Deep-learning enabled clustering of KSTAR ECEI data

R. Kube[1], R.M. Churchill[1], J. Choi [2], and M.J. Choi[3]

[1] *Princeton Plasma Physics Laboratory, Princeton, NJ 08540, USA*

[2] *Oak Ridge National Laboratory, Oak Ridge, TN 37831, USA*

[3] *Korea Institute of Fusion Energy, Daejeon 34133, Republic of Korea*

Abstract

A key enabler to scientific studies is the easy access to relevant measurements. As the corpus of measurement data increases geometrically, due to more measurements being taken at increasing fidelity, better ways to find relevant data are required. Approaching this problem, we explore how unsupervised machine learning models can be used to classify a corpus of electron cyclotron emission imaging (ECEI) data from the KSTAR tokamak. A dataset consisting of prominent MHD phenomena such as magnetic islands and ELM filaments is compiled and clustered using end-to-end trainable deep learning architectures. In particular we are using an approach based on generative adversarial networks and deep divergence-based clustering. On this reference dataset, the used methods achieve cluster accuracies of up to 90%.

Introduction

Fusion plasmas are regularly measured by a vast array of plasma diagnostics. As diagnostics are upgraded, new ones are added, or new generation of machines are commisioned, data streams increase in velocity and variety. To make optimal use of these ever increasing amounts of data for scientific discovery it is desirable to improve the tools available for scientists to query fusion data bases. Case in point, long-pulse operation of KSTAR generates of the order of 10 Gigabyte of measurement data per discharge. ITER is expected to generate approximately 2 Petabytes of data per day [1]. And with upcoming long pulse devices, automatic data labelling and extraction capabilities become desireable to aid scientific discovery by optimizing data access. Performance of deep learning algorithms are known to scale with the size of the training dataset [1]. However, manually labelling datasets for supervised learning is time consuming and infeasible on the turnaround time of fusion experiments [2]. To break free of expensive data labelling tasks, we explore how well end-to-end trainable deep learning architectures can cluster a dataset of KSTAR ECEI measurements of various MHD phenomena, including magnetic islands and ELM filaments. Such trained models may be used to easily assemble large datasets of specific MHD phenomena.

[1] https://www.iter.org/newsline/-/3534

Deep-learning enabled clustering

KSTAR's ECEI diagostics is regularly used to study MHD phenomena [3]. Our work aims to cluster ECEI images, such that each cluster represents a specific MHD phenomenon. For this, we compiled a dataset consisting of 18 shots where various MHD phenomena are visible. 12 shots show 2/1 magnetic islands, 5 shots show 3/2 magnetic islands, and 1 shots has visible ELM filaments. These phenomena are visible between 0.5 and 5 seconds, varying between the shots. All data is preprocessed with a bandpass filter, with a pass band between 5 and 9 kHz, then normalized to its mean and clamped to 15% fluctuation level.

We continue by discussing experiments on how well deep-learning enabled clustering algorithms perform on the dataset described in the previous paragraph. Generative Adversarial Networks approximate a generative model via an adversarial process [4]. As a central ingredient, a generator G and a discriminator D are trained in an adversarial fashion. G maps a latent space Z into the space where the data distribution we wish to learn, X, is embedded in. During training, D is rewarded for correctly discriminating between real samples from X and samples generated by G. And G is rewarded by producing examples that D decides are from X. In this setting, D acts as a binary classifier. To adapt GANs to perform multiclass classification [5], the discriminator outputs a probability distribution over K classes and the Shannon entropy H takes a central place in the loss functions. A uniform distribution over the class labels is assumed as a prior. The discriminator is then rewarded by utilizing the class labels equally, for being maximally certain in class assignments for real data and maximally uncertain in class assignments for generated data:

$$\mathscr{L}_D = \max_D H_X\left[p(y|D)\right] - \mathbb{E}_{x\sim X}\left[H(p(y|x,D))\right] + \mathbb{E}_{z\sim p_z(z)}\left[H(p(y|G(z),D))\right]. \tag{1}$$

G is also rewarded for utilizing class labels equally and for instances where D minimizes H when subject to generated data

$$\mathscr{L}_G = \max_D H_{G(z)}\left[p(y|D)\right] - \mathbb{E}_{z\sim p_z(z)}\left[H(p(y|G(z),D))\right]. \tag{2}$$

In order to work with ECEI data, we implement a discriminator that performs a series of three-dimensional convolutions using 16 filters of size $(5,3,3)$, followed by another set of 16 filters of size $(5,3,5)$, followed by 32 filters of size $(5,3,1)$ and another 32 filters of size $(9,1,1)$. The output of these layers is flattened and reduced into a 128 element vector through a fully connected layer. Here the code is subject to minibatch discrimination [6] before a final fully connected layer transforms it into a 3 element vector on which a softmax function is applied. The generator architecture up-samples from the latent dimension into a 768-dimensional space.

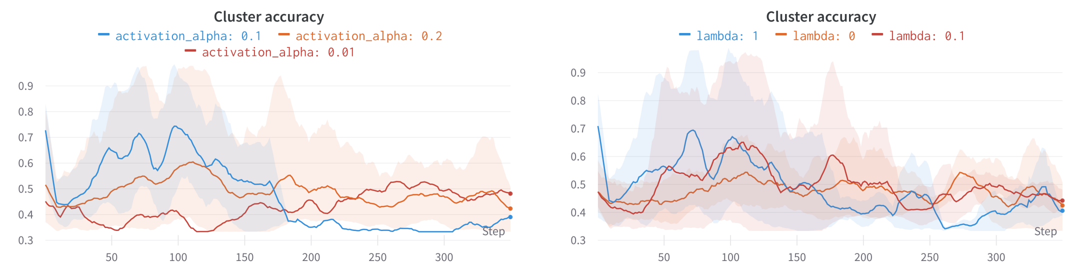

Figure 1: Clustering accuracy using Categorical GANs. Left: Average accuracy for various leak rates of the leaky ReLU activation function. Right: Average accuracy for different degrees of semi-supervision. The shaded area denotes maxima and minima of the loss curves and the solid line denotes the exponential moving average.

