Machine learning active-nematic hydrodynamics

Jonathan Colen[a,b,1], Ming Han[b,c,1], Rui Zhang[c,d], Steven A. Redford[b,e], Linnea M. Lemma[f,g], Link Morgan[g], Paul V. Ruijgrok[h], Raymond Adkins[g], Zev Bryant[h,i], Zvonimir Dogic[g], Margaret L. Gardel[a,b], Juan J. de Pablo[c,j,2], and Vincenzo Vitelli[a,b,2]

[a]Department of Physics, University of Chicago, Chicago, IL 60637; [b]James Franck Institute, University of Chicago, Chicago, IL 60637; [c]Pritzer School of Molecular Engineering, University of Chicago, Chicago, IL 60637; [d]Department of Physics, Hong Kong University of Science and Technology, Kowloon, Hong Kong SAR, People's Republic of China; [e]Graduate Program in Biophysical Sciences, University of Chicago, Chicago, IL 60637; [f]Department of Physics, Brandeis University, Waltham, MA 02454; [g]Department of Physics, University of California, Santa Barbara, CA 92111; [h]Department of Bioengineering, Stanford University, Stanford, CA 94305; [i]Department of Structural Biology, Stanford University Medical Center, Stanford, CA 94305; and [j]Center for Molecular Engineering, Argonne National Laboratory, Lemont, IL 60439

Edited by Pablo G. Debenedetti, Princeton University, Princeton, NJ, and approved December 28, 2020 (received for review August 6, 2020)

Hydrodynamic theories effectively describe many-body systems out of equilibrium in terms of a few macroscopic parameters. However, such parameters are difficult to determine from microscopic information. Seldom is this challenge more apparent than in active matter, where the hydrodynamic parameters are in fact fields that encode the distribution of energy-injecting microscopic components. Here, we use active nematics to demonstrate that neural networks can map out the spatiotemporal variation of multiple hydrodynamic parameters and forecast the chaotic dynamics of these systems. We analyze biofilament/molecular-motor experiments with microtubule/kinesin and actin/myosin complexes as computer vision problems. Our algorithms can determine how activity and elastic moduli change as a function of space and time, as well as adenosine triphosphate (ATP) or motor concentration. The only input needed is the orientation of the biofilaments and not the coupled velocity field which is harder to access in experiments. We can also forecast the evolution of these chaotic many-body systems solely from image sequences of their past using a combination of autoencoders and recurrent neural networks with residual architecture. In realistic experimental setups for which the initial conditions are not perfectly known, our physics-inspired machine-learning algorithms can surpass deterministic simulations. Our study paves the way for artificial-intelligence characterization and control of coupled chaotic fields in diverse physical and biological systems, even in the absence of knowledge of the underlying dynamics.

deep learning | active turbulence | liquid crystals | topological defects | biomaterials

Machine learning holds great promise as a tool capable of transforming quantitative modeling in the physical sciences (1, 2). It takes data generated from simulations or collected from experiments and uses powerful nonlinear fitting functions to find the characteristic features behind the data. It can be used as either a continuous regression tool to extract physical principles or a discrete classifier to identify states of matter. Among the wide variety of machine-learning techniques, neural networks (3, 4) have attracted much attention due to their strong predictive power and ability to establish complex models from common building blocks (see *SI Appendix* for a primer). Notable developments within condensed-matter physics have led to machine-learning algorithms capable of recognizing structural signatures of the glass transition (5, 6) or distinguishing phases of matter (7, 8). They have also unveiled intriguing connections between deep learning and renormalization group methods (9, 10). However, the use of machine learning as a tool for the experimental characterization and discovery of material properties is still in its infancy (11, 12). Active nematics (13–21) provide an ideal material platform for machine-learning methods. While sufficiently well characterized to be a reliable benchmark, their chaotic dynamics are hard to predict and rich in unexplored phenomena of relevance to both material science and biology (14, 22–30).

Unlike simple fluids, nematic liquid crystals are orientationally ordered media described by a director field $\mathbf{n}(\mathbf{r})$, which tracks the average orientation of their microscopic constituents (e.g., biofilaments or elongated molecules), in addition to the local velocity field $\mathbf{v}(\mathbf{r})$ (13, 14). Representative images of the director field are shown in Fig. 1 *A* and *B*. In equilibrium nematics, the filaments tend to align and gradients of $\mathbf{n}(\mathbf{r})$ are penalized by the Frank free energy, which in three dimensions reads

$$F = \frac{K}{2}\int \left[(\nabla\cdot\mathbf{n})^2 + (\mathbf{n}\cdot(\nabla\times\mathbf{n}))^2 + (\mathbf{n}\times(\nabla\times\mathbf{n}))^2\right] \mathrm{d}\mathbf{r} \quad [1]$$

where, for simplicity, all of the elastic constants are set equal to K.

The introduction of microscopic energy sources into these orientationally ordered fluids generates out-of-equilibrium systems called active nematics (13, 14). A common example is provided by cytoskeleton filaments, with molecular motors that generate active forces promoting interfilament sliding. The resulting active stress, σ^a_{ij}, can be macroscopically described by

Significance

Artificial intelligence holds considerable promise for transforming quantitative modeling in materials science. We illustrate this potential by developing machine-learning models of a paradigmatic class of biomaterials called active nematics. These hybrid materials can be viewed as artificial muscles composed of biological fibers and molecular motors. Here, the macroscopic coefficients characterizing energy injection by motors and material elasticity are not constant. They are unknown functions of space and time that we extract directly from experiments using neural networks. Our physics-inspired machine-learning algorithms can also forecast the evolution of these complex materials simply using image sequences from their past, without any knowledge of the governing dynamics.

Author contributions: Z.B., Z.D., M.L.G., J.J.d.P., and V.V. designed research; J.C., M.H., R.Z., S.A.R., L.M.L., L.M., P.V.R., and R.A. performed research; and J.C., M.H., R.Z., S.A.R., L.M.L., L.M., P.V.R., R.A., Z.B., Z.D., M.L.G., J.J.d.P., and V.V. wrote the paper.

The authors declare no competing interest.

This article is a PNAS Direct Submission.

See online for related content such as Commentaries.

[1]J.C. and M.H. contributed equally to this work.

[2]To whom correspondence may be addressed. Email: vitelli@uchicago.edu or depablo@uchicago.edu.

This article contains supporting information online at https://www.pnas.org/lookup/suppl/doi:10.1073/pnas.2016708118/-/DCSupplemental.

Published March 2, 2021.

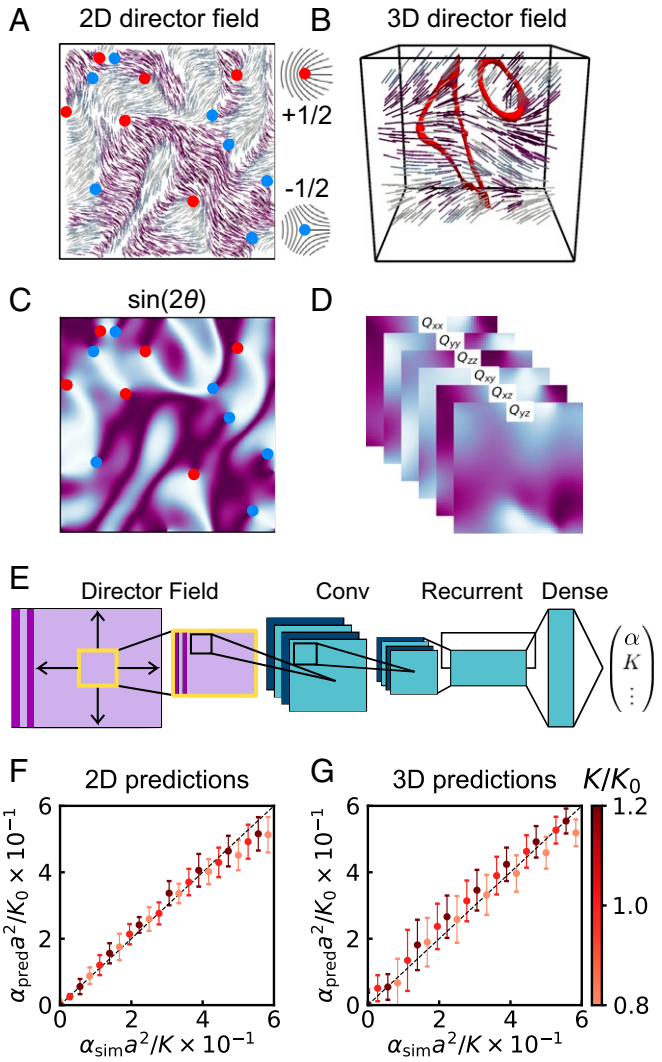

Fig. 1. Machine-learned hydrodynamic parameters in lattice Boltzmann simulations. (*A* and *B*) Nematic director fields in two (*A*) and three (*B*) dimensions. The $+1/2$ and $-1/2$ defects in 2D are marked as red and blue dots, respectively. Disclination loops are indicated in red. (*C* and *D*) Continuous representations of the director field used by the network. In 2D, the network can use $\sin 2\theta$ where θ is the angle of the director field. In 3D, the network uses the tensor $Q_{ij} = n_i n_j - 1/3$. Color indicates the magnitude of these continuous representations. (*E*) Schematic of neural network architecture. The full input images are divided into patches, which are then fed into a set of convolutional filters, a LSTM recurrent layer, and a fully connected dense layer. The outputs are averaged into a final estimate for hydrodynamic parameters. (*F* and *G*) Predictive accuracy of rescaled dimensionless activity in simulation data in 2D and 3D at different values of K. Networks were trained at $K = K_0$. Units such as K_0 are listed in *Materials and Methods*.

$$\boldsymbol{\sigma}^a_{ij} = \alpha\, n_i n_j, \qquad [2]$$

where $\alpha(\mathbf{r}, t)$ is an a priori unknown activity field related to the concentration of molecular motors or other energy sources, which is typically approximated as constant (13, 31).

Active-Nematic Hydrodynamics

The dynamical equations for active nematics involve a coupling between the director field $\mathbf{n}(\mathbf{r}, t)$ and the velocity field $\mathbf{v}(\mathbf{r}, t)$ (13, 14, 31, 32). In particular, the director field evolves according to the equation

$$(\partial_t + \mathbf{v}\cdot\nabla) n_i = \lambda_{ijk}\partial_j v_k - \frac{1}{\gamma}\left[\frac{\delta F}{\delta n_i} - \left(\frac{\delta F}{\delta n_j} n_j\right) n_i\right], \qquad [3]$$

where γ is the rotational viscosity. The tensor λ_{ijk} is given by

$$\lambda_{ijk} \equiv \frac{\lambda+1}{2} n_j \delta_{ik} + \frac{\lambda-1}{2} n_k \delta_{ij} - \lambda n_i n_j n_k \qquad [4]$$

where λ is a dimensionless flow-alignment parameter and δ_{ij} is the Kronecker delta.

