CELL-E: Biological Zero-Shot Text-to-Image Synthesis for Protein Localization Prediction

Emaad Khwaja[*‡] emaad@berkeley.edu

Yun S. Song[†‡§] yss@berkeley.edu

Bo Huang[§¶‖] bo.huang@ucsf.edu

Abstract

Predicting the cellular activities of proteins from their primary amino acid sequences is a highly desirable capability that could greatly enhance our functional understanding of the proteome. Here, we demonstrate CELL-E, a text-to-image transformer architecture, which given a protein sequence and a reference image for cell (or nucleus) morphology, can generate a 2D probability density map of the protein distribution within cells. Unlike previous *in silico* methods, which rely on existing, discrete class annotation of protein localization to predefined subcellular compartments, CELL-E uses imaging data directly, thus relying on a native description of protein localization relative to the cellular context.

[*]UC Berkeley - UCSF Joint Bioengineering Graduate Program
[†]Department of Statistics, UC Berkeley, CA 94720
[‡]Computer Science Division, UC Berkeley, CA 94720
[§]Chan Zuckerberg Biohub, San Francisco, CA 94158
[¶]Department of Pharmaceutical Chemistry, UCSF, San Francisco, CA 94143
[‖]Department of Biochemistry and Biophysics, UCSF, San Francisco, CA 94143

Preprint. Under review.

1 Introduction

Recent advancements in sequencing technologies have enabled cataloging of a vast amount of proteins and their amino acid sequences in many organisms [1]. Still, for many proteins, their actual function and cellular dynamics remain enigmatic. *In silico* prediction of these properties, including structure, interaction partners, subcellular localization and trafficking patterns, from their amino acid sequences, is thus a highly sought after path of research. Such tools could not only provide clues to decipher "dark matters" in the proteome but also enable large-scale computational screening prior to the much more costly experimental validations. The advent of these technologies would be highly beneficial to many avenues of biomedical research, such as: therapeutic target discovery, drug design, and biochemical pathway exploration [2].

In this work, we focus on predicting subcellular localization of proteins from the amino acid sequences. Eukaryotic cells contain several subcellular structures, defined by either membrane compartments or biomolecular interactions [3]. These subcellular structures serve as the spatial context for the functioning of specific proteins, and mislocalization of proteins is often implicated in cardiovascular, metabolic, and neurodegenerative diseases, as well as cancer [4]. Localization of a protein to a given structure could be driven by either active transport or passive diffusion[5] in conjugation with specific protein-protein interactions [2, 6, 7], often involving localization "signals" in its amino acid sequence [8, 9, 10]. In many cases, however, the exact cellular mechanism for sequence recognition and trafficking has yet to be fully elucidated [11]. There is an ongoing debate about the mechanism dictating import via the nuclear localization sequence (NLS), for example. Further discovery is needed to identify how the nuclear pore complex allows cargo protein to pass through the nuclear membrane [12]. Furthermore, not all NLSs within the proteome have been identified. Machine learning from existing knowledge of protein localization is therefore particularly useful.

Although computational prediction of protein subcellular localization from primary amino acid sequences is an active area of research, most works train the model with class annotation of subcellular compartments (e.g. nucleus, plasma membrane, endoplasmic reticulum, etc.) which are available from databases such as UniProt [13]. This approach has two major limitations. First, many proteins are present in different and variable amounts across multiple subcellular compartments. Second, protein localization could be highly heterogeneous and dynamic depending on the cell type and cell state (including cell cycle state). Neither of these two aspects have been captured by existing discrete class annotations. Consequently, machine-learning-based protein localization prediction still have limited applications [14]. Furthermore, to assist mechanistic discoveries, it is highly desirable for the machine learning models to be explainable.

To address these issues, we present CELL-E, a text-to-image based transformer which predicts the probability of protein localization on a per-pixel level from a given protein sequence and a conditional reference image for the cell morphology and location (we used nucleus image for this work). The intent of the reference image is to provide CELL-E with information on cell type and cell state context for CELL-E prediction. By generating a two-dimensional probability density function (2D PDF) atop the reference image, CELL-E naturally accounts for multi-compartment localization.

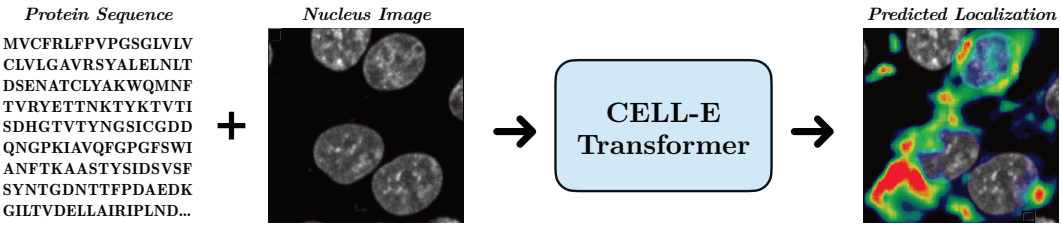

Figure 1: Given an input of amino acids and a reference nucleus image, CELL-E makes a prediction of protein localization with respect to the nucleus as a 2D probability density function. Lysosome-associated membrane glycoprotein 2 depicted above is expected to localize to vesicles [15]. **Protein Sequence:** Amino acid letter codes used as the text input shown. **Nucleus Image:** The nucleus image is a random crop of a microscope slide image. Four nuclei are shown, marked as gray circles with pixel intensity corresponding to DNA concentration via Hoescht stain. **Predicted Localization:** CELl-E prediction shown as heatmap, with color indicating relative confidence for each pixel.

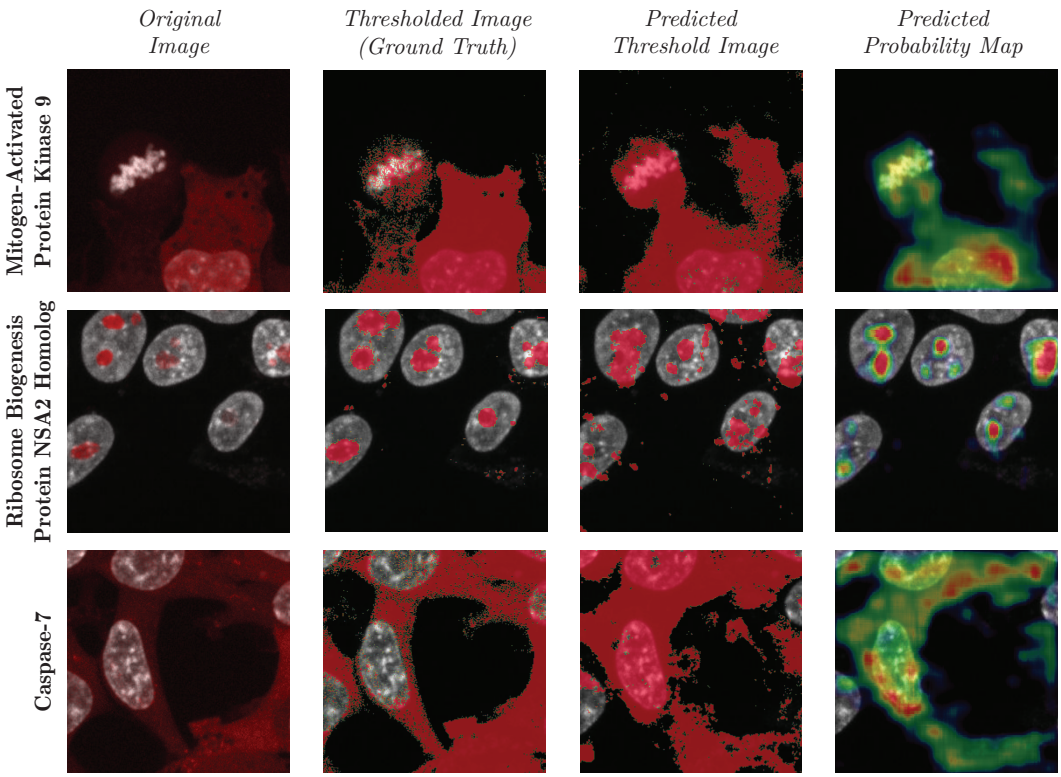

Figure 2: Prediction results of several types of proteins from the validation set, unseen to the model during training. The nucleus channel is depicted in grayscale, and the protein channel is shown as an overlay in red (Fig. S.1 for clarification). The thresholded image (Column 2) is designated "Ground Truth" because those are the types of images exposed to the model during training. Information on protein properties and expected localization of depicted results are available in Table S.1.

CELL-E is the first demonstration, to our knowledge, of text-to-image synthesis within the cell imaging domain. It relies on transfer learning via amino acid encodings from a pretrained language model, and two quantized image encoders trained from a live-cell imaging dataset. With CELL-E, one can visualize the probability distribution of any variable amino acid sequence (< 1000 amino acids) on a given cell (Fig. 1). CELL-E is capable of localization prediction for many types of proteins (Fig. 2, S.2). Our code is publicly available on our GitHub: `https://github.com/BoHuangLab/Protein-Localization-Transformer`.

2 Related Work

2.1 Protein Language Models

Natural language processing (NLP) has been an important starting point for protein sequence encoding due to the fact that, like long-form text, amino acids can have long contextual dependencies which materialize in a folded three dimensional structure [16]. Word-embedding models, such as GPT-3, will typically have a code book containing numbered indices corresponding to thousands of unique potential words [17]. Since a protein sequence uses a comparatively small vocabulary of 20 standard amino acids, representing protein sequences is more akin to character-level modelling [18].

There are multiple self-supervised models borrowed from the NLP field which have demonstrated state-of-the-art performance predicting protein properties solely from an amino acid sequence input [19, 20, 21, 22]. These language models are trained on millions of protein sequences from several established databases, such as BFD [23], Protein Data Bank [24], UniRef [25], and Pfam [26]. Language models have performed well in downstream tasks involving structure prediction, evolutionary understanding, and protein engineering [27]. LSTM and attention-based models have been particu-

larly successful [28]. UniRep is an example of an LSTM model, which are tasked with predicting the next amino acid token given a variable length string input [29]. BERT uses bidirectional masked language modelling, where instead of predicting the next token, it predicts the identity of randomly masked tokens throughout the entire sequence [30]. Facebook's Evolutionary Scale Model (ESM) is also an attention-based model pre-trained with 250M protein sequences [20].

2.2 Protein Localization Predictors

While traditional supervised approaches to predicting localization (such as stochastic modelling) have been restricted by limited feature representation or prohibitively long computation times, deep learning approaches have allowed for great strides in the predictive power of models [31]. The ability to represent profoundly complex patterns via optimization of millions of parameters is one reason for the success of deep neural networks [32]. This level of complexity can be difficult to achieve through manual feature extraction [33].

Protein localization has been primarily framed as a class-prediction task. One-dimensional (1D) localization predictors generally take a primary sequence as input and output a fixed-length vector, where each entry corresponds to a sub-cellular location (either organelle or sub-organelle) being considered. The values of each entry are probability values, and the predicted classification is selected based on entries with values larger than a predetermined threshold [34, 35]. The limitations of 1D methods are multi-fold. For one, discrete classifications for regions of the cell which are contiguous like the nuclear membrane, endoplasmic reticulum, and parts of the Golgi apparatus can be spatially ambiguous and have flawed annotation within established datasets (Fig. S.3) [36].

2.3 Text-To-Image Generation

Ramesh et al. [37] from OpenAI demonstrated true zero-shot text-to-image generation with DALL-E, in that DALL-E was able to make novel image predictions with high fidelity from prompts not found in the training data. DALL-E relies on an autoregressive framework which was trained to learn a joint distribution of text and image tokens [37]. Previous frameworks utilized variational autoencoder (VAE [38]) or General Adversarial Network (GAN [39])-based architectures, but performed poorly outside of the training data, suffering from distortion and image artifacts [40, 41, 42]. A similar approach was taken for CogView [43].

In this work, we demonstrate the prediction of protein localization as an image from an amino acid sequence and a cell morphology image precursor. To our knowledge, this is the first implementation of text-to-image synthesis within the biological domain. Neither DALL-E nor CogView specifically sought to incorporate a conditional image as part of the prompt, which we specifically demonstrate in this work. There have been models which have utilized additional sources of conditioning information beyond just text, including image inputs [**?** 44, 45]. These implementations have been restricted to GANs. As such, the input sequences were quite short (only consisting of few word captions), and image fidelity has paled when compared to results of the previously mentioned transformed-based models. Recently DALL-E 2, which substitutes dVAE for a diffusion-guided prior for creating image embeddings, demonstrated higher image fidelity when incorporating an image embedding in the prompt, but this conditoned synthesis merely served to help the model converge towards a solution in a similar region of latent space, producing an image of similar content [46]. Our model utilizes a nucleus image as a prompt, but generates dependent protein threshold images, which represent a wholly new image domain with markedly different image features.

The intent of models like DALL-E, CogView, and DALL-E 2 is to produce perceptually strong images with high fidelity, and they do not seek to produce images with any degree of numerical accuracy. While our method does similarly allow for truly zero-shot protein image prediction (Fig. S.4), our goal for image generation in this context is to extend beyond visual fidelity and provide a degree of spatial accuracy to inform scientists in research settings. Protein abundance represented in images is a dynamic process which fluctuates with respect to cell state and environmental factors. A snapshot view, therefore, is incapable of capturing a global estimation of accumulation within a cell. As such, rather than realistic looking images of fluorescently labelled protein, the model is tasked with predicting per-pixel binary probability representations of protein localization. These predictions can then be linearly combined with weighting to generate 2D PDFs of protein localization.

3 Dataset

Data availability in this domain is a large obstacle. DALL-E was trained on 250 million text-images pairs. Protein sequence databases for which there are corresponding cellular images are comparatively smaller. For example, the largest publicly available dataset with annotated protein images in human cells, Human Protein Atlas (HPA), contains 12,003 unique proteins with just 82,000 immunofluorescence staining and confocal microscopy (ICC-IF) images [47]. Additionally, image captions are often much shorter than full protein sequences sequences. CogView and DALL-E relied on text encoding which were only 256 word tokens in length. Protein sequences can contain upwards of 600 amino acid residues, where any portion could have functional and localization-critical domains [48]. Another difference is NLP utilize dictionaries which contain thousands of possible words, whereas a protein language model only needs to encode at least 20 entries for the canonical amino acids.

