AI algorithm that can simulate and assess shortcomings in the attacking algorithm.

Another safeguard against AI hacking may be the inherent difficulty in hacking political systems, regardless of the sophistication of the machine learner. After all, reliably achieving political outcomes is a task that generations of humanity's own most well-meaning and intelligent actors—as well as malignant and/or less intelligent actors—have failed at. There are many tasks at which modern machine learning systems simply fail to perform. Worse, there are many tasks that ML systems may appear to solve, yet will actually fail to

generalize to more complex or realistic examples [D'Amour et al. 2020; Geirhos et al. 2020].

A tool to recognize when a policy has been manipulated could be a further safeguard against AI hacking. Likewise, the advent of "deepfakes" (hyperrealistic computer-generated audio and video) has spurred development of fake-spotting systems and models [Wang et al. 2020a]. Notwithstanding the potential for a sufficiently advanced AI to fool the spotting system, the need for such techniques could again motivate the systematic study of AI hacking by benign researchers.

Lastly, we note a structural inequity in the challenge posed by AI hacking to democratic systems. If a polity fears that policy changes may have been dictated by a manipulative AI system, they may be inclined to resist change and to introduce additional friction into the policymaking process. This may indeed be a valid mitigating factor against AI hacking. But, in this way, fear of AI hacking may promote conservative modes of governing that are skeptical of progressive change. The legitimate risks associated with practical applications of AI hacking in the present day, and their growth over time, should be carefully considered in any systemic response.

ACKNOWLEDGMENTS

We thank Rebecca Tabasky and the Berkman Klein Center for Internet and Society for facilitating conversations about this topic at the May 2021 Festival of Ideas event.

REFERENCES

Kiran Adnan and Rehan Akbar. 2019. An analytical study of information extraction from unstructured and multidimensional big data. *Journal of Big Data* 6, 1 (Dec. 2019), 91. https://doi.org/10.1186/s40537-019-0254-8

Peter Asaro. 2019. What is an artificial intelligence arms race anyway. *ISJLP* 15 (2019), 45. Publisher: HeinOnline.

Shaojie Bai, J Zico Kolter, and Vladlen Koltun. 2018. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018).

Bitty Balducci and Detelina Marinova. 2018. Unstructured data in marketing. *Journal of the Academy of Marketing Science* 46, 4 (July 2018), 557–590. https://doi.org/10.1007/s11747-018-0581-x

Y. Bengio, A. Courville, and P. Vincent. 2013. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 8 (Aug. 2013), 1798–1828. https:

//doi.org/10.1109/TPAMI.2013.50

Corinne Cath. 2018. Governing artificial intelligence: ethical, legal and technical opportunities and challenges. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 376, 2133 (Nov. 2018), 20180080. https://doi.org/10.1098/rsta.2018.0080

Kakia Chatsiou and Slava Jankin Mikhaylov. 2020. Deep Learning for Political Science. *arXiv:2005.06540 [cs]* (May 2020). http://arxiv.org/abs/2005.06540 arXiv: 2005.06540.

Alexander D'Amour, Katherine Heller, Dan Moldovan, Ben Adlam, Babak Alipanahi, Alex Beutel, Christina Chen, Jonathan Deaton, Jacob Eisenstein, Matthew D. Hoffman, Farhad Hormozdiari, Neil Houlsby, Shaobo Hou, Ghassen Jerfel, Alan Karthikesalingam, Mario Lucic, Yian Ma, Cory McLean, Diana Mincu, Akinori Mitani, Andrea Montanari, Zachary Nado, Vivek Natarajan, Christopher Nielson, Thomas F. Osborne, Rajiv Raman, Kim Ramasamy, Rory Sayres, Jessica Schrouff, Martin Seneviratne, Shannon Sequeira, Harini Suresh, Victor Veitch, Max Vladymyrov, Xuezhi Wang, Kellie Webster, Steve Yadlowsky, Taedong Yun, Xiaohua Zhai, and D. Sculley. 2020. Underspecification Presents Challenges for Credibility in Modern Machine Learning. *arXiv:2011.03395 [cs, stat]* (Nov. 2020). http://arxiv.org/abs/2011.03395 arXiv: 2011.03395.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R. Arabnia. 2020. A Brief Review of Domain Adaptation. *arXiv:2010.03978 [cs]* (Oct. 2020). http://arxiv.org/abs/2010.03978 arXiv: 2010.03978.