The velocity field evolves according to the equation

$$\rho_0(\partial_t + \mathbf{v}\cdot\nabla) v_k = -\partial_k P + \eta\nabla^2 v_k + \partial_j\left(\lambda_{ijk}\frac{\delta F}{\delta n_i}\right) + \partial_j \sigma^a_{jk} \qquad [5]$$

where P is the hydrostatic pressure, η is the isotropic fluid viscosity, and σ^a is the active stress defined in Eq. **2**. The final three terms on the right-hand side of Eq. **5** correspond to the viscous, elastic, and active stresses, respectively. The coupled equations of motion Eqs. **3**–**5** can also be cast in terms of a symmetric and traceless order parameter $Q_{ij} = S(n_i n_j - \delta_{ij}/3)$ with a scalar field S that quantifies the orientational-order strength (*Materials and Methods*).

Determining how hydrodynamic parameters, such as α and K, vary in space and time as well as a function of adenosine triphosphate (ATP) or motor concentrations is challenging (18, 33–35). Even direct measurements (36–39) that rely on controlled flow experiments are difficult to devise if the underlying flows are chaotic. This is precisely what happens in active nematics, where the energy injected by molecular motors at the microscale cascades to macroscopic scales, leading to chaotic flows mediated by the proliferation of topological defects (13, 31). The mechanisms behind this process, loosely called active nematic turbulence, are not fully understood (15, 20, 30, 31, 40–45). Nonetheless, it is clear that most active nematic responses depend on the competition between active stresses that promote director or velocity gradients and viscoelastic stresses that resist them. As a consequence of this interplay, experimental measurements often access only nontrivial combinations of hydrodynamic parameters, e.g., α/K or ratios of elastic constants (18, 34). Furthermore, the derivation of these parameters from realistic microscopic models is often prohibitively difficult (39). This prompts us to seek approaches that bypass coarse graining and extract hydrodynamic parameters like $\alpha(\mathbf{r}, t)$ directly from experiments.

In this paper, we first design a neural network to extract hydrodynamic parameters directly from experimental or simulated movies. This requires prior knowledge of the underlying physics. For systems whose governing equations are unknown, we introduce another neural network capable of forecasting the time evolution of a nonequilibrium system solely based upon its past. Detailed workflows for training and testing these neural networks are provided in *SI Appendix*, Fig. S2.

Extracting Hydrodynamic Parameters as Fields

To make progress, we recast the task of estimating the spatiotemporal variations of multiple hydrodynamic parameters as a computer-vision problem that can be effectively addressed by artificial intelligence. We begin by generating a library of director fields (Fig. 1 *A* and *B*) for a wide range of activity in two and three dimensions using lattice Boltzmann simulations based on the continuum equations (18, 46) (*Materials and Methods*). Using the simulated library, we train neural networks on continuous representations of $\mathbf{n}$ that account for the nematic symmetry $\mathbf{n} = -\mathbf{n}$. For instance, in two dimensions we use $\sin(2\theta)$ with θ denoting the tilt angle of $\mathbf{n}$ (Fig. 1 *C* and *D*). The neural network

architecture, shown schematically in Fig. 1*E*, contains 1) a single convolutional layer used for image processing, 2) a recurrent layer that captures the system dynamics, and 3) a dense layer that identifies the hydrodynamic parameters (see *SI Appendix* primer on neural networks and *SI Appendix*, Fig. S1). We train these neural networks on simulation data for which we can obtain the exact values of the prescribed hydrodynamic parameters corresponding to each director-field configuration. This allows us to conduct supervised learning, which would not be possible using experimental data alone. Once trained, these neural networks can be used to obtain hydrodynamic parameters in simulations as well as experiments (*SI Appendix*, Fig. S2*A*).

We first apply this scheme to estimate a single parameter: the rescaled dimensionless activity $\alpha/K \times a^2$, where a denotes the pixel or voxel size for the director-field image. Comparison of the machine-learning predictions for the activity with the known values of α reveals good agreement for both two-dimensional (2D) and three-dimensional (3D) active nematics (Fig. 1 *F* and *G*). Although these networks are trained on data generated at a single value of $K = K_0$, their accuracy persists for samples where K differs from K_0. We also stress that our machine-learning model is robust and its predictive performance does not depend on the choice of K_0 (*SI Appendix*, Fig. S3).

Hydrodynamic theories suggest how α can be estimated from the characteristic length $\ell_d \propto \sqrt{K/\alpha}$ obtained by balancing the right-hand sides of Eqs. **1** and **2** (47). In 2D, ℓ_d can be interpreted (and experimentally extracted) as the average spacing between point disclinations, topological defects with index $\frac{+1}{2}$ and $\frac{-1}{2}$ shown in Fig. 1*A* as red and blue dots, respectively (13, 40). This procedure cannot be carried out in the low-activity regime, where disclinations are not always present in the field of view. Furthermore, it does not extend to 3D samples where the dominant excitations are charge-neutral disclination loops (19) (Fig. 1*B*) whose activity dependence is unclear. Extracting and measuring topological defects with a neural network would require multiple convolutional layers to achieve a sufficiently large receptive field, i.e., the total field of view for the last neural layer. Our networks, which contain just a single convolutional layer, simply exploit local spatial fluctuations of the director field.

While capable of achieving high accuracy even in 3D or at low α, neural networks designed for single-parameter estimation still predict only combinations of parameters such as $\ell_d \propto \sqrt{K/\alpha}$. To decouple α from K, one would need to resort to additional measurements (35) or ad hoc assumptions like the independence between K and α which is not always experimentally valid (34). However, our neural networks are also capable of extracting multiple parameters without the need to devise a set of experiments that disentangle the parameters' interdependence. In Fig. 2, we demonstrate the performance of a machine-learning model that uses the same architecture described above but is trained to predict α and K simultaneously.

Using a simulated dataset, we evaluate the performance of our machine-learning model by comparing it to high-throughput scans of parameter space inspired by ref. 35. While the exact approach implemented in ref. 35 uses all evolving fields, our machine-learning model does not rely on knowledge of the velocity field to which the director is coupled. Because complete information about the system state is inaccessible in many experimental systems, we compare our method against parameter scans which use only the nematic director field. These scans aim to find the set (α, K) best fitting the observed nematic correlation length and correlation time (*SI Appendix*, *SI Text* and Fig. S4). For each parameter, we quantify the model performance using the R^2 value of the linear fit between the predicted parameter and the ground truth. We find that the machine-learning model results in a higher R^2 than the parameter-scan approach (Fig. 2), indicating a more accurate prediction of the ground truth for each parameter.

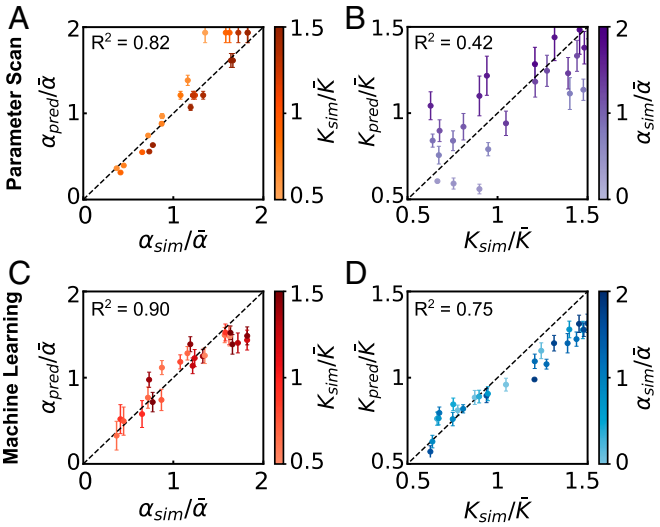

Fig. 2. Comparison of multiparameter estimation using neural networks and a high-throughput parameter scan. (*A* and *B*) Simultaneous estimation of α and K using a high-throughput parameter scan (*SI Appendix, SI Text*). (*C* and *D*) Multiparameter estimation using our neural network. The network estimator outperforms the parameter scan approach for the predictions of both α and K. Here $\bar{\alpha}$ and $\bar{K}$ are the mean values of α and K from the training dataset. We quantify the performance for each parameter using the R^2 of the linear fit between the predictions and the ground truth (dashed line).

So far, we have tested our machine-learning algorithms on numerical data. We now turn to experiments and begin by applying our multiparameter estimation trained on simulations to data obtained from microtubule–kinesin experiments (15, 48) (*Materials and Methods*). It is known that in this system the rescaled activity α/K increases with ATP concentration (33, 34). Here, we use machine learning to measure α and K independently, probing first how they vary with the ATP concentration, c. Inspection of Fig. 3*A* shows that the spatiotemporally averaged activity, α, predicted by our machine-learning algorithms increases with c while the elastic modulus, K, decreases. Similar results obtained from experiments on 3D microtubule–kinesin systems (19) and 2D actin–myosin systems (18, 27) are shown in *SI Appendix*, Figs. S5 and S6.

Before expanding on the capabilities of our parameter-estimation networks, we highlight their salient features. Our algorithms do not perform curve fitting by 1) identifying observables for which the underlying theory is solvable and 2) parameterizing them in terms of the sought-after coefficients. Instead, neural networks are trained on data obtained using whatever conditions are experimentally available without choosing in advance which reduced data representations to use (e.g., correlation functions or collective variables). Training neural networks differs from building lookup tables (or other discrete representations of the data), which are impractical for fields like $\mathbf{n}(\mathbf{r}, t)$ when the number of possible pixel (voxel) configurations vastly exceeds the number of available data points. When applied to previously unseen data, neural networks can still produce accurate predictions because they learn the smooth high-dimensional manifolds that map all possible realizations of the fields onto the corresponding hydrodynamic parameters.

Equipped with the ability to determine multiple hydrodynamic parameters from experimental data, we now proceed to put the hydrodynamic theory itself to the test. Here, we compare microtubule–kinesin experiments at different ATP concentrations to the evolution of lattice Boltzmann simulations calibrated using parameters machine learned from the same experiments.