In this work, we chose to use the OpenCell dataset [49] for training and validation because of its strong image quality, uniformity of imaging and cell conditions, and availability of consistent morphology reference images. OpenCell contains a library of 1,311 CRISPR-edited HEK293T human cell lines, each with one target protein fluorescently tagged via the split-mNeonGreen2 system. For each cell line, the OpenCell imaging dataset contain 4-5 confocal images of the tagged protein with accompanying DNA staining as the reference for nuclei morphology. The cells were imaged alive, which collectively represents a more accurate depiction of protein distribution within the cell than immunofluorescence. More information can be found in Section S.3.

4 Methods

Like DALL-E, our model autoregressively learns text and image tokens as a single stream of data. In our case, however, we utilize 3 separate embedding spaces corresponding to amino acid tokens, nucleus image tokens, and protein threshold image tokens. We also use a similar multi-phase training approach, but our model also uses pre-trained language-model input embeddings for the amino acid text sequences via BERT:

- **Phase 1** A Vector Quantized-Generative Adversarial Network (VQGAN) [50] is trained to represent a single channel 256×256 nucleus image as a grid comprised of 16 × 16 image tokens (Fig. S.6), each of which could be one of 512 tokens
- **Phase 2** A similar VQGAN is trained on images corresponding to binarized versions of protein images. These tokens represent the spatial distribution of the protein (Fig. S.8).
- **Phase 3** The VQGAN image tokens are concatenated to 1000 amino acid tokens for the autoregressive transformer which models a joint distribution over the amino acids, nucleus image, and protein threshold image tokens.

The optimization problem is modelled as maximizing the evidence lower bound (ELBO) [38, 51] on a joint likelihood distribution over protein threshold images u, nucleus images x, amino acids y, and tokens z for the protein threshold image: $p_{\theta,\psi}(u,x,y,z) = p_\theta(u|x,y,z)p_\psi(x,y,z)$. This is bounded by:

$$\ln\ p_{\theta,\psi}(u,x,y) \geq \underset{z\sim q_\phi(z|u)}{\mathbb{E}}[\ln\ p_\theta(u|x,y,z)] - \mathrm{KL}(q_\phi(x,y,z|u),\ p_\psi(x,y,z))$$

where q_ϕ is the distribution 16 × 16 image tokens from the VQGAN corresponding to the threshold protein image u, p_θ is the distribution over protein threshold generated by the VQGAN given the image tokens, and p_ψ indicates the joint distribution over the amino acid, nucleus, and protein threshold tokens within the transformer.

4.1 Nucleus Image Encoder

Training both image VQGANs maximizes ELBO with respect to ϕ and θ. The VQGAN improves upon existing quantized autoencoders by introducing a learned discriminator borrowed from GAN architectures [50]. The Nucleus Image Encoder is a VQGAN which represents 256 × 256 nucleus reference images as 256 16 × 16 image patches. The VQGAN codebook size was set to n =512 image patches. Further details can be found in Section S.4.2.

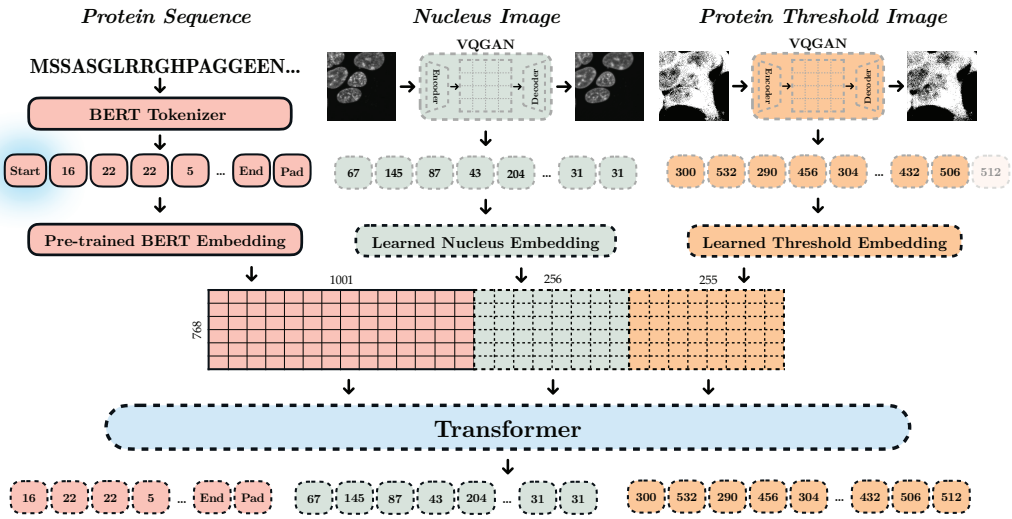

Figure 3: Graphical depiction of CELL-E. Solid lines correspond to pre-trained components. Gray dashed lines are learned in Phase 1 and 2 (Nucleus and Protein Threshold VQGANs). Black dashed lines correspond to components learned in Phase 3. A start token is prepended to the sequence and the final protein image token is removed. The protein sequence embedding from the model is preserved, and embedding spaces for the image tokens are cast in the same depth and concatenated with the protein sequence embedding. The transformer is tasked with reproducing the original sequence of tokens (e.g. the input sequence with start token shifted to the right one position).

4.2 Protein Threshold Image Encoder

The protein threshold image encoder learns a dimension reduced representation of a discrete binary PDF of per-pixel protein location, represented as an image image. We adopt a VQGAN architecture identical to the Nucleus VQGAN. The VQGAN serves to approximate the total set of binarized image patches. While in theory a discrete lookup of each pixel arrangement is possible, this would require $\sim 1.16 \times 10^{77}$ entries, which is computationally infeasible. Furthermore, some distributions of pixels might be so improbable that having a discrete entry would be a waste of space.

Protein images are binarized with respect to a mean-threshold, via:

$$\bar{u}_{i,j} = \begin{cases} 1, & u_{i,j} \geq \mu, \\ 0, & u_{i,j} < \mu, \end{cases}$$

$\forall$ pixels $u \in$ image U of size $i \times j$, where μ is the mean pixel intensity in the image (Fig. S.7).

The 16×16 image patches learned within the VQGAN codebook therefore correspond to local protein distributions. In Section 4.5, we detail how a weighted sum over these binarized image patches is used to determine a final probability density map. Hyperparameters and other training details can be found in Section S.4.3.

4.3 Amino Acid Embedding

For language transformers, it is necessary to learn both input embedding representations of a text vector as well as attention weights between embeddings [28]. In practice, this creates a need for very large datasets [52]. The OpenCell dataset contains 1,311 proteins, while the human body is estimated to contain upwards of 80,000 unique proteins [53]. It is unlikely that such a small slice could account for the large degrees of variability found in nature.

In order to overcome this obstacle, we opted for a transfer learning strategy, where fixed amino acid embeddings from a pretrained language model exposed to a much larger dataset were utilized. We found the strongest performance came from BERT embeddings [27], which was trained on thirty-one million protein sequences from pFAM [54]. Utilizing pretrained embeddings had the two-fold benefit of giving our model a larger degree of protein sequence context, as well as reducing the number of trained model parameters, which allowed us to scale the depth of our network.

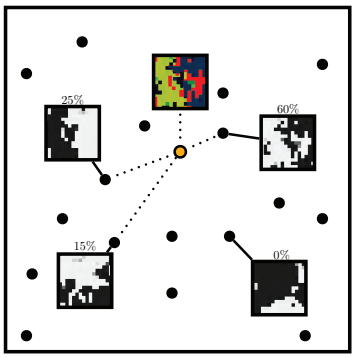

Figure 4: Simplified example of probability map calculation. Each circle corresponds to an image token within the quantized VQGAN embedding space. Each PDF patch (yellow) is obtained as a weighted sum over all protein threshold image VQGAN codebook vectors.

We tried training using random initialization for amino acid embeddings (See Section S.6.5), however, we noted overfitting on the validation set image reconstruction and high loss on validation sequences. We also experimented with other types of protein embeddings, including UniRep and ESM1-b, but observed top performance from BERT encodings (Table S.2).

4.4 CELL-E Transformer

The autoregressive transformer (p_ϕ) utilizes an input comprised of amino acid tokens, a 256 $\times$ 256 nucleus image crop, and the 256 $\times$ 256 corresponding protein image threshold crop. In this phase, ϕ and θ are fixed, and a prior over all tokens is learned by maximizing ELBO with respect to ϕ. It is a decoder-only model [55].

The model is trained on a concatenated sequence of text tokens, nucleus image tokens, and protein threshold image tokens, in order. Within the CELL-E transformer, image token embeddings were cast into the same dimensionality as the language model embedding to in order to maintain the larger protein context information, however the embeddings corresponding to the image tokens within this dimension are learned.

4.5 Probability Density Maps

When generating images, the model is provided with the protein sequence and nucleus image. The transformer autoregressively predicts the protein-threshold image. In order to select a token, the model outputs logits which contain probability values corresponding to the codebook identity of the next token. The image patch v_i is selected by filtering for the top 25% of tokens and applying sampling with gumbel noise [56].

Ordinarily, the final image is generated by converting the predicted codebook indices of the protein threshold image to the VQGANs decoder. However, to generate the probability density map $\bar{v}$, we include the full range of probability values corresponding to image patches, $p(v_i)$, obtained from the output logits. The values are clipped between 0 and 1 and multiplied by the embedding weights within the VQGAN's decoder, w_i:

$$\bar{v} = w \cdot \; p(v) = \sum_{i=1}^{n} w_i \, p(v_i)$$

This output is normalized and displayed as a heatmap (Fig. 4).

5 Results

Image Accuracy: We used several metrics to evaluate the reconstructive performance of CELL-E, summarized in Table 1. Performance is reported as mean $\pm$ standard deviation where applicable. The VQGAN columns indicate the value of these metrics evaluated on the ground truth threshold

Table 1: Image Accuracy

	Train		Validation	
	CELL-E	VQGAN	CELL-E	VQGAN
Nucleus Proportion Accuracy	0.94 ± 0.05	0.99 ± 0.01	0.81 ± 0.18	0.99 ± 0.01
Predicted Threshold Pixel Accuracy	0.77 ± 0.06	0.87 ± 0.08	0.77 ± 0.05	0.88 ± 0.08
Predicted 2D PDF Pixel Accuracy	0.68 ± 0.10		0.63 ± 0.10	
SSIM	0.32 ± 0.21	0.55 ± 0.25	0.25 ± 0.16	0.56 ± 0.25
IS	2.77 ± 0.07	4.17 ± 0.17	2.13 ± 0.07	3.87 ± 0.17
FID	107	15	156	23

Table 2: Nuclear Localization Prediction Accuracy

	Train	Validation
VQGAN	0.99 ± 0.08	0.99 ± 0.09
CELL-E	$\mathbf{0.89 \pm 0.31}$	0.72 ± 0.45
MuLoc	0.71 ± 0.45	$\mathbf{0.79 \pm 0.41}$
Subcons	0.43 ± 0.49	0.69 ± 0.46

image passed through the protein threshold VQGAN. As CELL-E selects tokens from this VQGAN to produce its outputs, these values represent the best possible performance for our model. Nucleus proportion accuracy measures how close the estimated proportion of pixel intensity within the nucleus is to the ground truth thresholded image. We believe this is the most relevant metric as it is not obscured by small spatial variations and nucleus boundaries can be obtained from the dataset. Predicted Threshold Pixel Accuracy and Predicted 2D PDF Pixel Accuracy are measured as $1-$ Root-Mean Squared-Error (RMSE). We also measure the models performance in structural similarity index matching (SSIM) [57], Inception Score (IS) [58], and Fréchet Inception Distance (FID) [59]. More information on the evaluation procedure can be found in Section S.5.1.

Overall we see DALL-E is a fairly strong performer in all categories. In Section S.6.5, we performed ablation studies with respect to the nucleus proportion accuracy on the validation set to understand the performance improvement introduced from each model component. We noted that a completely sequence-blind model was able to predict nucleus proportion with accuracy = .7140. This is likely due to class imbalance in the OpenCell dataset, which contains $\sim 55\%$ nuclear proteins. We do still observe a $\sim 10\%$ increase prediction accuracy with the inclusion of sequence information.

Classification Accuracy: While not specifically trained as a discrete localization classifier, we also performed naive comparison between CELL-E model and 1D protein localization classifiers MuLoc [60] and Subcons [61]. Results are summarized in Table 2.

We specifically focused on nuclear classification. For CELL-E, we used a naïve classification criteria. If >50% of pixel intensity for the 2D PDF was found in the nucleus, it was classified as a nuclear localizing protein. VQGAN data is shown again. We were surprised to observe a relatively high degree of accuracy from this method compared to the task-specific models. CELL-E outperformed on the training set, but was a close second for validation set proteins.

6 Discussion

Visualizing Attention: In Fig. 5, we visualize attention weights on the input amino acids, nuclei, and generated protein threshold tokens corresponding to the predicted localization of DNA Topoisomerase I (TOP1), a protein responsible for uncoiling during transcription and replication, and creating double stranded breaks during replication [62]. We display attention on significant tokens when nuclear localization is predicted, designated as those with weights higher than the mean of non-discarded sequence tokens weights. More details on this process can be found in Section S.6.1.

NLSs are short 7-20 amino acid sections which tag a protein to be transported into the nucleus and typically rich in basic amino acids (R, H, K) [63]. They have been of particular interest recently

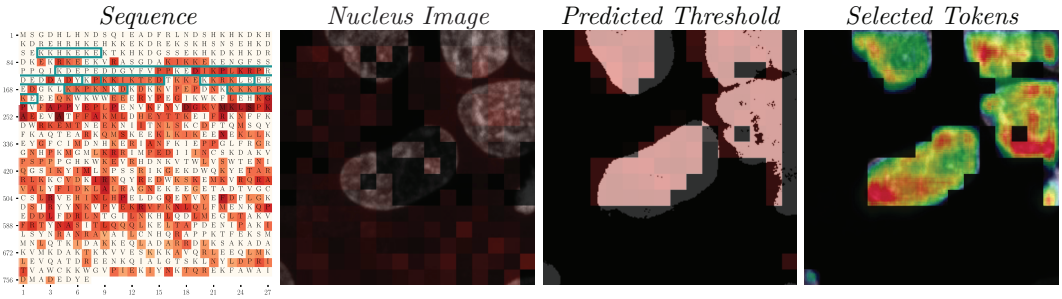

Figure 5: Attention weights for significant tokens when patches containing a large percentage of protein are selected (right column). Documented NLSs are boxed in blue (left column).

in the gene-editing space for optimizing CRISPR-Cas9 delivery to genetic material in the nucleus [64]. TOP1 is a nucleus-localizing protein, and we note high attention on several NLSs described in literature [65].