From there, a series of up-convolutional and convolutional layers transform the code into 8 24x8 images.

In experiments, we vary the size of the convolutional layers, the activation function, the size of the latent dimension, learning rates of D and G, the size of the minibatch discrimination layer, and the batch size. Additionally, we also investigate whether some degree of semi-supervision aids in the classification at hand. To evaluate the performance of the model we observe that multiple metrics need to be evaluated. First, and most importatnt, the cluster accuracy should be high. Then, the expected value of the entropy for real samples needs to be larger than for generated samples. And finally, histograms of the class probabilities for real samples should show clear peaks at 0 and 1, but have a much broader distribution for generated samples.

All experiments were performed with 0.1s data from one discharge class each in order to satisfy the uniform prior on the class labels. Results of training are shown in Fig.1, in all figures the x-axis is in units of step, which equals 25 mini-batches. We observe averaged cluster accuracies of up to 90% only when using leak rates fo $\alpha = 0.1$ or 0.2 for the used leaky-ReLU activation function. Smaller values result in poorer accuracy. Some degree of semi-supervision, parameterized by λ, is necessary to increase the average accuracy from 55% to 70%.

Using a batch size of 128, the cluster accuracy reaches 90% after about 50 steps, where each step corresponds to training with 25 mini-batches. This accuracy is reached only after about 125 steps when using a batch size of 64. The size of the latent dimension does not affect the achievable average cluster accuracy.

Deep-divergence based clustering (DDC) enables to discover cluster structure in unlabelled data by incorporating information theoretic divergence measures in the loss function [7]. Here

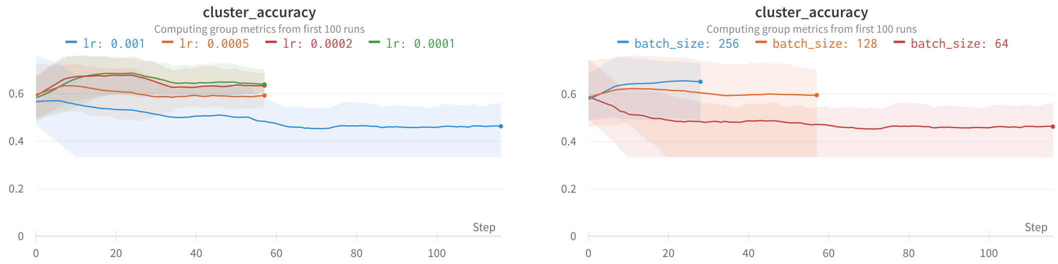

Figure 2: Clustering accuracy using DDC. Left: Average accuracy for various learning rates. Right: Average accuracy for different batch sizes.

the Cauchy-Schwarz divergence is estimated from intermediate representations in a convolutional architecture, enabling end-to-end training of the model. For experiments, we implemented the proposed architecture with 2 convolutional layers, followed by maxpool layers. The filter sizes were either $(3,3)$ or $(5,5)$ with 16 and 32, or 32 and 64 channels respectively. Sizes of the maxpool layers were fixed to $(2,2)$. Input image sequences of 5 and 10 layers were used. The networks were trained with the ADAM optimizer using batch sizes of 64, 128, 256 and learning rates of $\{1,2,5\} \times 10^{-4}$ and 10^{-3}. As shown in Fig.2, this architecture achieves average clustering accuracies of about 65%. The best accuracies were achieved with learning rates smaller than 0.0002 and a batch size of 128. The accuracy varied little over the scanned range of filter sizes, channels, or size of the final convolutional layer.

Future work aims to reduce the amount of manual intervention to perform unsupervised learning tasks. For one, we aim to explore pre-processing methods for ECEI data with less adjustable parameters than the bandpass method used for this work. And second, we aim to explore other proposed deep-learning enabled, end-to-end trainable clustering methods [8].

References

[1] C. Sun et al. arXiv 1707.02968 (2017); M. Johnson et al. Proc. 56th Annual Meeting of the Ass. for Comp. Ling. 450 - 455 (2018); J. Hestness et al. arXiv 1712.00409 (2017); M.Z. Alom et al. Electronics **8** 292 (2019)

[2] A. Jalalvand et al. Nucl. Fusion **62** 026007 (2022); J. Kates-Harbeck et al. Nature **568** 526-531 (2019); A. Bustos et al. Plasma Phys. Control. Fusion **63** 095001 (2021); K.J. Montes et al. Nucl. Fusion **61** 026022 (2021)

[3] G.S. Yun et al. Rev. Sci. Instrum.**83** 11D820 (2014); H. Park Adv. Phys. X **4** 1633956 (2019)

[4] I.J. Goodfellow et al. NeurIPS **27** (2014).

[5] J.T. Springenberg, arXiv **1511.06390** (2016)

[6] T. Salimans, et al. NeurIPS **29** (2016)

[7] M. Kampffmeyer et al. Neur. Networks **113** (2019)

[8] M. Caron et al. ECCV **132-149** (2018)