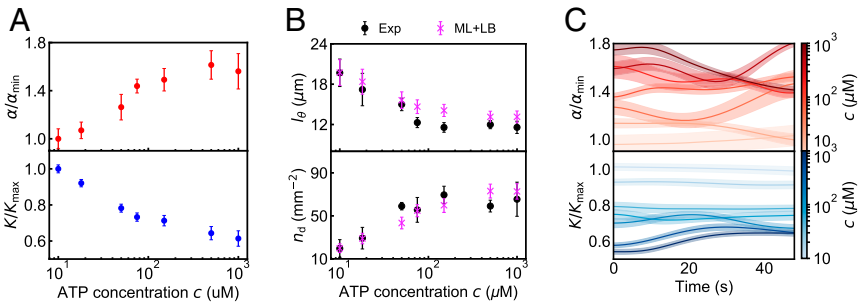

Fig. 3. Multiparameter estimation and dynamics in microtubule–kinesin experiments. (*A*) Dependence of spatiotemporally averaged activity and elastic modulus on ATP concentration. Here, $\alpha_{\min}$, $K_{\max}$ are the time-averaged predicted activity and elastic modulus at the lowest level of ATP concentration $c_{\min} = 10$ μM. (*B*) Comparison of director-field correlation length l_θ and defect spacing n_d in experiments (Exp) and machine-learning informed lattice Boltzmann simulations (ML + LB). (*C*) Simultaneous prediction of activity and elastic modulus over time at different levels of ATP concentration. The shaded regions represent the standard error of spatiotemporal fluctuations in the machine-learning predictions. ATP concentration c is indicated by the color bar.

As the chaotic nature of active nematics makes exact director-field comparisons unreliable over long times, we instead resort to properties of the dynamical steady state. Using the spatial correlation function $C_s(\mathbf{r})$ for the director field (*Materials and Methods*), we define the correlation length ℓ_θ such that $C_s(\ell_\theta) = 1/2$. We find that both the average correlation length and the defect density n_d calculated from machine-learning informed lattice Boltzmann simulations match experiments for a wide range of ATP concentrations (Fig. 3*B*). This procedure allows us to provide a partial experimental validation of the hydrodynamic theory with constant parameters. Note, however, that these hydrodynamic parameters are often taken as constant because measuring them locally is difficult.

Our machine-learning methods provide a rare glimpse into the spatiotemporal variations of these hydrodynamic parameters. For example, we can extract from the microtubule–kinesin experiments the activity field, $\alpha(\mathbf{r}, t)$, whose coarse-grained dynamics stem from heterogeneities in the motor distribution. We start by plotting in Fig. 3*C* examples of time series for the sample-averaged $\alpha(t)$ and $K(t)$. The mean values are plotted as solid lines and their uncertainties are marked with shaded regions over a wide range of ATP concentrations (denoted by color bars). We compare the parameters extracted from experiments with those from simulations, where constant parameters are prescribed so that the fluctuations in the extracted parameters arise solely from the uncertainty of the machine-learning predictions. We find that the fluctuations of the machine-learned α are markedly more pronounced in experiments than in simulations, while the variation of the predicted K is comparable in both cases (*SI Appendix*, Fig. S7). Our analysis also shows that the time variations in α and K are significantly larger than their spatial fluctuations. The machine-learned evidence discussed above suggests that a nonlinear fluctuating hydrodynamic theory may better explain our experimental observations. Heuristically, the strong disruption of fiber alignment at large activity can trigger motor detachment–reattachment events causing the time modulation of α inferred by our algorithms.

Our machine-learning models can be also applied to situations in which activity is engineered to deliberately vary in both time and space. We test this activity control scenario first in lattice Boltzmann simulations, where we prescribe spatiotemporal patterns of $\alpha(\mathbf{r}, t)$ (27). Remarkably, neural networks trained on the data with constant activity can still accurately estimate a time-varying activity coefficient as shown in Fig. 4 *A* and *B* where linear and sinusoidal activity profiles are probed. Since small director-field patches are sufficient to generate reliable predictions, we can generate a spatial activity map of $\alpha(\mathbf{r}, t)$ by applying our neural networks locally to each patch composing an image. By doing this, we are able to discern prescribed spatial activity patterns in lattice Boltzmann simulations, as demonstrated in Fig. 4*C* where activity is nonzero only in the central square.

We further test the capability of our machine-learning models in extracting parameter fields using actin–myosin experiments (18). In this system, one can alter the speed of some specialized molecular motors via selective exposure to light (49). This phenomenon, informally called gear shifting (*Materials and Methods*), allows for precise spatiotemporal control of active stresses (27). Inspection of Fig. 4*D* and Movie S1 reveals that our machine-learning models can successfully identify the marked increase in activity that occurs as light is turned on (indicated by the dashed line in Fig. 4*D*). Furthermore, our approach can identify the activity changes that occur in selectively illuminated spatial domains in these systems (Fig. 4 *E* and *F* and Movie S1). We can also extract the elastic modulus field $K(\mathbf{r}, t)$ in these experiments, but we find that K does not change significantly when light is applied to the gear-shifting motors (*SI Appendix*, Fig. S8). The performance of our machine-learning model in identifying spatiotemporally varying activity demonstrates its potential for 1) the control of engineered active materials and 2) the inference of biochemical processes that take place at the microscopic level (such as the different experimental configurations summarized in *SI Appendix*, Table S1).

Forecasting Time Evolution

We now ask, Can neural networks forecast the evolution of chaotic many-body systems solely from image sequences of their past? A time-honored approach to quantitative modeling relies on writing down equations and then solving them, analytically or via simulations, to make predictions. In what follows, we use the term machine-learning model to denote a very different approach (50). Instead of solving the equations, we train neural networks on existing data and then ask them to forecast the future behavior of the chaotic system. A feature that distinguishes our forecasting neural networks from the ones used for parameter estimation is that the former can be trained directly on experimental data while the latter rely on an underlying model. Our physics-inspired machine-learning approach to forecasting the dynamics of active nematics consists of iterating the following two steps. First, we perform next-frame predictions using a neural network that does not know anything about the physics of the system. Second, we reduce any noise generated in the previous step by applying to each frame a physics-inspired sharpening algorithm. This sharpening filter harnesses the known propensity of the fibers to align (i.e., it minimizes the elastic energy in Eq. **1**) while being agnostic about the active forces driving the nonequilibrium dynamics (*Materials and Methods*).

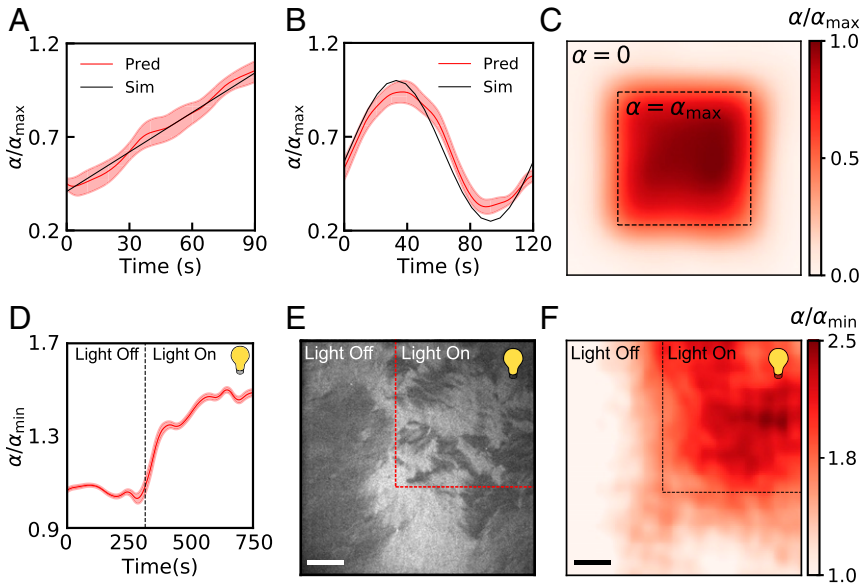

Fig. 4. Machine-learned activity field, $\alpha(\mathbf{r}, t)$, in simulations and actin–myosin experiments. (*A* and *B*) Machine-learning predicted activity on lattice Boltzmann simulations with spatially uniform activity prescribed to vary linearly (*A*) and sinusoidally (*B*) in time. (*C*) Machine-learning predicted activity on simulations where the central square (dashed line) is activated. (*D*) Machine-learning predicted activity vs. time on actin–myosin experiments where myosin motors are controlled through light-activated gear shifting. The dashed line indicates when light is switched on. (*E* and *F*) Direct image (*E*) and machine-learning predicted spatial activity profile (*F*) of a selectively illuminated actin nematic with light-activated gear-shifting motors. For *E* and *F* the experimental data are the dataset reported in figure 1 of Zhang et al. (27). Data for *D* are from the current study, following the approach used in ref. 27. (Scale bars, 20 μm.)

Our forecasting neural networks are a modification of the autoencoder architecture, a popular tool in the computer vision community (*SI Appendix* primer). A traditional autoencoder learns to compress an image to a feature vector which is then used to reconstruct the image. In our network, we insert a recurrent layer in between the encoder and the decoder to learn the system dynamics (*Materials and Methods*). Crucial for ensuring high performance is the following algorithmic trick: A residual architecture (52) is used in the recurrent layer to capture the difference between frames, rather than the images themselves. Denoting the output of the recurrent layer as $\mathrm{RNN}(X_0, X_1, \cdots, X_t)$, the predicted next frame can be written as

$$X_{t+1} = X_t + \mathrm{RNN}(X_0, X_1, \cdots, X_t). \quad [6]$$

Such a residual recurrent network resembles the discrete form of a general differential equation:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = f(X_0, X_1, \cdots, X_t). \quad [7]$$

As illustrated in Fig. 5*A*, the network encodes a time series of director-field images into a sequence of feature vectors. Next, it uses them to predict the future state of the system and finally decodes this state back into a director-field image (Fig. 5*A*). For large systems, the director field is divided into small overlapping domains. Machine-learning predictions are made within each domain and then stitched into a final prediction of the next director-field configuration. Although stitching could introduce artificial defects and image blurriness in the overlapping area between adjacent domains, these errors can be automatically corrected by the sharpening step (*Materials and Methods*). Once trained on either simulated or experimental image sequences, the machine-learning model can then forecast the defect dynamics in active nematics (*SI Appendix*, Fig. S2*B*).

We first examine the performance of our forecasting neural networks on simulation data. Given a particular sequence of nematic configurations, our algorithm can reliably learn the spatiotemporal evolution of the director field including singular events such as defect annihilation and nucleation (see Movie S2 or selected examples in *SI Appendix*, Fig. S9).

To systematically evaluate the accuracy of our machine-learning predictions, we first compare the time-evolved director fields generated by machine-learning and lattice Boltzmann simulations pixel by pixel. Such pixelwise comparison is only meaningful within the Lyapunov time, the characteristic timescale after which a nonlinear dynamical system becomes chaotic. Inspection of Fig. 5*B* shows that the pixelwise error rate of the predicted director field $1 - \langle |\mathbf{n}_{ML} \cdot \mathbf{n}_{LB}| \rangle$ remains small within the Lyapunov time (*Materials and Methods*). The Lyapunov time is the exponential diverging time for the evolution of a small difference in $\mathbf{n}(\mathbf{r}, t)$ introduced in the initial condition, which we find is equal to $t_\lambda \sim 3.6\tau_\mathrm{d}$ (τ_d denotes the average defect lifetime). Beyond the Lyapunov time (shaded region in Fig. 5*B*), even the lattice Boltzmann simulations are unreliable at the pixelwise level due to numerical precision.

