FATE: Feature-Agnostic Transformer-based Encoder for learning generalized embedding spaces in flow cytometry data

Lisa Weijler
TU Wien
lweijler@cvl.tuwien.ac.at

Florian Kowarsch
TU Wien
florian.kowarsch@gmail.com

Michael Reiter
TU Wien
rei@cvl.tuwien.ac.at

Pedro Hermosilla
TU Wien
phermosilla@cvl.tuwien.ac.at

Margarita Maurer-Granofszky
St. Anna CCRI
margarita.maurer@ccri.at

Michael Dworzak
St. Anna CCRI
michael.dworzak@stanna.at

Abstract

While model architectures and training strategies have become more generic and flexible with respect to different data modalities over the past years, a persistent limitation lies in the assumption of fixed quantities and arrangements of input features. This limitation becomes particularly relevant in scenarios where the attributes captured during data acquisition vary across different samples. In this work, we aim at effectively leveraging data with varying features, without the need to constrain the input space to the intersection of potential feature sets or to expand it to their union. We propose a novel architecture that can directly process data without the necessity of aligned feature modalities by learning a general embedding space that captures the relationship between features across data samples with varying sets of features. This is achieved via a set-transformer architecture augmented by feature-encoder layers, thereby enabling the learning of a shared latent feature space from data originating from heterogeneous feature spaces. The advantages of the model are demonstrated for automatic cancer cell detection in acute myeloid leukemia in flow cytometry data, where the features measured during acquisition often vary between samples. Our proposed architecture’s capacity to operate seamlessly across incongruent feature spaces is particularly relevant in this context, where data scarcity arises from the low prevalence of the disease. The code is available for research purposes at https://github.com/lisaweijler/FATE.

1. Introduction

In recent years, a prominent trend in the machine learning landscape has been to design more generic architectures and learning strategies. Examples are multi-modality learning, data modality agnostic learning, or task agnostic learning. Successes in this field can be attributed to the rise of transformer models [32], where (cross-)attention makes the combination of feature-embeddings of different modalities simple, as well as general pre-training and feature extraction methods such as the masked autoencoder (MAE), where training does not depend on data modality or task-specific data augmentations. Yet, those approaches rely on unimodal encoders to process the different types of features corresponding to each data modality.

In this work, we propose a novel architecture, which is agnostic to the number and order of features of the input data by learning a generalized feature space using only one feature-agnostic encoder. We build upon the flexibility of transformers with respect to varying input sequence lengths and ordering. By using a feature encoding similar to the idea of positional encoding and an MAE training strategy, the proposed model learns a general embedding space that captures the relationship between features across data samples. The common way of training an MAE is to mask parts of the input sample, such as patches of an image [12]. In contrast, we propose masking of single feature values to learn the relationship between different features across datasets.

An application, where this characteristic holds significant relevance is the automated processing of Flow Cytometry (FCM) data. An FCM sample is essentially a set of feature vectors (events), each corresponding to a single cell. Based on the properties measured, diverse cell populations can be detected and analyzed. The choice of the features measured and used for analysis is highly task-dependent and often varies even within one use case due to different medical protocols, labs, and routines. While this is no problem for manual analysis, where FCM samples are analyzed one by one, automated processing with e.g. deep learning requires a consistent set of features across datasets.
We evaluate the proposed architecture on the task of cancer cell detection in pediatric Acute Myeloid Leukemia (AML) patients. The proportion of remaining cancer cells during and after treatment, called the Measurable Residual Disease (MRD), is an important factor for risk stratification and the development of effective treatment plans in accordance with individual patients’ needs. This is a particularly challenging task of FCM data analysis, given the possible low proportions (down to 0.01%) and high heterogeneity of cancer cells due to patient-specific phenotype characteristics; sample- or patient-specific features are necessary to successfully distinguish healthy from cancerous cells. In addition, training data is scarce because of the low incidence of the disease (5-10 per mio. in Europe [6]). In summary, our contribution is threefold:

1. We introduce a novel architecture FATE (Feature-Agnostic Transformer-based Encoder), which, to the best of our knowledge, is the first one to enable processing of data with flexible input features.

2. We propose a pre-training strategy based on MAE with a novel masking strategy that improves the quality of the representations learned by our model and it is agnostic to the number and type of features of the datasets.

3. Using our proposed pre-training strategy we set a new State of The Art (SOTA) for the task of MRD detection in pediatric AML patients and gain a performance increase of approximately 60% compared to training from scratch.