Novel NLS Discovery: We noted several non-NLS amino acid residues with high attention weights, however it is unclear if these have any true effect on localization in practice. We decided to perform our first *in silico* experiment by simply inputting the protein nucleus image with some of the most highly attended-to sequences containing several basic amino acids. As a sanity check, we also passed a sequence with very low attention (See Fig. S.10). Our model predicted the DGKVMKLSPKAE sequence, in particular, would be highly localized to the nucleus. If true, this is a completely novel finding as the sequence does not resemble any entry found in the NLS database [66, 67]. We hope this model enables researches to pursue experimentation which elucidates their understanding of domains within proteins.

Reproducing Experimental Results *In Silico*: We embarked on reproducing results from literature involving protein sequence modifications. In Fig. S.11, we reproduce the findings of Mo et al. [65] by inputting sequences corresponding to the transfected eGFP plasmids used in the study. Each plasmid contained different portions of the TOP1 sequence removed.

We observe similar results to the wet lab experiments, save for pTI-8. In pTI-5 and pTI-10, we observe patterning (i.e. smaller circles) which may indicate subnuclear localization to the nucleolus, in tandem with comments made in the original study. It is important to acknowledge that they used HeLa cells in their experiment, which is markedly different in shape and likely different in protein expression some degree from the HEK293T cells the model was trained on [68].

Extra-domain Prediction: We tried appending known NLS sequences to GFP (Fig. S.12). This protein is ubiquitous in biological investigation, but non-native to humans. CELL-E struggles to make any prediction with high confidence from the base sequence. However, when we prepend known NLS signals (nucloplasmin and c-Myc NLSs) to the GFP sequence, the model predicts protein localization within the nucleus, demonstrating the model's awareness of their function despite being placed in a highly unfamiliar context. We visualize these results within a UMAP of joint CELL-E sequence-image encodings (Fig. S.13) and observe a transition from a cytosolic region to a nuclear region upon appending the NLSs.

7 Future Works

CELL-E's performance seems to be currently limited by the scope of the OpenCell dataset, which only accounts for a hand full of proteins within a single cell type and imaging modality. As the OpenCell project is an active development, we expect stronger performance as data becomes available. The availability of brightfield (e.g. phase-contrast) images as the morphological reference will also likely improve the prediction of cytoplasmic protein localization compared to using nuclei images. Furthermore, the utility of the model comes in terms of linking embededding spaces of dependent data. One could imagine follow up experiments where rather than images being the prediction, other signatures such as protein mass spec could be predicted. Additionally, other sources of information, such as structural embeddings could be incorporated to enhance CELL-E's capabilities.

Acknowledgments and Disclosure of Funding

B.H. is supported by the National Institutes of Health (R01GM131641). Y.S.S. and B.H. are Chan Zuckerberg Biohub Investigators.

SUPPLEMENTARY MATERIAL

S.1 Introduction - CELL-E Images and Figure Information

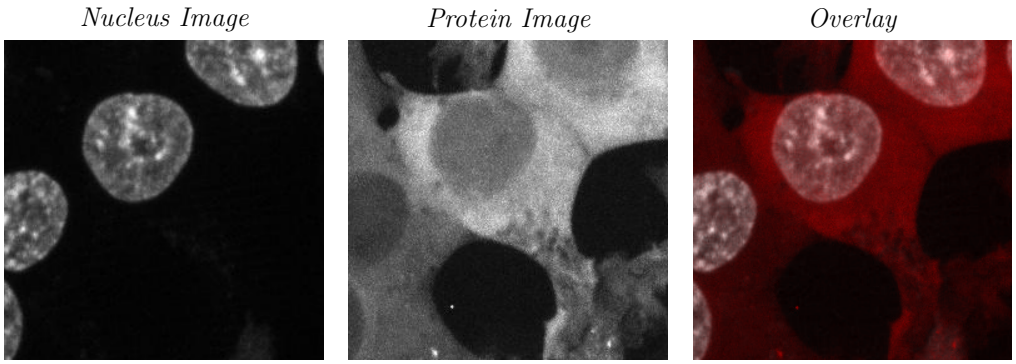

Figure S.1: Nucleus Image (left), Protein Image (middle), and Overlay (Right). The alpha value for the protein channel in the right column is set to .7. Overlay is used as the "Original Image" in Fig. 2 and Fig. S.2.

Table S.1: Depicted Results Localization Annotations

Figure	Protein Name	OpenCell Annotation
Fig. 2	Mitogen-Activated Protein Kinase 9	Cytoplasmic, Nucleoplasm
Fig. 2	Ribosome Biogenesis Protein NSA 2 Homolog	Nucleolus - GC
Fig. 2	Caspase-7	Cytoplasmic, Nucleoplasm, Aggregates
Fig. S.2	DNA Topoisomerase 2-Alpha	Chromatin, Nucleoplasm, Nuclear punctae, Nuclear Membrane
Fig. S.2	Ankyrin Repeat Domain-Containing Protein 17	Cytoplasmic, Nucleoplasm
Fig. S.2	Golgi Reassembly-Stacking Protein 2	Golgi
Fig. S.2	UDP-xylose and UDP-N-acetylglucosamine Transporter	Vesicles, Golgi

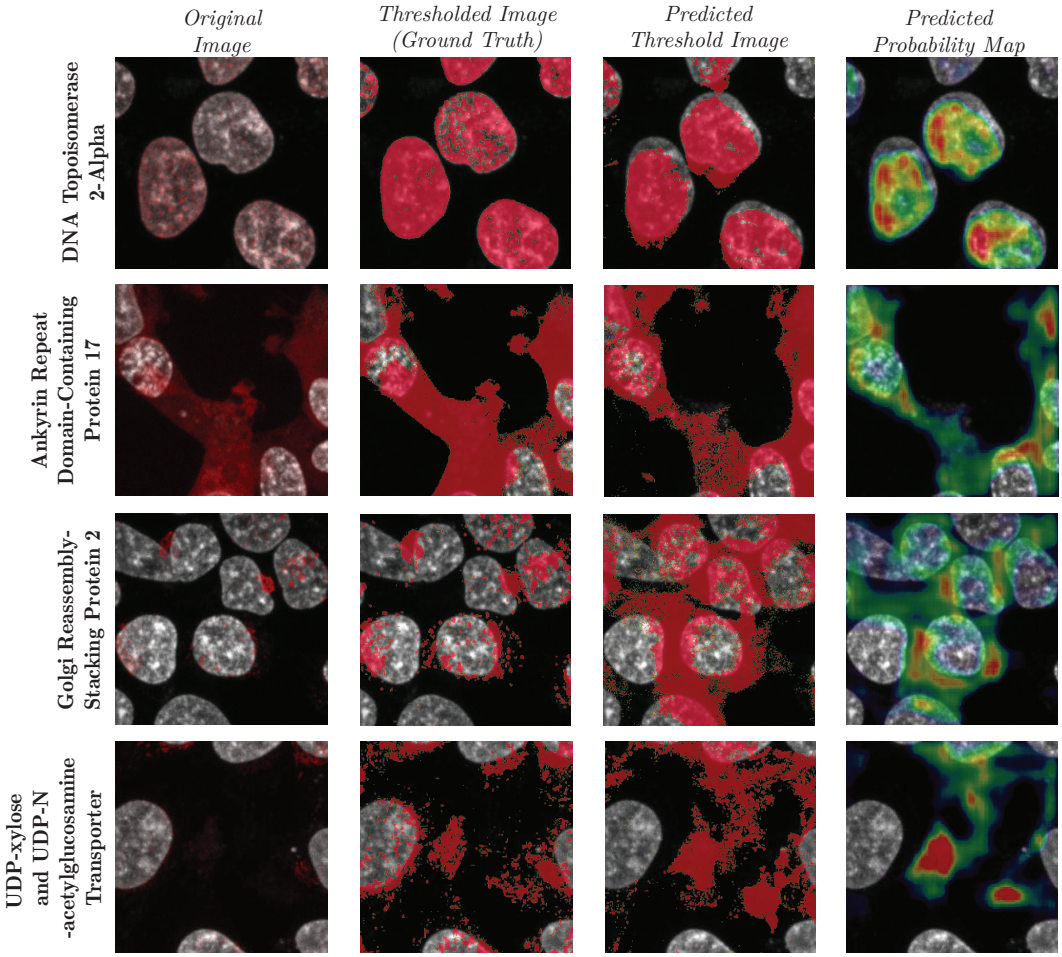

Figure S.2: More prediction results from the validation set. We observe a high degree of spatial awareness from the model, notably in UDP-xylose-acetylglucosamine Transporter, which accurately predicts signal between cell nucei with high confidence.

S.2 Related Works

S.2.1 Text-to-Image Generation

The following images (Fig. S.4, Fig. S.5) are from text-to-image models architecturally similar to CELL-E, but replace the Protein Threshold VQGAN with a similarly trained Protein Image VQ-GAN.

The sequence-to-image transformer is similar to CELL-E and does not take a nucleus image as input. This produces perceptually similar images (Fig. S.4). Within the training data, the corresponding protein image is merely a snapshot in time and could be markedly different if taken at another time-point. For this reason, we do not believe a image prediction without confidence provides scientific utility. Fig. S.5 shows model outputs model similar to Fig. S.4 above, but does include image synthesis conditioned on a nucleus image input (via Nucleus Image VQGAN). The predicted outputs are perceptually more similar to the ground truth protein image, but the questions of accuracy and scientific utility mentioned previously continue to be a factor within this paradigm.

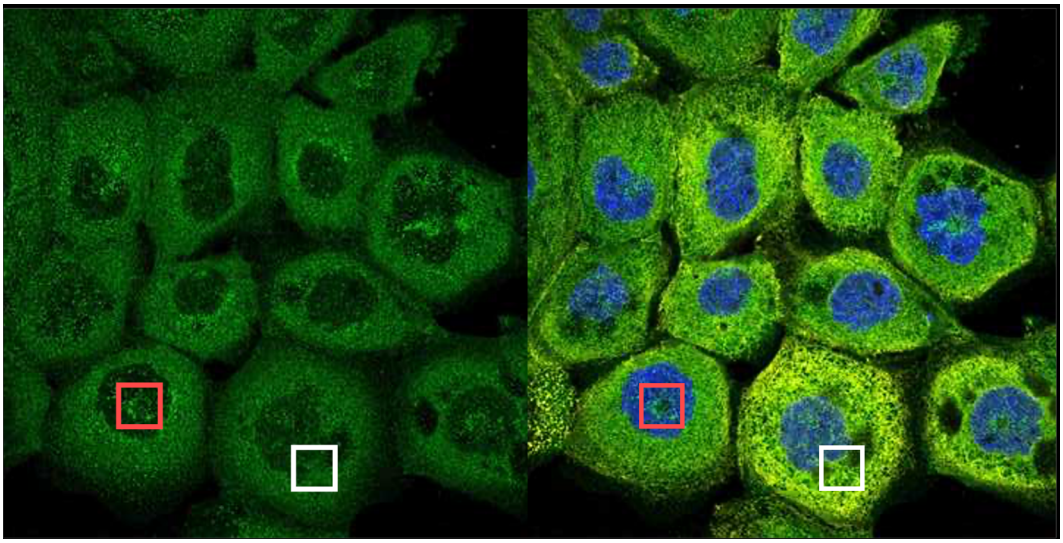

Figure S.3: Left: Images of ADP ribosylation factor 1 (green) from Human Protein Atlas [47]. Right: Left image with nucleus channel (blue) and ER channel (channel) visible. Although the HPA localization label is "Cytosol, Membrane," protein signal is clearly present in the nucleus (red box) and ER (white box).

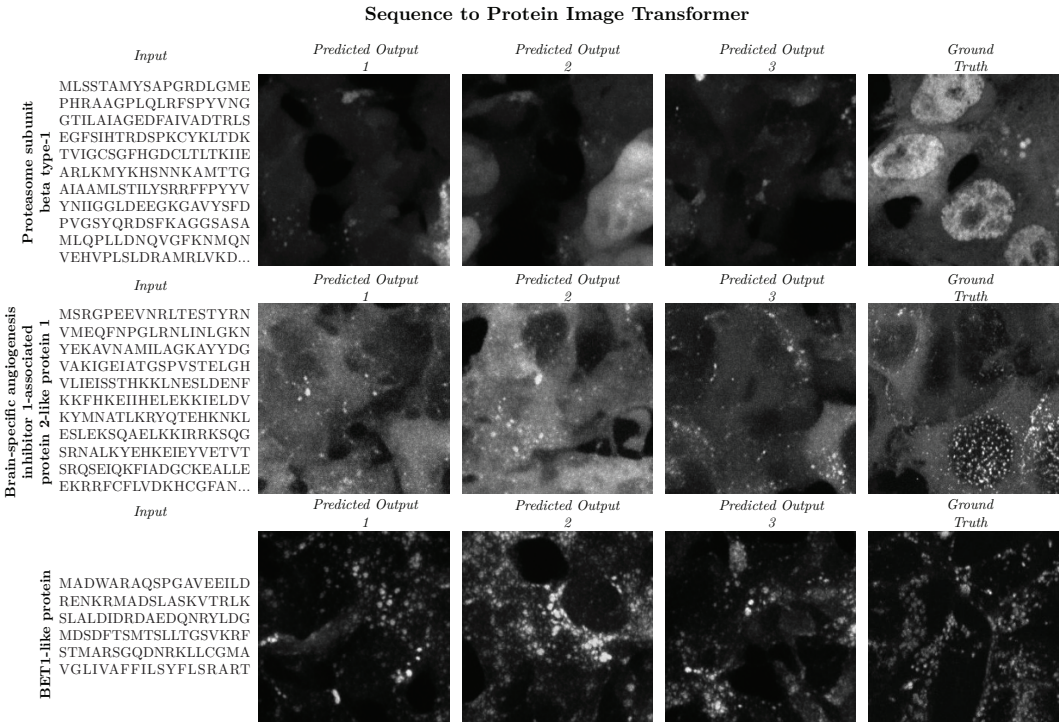

Figure S.4: DALL-E-like model with only amino acid sequence as the input. The sequence (left column) is used as input. The middle 3 columns show separate predicted images from random initialization. The true protein image is shown in the right column.