Our forecasting neural network predicts the evolution of the director field without knowing the velocity field $\mathbf{v}(\mathbf{r}, t)$ or ordering magnitude S. We compare the error rates of our algorithm to those of a lattice Boltzmann simulation, where similarly no initial velocity or ordering magnitude information is provided (Fig. 6*A*). Our machine-learning model achieves lower error rates up to the Lyapunov time, except at the lowest levels of activity when the system is nearly passive.

To evaluate the predictive accuracy of our machine-learning methods for even longer times, we turn to properties of the dynamical steady state such as the director-field correlation length ℓ_θ defined before and the correlation time t_θ defined by setting the time correlation function $C_t(t_\theta) = 1/2$ (*Materials and Methods*). Previous numerical studies have shown that the quantity ℓ_θ is proportional to $\sqrt{K/\alpha}$ (47). When comparing the predictions of our machine-learning model against lattice Boltzmann simulations, we find that machine learning correctly captures the activity dependence of the characteristic length ℓ_θ

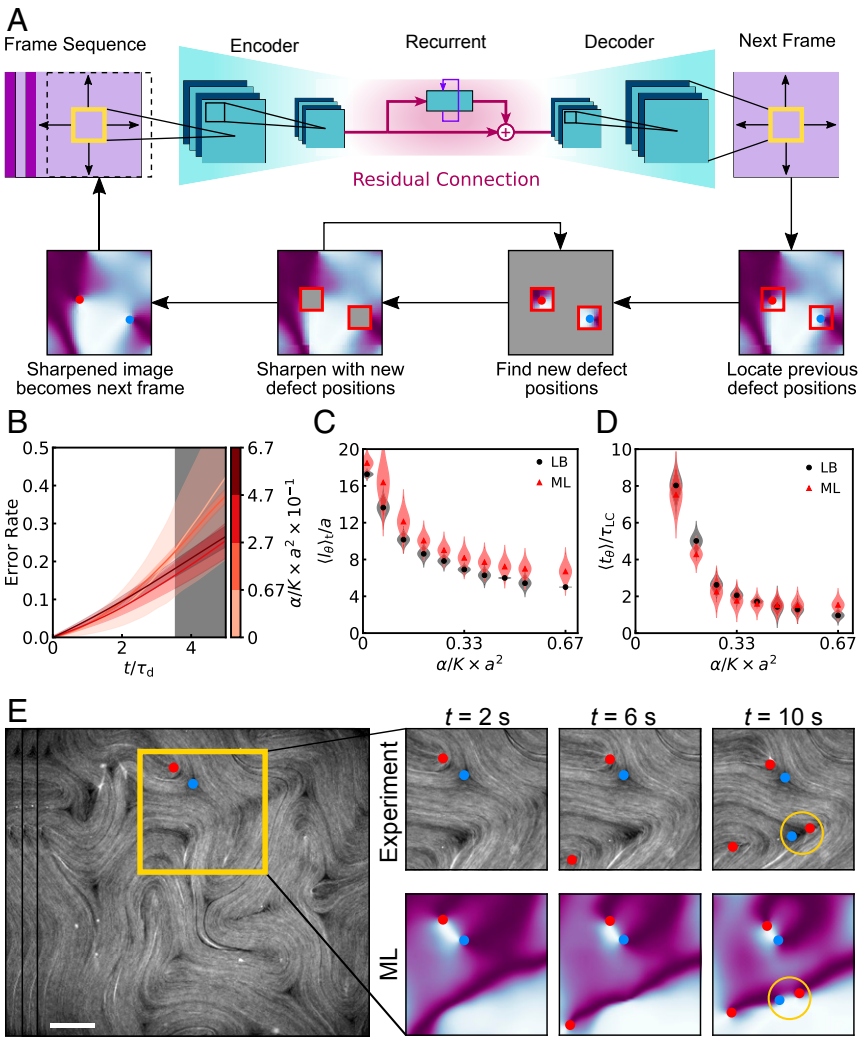

Fig. 5. Neural networks with a residual architecture as surrogate models of time evolution. (*A*) Schematic for predicting the time evolution of an active nematic. Individual images are first compressed by a convolutional encoder into a feature vector to reduce the data dimension by fivefold. Time evolution is predicted in the feature space using a residual block, which is composed of a direct shortcut (straight arrow) to preserve the memory from the previous frame and a recurrent layer (cyan box with a purple looped arrow) to capture the change between frames. Based upon the sequence of feature vectors for the past, the residual block generates the next feature vector for the future, which is then translated by a convolutional decoder into an output image. The output is sharpened by first using relaxational dynamics to update defect positions and then using the updated defect positions to sharpen the entire director field. This procedure is iterated and the sharpened image is used as the next frame. (*B*) Pixelwise error rate $1 - \langle |\mathbf{n}_{ML} \cdot \mathbf{n}_{LB}| \rangle$ of the predictive model versus time, for different groups of activity in lattice Boltzmann simulations. The gray area shows regions beyond the Lyapunov time for the lattice Boltzmann simulations (*Materials and Methods*). Here $\tau_d = \eta/\alpha$, the characteristic defect lifetime. We observe that measuring the Lyapunov time in units of τ_d yields a common value of $t_\lambda \sim 3.6\tau_d$. (*C*) Comparison of time-averaged correlation length in machine-learning and lattice Boltzmann simulations. (*D*) Comparison of average director-field correlation time t_θ in machine-learning and lattice Boltzmann simulations. Here, τ_{LC} is an activity-independent viscoelastic relaxation timescale defined as $\tau_{LC} = \gamma a^2/K$, where γ is the rotational viscosity (51). (*E*) A defect nucleation event as seen in experiment and as predicted by the machine-learning model trained on microtubule–kinesin experimental data. Machine–learning predictions depict the magnitude of $\sin(2\theta)$, where θ is the angle of the director field. The $+1/2$ and $-1/2$ defects are marked as red and blue dots, respectively. (Scale bar, 100 μm.)

(Fig. 5*C*) (corresponding results for the mean defect spacing ℓ_d are shown in *SI Appendix*, Fig. S10). We stress that while ℓ_θ at steady state is plotted in both Figs. 3*B* and 5*C*, the former is generated from lattice Boltzmann simulations with machine-learned parameters whereas the latter is generated solely using our time-evolution neural network. Furthermore, our networks also reproduce the same activity dependence for t_θ as the lattice Boltzmann simulations (Fig. 5*D*), suggesting that they have learned to reproduce the correct dynamics expected at each level of activity.

After training a neural network exclusively on experimental data, we can successfully forecast the time evolution of the nematic director field including singular events such as topological defect nucleations or annihilations; see Fig. 5*E* for an example. Here we choose microtubule–kinesin experiments as a paradigmatic example, as they often exhibit clear defect dynamics. In Movie S3 we show the corresponding experimental video next to the one generated using machine learning. Inspection of these movies show no discernible differences in the defect dynamics between experiments and machine-learning predictions. Similar agreement is obtained when our time-evolution neural networks are trained on lattice Boltzmann simulation data (Movie S3).

As an alternative approach, one can also forecast the evolution of active nematics by first extracting the parameters (α, K) using our parameter-estimation network and then plugging them

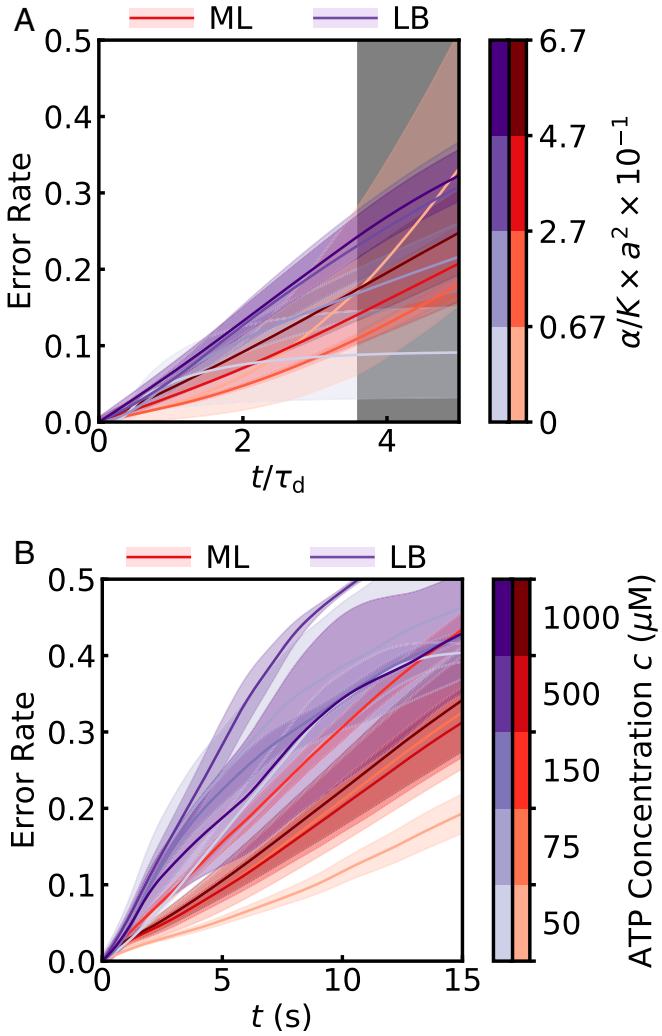

Fig. 6. Performance comparison between physics-inspired machine-learning model and lattice Boltzmann simulations with parameters extracted using the multiparameter estimation network. To make a fair comparison as well as mimic the true experimental constraints, both approaches take director fields only as input, with no prior knowledge of the velocity field. We quantify their performance by measuring the pixel-wise error rate $1 - \langle |\mathbf{n}_{\text{prediction}} \cdot \mathbf{n}_{\text{ground truth}}| \rangle$. (*A*) Error rates for predicting the simulated nemato-hydrodynamics at different levels of activity. Red curves show results for the machine-learning model, while purple curves show results for velocity-uninformed lattice Boltzmann predictions. The gray area shows regions beyond the Lyapunov time for the lattice Boltzmann simulations (*Materials and Methods*). (*B*) Error rates for predicting the evolution of microtubule–kinesin experiments at different ATP concentrations. Here we emphasize that unlike lattice Boltzmann simulations, the machine-learning model does not implement any physical theory. Nevertheless, its performance matches or exceeds that of lattice Boltzmann simulations.

into lattice Boltzmann simulations to model the corresponding hydrodynamics. We stress that this approach requires prior knowledge of the underlying physics. Notably, we find that our time-evolution network outperforms the physics-informed simulations in the context of practical applications to experimental data, where complete knowledge of the initial state is inaccessible (Fig. 6*B*). To evaluate the long-term validity of our predictions beyond specific realizations, we systematically check (as we did in Figs. 3*B* and 5*C*) that the steady-state values of ℓ_θ and n_d extracted from the machine-learned nematic director are in good agreement with experiments over a wide range of ATP concentrations (*SI Appendix*, Fig. S11).