2. Related Work

In this section, we describe the related work to this paper in different fields: Multimodal deep learning, MAE, and FCM data processing.

Multimodal deep learning focuses on processing multiple input modalities such as images and text, to learn richer semantic features than unimodal data can offer, as well as for cross-modality tasks such as text-to-image generation. A prominent example is CLIP [26], which is trained on text-image pairs to learn a cross-modality embedding used for different downstream tasks [34,37]. Another pioneering work in this area is DALL-E [27] for text-to-image generation and several follow-up models. This is also a common practice in the medical field, where it enables the extraction of potentially complimenting features of diverse healthcare data modalities [20,25,30,31]. More recently ImageBind [13] was proposed, which aims at learning a general embedding space for several data modalities such as audio, video, images and text. Those approaches, however, use several unimodal encoders or feature concatenation to process the features related to each modality and therefore expect a fixed number and ordering of features. In contrast, our model allows flexibility with respect to the features used.

Masked autoencoding has regained attention due to its capacity to learn useful embeddings from unlabelled data serving as an effective pre-training strategy for Deep Learning (DL) models with an increasing number of parameters and capacities. MAE are essentially a general form of denoising autoencoders [33] with a simple idea: mask i.e. remove a part of the input and let the model predict the missing part. This self-supervised training strategy does not depend on data-specific augmentations and is agnostic to the data modality. Starting from its success for natural language models such as BERT [7], it has made its way into vision [9,12,36], 3D data processing [14,16,38], and many other domains [4,10,13,21]. MAE also has been successfully employed in multimodal learning [4,5,11], yet, again they rely on different encoders for each data modality.

Automated FCM data processing has a wide range of applications in a number of fields and is usually concerned with the detection and classification of novel or targeted cell populations. Early works of targeted cell population detection pool events of different samples from the training set together and train a classifier using pairs of single events and corresponding labels [2,19,23]. Methods using single events as input are restricted to learning fixed decision regions, while the relational positioning of cell populations to each other has proven as important information for successfully detecting rare or aberrant cells such as in MRD detection [29]. Therefore, approaches that process a whole FCM sample at once have emerged e.g. using Gaussian Mixture Models [28] or transforming samples to images and applying convolutional neural networks [3]. A more natural solution, without loss of information due to prior transformations are methods based on the attention mechanism [15,32,35]. Attention-based models are a way for event-level classification that learns and incorporates the relevance of other cell populations in a sample for the specific task. The current SOTA for automated MRD detection [35] is based on the Set Transformer (ST) [17], an efficient variation of the Transformer model specifically designed for sets. Those approaches assume, however, a fixed set of features during training and inference. FCM data features are highly variable even within one dataset; reducing them to the greatest common intersection for consolidated processing dismisses discriminative information for the successful identification of targeted cells. There exists a work that aims at combining the features of samples by using nearest neighbor imputation [1,18,24] but a recent study shows that those methods have severe limitations due...
to the questionable accuracy of imputed values for downstream analysis [22].

3. Methods

Our feature-agnostic model has a simple architecture depicted as an encoder and a prediction head. The transformer architecture is equivariant to permutations of the input sequence and can handle different sequence lengths. We exploit those properties in two ways. First, to be able to process events with different numbers and ordering of features, and second to be able to process samples with varying amounts of events. Fig. 1 shows the architecture described in Sec. 3.2.

For pre-training of our FATEncoder we use an MAE that is trained to reconstruct the measurement values of masked features. Note that this differs from standard masking where whole instances of the data sample such as patches or pixels of images are masked. An illustration of the proposed MAE is given in Fig. 2, and in Sec. 3.3 its components are explained in detail.

3.1. Preliminaries

Formally, we denote an input sample (set) $X_i$ to our model as

$$X_i = \{x_{ki} \in \mathbb{R}^{F_i}\}_{k=1}^{n_i},$$

where $F_i$ is the dimension of the feature vectors i.e. the number of features, and $n_i$ is the number of elements in the input set. One dataset constituted of $N$ samples is a set of sets

$$\{X_i \in \mathbb{R}^{n_i \times F_i}\}_{i=1}^{N}.$$  

In the context of FCM data $x_{ki}$ is one event and $F_i$ the features measured by FCM during acquisition. The number of events as well as the feature space can vary between samples; $n_i$ is typically between $10^4$ and $10^6$ and $F_i$ between 10 and 20 (limited by the properties of FCM acquisition).