S.3 Dataset

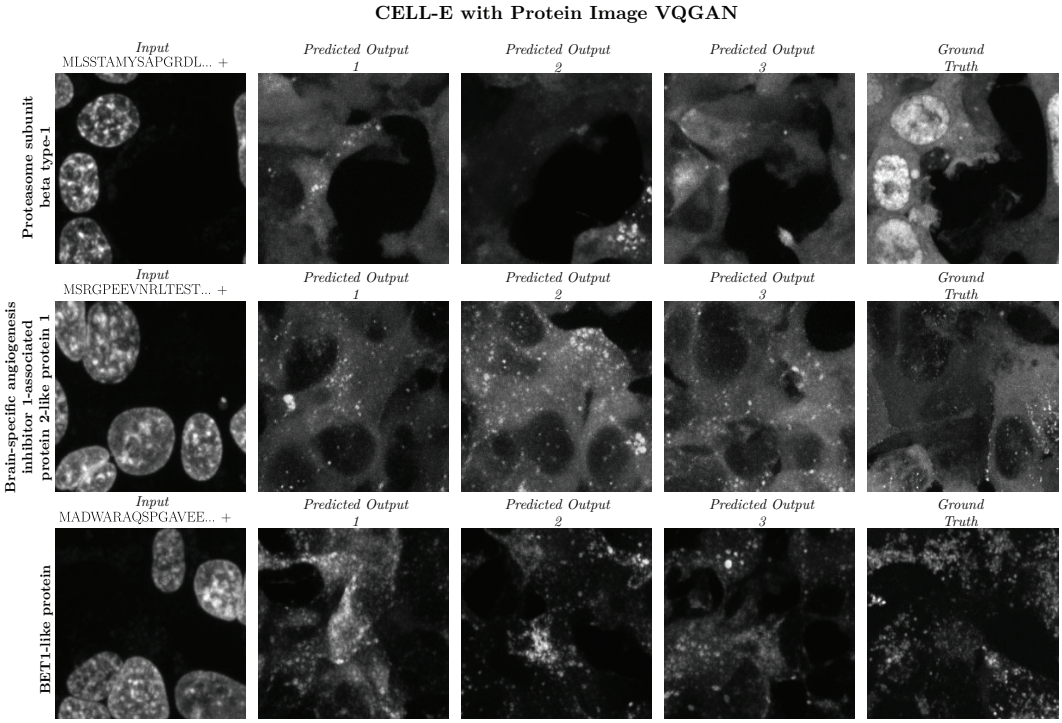

Figure S.5: CELL-E model with Protein Threshold VQGAN replaced with Protein Image VQGAN

Each protein entry in OpenCell is accompanied by multiple high-resolution 3D confocal images containing multiple cells [49]. Having multiple live cells enables the potential for protein distribution to be captured at several time points within a cell's lifetime. To reduce computational cost for out demonstration, we converted a 3D z-stack into a 2D maximum intensity projection [69], which still clearly depicts most subcellular structures and allow subtle subcellular protein localization differences to be distinguished from the OpenCell images [49].

The OpenCell dataset was selected because the split-fluorescent protein fusion system allows for tagging endogenous genomic proteins, maintaining local genomic context, and the preservation of native expression regulation [49]. This last point is specifically important when compared to the previously mentioned HPA, which contains $\sim 10\times$ more proteins and images. ICC-IF, which is the technique used for obtaining HPA images, requires several rounds of fixation and washing [70]. This means the proteins are not observed in a live cell, are subject to signal loss, artifacts, and/or relocalization events, and therefore does not represent the true nature of protein expression and distribution within a cell [71].

Training and validation sets were generated by randomly splitting the OpenCell dataset by protein 80%-20% training-validation For every stage of training, models were blind to sequences, nuclei, and protein images contained within the validation set. We utilize data augmentation techniques such as random horizontal and vertical flips on images during training.

S.4 Methods

S.4.1 Training

We utilized 4×NVIDIA RTX 3090 TURBO 24G GPUs for this study. 2 GPUs were utilized for training VQGANS via distributed training. Only a single GPU is ever used to train CELL-E models.

Our computer also contained 2×Intel Xeon Silver and 8×32768mb 2933MHz DR×4 Registered ECC DDR4 RAM.

S.4.2 Nucleus Image Encoder

VQGAN code was obtained from Esser et al. [50], noa [72], which was available via MIT license (Copyright (c) 2020 Patrick Esser and Robin Rombach and Björn Ommer).

The model was trained on random 256 $\times$ 256 crops of 512 $\times$ 512 nuclei images. Adam Optimizer was used with learning rate set to 4.5×10^{-6}. The model was initially trained solely using mean-squared error reconstruction loss. After 50,000 steps, $\sim$ 7 epochs, the discriminator loss term was introduced. This terms helps with reducing the blurriness typically associated with VAEs. 512 discrete image codes were learned. Training occurred until the model reached convergence (at 344 epochs).

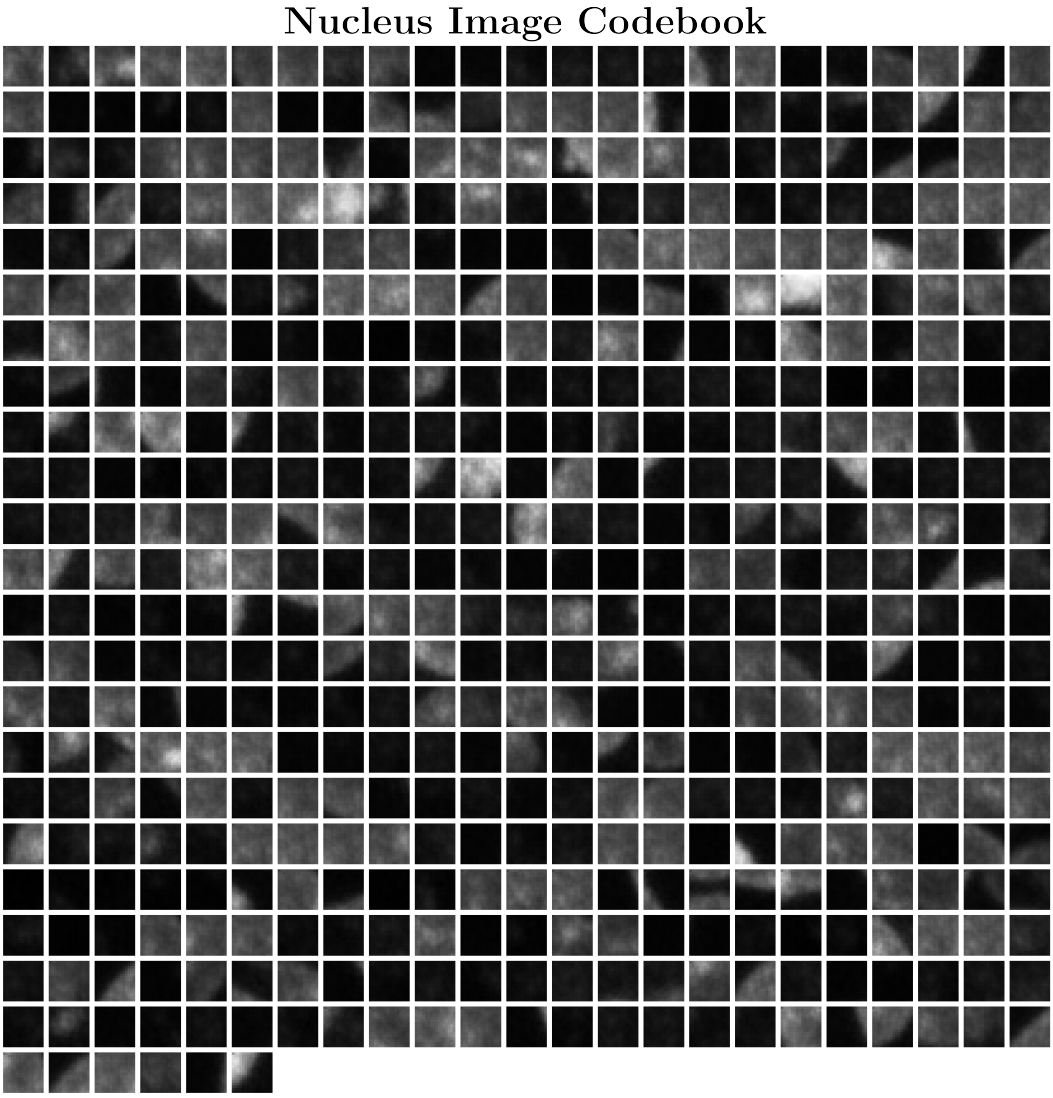

Figure S.6: 512 image patches extracted from the nucleus reference VQGAN

S.4.3 Protein Threshold Image Encoder

The model was trained on random 256 $\times$ 256 crops of 512 $\times$ 512 nuclei images. Adam Optimizer was used with learning rate set to 4.5×10^{-6}. The model was initially trained solely using mean-squared error reconstruction loss. After 50,000 steps, $\sim$ 7 epochs, the discriminator loss term was introduced. This terms helps with reducing the blurriness typically associated with VAEs. 512

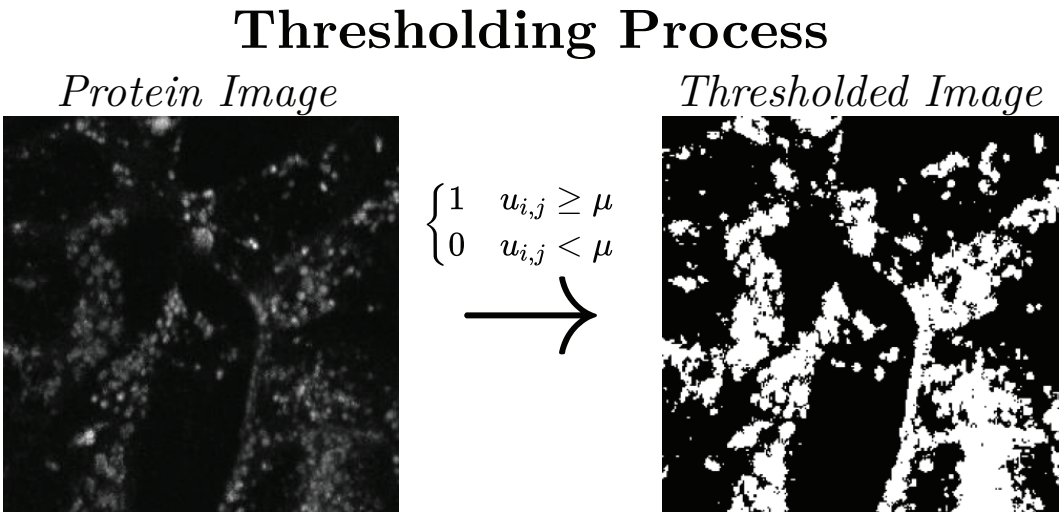

Figure S.7: Example of thresholding process to convert protein image (left) to thresholded ground truth image (right) for CELL-E model

discrete image codes were learned. Training occurred until the model reached convergence (at 371 epochs).

S.4.4 CELL-E Transformer

Our model is a heavily modified version of an open source text-to-image transformer [73], available via the MIT license (Copyright (c) 2021 Phil Wang). Our code is available at `https://github.com/BoHuangLab/Protein-Localization-Transformer` via the MIT license (Copyright (c) 2022 Emaad Khwaja, Yun Song, Bo Huang).

Protein sequences were converted to indices via the selected language tokenizers. Unless otherwise stated, all results in this work utilized the IUPAC tokens and BERT language embeddings. CELL-E uses encodings from BERT, There are 30 possible codebook values for amino acids within this model, with 25 corresponding to amino acids and 5 corresponding to special tokens (i.e. padding). Protein sequence length was limited to 1000 amino acids, which is longer than 96% of sequences within the dataset. For protein sequences shorter than 1000 amino acids, an end token (if utilized by the language model) was appended, followed by padding tokens. For protein sequences longer than 1000 amino acids, we randomly cropped a 1000 length subsection. If the right end of the crop ended before the true end of the protein sequence, no end token was applied. A start token is then prepended to all 1000 length sequence. The BERT model used represents input embeddings as vectors with dimension $n\times$ 768, where n is the number of amino acids. The sequence embedding for the BERT based models therefore had embedding vector sizes of 1001×768. Input amino acids were tokenized and their embeddings were retrieved from the language models. This input embedding is fixed. We also explored other embeddings (UniRep, ESM1b, One-hot encoding, and chemical descriptors) in Section S.2.

CELL-E was trained with an Adam optimizer with learning rate set to 3×10^{-4}.

The images were passed through the encoders of their respective VQGANs to obtain codebook tokens, and the final protein threshold image token is removed. We utilized data augmentation techniques including random cropping and random flips, just as was performed when training the VQGAN models. Within the CELL-E transformer, image token embeddings were cast into the same dimensionality as the language model embedding to in order to maintain the larger protein context information, however the embeddings corresponding to the image tokens within this dimension are learned. This ultimately creates a full sequence embeddings of 1512 $\times$768 (1001 $\times$ 768 for text, 256 $\times$ 768 for nucleus images and 255 $\times$ 768 for protein images) (Fig. 3).

A rotary positional embedding [74] is then applied to the input embeddings.