The successful performance of our time-evolution networks relies on the combination of autoencoders and recurrent networks that adopt a residual architecture. The convolutional autoencoders compress input images into feature vectors, which represent the underlying physics using fewer variables. This dimensional reduction not only diminishes the burden of the machine-learning model, but also enables training on limited experimental datasets without overfitting. We stress that our recurrent networks predict the correct long-term dynamics at steady state only when supplemented with a residual architecture (*SI Appendix*, Fig. S12). More generally, we expect the use of a residual architecture capturing the difference between frames to be crucial for all dynamical systems that are naturally governed by differential equations (Eqs. **6** and **7**).

Conclusion

The machine-learning framework proposed here can estimate hydrodynamic parameters using only movies of the director field without requiring knowledge of the velocity field, even if the two are coupled. Because the framework primarily exploits local spatial fluctuations of the nematic director, one can map out the spatiotemporal variation of hydrodynamic parameters and promote them from constants to fields. This provides a generic way to quantitatively connect experimental observations and theories.

When trained only on experimental data, neural networks are shown to be capable of forecasting the future, without theoretical knowledge of the underlying dynamics. This scenario is particularly intriguing for experimental systems that, unlike active nematics, lack a quantitative description of their coarse-grained dynamics. In addition, the simplicity of our time-evolution neural network system makes it suitable for implementation of artificial-intelligence informed control of such systems. For example, induced spatiotemporal variations of active parameters combined with machine-learning techniques could enable efficient control of complex flows and pattern formation in synthetic (26–28, 32, 53 and biological systems (22–25, 56). Beyond active and soft matter, our neural-network models could be employed in other contexts where coupled chaotic fields naturally occur, such as turbulent flows or magnetohydrodynamics (57–59).

Materials and Methods

Active Nematohydrodynamics and Lattice Boltzmann Simulation. Simulation data for training and testing were generated using a hybrid lattice Boltzmann method which has been used in prior studies of different types of active nematics (18, 27, 60, 61). The symmetric and traceless tensorial order parameter of the nematic is defined as

$$\mathbf{Q} = S(\mathbf{nn} - \mathbf{I}/3) \qquad [8]$$

with S being the scalar order parameter, $\mathbf{n}$ being the unit vector describing the local nematic orientation, and $\mathbf{I}$ being an identity tensor. The following governing equation of the nematic microstructure, namely Beris–Edwards Eq. **9**, reads

$$(\partial_t + \mathbf{u} \cdot \nabla)\mathbf{Q} - \mathbf{S}(\mathbf{W}, \mathbf{Q}) = \Gamma \mathbf{H} \qquad [9]$$

where $\mathbf{u}$ is the velocity vector, $\mathbf{W}$ is the velocity gradient $\nabla\mathbf{u}$, and Γ is related to the rotational viscosity γ_1 via $\Gamma = 2S_0^2/\gamma_1$ with S_0 the equilibrium scalar order parameter (62). Here, the generalized advection term $\mathbf{S}(\mathbf{W}, \mathbf{Q})$ is defined as

$$\begin{aligned}\mathbf{S}(\mathbf{W}, \mathbf{Q}) = &(\xi\mathbf{A} + \Omega)(\mathbf{Q} + \mathbf{I}/3) \\ &+ (\mathbf{Q} + \mathbf{I}/3)(\xi\mathbf{A} - \Omega) \\ &- 2\xi(\mathbf{Q} + \mathbf{I}/3))\mathrm{Tr}(\mathbf{QA})\end{aligned} \qquad [10]$$

with $\mathbf{A} = (\mathbf{W} + \mathbf{W}^T)/2$ being the strain rate tensor, $\Omega = (\mathbf{W} - \mathbf{W}^T)/2$ being the vorticity, and ξ being a flow-alignment parameter setting the Leslie angle. The molecular field $\mathbf{H}$ is a symmetric, traceless projection of the functional derivative of the free energy of the nematic. Its index form reads

$$H_{ij} = \frac{1}{2}\left(\frac{\delta F}{\delta Q_{ij}} + \frac{\delta F}{\delta Q_{ji}}\right) - \frac{\delta_{ij}}{3}\mathrm{Tr}\left(\frac{\delta F}{\delta Q_{kl}}\right) \qquad [11]$$

in which the free-energy functional is $F = \int_V f dV$. Its density f takes the form (63)

$$f = \frac{A_0}{2}\left(1 - \frac{U}{3}\right)Q_{ij}Q_{ij} - \frac{A_0 U}{3}Q_{ij}Q_{jk}Q_{ki} + \frac{A_0 U}{4}(Q_{ij}Q_{ij})^2 + \frac{1}{2}L\partial_k Q_{ij}\partial_k Q_{ij} \quad [12]$$

where A_0, U are material constants and L is related to the Frank elastic constant under the one-constant approximation. Eq. **9** is solved using a finite-difference method.

The hydrodynamic flow is governed by a momentum equation

$$\rho(\partial_t + u_j\partial_j)u_i = \partial_j\Pi_{ij} + \eta\partial_j\left[\partial_i u_j + \partial_j u_i + (1 - 3\partial_\rho P_0)\partial_\gamma u_\gamma \delta_{ij}\right] \quad [13]$$

where ρ is density, η is the isotropic viscosity, and $P_0 = \rho T - f$ is the hydrostatic pressure with T being the temperature. The additional stress has two contributions, $\Pi_{ij} = \Pi^p_{ij} + \Pi^a_{ij}$, where the first term is passive in its nature accounting for the anisotropy and is defined as

$$\begin{aligned}\Pi^p_{ij} = &- P_0\delta_{ij} - \xi H_{ik}\left(Q_{kj} + \frac{1}{3}\delta_{kj}\right) \\ &- \xi\left(Q_{ik} + \frac{1}{3}\delta_{ik}\right)H_{kj} \\ &+ 2\xi\left(Q_{ij} + \frac{1}{3}\delta_{ij}\right)Q_{kl}H_{kl} \\ &- \partial_j Q_{kl}\frac{\delta F}{\delta\partial_i Q_{kl}} + Q_{ik}H_{kj} - H_{ik}Q_{kj}.\end{aligned} \quad [14]$$

The active stress that drives the system out of equilibrium reads (31, 46)

$$\Pi^a_{ij} = -\alpha Q_{ij} \quad [15]$$

in which $\alpha > 0$ describes an extensile active nematic, as is the case for the experimental systems discussed in this paper. Eq. **13** is solved simultaneously via a lattice Boltzmann method over a D3Q15 grid (64). Additional details on this method can be found in ref. 65.

Typical simulation parameters were $\Gamma = 0.13$, $\eta = 0.33$, $A = 0.1$, and $U = 3.5$, leading to $S_0 \sim 0.62$. For Figs. 1 and 5, simulations were trained on $K = 0.075$, $\alpha \in [0, 0.05]$. The range of K for testing in Fig. 1 *F* and *G* was $K \in [0.06, 0.09]$. For the multiparameter estimator used in Figs. 2–4 and *SI Appendix*, Fig. S6–S8, $K \in [0.06, 0.20]$ and $\alpha \in [0, 0.05]$. For the experimental prediction in 3D (*SI Appendix*, Fig. S5), models were trained on $K = 0.1$, $\alpha \in [0, 0.09]$ as initial predictions indicated that the range of α first used for training was insufficient. For all of the simulations prescribed with constant activity, the director field was recorded after the system reached a dynamical steady state.

Machine-Learning Details

Neural networks are implemented in Python using the Pytorch library. Code for data preparation, network implementation, training, and evaluation is available online (66). The machine-learning training and testing workflows are summarized in *SI Appendix*, Fig. S2.

Parameter Estimation. Parameter estimation networks contain between one and two convolutional layers with hyperbolic tangent activation functions, each of which is followed by a max-pooling layer and a dropout layer with dropout probability of 0.15. The convolutional layers are further connected with a single recurrent layer implemented with a long short-term memory cell. Finally, a dense layer with linear activation function is added to output the predicted parameters. An example architecture is shown in Fig. 1*E*. To make predictions on large director-field images, the network randomly selects patches and ensemble averages the results into a final prediction. For networks using a recurrent layer, the model accepts a sequence of director-field frames, rather than a single frame.

Three parameter estimation network architectures are used in this paper. The first, used to predict activity in 2D nematics (Fig. 1*F* and *SI Appendix*, Fig. S3), has a single convolutional layer with 32 filters of size 3×3, a single 2×2 max-pooling layer, a recurrent layer implemented using a long short-term memory (LSTM) with hidden size 32, and a fully connected layer with 32 neurons. This model accepts input sequences of 32×32 pixel image patches and was trained on a dataset of 6,000 director-field frames separated by 10 simulation timesteps, at 12 different levels of activity. The second model, used to predict activity in 3D nematics (Fig. 1*G* and *SI Appendix*, Figs. S3 and S5), has a similar architecture, but with $5 \times 5 \times 5$ convolutional filters and no recurrent layer. This model accepts image volumes of size $32 \times 32 \times 32$ and was trained on a dataset of 6,000 director-field configurations, separated by 100 timesteps, at 12 levels of activity. The third model is used for simultaneous prediction of activity and elastic modulus in 2D nematics (Figs. 2–4 and *SI Appendix*, Figs. S6–S8). This network has the same structure as the other 2D model, but outputs two values and was trained on a dataset of 15,000 image frames, generated with 30 different combinations of activity and elastic modulus. The accuracy of this multiparameter estimator is summarized in Fig. 2 *C* and *D*.

The networks used on 2D active nematics were trained on continuous representations of the 2D director field such as $\sin 2\theta$ or $\cos 2\theta$. The choice of continuous representation had no effect on the predictive accuracy of the model. The networks used on 3D active nematics were trained on the traceless tensor $Q_{ij} = n_i n_j - 1/3$, which is the nematic order parameter without ordering magnitude coefficient.

Networks were trained for 100 epochs on director-field configurations generated using lattice Boltzmann simulations. Each frame of training data was a 200×200 director-field image with periodic boundary conditions. These datasets were augmented by applying random rotations, flips, and shifts during the training procedure. During each epoch, each input frame was randomly cropped to the predictive network input size. During training, we used an 80 to 20 training-validation split on the input dataset.