We use standard attention as proposed in [32] taking queries $Q$, keys $K$ and values $V$ as input. For self-attention $Q$, $K$ and $V$ are linear projections of the same input set $X_i$, while for cross-attention $Q$ is either a learnable vector or our feature encodings as explained below, $K$ and $V$ are again linear projections of the same input set $X_i$.

3.2. FATEncoder

The proposed encoder creates a common embedding space of fixed dimension $F_c$, regardless of the number and order of features of the input sample. It can be subdivided into two parts, which we call the Feature-Encoder and the Set-Encoder. The first part is responsible for being able to process inputs with different ordering and number of features, while the second part provides the context of all other events within one sample.

In order to achieve the feature agnostic property, the Feature-Encoder treats the feature values of single events as input sequences, i.e. a sequence of scalar values. Given a sample $X_i$, each event $x_{ki} \in \mathbb{R}^{F_i}$ thus turns into an input sequence $[x_{k_1}, \ldots, x_{k_{F_i}}]^T \in \mathbb{R}^{1 \times F_i}$. (3)

We employ feature encoding similar to the idea of positional encoding, so the model knows what feature the scalar value in the sequence belongs to, and hence is able to relate them to others and learn their semantic meaning in the context of the dataset. The encoding is concatenated to the scalar measurement value of its corresponding feature, making the final input to our FATEncoder a sequence of vectors

$$[\phi_E(x_{ki})_1, \ldots, \phi_E(x_{ki})_{F_i}]^T \in \mathbb{R}^{1+D_E \times F_i},$$

$$\phi_E(x_{ki})_j := [x_{k_j}, E_j] \in \mathbb{R}^{1+D_E},$$  

with $E_j$ as the feature encoding vector for feature $j$ and $D_E$ as its dimension.

Initially, self-attention is applied between those input vectors followed by cross-attention with a learned query vector $q \in \mathbb{R}^{D_{mask}}$ that summarizes the information of all features and transforms it into one single embedding vector of fixed dimension $z_{ki} \in \mathbb{R}^{D_{mask}}$,

$$z_{ki} = \text{CrossAttn}(q, \text{SelfAttn}(\phi_E(x_{ki}))),$$

$$\phi_E(x_{ki}) := [\phi_E(x_{ki})_1, \ldots, \phi_E(x_{ki})_{F_i}].$$

After all elements of one sample have passed through the Feature-Encoder\footnote{A full sample $X_i \in \mathbb{R}^{n_i \times F_i}$ can be efficiently processed as batch by using a tensor of dimension $n_i \times F_i \times 1 + D_E$ as input.}, the Set-Encoder puts the event-embeddings $z_{ki}$ of one sample in context to each other. For this we utilize the Induced Self-Attention Blocks (ISABs) as introduced with the ST [17] (see supplementary materials). The ISAB uses learned queries (induced points) to reduce the $O(n^2)$ complexity of self-attention to $O(nm)$ with $n$ denoting the input sequence length and $m$ the number of induced points. Finally, a linear layer maps the embedding to its final embedding dimension $F_c$. The Set-Encoder can be summarized as,

$$\{z_{ki}\}_{i=1}^{n_i} \leftarrow \text{LinLayer}(\text{ISAB}^{(l)}(\{z_{ki}\}_{i=1}^{n_i})),$$

with $l$ being the number of ISAB layers applied.

Feature encoding In contrast to the positional encoding used in e.g. natural language processing, where ordinal encodings i.e. encodings that have a ranked ordering are used, our feature encoding should be nominal, thus not imposing an artificial relationship between values. A typical choice is...
Figure 1. Illustration of the architecture of our model. **Left:** Each input feature of each event is concatenated with a positional encoding descriptor. These augmented features are updated using the other features of each event (Feature-Encoder) and additional events from the same sample (Set-Encoder). The resulting features, which are independent of the number and type of features used in the input, are used for the final prediction. **Right:** Detailed illustration of the Feature-Encoder and Set-Encoder networks. We use Self- and Cross-Attention one-hot encoding. However, with this choice, the dimension grows with the number of features covered and after training this number is locked-in given that adding new features would increase the dimension of the input vectors. This is why we chose learned encoding vectors of fixed dimension as used in [8] i.e. a learnable function

\[
e : \{0, \ldots, M - 1\} \mapsto \mathbb{R}^{DE},
\]

where \( M \) denotes the number of different features present in the training data set.