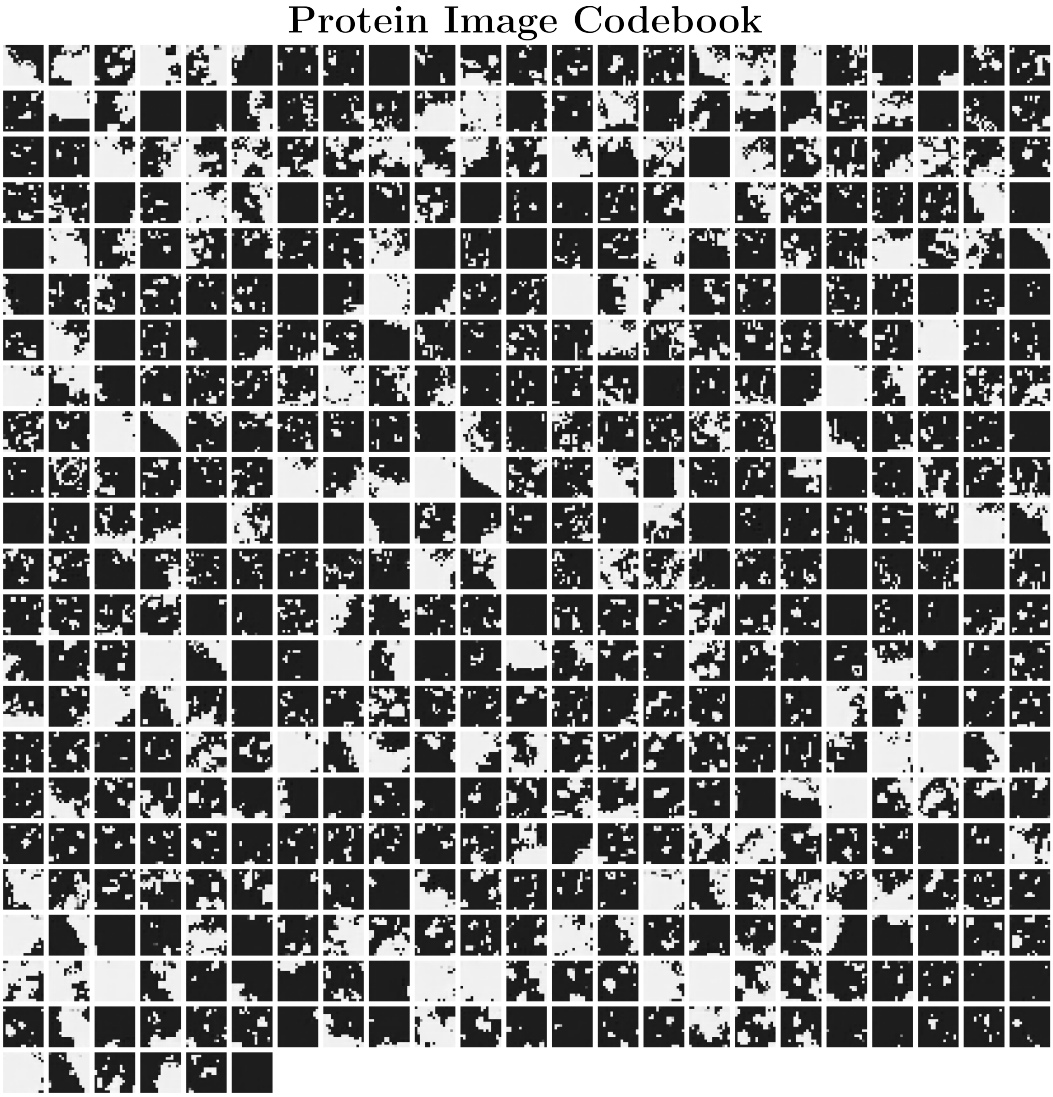

Figure S.8: 512 image patches extracted from the protein threshold VQGAN

We noted improved performance by shifting embeddings over by 1 (time-shifting [75]) in the feature dimension, but only for image tokens. Image token embeddings were shifted one position from the top and one position from the left.

A full attention scheme [76] is used where future tokens are masked in order to retain full sequence and image context. The output of the attention layers is passed through a block consisting of a linear and softmax layer to produce logits for predicted tokens at each position. Selective masking is applied so the model is unable to select anything but amino acid tokens for amino acid positions, nucleus image tokens for the nucleus image positions, and protein image tokens for the protein image positions.

Cross-entropy loss is used to measure the model's ability to reconstruct the original input vector (without the prepended start token and including the removed final protein threshold image token). The cross entropy is initially scaled by the length of the input, but further weighting is placed to emphasize the output protein image threshold tokens. We used weightings of 1/9 for the amino acid tokens, 1/9 for nucleus image tokens, and 8/9 for the protein threshold image tokens.

The main CELL-E model had a depth of 32, indicating 32 consecutive attention and feed forward blocks, and 16 attention heads with dimension = 64. We used attention and feed forward attention

dropout both = .1 during training. The language embedding was fixed. Model convergence occurred at 130 epochs and these weights were used for study.

S.5 Results

To assess performance, we generated a single prediction per image found in the OpenCell set. Each image was randomly cropped and flipped similar to training, but cropped regions and flips were maintained between models.

S.5.1 Image Accuracy

Nucleus Proportion Accuracy To calculate the proportion of intensity in the nucleus, we first create a mask (Fig. S.9 of the nucleus channel using Cellpose [77]. We take a sum over the predicted 2D PDF pixels found within the nucleus mask, and divide this by the sum of pixels across the image.

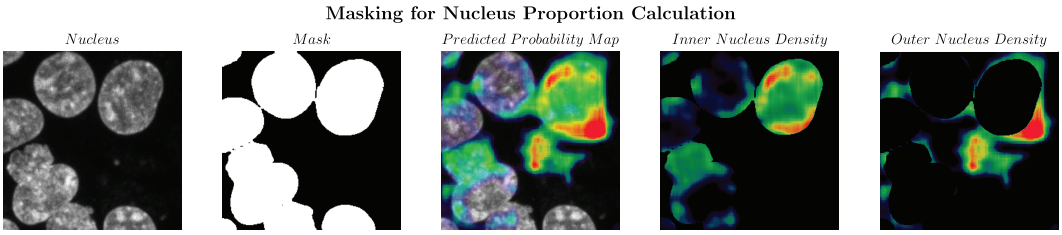

Figure S.9: Masking procedure depicted.

For the the ground truth, we use a similar masking calculation, but consider the values of the ground truth protein image. These values are subtracted to calculate a total root-mean square error (RMSE). Since the maximum possible value is 1 and minimum possible value is 0, we report accuracy as 1 - RMSE.

Predicted Threshold Pixel Accuracy We simple calculate a pixel-wise RMSE between the predicted protein threshold image of CELL-E and the ground truth protein threshold image.

Predicted 2D PDF Pixel Accuracy This metric is similar to Predicted Threshold Pixel Accuracy, except we evaluate the difference using the predicted 2D PDF, rather than the predicted protein threshold image. We expect this number to be less accurate as tokens with less confidence will reduce the pixel value, while all values in the protein threshold image are 0 or 1.

SSIM SSIM is a measure of local perceptual similarity between images. It considers neighboring pixels to evaluate loss contextually by incorporating luminance and contrast information. SSIM values range between 0, indicating no similarity, and 1, indicating maximum similarity.

IS Inception score is often used to evaluate the image outputs of GANs as a measure of "realisticness." It rewards image variety and similarity to real-life data. Performance evaluation is based on the magnitude of the IS score.

FID FID is another popular metric for evaluating the quality of images from generative models. It compaers the distributions between generated and ground truth images as the squared Wasserstein metric between two multidimensional Gaussian distributions. For this study FID was scored against the training or validation sets when applicable, rather than the entire OpenCell dataset.

S.6 Discussion

S.6.1 Visualizing Attention

To obtain Fig. 5, we first split the 16x16 generated threshold image patches into 2 groups, one where protein is primarily determined to be present $\bar{u}_{i,j} > .75$ and another where background tokens are

primarily selected $\bar{u}_{i,j} < .25$. For each respective group, we calculate a median of the attention matrices and used attention rollout [78] to recursively multiply across 32 layers. The final layers of both groups are then compared. We initially look at tokens with higher weightings for the present protein group, and discard the rest.

We show the entire image generation process, with frames corresponding to time-steps, in the attached video file: `DNAtopoisomerase1.mp4`.

S.6.2 Novel NLS Discovery

To predict the effect of localization using the intended sequences, we follow a procedure similar to that of Mo et al. [65] used when creating genomic edits in a wet lab setting, however we do not append an eGFP sequence. In short, for each sequence, the first amino acid is the starting "M" - methionine from the original TOP1 sequence. Next, we include only the potential NLS being considered, marked above the images in Fig. S.10. The farthest amino acid in the sequence is the final "E", which corresponds to position 254. All amino acids from position 255 onward are then included.

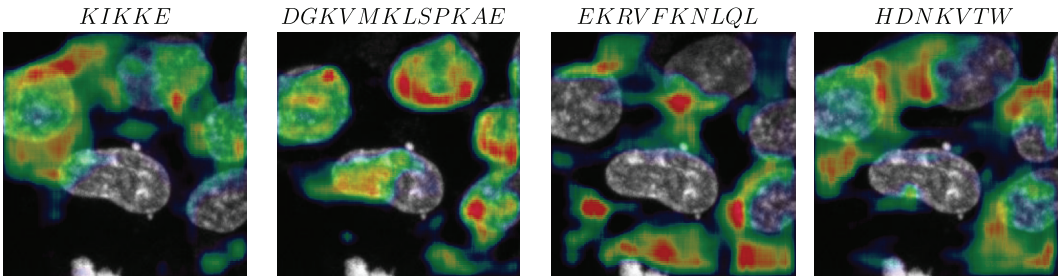

Figure S.10: Predicted probability distribution of TOP1 subsequences. Note the high degree of nuclear localization from DGKVMKLSPKAE, which is highly attended to in the sequence.

S.6.3 Reproducing Experimental Results *In Silico*

We reproduce results from Mo et al. [65] in Fig. S.11. This was done by using the reference sequences and a randomly selected nucleus image from the validation set as input.

pT104/B1, a negative control which only contains eGFP. pTI-5 contains eGFP + TOP1. The authors note the presence of nuclolus localization, and we similarly observe circular cluster in our prediction. pTI-7 deletes amino acids 148-199 from the TOP1 portion. It is similarly localized as pTI-5, but does not appear to contain any nucleolar emphasis. pTI-8 deletes 68-199. While we do not observe similar results, we do note a larger degree of protein predicted outside the top left nucleus, and signal completely omitted from the bottom right cell. pTI-10 deletes amino acids 1-148. We observe similar nucleolar patterning to pTI-5, although less pronounced.

S.6.4 Extra-Domain Prediction

We visualize the encodings of the inputs of Extra-Domain Prediction. NLS sequeunces were appended to the N' terminus of the GFP. The random sequence was generated by randomly selected proteins of length equal to the Nucleoplasmin NLS sequence. We observe strong predictive knowledge from the presence of the NLS sequences.

To further understand this prediction, we visualized the CELL-E latent space using a UMAP (Fig. S.13). We plotted encodings for sequence, nucleus, image sets on a dimension reduced UMAP for proteins which selectively isolate to particular subcellular organelles. The encodings were obtained from the output of the final feed-forward layer of the transformer. These labels were published in the OpenCell paper [49]. We note regions clustered together which correspond to extra-nuclear regions (golgi, cytoplasmic, vesicles, endoplasmic reticulum), and nuclear regions (nuclear membrane, nucleoplasm, nucleolus, and chromatin). We also observe continutiy between the nuclear membrane and golgi proteins, as well as nucleoplasmic and cytoplasmic proteins.

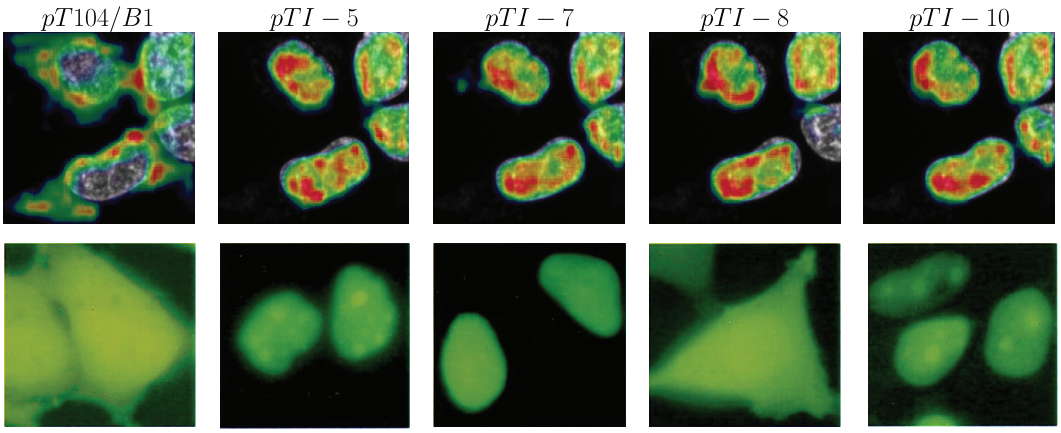

Figure S.11: Predicted localization of eGFP fusions from [65] and corresponding figures from the original paper.

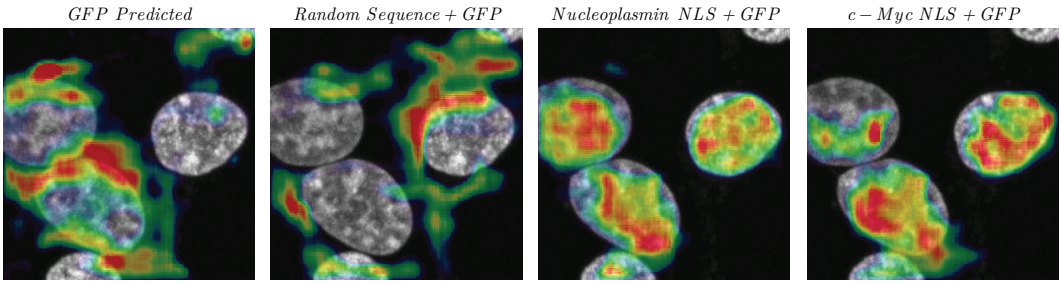

Figure S.12: Predicted localization of GFP and modified-GFP sequences..

We note that the base GFP and Random Seuqence appended to GFP are founded well within the cytoplasmic neighborhood. When the nucleoplasm and c-Myc NLSs are appended, the encodings can be found amongst nuclear membrane and nucleoplasm proteins. Note that c-Myc + GFP is closer to the edge of the cytoplasmic neighborhood, and we correspondingly observe more signal predicted outside of the nucleus.

Similar to in Section S.6.1, we show image generation process for a base GFP (`GreenFluorescentProtein.mp4`) and Nucleoplasmin NLS + GFP (`NucleoplasminNLS+ GreenFluorescentProtein.mp4`).

S.6.5 Ablation

In order to gauge the importance of each component of the model, we performed ablation by training several versions of the model with the same initialization. We specifically chose to look at performance in nuclear proportion accuracy over the validation set (Fig. S.14).

On the top row, performance from the VQGAN is used as a reference of possible top performance, just as in Table 1. The second row depicts our main CELL-E (with depth = 32 and a fixed language embedding) used for this study. We found the performance in both cross-entropy and nuclear proportion accuracy increased with model depth when compared to similar models of smaller depth (third, fourth, and fifth rows).

To understand the effect of using fixed language models, we trained 2 versions of CELL-E of same depth. The first (second row, light blue) had a fixed language embedding, while the second (seventh row, red) was free to change during training. We also introduced a model with a randomly initialized language embedding (sixth row, light green). While we note fairly high performance from the unfixed models on the training data, they performed quite poorly on the validation, indicating severe overfitting. This is a result of the comparatively small number of proteins represented within the OpenCell dataset when compared to the large pFam database used to train BERT.