Predicting Time Evolution

Autoencoder Architecture. The neural network for predicting time evolution, depicted in Fig. 5*A*, is composed of three parts: an encoder, recurrent layers, and a decoder. The encoder uses a sequence of convolutional layers to down-sample input images into feature vectors. The decoder accepts feature vectors and uses convolutional layers to up-sample those feature vectors back into images. A traditional autoencoder is composed of these two layers only and is an effective method of reducing data dimensionality. In our model, we insert the recurrent layers in between the encoder and decoder, so that dynamics can be computed on the encoded feature vectors. A benefit of this approach is that the dimensional reduction achieved by the encoder allows for smaller recurrent layers, reducing network complexity and improving performance.

The models reported in this paper accept director-field images processed into the two-channel input $(\sin(2\theta), \cos(2\theta))$, where θ is the local orientation angle of the director field. The encoder contains two convolutional layers of stride 2 with four and six output channels, respectively. The decoder architecture mirrors that of the encoder, accepting a six-channel feature vector and using two stride-2 transposed convolutional layers with four and two output channels, respectively. All convolutional layers use 4×4 kernels and are followed by batch normalization, which improves training performance, as well as hyperbolic tangent activation. The recurrent portion is a two-layer LSTM unit implemented as a residual network (resnet), with a shortcut that directly connects input and output of the entire LSTM cell. Given a sequence of feature vectors, the resnet computes a small residual to be added to the input, rather than computing a full output feature vector from scratch. For input sequences with small time separations,

the residual vector is sparse, which helps improve training performance and predictive accuracy (*SI Appendix*, Fig. S12 *A* and *B*). In particular, the residual architecture allowed the model to more effectively learn the activity-dependent temporal dynamics present in this system (*SI Appendix*, Fig. S12*C*).

These models were trained using a two-step training procedure. First, the encoder, resnet, and decoder were trained together for 100 epochs. Next, the weights in the resnet were frozen and the encoder and decoder were trained together for 50 epochs. Training data were generated either using lattice Boltzmann simulations or directly from experiments. The lattice Boltzmann training data consisted of 200×200 director-field images with periodic boundary conditions, separated by 6, 10, and 25 simulation timesteps. Each simulation dataset contained 6,000 director-field configurations at 12 levels of activity and was augmented during training using random flips, shifts, and crops. As before, we used an 80 to 20 training/validation split. Different models were trained on each dataset, with input image sizes of 48×48, 64×64, and 120×120. In the main text, we report results from the best performing of these models, which were trained on data with a frame separation of 10 timesteps and use 48×48 input image size. All predictions are made using input sequences of seven frames.

The experimental data consisted of 1,500 director-field configurations extracted from microtubule–kinesin experiments at five different ATP concentrations (*Experimental Methods*). We did not train on experiments with ATP concentrations of 10 μM and 18 μM as the time between snapshots was five times longer than for the other ATP concentrations. Here, we also used an 80 to 20 training validation split and augmented data using random flips and crops. The results reported in this paper are for a model with an input size of 48×48.

Stitching Predictions. While the models were trained to predict the evolution of director-field patches, the error rates and characteristic length and time scales reported in Fig 5 are computed for full images in the testing dataset. To obtain the predicted configuration of the full director field, the model stitches together predictions made on overlapping subdomains of the image. Here, each pixel will appear in the prediction for multiple subdomains. The final prediction for each pixel is given by the weighted average of predictions from each subdomain. For a pixel located at $\mathbf{r} = (x, y)$, the weight given to the predicted value from a subdomain centered at $\mathbf{r_0} = (x_0, y_0)$ is the Gaussian weight with $\sigma = R$, the radius of the subdomain. Thus, more credence is given to domains in which the pixel is farther from the boundary. For all results reported in this paper, predictions were stitched together from 48×48 ($R = 24\sqrt{2}$) domains which overlapped by 8 pixels.

Sharpening Algorithm. To reduce any noise or artificial defects that arise from stitching together neural network predictions, we use a physically motivated sharpening procedure. This algorithm exploits only the fact that the system is composed of fibers which tend to align while knowing nothing about the active forces present in the system.

The sharpening procedure minimizes the elastic free energy of a system composed of such fibers. Following refs. 67 and 68, we write the elastic free-energy density as

$$f_d = \frac{1}{2}K_1(\nabla \cdot \mathbf{n})^2 + \frac{1}{2}K_2(\mathbf{n} \cdot \nabla \times \mathbf{n})^2 + \frac{1}{2}K_3(\mathbf{n} \times (\nabla \times \mathbf{n}))^2. \quad [16]$$

Assuming a two-dimensional system parameterized by an angle θ as $\mathbf{n} = (\cos\theta, \sin\theta)$, this becomes

$$f_d = \frac{1}{2}K_1(\sin\theta\partial_x\theta - \cos\theta\partial_y\theta)^2 + \frac{1}{2}K_3(\cos\theta\partial_x\theta + \sin\theta\partial_y\theta)^2. \quad [17]$$

In the one elastic constant approximation $K_1 = K_3 = K$, this reduces to

$$f_d = \frac{1}{2}K\left[(\partial_x\theta)^2 + (\partial_y\theta)^2\right]. \quad [18]$$

The elastic free energy is minimized by setting $\frac{\delta f_d}{\delta\theta} = 0$, leading to the Laplace equation

$$\nabla^2\theta = 0. \quad [19]$$

Thus, the elastic free-energy minimization can be accomplished by applying relaxational dynamics to the director field. We implement this using a standard finite-differences approach, slightly modified to account for the nematic symmetry $\mathbf{n} = -\mathbf{n}$.

We first apply relaxational dynamics in a small box surrounding the topological defect positions from the previous director-field frame. Because the winding number around the boundary of this box is fixed and nonzero, this sharpens the director field around each defect without risking removing the defect. Next, the director field is fixed inside the box and relaxational dynamics are applied in the defect-free region. This procedure is applied iteratively to sharpen the raw predicted image.

This procedure will work if the defect has not moved outside of the box between image frames. Assuming a timestep of τ, box size of R, and characteristic defect velocity v_d, this condition is satisfied if $v_d\tau < R$. We can approximate v_d using the relation provided by ref. 51 for an isolated $+1/2$ defect, $v_d \approx \alpha l_d/\eta$. Here, we insert $l_d \approx \sqrt{K/\alpha}$, the mean defect spacing, as the radius of the defect-free region surrounding the $+1/2$ defect. Thus, the defect will remain in the box if $\sqrt{\alpha K}\tau/\eta < R$. The simulation data used in Fig. 5 had $K = 0.075$, $\eta = 0.33$, $\alpha_{\max} = 0.05$, and $\tau = 10$, leading to $R > 2$. The data reported in the main text were generated using a 5×5 box, corresponding to $R \in [2.5, 3.5]$. We chose the smallest possible value for R above this threshold, as it prevented the immediate annihilation of recently nucleated defect pairs, which would otherwise be close enough to be enclosed by the same box. This would result in a net zero winding around the boundary, leading to their removal by the sharpening procedure.

Applications to Experiment

Before being fed into parameter-prediction models, the actin–myosin images are adjusted in ImageJ to remove outliers using a median filter. The fixed orientation of the dye along the actin filaments and the polarization of the laser yield a polarized image whose intensity is proportional to $\cos^2\theta$, where θ is the director-field orientation. From this we extract the continuous director representation $\cos 2\theta$ and down-sample. For actin–myosin, we down-sample by a factor of 6 to an effective pixel width of $a = 1$ μm, a convention that has been used in the past when comparing this lattice Boltzmann code with actin–myosin nematics (18, 69).

The microtubule–kinesin orientation fields were extracted from fluorescence images using the ImageJ plugin OrientationJ, which determines the structure tensor from intensity gradients. We use the extracted director field to obtain continuous director representations and then down-sample them. Microtubule–kinesin data were down-sampled to an effective pixel width of $a = 2.6$ μm, as the length scale of spatial variations in the raw data is larger. As a machine-learning model will be less reliable when applied to data significantly different from its training set, we aimed to down-sample data such that the correlation length fell within the range of our simulation training dataset.

Determination of Lyapunov Time

Active nematics are a nonlinear system characterized by a positive Lyapunov exponent. As a result, direct comparison of time-evolved director-field configurations is not necessarily valid for long time scales. Pixelwise accuracy should not be expected beyond the Lyapunov time, particularly as our predictive model lacked complete information about the system. While we report pixelwise accuracy in the main text (Fig. 5*B*), knowledge of the chaotic dynamics of these systems is important to contextualize these results.

The grayed-out region in Fig. 5*B* is bounded by the Lyapunov time as determined from lattice Boltzmann simulations. To find this time scale, we ran lattice Boltzmann simulations at different levels of activity and saved an intermediate state of the system. We then perturbed this state and continued the simulation. At each level of activity, we ran 10 trials from 10 separate intermediate states. Each simulation was time evolved on a 200×200 grid, from which 100 points were randomly selected and tracked over time. By comparing these randomly selected pixels as a function of time, we extracted the Lyapunov exponent which was inverted to obtain the Lyapunov time. This quantity was dependent on activity, with more active systems exhibiting a shorter Lyapunov time. However, when we rescaled by the characteristic defect lifetime $\tau_d = \eta/\alpha$, we found that all values coalesced to approximately $\tau = 3.6\tau_{\mathrm{d}}$.

As infinitesimal pixelwise changes would be eliminated by relaxational dynamics, we used a more global method of perturbing the intermediate state. First, we computed the singular-value decomposition of the order parameter tensor field $Q_{ij}(\mathbf{r})$. We then fractionally changed 10 elements of the singular matrix by random amounts between –10% and +10% and used the new matrix to reconstruct the perturbed-order parameter field. This method of globally varying the intermediate state yielded nonvanishing pixelwise deviations that showed exponentially growing behavior.

Characteristic Length and Time Scales

Direct comparisons of the machine-learning predicted director field to lattice Boltzmann simulations are unreliable beyond the Lyapunov time. To evaluate the validity of our predictions over longer time scales, we compare instead characteristic length and time scales of the machine-learning predicted dynamical steady state. For a given order parameter configuration $Q_{ij}(\mathbf{r}, t)$, with i and j running over x, y, we define the spatial correlation function $C_s(\mathbf{r}, t)$ as

$$C_s(\mathbf{r}, t) = \frac{\langle Q_{ij}(\mathbf{r}, t)\, Q_{ij}(0, t)\rangle}{\langle Q_{ij}(0, t)\, Q_{ij}(0, t)\rangle} \qquad [20]$$

and the time correlation function $C_t(\mathbf{r}, t)$ as

$$C_t(\mathbf{r}, t) = \frac{\langle Q_{ij}(\mathbf{r}, t)\, Q_{ij}(\mathbf{r}, 0)\rangle}{\langle Q_{ij}(\mathbf{r}, 0)\, Q_{ij}(\mathbf{r}, 0)\rangle} \qquad [21]$$

where indexes i and j are contracted following the Einstein summation convention. Using Eqs. **20** and **21**, we define the director-field correlation length ℓ_θ such that $C_s(\ell_\theta, t) = 1/2$ and the correlation time t_θ such that $C_t(r, t_\theta) = 1/2$.