See supplementary materials for a full list of features used.

**Prediction head** As prediction head, we use a 2-layer MLP with GELU as an activation function and a hidden layer dimension equal to the common feature space dimension \( F_c \).

### 3.3. FATE-MAE

We propose an MAE strategy for pre-training our model. Like all autoencoders, our method uses an encoder that transforms the observed signal into a latent representation, and a decoder that reconstructs the original signal from this latent representation. However, our encoder is specifically designed to be able to process varying input features as described in Sec. 3.2. Consequently, our decoder has to be able to reconstruct a varying amount of features. The masking strategy and decoder architecture are described in the following; Fig. 2 gives an overview of the MAE. A detailed illustration of our proposed decoder architecture is given in the supplementary materials.
Masking Contrary to common masking strategies, where full instances e.g. patches of an image are masked, we mask a proportion of feature measurement values per element of the input set $X_i$, i.e. per feature measurement vector. The proposed masking strategies allow us to learn the relationship between features in the context of the composition of cell populations within and across samples, in contrast to learning the relationship between cell populations given a fixed set of features, which is standard masking. The strategy is straightforward: We define a fixed ratio of features to mask (i.e., remove); a masking ratio of 0.5 would mean removing half of the feature measurement values in each feature measurement vector of all samples. What features are to be masked is chosen randomly following a uniform distribution and can be different for each $x_{k_i} \in X_i$.

Decoder The decoder is symmetric to the encoder. Following the structure of the encoder, we divide it into the Set-Decoder and Feature-Decoder. The Set-Decoder consists of ISAB layers to capture the relation between the embeddings of one sample $\{z_{k_i}\}_{i=1}^n$. The Feature-Decoder reconstructs the original feature measurement values of each set element $x_{k_i} \in \mathbb{R}^{F}$ from its corresponding embedding vector $z_{k_i} \in \mathbb{R}^{F_c}$ given the context of all elements in the input set. This is not trivial due to the varying number of features between samples. To achieve this we condition the decoder on the features of which we want their measurement values to be reconstructed by cross-attending their feature encodings with the output of the set-decoder. Finally, similar to the feature-encoder we apply self-attention between the reconstructed feature measurement value vector of each $x_{k_i}$ and map them with a linear layer to a scalar value. For a detailed illustration of the proposed decoder see supplementary materials.

4. Experimental Setup

In this section, we give an overview of the experiments conducted, the implementation details, and the datasets used.

We focus on the task of MRD detection of FCM samples in pediatric AML patients given that it is one of the most challenging tasks in FCM data analysis with increased varying features between samples. AML depicts a particularly high heterogeneity of cancer cells in comparison to other sub-types such as b-Cell Lymphoblastic Leukemia (b-ALL). Consequently, the features necessary to successfully separate healthy from cancer cells are highly diverse. In addition, it is a very rare disease resulting in a natural data scarcity. Hence, the ability to use other datasets for pre-training while being flexible with respect to input features is crucial.

We treat the problem as a binary classification, cancerous or healthy, and use binary cross-entropy as loss function for supervised training. Precision $p$, recall $r$, and F-score $F_1$ with cancer cells as positives are used as evaluation metrics. The metrics are calculated per sample, then averaged to determine the final metrics of the test set.

4.1. Datasets

The main data set (AML-MRD) for the downstream task of MRD detection consists of 71 FCM samples coming from 12 different pediatric AML patients. The samples were collected at different time points during therapy between the years 2021 and 2022. The number of markers used for samples of this dataset varies between 10 and 14. Additionally, the forward- and side-scatter of the laser light, which reflect the physical properties of the cells, are used as features. Between all samples of this dataset, 6 markers are shared. For a full listing of the number of samples per patient and the corresponding features see the supplementary materials.

For pre-training we use data from pediatric b-ALL patients at day 15 of therapy, data from pediatric AML patients at diagnosis, where cancerous cells have replaced most of the healthy cells, as well as FCM samples from screening cases or recovered patients, that are free of cancer cells (MRD negative). The b-ALL data is a publicly available dataset (ALL-MRD) collected from three different laboratories in Berlin, Buenos Aires, and Vienna. The diagnosis (DIA) as well as the MRD negative dataset (CONTROL) are a collection of samples from different laboratories in Vienna, Padua and Essen between the years 2016-2022. CONTROL and DIA have 4 features with AML-MRD in common, while ALL-MRD only has 2 excluding forward- and side-scatter. Supplementary materials give an overview of all features and the overlaps between datasets.