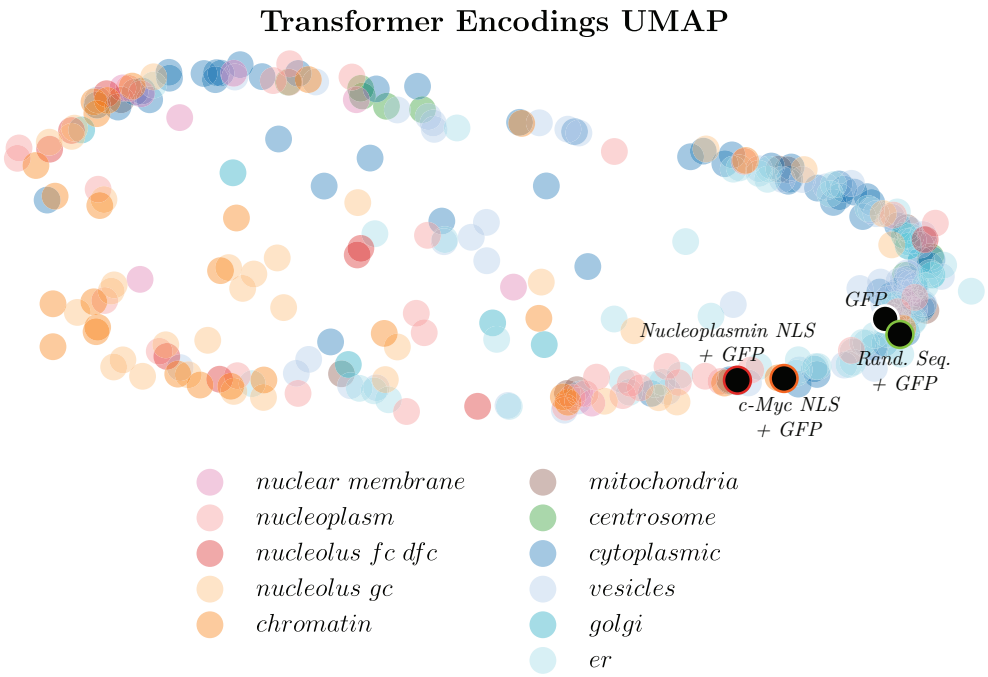

Figure S.13: UMAP of encodings. We observe a largely nuclear protein "neighborhood" (left) and a cytoplasmic neighborhood (right). Prepending the NLS to GFP causes a shift towards the nuclear protein neighborhood.

We also trained a versions of the model which did not use a nucleus input (second to last row, pink) like DALL-E, and a model that only used a nucleus input and no sequence (last row, purple), although a start token was still prepended.

Overall, we observe a distributional shift to the right, indicating more accurate predictions, as the depth of the trnasformer is sclaed. Full results for both training and validation sets can be seen in Table S.2. We also evaluated the performance of CELL-E using different protein embedding spaces. These were configured such that they were either at the same depth as the BERT model, or the depth was scaled as deep as possible such that the GPU memory was saturated during training. All model were trained until convergence on the validation set.

Alongside language embeddings, we also used one-hot and amino acid chemical descriptors as embedding features. The amino acid chemical descriptors come from Osorio et al. [79], which contains amino numerical descriptions of amino acid properties from various literature sources [80, 81, 82, 83, 84, 85, 86, 87, 88, 89]. Using one-hot and amino acid descriptors allowed us to scale to deeper model depths, but we did not see much improvement from doing so using these embeddings. These encodings likely do not contain sufficient information complexity about local environments that BERT, UniRep, and ESM1b contain. While we do not see a consistent top performer on the training data, BERT-based models generally performed the best across all metrics on the validation set, indicating a higher degree of generalizability.

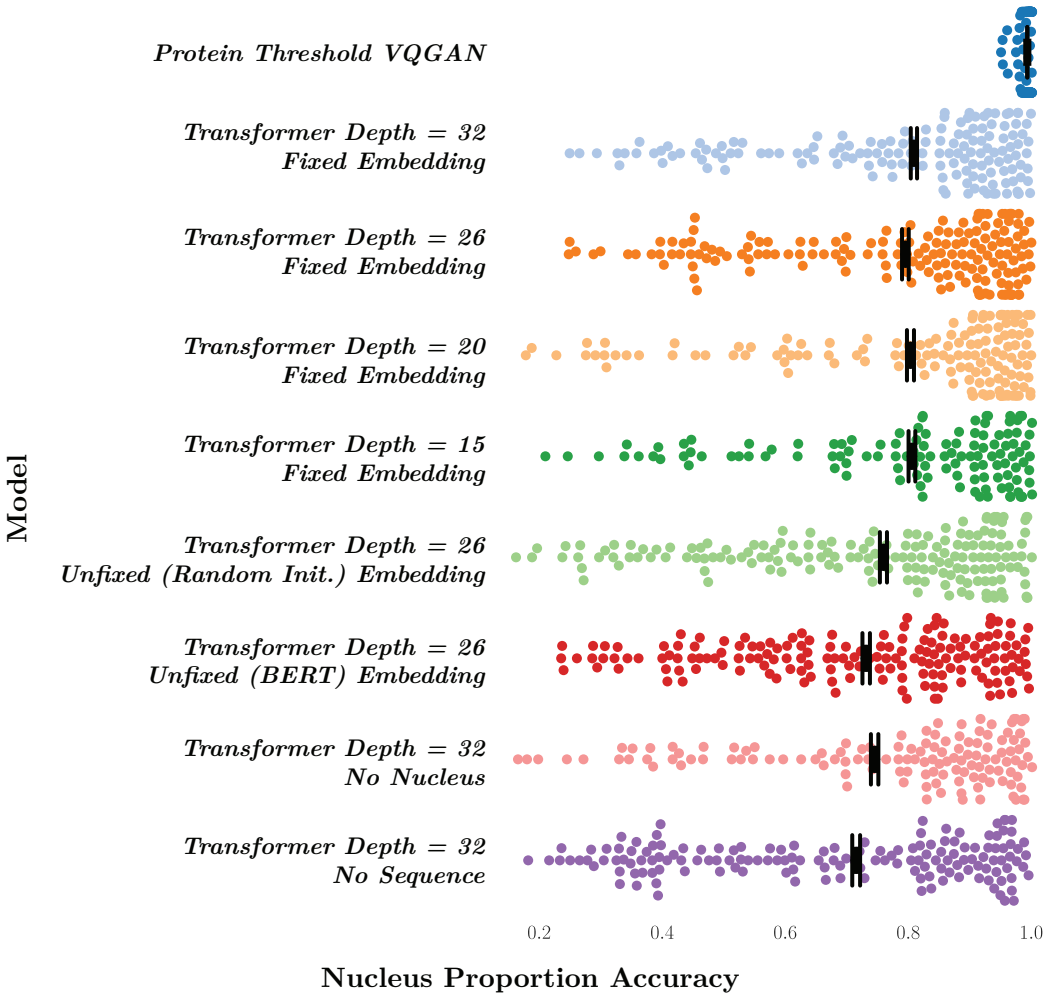

Figure S.14: Ablation Plot. "Fixed" and "Unfixed" embedding refer exclusively to the amino acid embeddings. Image embeddings are always unfixed. Mean values and standard deviation are marked in black. ~ 150 points are randomly selected for display out of 1303 total predictions per model.

Table S.2: Full Results Table

Train											
Language Emb.	Model Depth	Emb. Dim.	Fixed Lang. Emb?	Epoch	Nucleus Proportion Accuracy	Predicted Threshold Pixel Acc.	Predicted 2D PDF Pixel Acc.	SSIM	IS	FID	Nucleus Loc. Pred. Accy.
N/A (VQGAN)				371	.9927± .0084	.8749± .0817		.5523± .2528	4.174± .1716	14.818	.9933± .0810
BERT	32	768	✓	130	.9396± .0535	.7678± .0647	.6758± .1047	.3154± .2100	2.7684± .0681	107.4147	.8946± .3071
BERT	26	768	✓	136	.9475± .0465	.7551± .0726	**.6849± .1051**	.3146± .2195	2.9021± .0067	81.0061	.9059± .2920
BERT	26	768	✗	49	.9260± .0656	.7568± .0664	.6131± .0928	.2266± .1485	2.3797± .0742	140.339	.8730± .3330
BERT	20	768	✓	63	.9433± .0501	.7647± .0719	.6815± .1081	.3138± .2234	2.8539± .0474	98.5444	.9004± .2994
BERT	15	768	✓	84	.9456± .0469	.7485± .0704	.6730± .1041	.2955± .2122	2.687± .0742	112.4738	**.9095± .2869**
BERT (No Nuc.)	32	768	✓	91	.9115± .0762	.7609± .0762	.5743± .0887	.2092± .1609	2.2793± .3975	137.8209	.8516± .3554
ESM1b	20	1280	✓	54	.9403± .0523	.7610± .0664	.6633± .1027	.2855± .2057	2.6059± .1153	111.8584	.8902± .3127
UniRep	15	1900	✓	8	.6591± .1547	.7420± .0632	.6011± .0803	.2108± .1410	2.1317± .0432	193.1455	.8651± .3417
AA Descriptors	58	66	✓	60	.9439± .0528	.7674± .0690	.6815± .1083	.3220± .2261	2.8121± .0529	94.2488	.9044± .2941
AA Descriptors	32	66	✓	66	.9472± .0499	.7622 ±.0792	.6825± .1114	**.3293± .2420**	**3.144± .1315**	**65.5666**	.9082± .2888
One-Hot	59	25	✓	100	.9469± .0487	.7524± .0681	.6791± .1077	.3141± .2212	2.7269± .0917	102.8544	.9068± .2907
One-Hot	32	25	✓	70	**.9478± .0476**	.7475± .0705	.6769± .1073	0.3041± .2287	2.7469± .0905	90.0424	.9057± .2922
Random Initialization	26	768	✗	71	.9348± .0567	**.7693± .0668**	.6606± .1093	.2943± .2144	2.7503± .0852	110.9421	,8885± .3147
No Sequence	32	768		51	.9235± .0678	.7578± .0749	.6113± .0930	.2196± .1517	2.2836± .0658	159.0451	.8628± .3440

					Validation						
Language Emb.	Model Depth	Emb. Dim.	Fixed Lang. Emb?	Epoch	Nucleus Proportion Accuracy	Predicted Threshold Pixel Acc.	Predicted 2D PDF Pixel Acc.	SSIM	IS	FID	Nucleus Loc. Pred. Accy.
N/A (VQGAN)				371	0.9921± .0091	.8756± .0824		.5567± .2540	3.8718± .1662	25.9567	.9923± .0872
BERT	32	768	✓	130	**.8078± .1837**	.7653± .0520	.6342± .0964	.2536± .1629	2.1300± .0704	155.7741	.7155± .4511
BERT	26	768	✓	136	.7943± .1992	.7574± .0607	.6377± .1026	.2446± .1689	2.2708± .1303	136.294	.6979± .4594
BERT	26	768	✗	49	.7300± .2229	.7505± .0648	.6092± .0918	.2148± .1436	2.2962± .1740	155.4038	.5905± .4917
BERT	20	768	✓	63	.8024± .2071	**.7739± .0613**	**.6388± .1088**	.2547± .1821	2.2641± .1379	156.3681	.7017± .4575
BERT	15	768	✓	84	.8044± .1927	.7422± .0554	.6267± .0914	.2226± .1546	2.0341± .0848	170.5773	**.7247± .4467**
BERT (No Nuc.)	32	768	✓	91	.7742± .2239	.7733± .0660	.5688± .0739	.1956± .1311	**3.3204± .1737**	**45.6887**	.6288± .4833
ESM1b	20	1280	✓	54	.8044± .1851	.7465± .0602	.6235± .0916	.2207± .1604	2.1925± .0868	146.5215	.7224± .4478
UniRep	15	1900	✓	8	.7474± .2063	.7393± .0616	.5982± .0789	.2009± .1332	2.1287± .1231	206.3325	.6173± .4862
AA Descriptors	58	66	✓	60	.7854± .2234	.7642± .0602	.6340± .1020	.2487± .1736	2.2561± .1030	144.271	.7002± .4582
AA Descriptors	32	66	✓	66	.7688± .2435	.7676± .0761	.6350± .1163	**.2662± .2016**	2.8155± .1382	88.4847	.6817± .4658
One-Hot	59	25	✓	100	.7688± .1989	.7347± .0535	.6207± .0904	.2209± .1484	2.1214± .1182	159.35	.6457± .4783
One-Hot	32	25	✓	70	.7714± .2094	.7383± .0599	.6255± .1032	.2252± .1703	2.2802± .1170	129.3922	.6419± .4795
Random Initialization	26	768	✗	71	.7587 .2169	.7605± .0517	.6204± .0948	.2287± .1543	2.083± .1373	167.6092	.6250± .4841
No Sequence	32	768		51	.7140± .2359	.7528± .0733	.6054± .0889	.2032± .1387	2.2473 .1400	169.204	.5721± .4948

References

[1] Taishan Hu, Nilesh Chitnis, Dimitri Monos, and Anh Dinh. Next-generation sequencing technologies: An overview. *Human Immunology*, 82(11):801–811, November 2021. ISSN 0198-8859. doi: 10.1016/j.humimm.2021.02.012. URL https://www.sciencedirect.com/science/article/pii/S0198885921000628.

[2] Carlos-Andres Palma, Marco Cecchini, and Paolo Samorì. Predicting self-assembly: from empirism to determinism. *Chemical Society Reviews*, 41(10):3713–3730, April 2012. ISSN 1460-4744. doi: 10.1039/C2CS15302E. URL https://pubs.rsc.org/en/content/articlelanding/2012/cs/c2cs15302e. Publisher: The Royal Society of Chemistry.

[3] Gerald Karp. *Cell and Molecular Biology: Concepts and Experiments*. Wiley, Hoboken, NJ, 6th edition edition, October 2009. ISBN 978-0-470-48337-4.

[4] Mien-Chie Hung and Wolfgang Link. Protein localization in disease and therapy. *Journal of Cell Science*, 124(Pt 20):3381–3392, October 2011. ISSN 1477-9137. doi: 10.1242/jcs.089110.