In Fig. 5 *C* and *D*, we compare the average values of ℓ_θ, t_θ as found in machine-learning predicted director-field frames to those of lattice Boltzmann simulations. Here, we iterate the predictive model to predict large (200×200) image frames over a long time ($t = 30\ \tau_{\mathrm{LC}}$) and compute the time-averaged correlation length and spatially averaged correlation time. Correlation lengths and correlation times are computed using only the machine-learning–generated image frames. In *SI Appendix*, Fig. S10, we report the time-averaged mean-defect spacing, defined as $\ell_{\mathrm{d}} = 1/\sqrt{n_{\mathrm{d}}}$, where n_{d} is the defect density.

Experimental Methods

Actin–Myosin Nematics. Experiments for Fig. 4*D* and *SI Appendix*, Fig. S6 were performed as in ref. 27 using the method originally described in ref. 18; the experimental data for Fig. 4 *E* and *F* are taken directly from ref. 27. *SI Appendix*, Table S1 contains a full enumeration of the conditions in each experiment but the general method is summarized here. Fluorescent (tetramethylrhodamine labeled) and nonfluorescent actin monomers are mixed at a ratio of 1:5 and allowed to polymerize in F buffer (1 mM MgCl2, 50 mM KCl, 0.2 mM egtazic acid [EGTA]) containing either imidazole (10 mM, pH 7.5) or Hepes (10 mM, pH 7.5) as a buffering reagent. Also present is f-actin capping protein to limit the length of nascent filaments to 2 μm. Additionally the mix contains a glucose oxidase/catalase oxygen scavenging system (2.7 mg/mL glucose oxidase, 1,700 units/mL catalase, 4.5 mg/mL glucose, 0.5% vol/vol β-mercaptoethanol) to limit photodamage. Finally, the polymerization mix contains methylcellulose (0.3% wt/vol in water) as a crowding agent and ATP as a source of chemical fuel.

The experiment is performed in a chamber composed of a glass cloning cylinder attached to a glass coverslip via 5 min epoxy. Before adding the sample to the chamber, the bottom is first coated with a thin layer of oil that contains a surfactant (perfluoropolyether–polyethylene glycol–perfluoropolyether [PFPE-PEG-PFPE] surfactant) to prevent the filaments from sticking to the glass. The sample is allowed to settle as the methylcellulose crowds the filaments to the oil–water interface, forming a nematic liquid crystal. The sample is imaged on an inverted spinning-disk confocal microscope.

Activity is introduced via the addition of synthetic myosin motors after the sample has formed a liquid crystal at the oil–water interface. Two different motor constructs were used. The synthetic motors used for *SI Appendix*, Fig. S6 are the engineered myosin tetramers CM11CD$_{746}$2R∼1R∼TET from ref. 70, referred to in this study as "fixed-gear motor." These motors are constructed from the catalytic domain of a fast algal myosin (*Chara corallina* myosin XI) fused to an artificial lever arm consisting of two spectrin repeats. The motors oligomerize into tetramers with flexible linkages using the engineered leucine zipper variant pLI2 and contain a spectrin repeat flanked by flexible linkers. The synthetic motor used for the experiment in Fig. 4 *E* and *F* is MyLOVChar4∼1R∼TET (71), referred to in this study as "gear-shifting" motor, which is the same construct used for experiments in ref. 27 including the data from that study analyzed in Fig. 4 *E* and *F*. This motor construct is also based on the catalytic domain of *C. corallina* myosin XI, but utilizes an artificial lever arm that contains the light-activatable LOV2 domain that unfolds upon stimulation with blue light. The conformational change of the lever arm results in a light-dependent stroke vector for the motor (71) which, in the context of the active nematic system in this study, results in higher activity in the presence of blue light (27). The fixed-gear and gear-shifting motors were purified, flash frozen, and stored at –80 °C as described in ref. 71. Time-varying activity movies were stimulated over the entire sample via a 491-nm solid-state laser, while spatially varying activation was achieved by targeting a 470-nm light-emitting diode to one location in the sample using a digital micromirror array (27).

Microtubule–Kinesin Nematics. The microtubule and kinesin motor-based active nematics were assembled according to previously published methods (15, 48). Briefly, K401-BIO-HIS purified from *Escherichia coli* (72) was incubated with streptavidin for 30 min to create motor clusters. A mixture containing salt (5 mM $MgCl_2$), an ATP regeneration system (26.6 mM phosphenol pyruvate (Beantown Chemicals), pyruvate kinase/lactic dehydrogenase), the depletion agent (0.8% wt/vol 20 kDa polyethylene glycol), and an antioxidant system (6.7 mg/mL glucose, 0.4 mg/mL glucose catalase, 0.08 mg/mL glucose oxidase, and 2 mM

trolox) was combined with the kinesin motor clusters in a buffer of M2B (80 mM Pipes, pH 6.8, 1 mM EGTA, 2 mM $MgCl_2$). The ATP was added in the desired concentration (from 5 μM to 1,000 μM) to individual aliquots before flash freezing in liquid nitrogen. The concentration of ATP controls the stepping rate of the motors, which in turn tunes the level of activity in the system.

The tubulin was purified from bovine brain (73). Tubulin labeled with NHS-Alexa 647 (74) was copolymerized with unlabeled tubulin in the presence of Guanosine-5-[(α,β)-methyleno]triphosphate GMPCPP (Jena Bioscience NU-405L) to create microtubules with a final fraction of 3% labeled tubulin and an average length of 2.5 μm. The activity in the nematics is sensitive to the particular protein preparation, microtubule (MT) length distribution, and the chemical properties of the active mixture. Therefore, it was important to use stocks from the same preparation and polymerization to get quantitatively consistent results.

The experiments were performed in a flow chamber with dimensions $18 \times 3 \times 0.6$ mm made of double-sided tape sandwiched between a glass slide and a coverslip. The glass slide was treated with commercially available Aquapel to create a hydrophobic surface. The coverslip was passivated with acrylamide.

On the day of experiments, the premixed active components and MTs were thawed rapidly and combined. To form a large flat interface, first oil (HFE 7500) stabilized with a fluoro-surfactant PFPE-PEG-PFPE (1.8% RAN Biotech) was flowed into the chamber and then the aqueous active mixture. The hydrophobic treatment left a thin layer of oil for the MTs to sediment onto. The chamber was sealed with Norland Optical Adhesive and cured under ultraviolet light for 1 min. The sedimentation of the microtubules onto the oil–water interface was aided by spinning the sample in a swinging bucket centrifuge (Sorval Legend RT rotor 6434) at 1,000 rpm for 20 min.

The samples were imaged using epifluorescence microscopy on a Nikon Ti-Eclipse equipped with a CMOS camera (Andor Zyla). The orientation field was extracted from the fluorescence images using the Image-J plugin Orientation-J which finds that structure tensor from gradients in intensity.

Data Availability. Machine-learning code and lattice Boltzmann simulation code data have been deposited in Zenodo (DOI: 10.5281/zenodo.4541607). Some study data are available upon request.

ACKNOWLEDGMENTS. V.V. and J.C. acknowledge support from the Complex Dynamics and Systems Program of the Army Research Office under Grant W911NF-19-1-0268. We acknowledge funding from NSF DMR-1905675 (to M.L.G.) and Army Research Office Multi-University Research Initiative W911NF1410403 (to M.L.G. and V.V.). M.H. acknowledges support from the University of Chicago Materials Research Science and Engineering Center (MRSEC) through Kadanoff-Rice postdoctoral fellowships. Z.B. acknowledges support from NIH Grant GM114627 and NSF Grant DMR-1905675. J.J.d.P. acknowledges support from NSF Grant DMR-1710318 for development of the models and codes used to simulate active nematics. J.J.d.P. also acknowledges support from the Army Research Office (MURI: W911NF-15-1-0568) for the application of active nematics in triggerable materials systems. L.M.L. and Z.D. were supported by the Department of Energy Basic Energy Science through Award DE-SC0019733. This work was partially supported by the University of Chicago MRSEC, which is funded by the NSF under Award DMR-2011854. The calculations presented here were performed on the GPU cluster supported by the NSF under Grant DMR-1828629.