Henry John Farrell and Bruce Schneier. 2018. *Common-Knowledge Attacks on Democracy.* Technical Report 2018-7. Berkman Klein Center, Harvard University. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3273111

James Alan Fox, Nathan E Sanders, Emma E Fridel, Grant Duwe, and Michael Rocque. 2021. The Contagion of Mass Shootings: The Interdependence of Large-Scale Massacres and Mass Media Coverage. *Statistics and Public Policy* just-accepted (2021), 1–22. Publisher: Taylor & Francis.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence* 2, 11 (Nov. 2020), 665–673. https://doi.org/10.1038/s42256-020-00257-z

Liyu Gong and Qiang Cheng. 2019. Exploiting edge features for graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9211–9219.

Ross Gruetzemacher, David Paradice, and Kang Bok Lee. 2019. Forecasting Transformative AI: An Expert Survey. *arXiv:1901.08579 [cs]* (July 2019). http://arxiv.org/abs/1901.08579 arXiv: 1901.08579.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2020. Open graph benchmark: Datasets for machine learning on graphs. *arXiv preprint arXiv:2005.00687* (2020).

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for sequence tagging. *arXiv preprint arXiv:1508.01991* (2015).

Anna Jobin, Marcello Ienca, and Effy Vayena. 2019. The global landscape of AI ethics guidelines. *Nature Machine Intelligence* 1, 9 (Sept. 2019), 389–399. https://doi.org/10.1038/s42256-019-0088-2

Samina Khalid, Tehmina Khalil, and Shamila Nasreen. 2014. A survey of feature selection and feature extraction techniques in machine learning. In *2014 science and information conference*. IEEE, 372–378.

Kenji Kira and Larry A Rendell. 1992. A practical approach to feature selection. In *Machine learning proceedings 1992*. Elsevier, 249–256.

Wouter M Kouw and Marco Loog. 2018. An introduction to domain adaptation and transfer learning. *arXiv preprint arXiv:1812.11806* (2018).

Yifeng Li, Fang-Xiang Wu, and Alioune Ngom. 2016. A review on machine learning principles for multi-view biological data integration. *Briefings in Bioinformatics* (Dec. 2016), bbw113. https://doi.org/10.1093/bib/bbw113

Herbert Lin and Jaclyn Kerr. 2019. *On Cyber-Enabled Information Warfare and Information Operations.* Working Paper. Center for International Security and Cooperation (CISAC). https://ssrn.com/abstract=3015680

Scott W. Linderman and Ryan P. Adams. 2015. Scalable Bayesian Inference for Excitatory Point Process Networks. *arXiv:1507.03228 [stat]* (July 2015). http://arxiv.org/abs/1507.03228 arXiv: 1507.03228.

Andrew McCallum. 2005. Information extraction: Distilling structured data from unstructured text. *Queue* 3, 9 (2005), 48–57. Publisher: ACM New York, NY, USA.

Sara Meftah, Nasredine Semmar, Mohamed-Ayoub Tahiri, Youssef Tamaazousti, Hassane Essafi, and Fatiha Sadat. 2020. Multi-Task Supervised Pretraining for Neural Domain Adaptation. In *Proceedings of the Eighth International Workshop on Natural Language Processing for Social Media*. Association for Computational Linguistics, Online, 61–71. https://doi.org/10.18653/v1/2020.socialnlp-1.8

Giulia Muzio, Leslie O'Bray, and Karsten Borgwardt. 2021. Biological network analysis with deep learning. *Briefings in Bioinformatics* 22, 2 (March 2021), 1515–1530. https://doi.org/10.1093/bib/bbaa257

Jiquan Ngiam, Aditya Khosla, Mingyu Kim, Juhan Nam, Honglak Lee, and Andrew Y. Ng. 2011. Multimodal Deep Learning. In *ICML*. 689–696. https://icml.cc/2011/papers/399_icmlpaper.pdf

Simon Parkin. 2020. *A game of Birds and Wolves: the ingenious young women whose secret board game helped Win World War II.* Little, Brown and Company, New York.