[5] P. L. Paine, L. C. Moore, and S. B. Horowitz. Nuclear envelope permeability. *Nature*, 254(5496):109–114, March 1975. ISSN 0028-0836. doi: 10.1038/254109a0.

[6] Maxence V. Nachury, E. Scott Seeley, and Hua Jin. Trafficking to the ciliary membrane: how to get across the periciliary diffusion barrier? *Annual Review of Cell and Developmental Biology*, 26:59–87, 2010. ISSN 1530-8995. doi: 10.1146/annurev.cellbio.042308.113337.

[7] Elizabeth J. Tran, Megan C. King, and Anita H. Corbett. Macromolecular transport between the nucleus and the cytoplasm: Advances in mechanism and emerging links to disease. *Biochimica Et Biophysica Acta*, 1843(11):2784–2795, November 2014. ISSN 0006-3002. doi: 10.1016/j.bbamcr.2014.08.003.

[8] Agnieszka Chacinska, Carla M. Koehler, Dusanka Milenkovic, Trevor Lithgow, and Nikolaus Pfanner. Importing Mitochondrial Proteins: Machineries and Mechanisms. *Cell*, 138(4):628–644, August 2009. ISSN 0092-8674. doi: 10.1016/j.cell.2009.08.005. URL https://www.sciencedirect.com/science/article/pii/S0092867409009672.

[9] Kenichiro Imai and Kenta Nakai. Prediction of subcellular locations of proteins: where to proceed? *Proteomics*, 10(22):3970–3983, November 2010. ISSN 1615-9861. doi: 10.1002/pmic.201000274.

[10] Hazem Radwan Ahmed and Janice Glasgow. A Novel Particle Swarm-Based Approach for 3D Motif Matching and Protein Structure Classification. In Marina Sokolova and Peter van Beek, editors, *Advances in Artificial Intelligence*, Lecture Notes in Computer Science, pages 1–12, Cham, 2014. Springer International Publishing. ISBN 978-3-319-06483-3. doi: 10.1007/978-3-319-06483-3_1.

[11] Jennifer L. Gardy and Fiona S. L. Brinkman. Methods for predicting bacterial protein subcellular localization. *Nature Reviews Microbiology*, 4(10):741–751, October 2006. ISSN 1740-1534. doi: 10.1038/nrmicro1494. URL https://www.nature.com/articles/nrmicro1494. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 10 Primary_atype: Reviews Publisher: Nature Publishing Group.

[12] Juane Lu, Tao Wu, Biao Zhang, Suke Liu, Wenjun Song, Jianjun Qiao, and Haihua Ruan. Types of nuclear localization signals and mechanisms of protein import into the nucleus. *Cell Communication and Signaling*, 19(1):60, May 2021. ISSN 1478-811X. doi: 10.1186/s12964-021-00741-y. URL https://doi.org/10.1186/s12964-021-00741-y.

[13] UniProt: the universal protein knowledgebase. *Nucleic Acids Research*, 45(Database issue):D158–D169, January 2017. ISSN 0305-1048. doi: 10.1093/nar/gkw1099. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5210571/.

[14] Sumit Kumar Jha, Arvind Ramanathan, Rickard Ewetz, Alvaro Velasquez, and Susmit Jha. Protein Folding Neural Networks Are Not Robust. *arXiv:2109.04460 [cs, q-bio]*, September 2021. URL http://arxiv.org/abs/2109.04460. arXiv: 2109.04460.

[15] OpenCell, . URL https://opencell.czbiohub.org/target/CID000889.

[16] Jesse Vig, Ali Madani, Lav R. Varshney, Caiming Xiong, Richard Socher, and Nazneen Fatema Rajani. BERTology Meets Biology: Interpreting Attention in Protein Language Models. June 2020. doi: 10.48550/arXiv.2006.15222. URL https://arxiv.org/abs/2006.15222v3.

[17] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models are Few-Shot Learners. *arXiv:2005.14165 [cs]*, July 2020. URL http://arxiv.org/abs/2005.14165. arXiv: 2005.14165.

[18] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M. Rush. Character-Aware Neural Language Models. In *Thirtieth AAAI Conference on Artificial Intelligence*, March 2016. URL https://www.aaai.org/ocs/index.php/AAAI/AAAI16/paper/view/12489.

[19] Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. Big Bird: Transformers for Longer Sequences. July 2020. doi: 10.48550/arXiv.2007.14062. URL https://arxiv.org/abs/2007.14062v2.

[20] Alexander Rives, Joshua Meier, Tom Sercu, Siddharth Goyal, Zeming Lin, Jason Liu, Demi Guo, Myle Ott, C. Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *Proceedings of the National Academy of Sciences*, 118(15):e2016239118, April 2021. doi: 10.1073/pnas.2016239118. URL https://www.pnas.org/doi/10.1073/pnas.2016239118. Publisher: Proceedings of the National Academy of Sciences.

[21] Ahmed Elnaggar, Michael Heinzinger, Christian Dallago, Ghalia Rehawi, Yu Wang, Llion Jones, Tom Gibbs, Tamas Feher, Christoph Angerer, Martin Steinegger, Debsindhu Bhowmik, and Burkhard Rost. ProtTrans: Towards Cracking the Language of Lifes Code Through Self-Supervised Deep Learning and High Performance Computing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2021. ISSN 0162-8828, 2160-9292, 1939-3539. doi: 10.1109/TPAMI.2021.3095381. URL https://ieeexplore.ieee.org/document/9477085/.

[22] Yanbin Wang, Zhu-Hong You, Shan Yang, Xiao Li, Tong-Hai Jiang, and Xi Zhou. A High Efficient Biological Language Model for Predicting ProteinProtein Interactions. *Cells*, 8(2):122, February 2019. ISSN 2073-4409. doi: 10.3390/cells8020122. URL https://www.mdpi.com/2073-4409/8/2/122. Number: 2 Publisher: Multidisciplinary Digital Publishing Institute.

[23] Martin Steinegger, Milot Mirdita, and Johannes Söding. Protein-level assembly increases protein sequence recovery from metagenomic samples manyfold. *Nature Methods*, 16(7):603–606, July 2019. ISSN 1548-7105. doi: 10.1038/s41592-019-0437-4. URL https://doi.org/10.1038/s41592-019-0437-4.

[24] Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, January 2000. ISSN 0305-1048. doi: 10.1093/nar/28.1.235. URL https://doi.org/10.1093/nar/28.1.235.

[25] Baris E. Suzek, Yuqi Wang, Hongzhan Huang, Peter B. McGarvey, and Cathy H. Wu. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics*, 31 (6):926–932, March 2015. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu739. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4375400/.

[26] Jaina Mistry, Sara Chuguransky, Lowri Williams, Matloob Qureshi, GustavoăA Salazar, Erik L L Sonnhammer, Silvio C E Tosatto, Lisanna Paladin, Shriya Raj, Lorna J Richardson, Robert D Finn, and Alex Bateman. Pfam: The protein families database in 2021. *Nucleic Acids Research*, 49(D1):D412–D419, January 2021. ISSN 0305-1048. doi: 10.1093/nar/gkaa913. URL https://doi.org/10.1093/nar/gkaa913.

[27] Roshan Rao, Nicholas Bhattacharya, Neil Thomas, Yan Duan, Xi Chen, John Canny, Pieter Abbeel, and Yun S. Song. Evaluating Protein Transfer Learning with TAPE. June 2019. doi: 10.48550/arXiv.1906.08230. URL https://arxiv.org/abs/1906.08230v1.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[29] Ethan C. Alley, Grigory Khimulya, Surojit Biswas, Mohammed AlQuraishi, and George M. Church. Unified rational protein engineering with sequence-based deep representation learning. *Nature Methods*, 16(12):1315–1322, December 2019. ISSN 1548-7105. doi: 10.1038/s41592-019-0598-1. URL https://www.nature.com/articles/s41592-019-0598-1. Number: 12 Publisher: Nature Publishing Group.

[30] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*, May 2019. URL http://arxiv.org/abs/1810.04805. arXiv: 1810.04805.

[31] Gaofeng Pan, Chao Sun, Zijun Liao, and Jijun Tang. Machine and Deep LearningDeep learning (DL) for Prediction of Subcellular Localization. In Daniela Cecconi, editor, *Proteomics Data Analysis*, Methods in Molecular Biology, pages 249–261. Springer US, New York, NY, 2021. ISBN 978-1-07-161641-3. doi: 10.1007/978-1-0716-1641-3_15. URL https://doi.org/10.1007/978-1-0716-1641-3_15.

[32] James Zou, Mikael Huss, Abubakar Abid, Pejman Mohammadi, Ali Torkamani, and Amalio Telenti. A primer on deep learning in genomics. *Nature Genetics*, 51(1):12–18, January 2019. ISSN 1546-1718. doi: 10.1038/s41588-018-0295-5. URL https://www.nature.com/articles/s41588-018-0295-5. Bandiera_abtest: a Cg_type: Nature Research Journals Number: 1 Primary_atype: Reviews Publisher: Nature Publishing Group Subject_term: Computational biology and bioinformatics;Genetics;Genome informatics Subject_term_id: computational-biology-and-bioinformatics;genetics;genome-informatics.

[33] Kyongsik Yun, Alexander Huyen, and Thomas Lu. Deep Neural Networks for Pattern Recognition. *arXiv:1809.09645 [cs]*, September 2018. URL http://arxiv.org/abs/1809.09645. arXiv: 1809.09645.

[34] Long Pang, Junjie Wang, Lingling Zhao, Chunyu Wang, and Hui Zhan. A Novel Protein Subcellular Localization Method With CNN-XGBoost Model for Alzheimer's Disease. *Frontiers in Genetics*, 9:751, 2019. ISSN 1664-8021. doi: 10.3389/fgene.2018.00751. URL https://www.frontiersin.org/article/10.3389/fgene.2018.00751.

[35] Wen-Yun Yang, Bao-Liang Lu, and Yang Yang. A Comparative Study on Feature Extraction from Protein Sequences for Subcellular Localization Prediction. In *2006 IEEE Symposium on Computational Intelligence and Bioinformatics and Computational Biology*, pages 1–8, Toronto, ON, Canada, September 2006. IEEE. ISBN 978-1-4244-0623-4 978-1-4244-0624-1. doi: 10.1109/CIBCB.2006.330991. URL http://ieeexplore.ieee.org/document/4133173/.

[36] K. M. Hager, B. Striepen, L. G. Tilney, and D. S. Roos. The nuclear envelope serves as an intermediary between the ER and Golgi complex in the intracellular parasite Toxoplasma gondii. *Journal of Cell Science*, 112 (Pt 16):2631–2638, August 1999. ISSN 0021-9533. doi: 10.1242/jcs.112.16.2631.

[37] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-Shot Text-to-Image Generation. *arXiv:2102.12092 [cs]*, February 2021. URL http://arxiv.org/abs/2102.12092. arXiv: 2102.12092.

[38] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv:1312.6114 [cs, stat]*, May 2014. URL http://arxiv.org/abs/1312.6114. arXiv: 1312.6114.

[39] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html.

[40] Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Generating Images from Captions with Attention. *arXiv:1511.02793 [cs]*, February 2016. URL http://arxiv.org/abs/1511.02793. arXiv: 1511.02793.

[41] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative Adversarial Text to Image Synthesis. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1060–1069, New York, New York, USA, June 2016. PMLR. URL https://proceedings.mlr.press/v48/reed16.html.

[42] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks. *arXiv:1711.10485 [cs]*, November 2017. URL http://arxiv.org/abs/1711.10485. arXiv: 1711.10485.

[43] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. CogView: Mastering Text-to-Image Generation via Transformers. *arXiv:2105.13290 [cs]*, November 2021. URL `http://arxiv.org/abs/2105.13290`. arXiv: 2105.13290.

[44] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-Driven Text-To-Image Synthesis via Adversarial Training. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12166–12174, Long Beach, CA, USA, June 2019. IEEE. ISBN 978-1-72813-293-8. doi: 10.1109/CVPR.2019.01245. URL `https://ieeexplore.ieee.org/document/8953291/`.

[45] Jing Yu Koh, Jason Baldridge, Honglak Lee, and Yinfei Yang. Text-to-Image Generation Grounded by Fine-Grained User Attention. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 237–246, Waikoloa, HI, USA, January 2021. IEEE. ISBN 978-1-66540-477-8. doi: 10.1109/WACV48630.2021.00028. URL `https://ieeexplore.ieee.org/document/9423212/`.

[46] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. page 26.

[47] Peter J. Thul and Cecilia Lindskog. The human protein atlas: A spatial map of the human proteome. *Protein Science: A Publication of the Protein Society*, 27(1):233–244, January 2018. ISSN 1469-896X. doi: 10.1002/pro.3307.

[48] Axel Tiessen, Paulino Pérez-Rodríguez, and Luis José Delaye-Arredondo. Mathematical modeling and comparison of protein size distribution in different plant, animal, fungal and microbial species reveals a negative correlation between protein size and protein number, thus providing insight into the evolution of proteomes. *BMC Research Notes*, 5(1):1–23, December 2012. ISSN 1756-0500. doi: 10.1186/1756-0500-5-85. URL `https://bmcresnotes.biomedcentral.com/articles/10.1186/1756-0500-5-85`. Number: 1 Publisher: BioMed Central.

[49] Nathan H. Cho, Keith C. Cheveralls, Andreas-David Brunner, Kibeom Kim, André C. Michaelis, Preethi Raghavan, Hirofumi Kobayashi, Laura Savy, Jason Y. Li, Hera Canaj, James Y. S. Kim, Edna M. Stewart, Christian Gnann, Frank McCarthy, Joana P. Cabrera, Rachel M. Brunetti, Bryant B. Chhun, Greg Dingle, Marco Y. Hein, Bo Huang, Shalin B. Mehta, Jonathan S. Weissman, Rafael Gómez-Sjöberg, Daniel N. Itzhak, Loic A. Royer, Matthias Mann, and Manuel D. Leonetti. OpenCell: proteome-scale endogenous tagging enables the cartography of human cellular organization. Technical report, March 2021. URL `https://www.biorxiv.org/content/10.1101/2021.03.29.437450v1`. Company: Cold Spring Harbor Laboratory Distributor: Cold Spring Harbor Laboratory Label: Cold Spring Harbor Laboratory Section: New Results Type: article.

[50] Patrick Esser, Robin Rombach, and Björn Ommer. Taming Transformers for High-Resolution Image Synthesis. *arXiv:2012.09841 [cs]*, June 2021. URL `http://arxiv.org/abs/2012.09841`. arXiv: 2012.09841.