1. P. Mehta *et al.*, A high-bias, low-variance introduction to machine learning for physicists. *Phys. Rep.* **810**, 1–124 (2019).
2. G. Carleo *et al.*, Machine learning and the physical sciences. *Rev. Mod. Phys.* **91**, 45002 (2019).
3. Y. Lecun, Y. Bengio, G. Hinton, Deep learning. *Nature* **521**, 436–444 (2015).
4. J. Schmidhuber, Deep learning in neural networks: An overview. *Neural Network.* **61**, 85–117 (2015).
5. S. Schoenholz, E. Cubuk, D. Sussman, E. Kaxiras, A. Liu, A structural approach to relaxation in glassy liquids. *Nat. Phys.* **12**, 469–471 (2016).
6. V. Bapst *et al.*, Unveiling the predictive power of static structure in glassy systems. *Nat. Phys.* **16**, 448–454 (2020).
7. J. Carrasquilla, R. G. Melko, Machine learning phases of matter. *Nat. Phys.* **13**, 431–434 (2017).
8. E. P. Van Nieuwenburg, Y. H. Liu, S. D. Huber, Learning phase transitions by confusion. *Nat. Phys.* **13**, 435–439 (2017).
9. P. Mehta, D. J. Schwab, An exact mapping between the variational renormalization group and deep learning. arXiv:1410.3831 (14 October 2014).
10. M. Koch-Janusz, Z. Ringel, Mutual information, neural networks and the renormalization group. *Nat. Phys.* **14**, 578–582 (2018).
11. E. D. Cubuk *et al.*, Structure-property relationships from universal signatures of plasticity in disordered solids. *Science* **358**, 1033–1037 (2017).
12. A. Radovic *et al.*, Machine learning at the energy and intensity frontiers of particle physics. *Nature* **560**, 41–48 (2018).
13. M. C. Marchetti *et al.*, Hydrodynamics of soft active matter. *Rev. Mod. Phys.* **85**, 1143–1189 (2013).
14. A. Doostmohammadi, J. Ignés-Mullol, J. M. Yeomans, F. Sagués, Active nematics. *Nat. Commun.* **9**, 3246 (2018).
15. T. Sanchez, D. T. Chen, S. J. DeCamp, M. Heymann, Z. Dogic, Spontaneous motion in hierarchically assembled active matter. *Nature* **491**, 431–434 (2012).
16. F. C. Keber *et al.*, Topology and dynamics of active nematic vesicles. *Science* **345**, 1135–1139 (2014).
17. P. W. Ellis *et al.*, Curvature-induced defect unbinding and dynamics in active nematic toroids. *Nat. Phys.* **14**, 85–90 (2018).
18. N. Kumar, R. Zhang, J. J. De Pablo, M. L. Gardel, Tunable structure and dynamics of active liquid crystals. *Sci. Adv.* **4**, eaat7779 (2018).
19. G. Duclos *et al.*, Topological structure and dynamics of three-dimensional active nematics. *Science* **367**, 1120–1124 (2020).
20. A. J. Tan *et al.*, Topological chaos in active nematics. *Nat. Phys.* **15**, 1033–1039 (2019).
21. J. Hardoüin *et al.*, Reconfigurable flows and defect landscape of confined active nematics. *Commun. Phys.* **2**, 121 (2019).
22. T. B. Saw *et al.*, Topological defects in epithelia govern cell death and extrusion. *Nature* **544**, 212–216 (2017).
23. G. Duclos, C. Erlenkämper, J. F. Joanny, P. Silberzan, Topological defects in confined populations of spindle-shaped cells. *Nat. Phys.* **13**, 58–62 (2017).
24. Y. Maroudas-Sacks *et al.*, Topological defects in the nematic order of actin fibers as organization centres of *Hydra* morphogenesis. *Nat. Phys.* **17**, 251–259 (2021).
25. R. Mueller, J. M. Yeomans, A. Doostmohammadi, Emergence of active nematic behavior in monolayers of isotropic cells. *Phys. Rev. Lett.* **122**, 048004 (2019).
26. K. T. Wu *et al.*, Transition from turbulent to coherent flows in confined three-dimensional active fluids. *Science* **355**, eaal1979 (2017).
27. R. Zhang *et al.*, Structuring stress for active materials control. arXiv [Preprint] (2019). https://arxiv.org/abs/1912.01630 (Accessed 19 February 2021).
28. L. Giomi, Ž. Kos, M. Ravnik, A. Sengupta, Cross-talk between topological defects in different fields revealed by nematic microfluidics. *Proc. Natl. Acad. Sci. U.S.A.* **114**, E5771–E5777 (2017).
29. G. Duclos *et al.*, Spontaneous shear flow in confined cellular nematics. *Nat. Phys.* **14**, 728–732 (2018).
30. C. Blanch-Mercader *et al.*, Turbulent dynamics of epithelial cell cultures. *Phys. Rev. Lett.* **120**, 208101 (2018).
31. R. Aditi Simha, S. Ramaswamy, Hydrodynamic fluctuations and instabilities in ordered suspensions of self-propelled particles. *Phys. Rev. Lett.* **89**, 058101 (2002).
32. R. Green, J. Toner, V. Vitelli, Geometry of thresholdless active flow in nematic microfluidics. *Phys. Rev. Fluids* **2**, 104201 (2017).
33. A. Joshi, E. Putzig, A. Baskaran, M. F. Hagan, The interplay between activity and filament flexibility determines the emergent properties of active nematics. *Soft Matter* **15**, 94–101 (2019).
34. L. M. Lemma, S. J. DeCamp, Z. You, L. Giomi, Z. Dogic, Statistical properties of autonomous flows in 2D active nematics. *Soft Matter* **15**, 3264–3272 (2019).
35. H. Li *et al.*, Data-driven quantitative modeling of bacterial active nematics. *Proc. Natl. Acad. Sci. U.S.A.* **116**, 777–785 (2019).
36. D. Geyer, A. Morin, D. Bartolo, Sounds and hydrodynamics of polar active fluids. *Nat. Mater.* **17**, 789–793 (2018).
37. N. Bain, D. Bartolo, Dynamic response and hydrodynamics of polarized crowds. *Science* **363**, 46–49 (2019).
38. V. Soni *et al.*, The odd free surface flows of a colloidal chiral fluid. *Nat. Phys.* **15**, 1188–1194 (2019).
39. A. Bricard, J. B. Caussin, N. Desreumaux, O. Dauchot, D. Bartolo, Emergence of macroscopic directed motion in populations of motile colloids. *Nature* **503**, 95–98 (2013).
40. L. Giomi, M. J. Bowick, X. Ma, M. C. Marchetti, Defect annihilation and proliferation in active nematics. *Phys. Rev. Lett.* **110**, 228101 (2013).
41. R. Voituriez, J. F. Joanny, J. Prost, Spontaneous flow transition in active polar gels. *Europhys. Lett.* **70**, 404–410 (2005).
42. S. Shankar, S. Ramaswamy, M. C. Marchetti, M. J. Bowick, Defect unbinding in active nematics. *Phys. Rev. Lett.* **121**, 108002 (2018).
43. S. Shankar, M. C. Marchetti, Hydrodynamics of active defects: From order to chaos to defect ordering. *Phys. Rev. X* **9**, 041047 (2019).
44. R. Alert, J. F. Joanny, J. Casademunt, Universal scaling of active nematic turbulence. *Nat. Phys.* **16**, 682–688 (2020).
45. L. Giomi, Geometry and topology of turbulence in active nematics. *Phys. Rev. X* **5**, 1–11 (2015).

46. D. Marenduzzo, E. Orlandini, J. M. Yeomans, Hydrodynamics and rheology of active liquid crystals: A numerical investigation. *Phys. Rev. Lett.* **98**, 118102 (2007).
47. E. J. Hemingway, P. Mishra, M. C. Marchetti, S. M. Fielding, Correlation lengths in hydrodynamic models of active nematics. *Soft Matter* **12**, 7943–7952 (2016).
48. S. J. DeCamp, G. S. Redner, A. Baskaran, M. F. Hagan, Z. Dogic, Orientational order of motile defects in active nematics. *Nat. Mater.* **14**, 1110–1115 (2015).
49. M. Nakamura *et al.*, Remote control of myosin and kinesin motors using light-activated gearshifting. *Nat. Nanotechnol.* **9**, 693–697 (2014).
50. J. Pathak, B. Hunt, M. Girvan, Z. Lu, E. Ott, Model-free prediction of large spatiotemporally chaotic systems from data: A reservoir computing approach. *Phys. Rev. Lett.* **120**, 024102 (2018).
51. L. Giomi, M. J. Bowick, P. Mishra, R. Sknepnek, M. C. Marchetti, Defect dynamics in active nematics. *Philos. Trans. Math. Phys. Eng. Sci.* **372**, 20130365 (2014).
52. K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition. arXiv [Preprint] (2015). https://arxiv.org/abs/1512.03385 (Accessed 19 February 2021).
53. J. Palacci, S. Sacanna, A. P. Steinberg, D. J. Pine, P. M. Chaikin, Living crystals of light-activated colloidal surfers. *Science* **339**, 936–940 (2013).
54. C. Peng, T. Turiv, Y. Guo, Q. H. Wei, O. D. Lavrentovich, Command of active matter by topological defects and patterns. *Science* **354**, 882–885 (2016).
55. S. Čopar, Ž. Kos, T. Emeršič, U. Tkalec, Microfluidic control over topological states in channel-confined nematic flows. *Nat. Commun.* **11**, 59 (2020).
56. J. Gautrais *et al.*, Deciphering interactions in moving animal groups. *PLoS Comput. Biol.* **8**, e1002678 (2012).
57. J. Ling, A. Kurzawski, J. Templeton, Reynolds averaged turbulence modelling using deep neural networks with embedded invariance. *J. Fluid Mech.* **807**, 155–166 (2016).
58. K. Duraisamy, G. Iaccarino, H. Xiao, Turbulence modeling in the age of data. *Annu. Rev. Fluid Mech.* **51**, 357–377 (2019).
59. S. L. Brunton, B. R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics. *Annu. Rev. Fluid Mech.* **52**, 477–508 (2020).
60. R. Zhang, Y. Zhou, M. Rahimi, J. J. De Pablo, Dynamic structure of active nematic shells. *Nat. Commun.* **7**, 13483 (2016).
61. A. Sokolov, A. Mozaffari, R. Zhang, J. J. De Pablo, A. Snezhko, Emergence of radial tree of bend stripes in active nematics. *Phys. Rev. X* **9**, 031014 (2019).
62. C. Denniston, E. Orlandini, J. M. Yeomans, Lattice Boltzmann simulations of liquid crystal hydrodynamics. *Phys. Rev. E* **63**, 056702 (2001).
63. M. Ravnik, S. Žumer, Landau–de Gennes modelling of nematic liquid crystal colloids. *Liq. Cryst.* **36**, 1201–1214 (2009).
64. C. Denniston, D. Marenduzzo, E. Orlandini, J. M. Yeomans, Lattice Boltzmann algorithm for three-dimensional liquid-crystal hydrodynamics. *Philos. Trans. Math. Phys. Eng. Sci.* **362**, 1745–1754 (2004).
65. R. Zhang, T. Roberts, I. S. Aranson, J. J. De Pablo, Lattice Boltzmann simulation of asymmetric flow in nematic liquid crystals with finite anchoring. *J. Chem. Phys.* **144**, 084905 (2016).
66. J. Colen, M. Han, R. Zhang, ML_ActiveNematics. Zenodo. https://doi.org/10.5281/zenodo.4541607. Deposited 15 February 2021.
67. P. G. de Gennes, J. Prost, *The Physics of Liquid Crystals* (Oxford University Press, 1995).
68. P. M. Chaikin, T. C. Lubensky, *Principles of Condensed Matter Physics* (Cambridge University Press, 1995).
69. R. Zhang, N. Kumar, J. L. Ross, M. L. Gardel, J. J. De Pablo, Interplay of structure, elasticity, and dynamics in actin-based nematic materials. *Proc. Natl. Acad. Sci. U.S.A.* **115**, E124–E133 (2017).
70. T. D. Schindler, L. Chen, P. Lebel, M. Nakamura, Z. Bryant, Engineering myosins for long-range transport on actin filaments. *Nat. Nanotechnol.* **9**, 33–38 (2014).
71. P. V. Ruijgrok *et al.*, Optical control of fast and processive engineered myosins in vitro and in living cells. *Nat. Chem. Biol.*, 10.1038/s41589-021-00740-7 (2021).
72. R. Subramanian, J. Gelles, Two distinct modes of processive kinesin movement in mixtures of ATP and AMP-PNP. *J. Gen. Physiol.* **130**, 445–455 (2007).
73. M. Castoldi, A. V. Popov, Purification of brain tubulin through two cycles of polymerization–depolymerization in a high-molarity buffer. *Protein Expr. Purif.* **32**, 83–88 (2003).
74. A. Hyman *et al.*, "Preparation of modified tubulins" in *Molecular Motors and the Cytoskeleton Methods in Enzymology*, R. B. Vallee, Ed. (Academic Press, 1991), vol. 196, pp. 478–485.