Anaïs Rességuier and Rowena Rodrigues. 2020. *AI ethics should not remain toothless!* A call to bring back the teeth of ethics. *Big Data & Society* 7, 2 (July 2020), 205395172094254. https://doi.org/10.1177/2053951720942541

Lawrence A Rowe and Michael R Stonebraker. 1987. *The POSTGRES data model.* Technical Report. CALIFORNIA UNIV BERKELEY DEPT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (May 2019), 206–215. https://doi.org/10.1038/s42256-019-0048-x

Thorsten Ruprechter, Foaad Khosmood, and Christian Guetl. 2020. Deconstructing Human-assisted Video Transcription and Annotation for Legislative Proceedings. *Digital Government: Research and Practice* 1, 3 (Dec. 2020), 1–24. https://doi.org/10.1145/3395316

Bruce Schneier. 2021. *The Coming AI Hackers.* Technical Report. Harvard Kennedy School, Belfer Center for Science and International Affairs.

Nathalie A. Smuha. 2021. From a 'race to AI' to a 'race to AI regulation': regulatory competition for artificial intelligence. *Law, Innovation and Technology* 13, 1 (Jan. 2021), 57–84. https://doi.org/10.1080/17579961.2021.1898300

Jonathan Stray. 2019. Making Artificial Intelligence Work for Investigative Journalism. *Digital Journalism* 7, 8 (Sept. 2019), 1076–1097. https://doi.org/10.1080/21670811.2019.1630289

Mariarosaria Taddeo and Luciano Floridi. 2018. Regulate artificial intelligence to avert cyber arms race. *Nature* 556, 7701 (April 2018), 296–298. https://doi.org/10.1038/d41586-018-04602-6

Jindong Wang, Yiqiang Chen, Shuji Hao, Wenjie Feng, and Zhiqi Shen. 2017. Balanced Distribution Adaptation for Transfer Learning. In *2017 IEEE International Conference on Data Mining (ICDM)*. IEEE, New Orleans, LA, 1129–1134. https://doi.org/10.1109/ICDM.2017.150

Run Wang, Felix Juefei-Xu, Lei Ma, Xiaofei Xie, Yihao Huang, Jian Wang, and Yang Liu. 2020a. FakeSpotter: A Simple yet Robust Baseline for Spotting AI-Synthesized Fake Faces. *arXiv:1909.06122 [cs]* (July 2020). http://arxiv.org/abs/1909.06122 arXiv: 1909.06122.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020b. Generalizing from a Few Examples: A Survey on Few-shot Learning. *Comput. Surveys* 53, 3 (July 2020), 1–34. https://doi.org/10.1145/3386252

Andrew Gordon Wilson and Pavel Izmailov. 2020. Bayesian Deep Learning and a Probabilistic Perspective of Generalization. *arXiv:2002.08791 [cs, stat]* (April 2020). http://arxiv.org/abs/2002.08791 arXiv: 2002.08791.

Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. 2019. Zero-Shot Learning—A Comprehensive Evaluation of the Good, the Bad and the Ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 41, 9 (Sept. 2019), 2251–2265. https://doi.org/10.1109/TPAMI.2018.2857768

Yunlong Xiao, Yang Gu, Jiwei Wang, and Tong Wu. 2019. A Collaborative Multi-modality Selection Method Based on Data Utility Assessment. In *2019 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*. IEEE, Leicester, United Kingdom, 454–459. https://doi.org/10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00120

Doris Xin, Hui Miao, Aditya Parameswaran, and Neoklis Polyzotis. 2021. Production Machine Learning Pipelines: Empirical Analysis and Optimization Opportunities. *arXiv:2103.16007 [cs]* (March 2021). http://arxiv.org/abs/2103.16007 arXiv: 2103.16007.

Hao Yuan, Haiyang Yu, Jie Wang, Kang Li, and Shuiwang Ji. 2021. On Explainability of Graph Neural Networks via Subgraph Explorations. *arXiv:2102.05152 [cs]* (May 2021). http://arxiv.org/abs/2102.05152 arXiv: 2102.05152.