[51] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic Backpropagation and Approximate Inference in Deep Generative Models. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1278–1286. PMLR, June 2014. URL `https://proceedings.mlr.press/v32/rezende14.html`. ISSN: 1938-7228.

[52] Martin Popel and Ondej Bojar. Training Tips for the Transformer Model. *The Prague Bulletin of Mathematical Linguistics*, 110(1):43–70, April 2018. ISSN 1804-0462. doi: 10.2478/pralin-2018-0002. URL `http://content.sciendo.com/view/journals/pralin/110/1/article-p43.xml`.

[53] G. D. Schuler, M. S. Boguski, E. A. Stewart, L. D. Stein, G. Gyapay, K. Rice, R. E. White, P. Rodriguez-Tomé, A. Aggarwal, E. Bajorek, S. Bentolila, B. B. Birren, A. Butler, A. B. Castle, N. Chiannilkulchai, A. Chu, C. Clee, S. Cowles, P. J. Day, T. Dibling, N. Drouot, I. Dunham, S. Duprat, C. East, C. Edwards, J. B. Fan, N. Fang, C. Fizames, C. Garrett, L. Green, D. Hadley, M. Harris, P. Harrison, S. Brady, A. Hicks, E. Holloway, L. Hui, S. Hussain, C. Louis-Dit-Sully, J. Ma, A. MacGilvery, C. Mader, A. Maratukulam, T. C. Matise, K. B. McKusick, J. Morissette, A. Mungall, D. Muselet, H. C. Nusbaum, D. C. Page, A. Peck, S. Perkins, M. Piercy, F. Qin, J. Quackenbush, S. Ranby, T. Reif, S. Rozen, C. Sanders, X. She, J. Silva, D. K. Slonim, C. Soderlund, W. L. Sun, P. Tabar, T. Thangarajah, N. Vega-Czarny, D. Vollrath, S. Voyticky, T. Wilmer, X. Wu, M. D. Adams, C. Auffray, N. A. Walter, R. Brandon, A. Dehejia, P. N. Goodfellow, R. Houlgatte, J. R. Hudson, S. E. Ide, K. R. Iorio, W. Y. Lee, N. Seki, T. Nagase, K. Ishikawa, N. Nomura, C. Phillips, M. H. Polymeropoulos, M. Sandusky, K. Schmitt, R. Berry, K. Swanson, R. Torres, J. C. Venter, J. M. Sikela, J. S. Beckmann, J. Weissenbach, R. M. Myers, D. R. Cox, M. R. James, D. Bentley, P. Deloukas, E. S. Lander, and T. J. Hudson. A gene map of the human genome. *Science (New York, N.Y.)*, 274(5287):540–546, October 1996. ISSN 0036-8075.

[54] Robert D. Finn, Alex Bateman, Jody Clements, Penelope Coggill, Ruth Y. Eberhardt, Sean R. Eddy, Andreas Heger, Kirstie Hetherington, Liisa Holm, Jaina Mistry, Erik L. L. Sonnhammer, John Tate, and Marco Punta. Pfam: the protein families database. *Nucleic Acids Research*, 42(Database issue):D222–D230, January 2014. ISSN 0305-1048. doi: 10.1093/nar/gkt1223. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3965110/.

[55] Peter J. Liu*, Mohammad Saleh*, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating Wikipedia by Summarizing Long Sequences. February 2018. URL https://openreview.net/forum?id=Hyg0vbWC-.

[56] Eric Jang, Shixiang Gu, and Ben Poole. Categorical Reparameterization with Gumbel-Softmax. *arXiv:1611.01144 [cs, stat]*, August 2017. URL http://arxiv.org/abs/1611.01144. arXiv: 1611.01144.

[57] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, April 2004. ISSN 1941-0042. doi: 10.1109/TIP.2003.819861. Conference Name: IEEE Transactions on Image Processing.

[58] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved Techniques for Training GANs. *arXiv:1606.03498 [cs]*, June 2016. URL http://arxiv.org/abs/1606.03498. arXiv: 1606.03498.

[59] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://papers.nips.cc/paper/2017/hash/8a1d694707eb0fefe65871369074926d-Abstract.html.

[60] Yuexu Jiang, Duolin Wang, Weiwei Wang, and Dong Xu. Computational methods for protein localization prediction. *Computational and Structural Biotechnology Journal*, 19:5834–5844, January 2021. ISSN 2001-0370. doi: 10.1016/j.csbj.2021.10.023. URL https://www.sciencedirect.com/science/article/pii/S2001037021004451.

[61] M Salvatore, P Warholm, N Shu, W Basile, and A Elofsson. SubCons: a new ensemble method for improved human subcellular localization predictions. *Bioinformatics*, 33(16):2464–2470, August 2017. ISSN 1367-4803. doi: 10.1093/bioinformatics/btx219. URL https://doi.org/10.1093/bioinformatics/btx219.

[62] James C. Wang. Cellular roles of DNA topoisomerases: a molecular perspective. *Nature Reviews. Molecular Cell Biology*, 3(6):430–440, June 2002. ISSN 1471-0072. doi: 10.1038/nrm831.

[63] Roger Hull. Chapter 10 - Movement of Viruses Within Plants. In Roger Hull, editor, *Plant Virology (Fifth Edition)*, pages 531–603. Academic Press, Boston, January 2014. ISBN 978-0-12-384871-0. doi: 10.1016/B978-0-12-384871-0.00010-8. URL https://www.sciencedirect.com/science/article/pii/B9780123848710000108.

[64] Ignazio Maggio, Hidde A. Zittersteijn, Qian Wang, Jin Liu, Josephine M. Janssen, Ivan Toral Ojeda, Silvère M. van der Maarel, Arjan C. Lankester, Rob C. Hoeben, and Manuel A. F. V. Gonçalves. Integrating gene delivery and gene-editing technologies by adenoviral vector transfer of optimized CRISPR-Cas9 components. *Gene Therapy*, 27(5):209–225, May 2020. ISSN 1476-5462. doi: 10.1038/s41434-019-0119-y. URL https://www.nature.com/articles/s41434-019-0119-y. Number: 5 Publisher: Nature Publishing Group.

[65] Yin-Yuan Mo, Chengyi Wang, and William T. Beck. A Novel Nuclear Localization Signal in Human DNA Topoisomerase I*. *Journal of Biological Chemistry*, 275(52):41107–41113, December 2000. ISSN 0021-9258. doi: 10.1074/jbc.M003135200. URL https://www.sciencedirect.com/science/article/pii/S0021925819556435.

[66] Rajesh Nair, Phil Carter, and Burkhard Rost. NLSdb: database of nuclear localization signals. *Nucleic Acids Research*, 31(1):397–399, January 2003. ISSN 1362-4962. doi: 10.1093/nar/gkg001.

[67] Michael Bernhofer, Tatyana Goldberg, Silvana Wolf, Mohamed Ahmed, Julian Zaugg, Mikael Boden, and Burkhard Rost. NLSdbmajor update for database of nuclear localization signals and nuclear export signals. *Nucleic Acids Research*, 46(D1):D503–D508, January 2018. ISSN 0305-1048. doi: 10.1093/nar/gkx1021. URL https://doi.org/10.1093/nar/gkx1021.

[68] D A McSwiggan, S Darougar, A F Rahman, and J A Gibson. Comparison of the sensitivity of human embryo kidney cells, HeLa cells, and WI38 cells for the primary isolation of viruses from the eye. *Journal of Clinical Pathology*, 28(5):410–413, May 1975. ISSN 0021-9746. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC475723/.

[69] J.W. Wallis, T.R. Miller, C.A. Lerner, and E.C. Kleerup. Three-dimensional display in nuclear medicine. *IEEE Transactions on Medical Imaging*, 8(4):297–230, December 1989. ISSN 1558-254X. doi: 10.1109/42.41482. Conference Name: IEEE Transactions on Medical Imaging.

[70] Peter J. Thul, Lovisa Åkesson, Mikaela Wiking, Diana Mahdessian, Aikaterini Geladaki, Hammou Ait Blal, Tove Alm, Anna Asplund, Lars Björk, Lisa M. Breckels, Anna Bäckström, Frida Danielsson, Linn Fagerberg, Jenny Fall, Laurent Gatto, Christian Gnann, Sophia Hober, Martin Hjelmare, Fredric Johansson, Sunjae Lee, Cecilia Lindskog, Jan Mulder, Claire M. Mulvey, Peter Nilsson, Per Oksvold, Johan Rockberg, Rutger Schutten, Jochen M. Schwenk, Åsa Sivertsson, Evelina Sjöstedt, Marie Skogs, Charlotte Stadler, Devin P. Sullivan, Hanna Tegel, Casper Winsnes, Cheng Zhang, Martin Zwahlen, Adil Mardinoglu, Fredrik Pontén, Kalle von Feilitzen, Kathryn S. Lilley, Mathias Uhlén, and Emma Lundberg. A subcellular map of the human proteome. *Science*, 356(6340):eaal3321, May 2017. doi: 10.1126/science.aal3321. URL https://www.science.org/doi/10.1126/science.aal3321. Publisher: American Association for the Advancement of Science.

[71] Ulrike Schnell, Freark Dijk, Klaas A Sjollema, and Ben N G Giepmans. Immunolabeling artifacts and the need for live-cell imaging. *Nature Methods*, 9(2):152–158, February 2012. ISSN 1548-7105. doi: 10.1038/nmeth.1855. URL https://doi.org/10.1038/nmeth.1855.

[72] Taming Transformers for High-Resolution Image Synthesis, May 2022. URL https://github.com/CompVis/taming-transformers. original-date: 2020-12-17T14:47:06Z.

[73] Phil Wang. DALL-E in Pytorch, April 2022. URL https://github.com/lucidrains/DALLE-pytorch. original-date: 2021-01-05T20:35:16Z.

[74] Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, and Yunfeng Liu. RoFormer: Enhanced Transformer with Rotary Position Embedding. *arXiv:2104.09864 [cs]*, October 2021. URL http://arxiv.org/abs/2104.09864. arXiv: 2104.09864.

[75] Transformer - , . URL https://zhuanlan.zhihu.com/p/191393788.

[76] [1904.10509] Generating Long Sequences with Sparse Transformers, . URL https://arxiv.org/abs/1904.10509.

[77] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature Methods*, 18(1):100–106, January 2021. ISSN 1548-7105. doi: 10.1038/s41592-020-01018-x. URL https://www.nature.com/articles/s41592-020-01018-x. Number: 1 Publisher: Nature Publishing Group.

[78] Samira Abnar and Willem Zuidema. Quantifying Attention Flow in Transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.385. URL https://aclanthology.org/2020.acl-main.385.

[79] Daniel Osorio, Paola Rondón-Villarreal, and Rodrigo Torres Sáez. Peptides: A Package for Data Mining of Antimicrobial Peptides. *The R Journal*, 7:4–14, June 2015. doi: 10.32614/RJ-2015-001.

[80] Akinori Kidera, Yasuo Konishi, Masahito Oka, Tatsuo Ooi, and Harold A. Scheraga. Statistical analysis of the physical properties of the 20 naturally occurring amino acids. *Journal of Protein Chemistry*, 4 (1):23–55, February 1985. ISSN 1573-4943. doi: 10.1007/BF01025492. URL https://doi.org/10.1007/BF01025492.

[81] M Sandberg, L Eriksson, J Jonsson, M Sjöström, and S Wold. New chemical descriptors relevant for the design of biologically active peptides. A multivariate characterization of 87 amino acids. *Journal of medicinal chemistry*, 41(14):2481–2491, July 1998. ISSN 0022-2623. doi: 10.1021/jm9700575. URL https://doi.org/10.1021/jm9700575.

[82] Gabriele Cruciani, Massimo Baroni, Emanuele Carosati, Monica Clementi, Roberta Valigi, and Sergio Clementi. Peptide studies by means of principal properties of amino acids derived from MIF descriptors. *Journal of Chemometrics*, 18(3-4):146–155, 2004. ISSN 1099-128X. doi: 10.1002/cem.856. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/cem.856. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/cem.856.

[83] Guizhao Liang and Zhiliang Li. Factor Analysis Scale of Generalized Amino Acid Information as the Source of a New Set of Descriptors for Elucidating the Structure and Activity Relationships of Cationic Antimicrobial Peptides. *QSAR & Combinatorial Science*, 26(6):754–763, 2007. ISSN 1611-0218. doi: 10.1002/qsar.200630145. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/qsar.200630145. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/qsar.200630145.

[84] Feifei Tian, Peng Zhou, and Zhiliang Li. T-scale as a novel vector of topological descriptors for amino acids and its application in QSARs of peptides. *Journal of Molecular Structure*, 830(1):106–115, March 2007. ISSN 0022-2860. doi: 10.1016/j.molstruc.2006.07.004. URL `https://www.sciencedirect.com/science/article/pii/S0022286006006314`.

[85] Hu Mei, Zhi H. Liao, Yuan Zhou, and Shengshi Z. Li. A new set of amino acid descriptors and its application in peptide QSARs. *Biopolymers*, 80(6):775–786, 2005. ISSN 0006-3525. doi: 10.1002/bip.20296.

[86] Gerard JP van Westen, Remco F Swier, Jörg K Wegner, Adriaan P IJzerman, Herman WT van Vlijmen, and Andreas Bender. Benchmarking of protein descriptor sets in proteochemometric modeling (part 1): comparative study of 13 amino acid descriptor sets. *Journal of Cheminformatics*, 5:41, September 2013. ISSN 1758-2946. doi: 10.1186/1758-2946-5-41. URL `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3848949/`.

[87] Li Yang, Mao Shu, Kaiwang Ma, Hu Mei, Yongjun Jiang, and Zhiliang Li. ST-scale as a novel amino acid descriptor and its application in QSAM of peptides and analogues. *Amino Acids*, 38(3):805–816, March 2010. ISSN 1438-2199. doi: 10.1007/s00726-009-0287-y.

[88] Alexander G. Georgiev. Interpretable numerical descriptors of amino acid space. *Journal of Computational Biology: A Journal of Computational Molecular Cell Biology*, 16(5):703–723, May 2009. ISSN 1557-8666. doi: 10.1089/cmb.2008.0173.

[89] A. Zaliani and E. Gancia. MS-WHIM Scores for Amino Acids: A New 3D-Description for Peptide QSAR and QSPR Studies. *J. Chem. Inf. Comput. Sci.*, 1999. doi: 10.1021/ci980211b.