Sampling and research were approved by local Ethics Committees, and informed consent was obtained from patients’ or patients’ parents or legal guardians according to the Declaration of Helsinki. Ground truth was obtained using manual analysis by at least two experts. Whenever available, results were confirmed using an independent molecular methodology (RT-PCR).

The datasets we use for pre-training are not suitable as training data for the downstream task of MRD detection in pediatric AML patients, given the different types of cancer cells (ALL-MRD), composition of data (DIA), or lack of cancer cells (CONTROL). However, they can be successfully utilized for pre-training by our proposed model as we show in Sec. 5.

Tab. 1 provides a tabular overview of the data sets.

4.2. Experiments

We conduct several experiments to evaluate the proposed FATE architecture and MAE. The current SOTA for auto-
serves as a baseline using the official implementation of MRD detection in FCM data based on the ST [35] and the FATE model supervised with the CONTROL dataset. We test our proposed pre-training method based on MAE using the CONTROL, DIA and ALL-MRD datasets.

**Implementation details** For the training-from-scratch experiments, we train our model for 400 epochs and use early stopping after 300 epochs if there is no improvement of the $F_1$-score on the validation set. The baseline model is trained for 200 epochs with early stopping after 100 epochs since it showed a tendency to overfit on the small datasets.

For the supervised pre-training we use 1500 epochs and finetune for 100 epochs using the pretrained weights as initialization. Given that the task of finetuning is the same as for pre-training we do not remove the prediction head. For pre-training of the baseline we conduct two experiments. First, we only use the 4 features plus forward- and side-scatter that the CONTROL and DIA have in common with the AML-MRD dataset and second, we use the union over all occurring features in all three datasets as input imputing the missing ones with zeros. For the FATE model each input sample’s individual features can be utilized.

Pre-training of the FATE MAE is conducted for 2000 epochs for all experiments and pre-trained weights are again used as initialization for training of the downstream task. The model is finetuned for 300 epochs with early stopping after 200 and different masking-ratios are evaluated.

The $L_1$-loss is used and calculated between the true measurement values of masked features and those reconstructed by the decoder. With respect to the learning rate for finetuning and from-scratch experiments, we opt for the cosine-annealing learning rate scheduler with a starting value of 0.001, a minimum value of 0.0002, and a maximum of 10 iterations. Throughout all pre-training experiments, we use the same scheduler with a starting value of 0.001, a minimum value of 0.00002, and a maximum of 100 iterations. AdamW optimizer is employed for all experiments. A batch size of 32 and 8 is used for pre-training and finetuning, respectively. Training is conducted on an NVIDIA GeForce RTX 3090.

All models are trained at least three times with different initialization and mean and standard deviations are reported.

**Patient cross-validation** Since the target dataset AML-MRD is small with 71 samples and several samples can originate from the same patient we employ a cross-validation to maximize the amount of training and evaluation data. The model should be capable of detecting MRD in FCM data of new patients, meaning for a fair assessment, samples taken from one patient must not be split up between the training, evaluation, and test set. Therefore we conducted a “patient-cross-validation”, where each patient generates one split, i.e., the patient’s samples are held out as test data. All remaining samples from other patients are divided into training and evaluation sets with a ratio of 0.8 and 0.2, respectively. The reported metrics are the average of the resulting values over all samples. See supplementary materials for a full list of splits and the number of training, validation, and test samples.

**Model architecture** For the baseline model we use the same architecture as proposed in [35], namely 4 ISAB layers with a hidden dimension of 32 and 16 induced points. For our FATE model, we use the architecture as described in Sec. 3 with 3 ISAB layers in the encoder and decoder, a hidden dimension of 32, and an embedding space dimension $F_e = 8$. The learned feature encodings have a dimension of $D_E = 10$.

### 5. Results

Tab. 2 shows the results of the experiments conducted. Training from scratch yields similar results for all three setups, ST with the intersection of all features, ST with the union over all features in all datasets, and our FATE model. The ST using all features as input, where missing ones are imputed with 0, performs slightly better showing the benefit of sample-specific features. Our FATE model needs to learn the feature encoding and the common embedding space, which is hard given the little training set when training from scratch. Yet it holds comparable results with higher mean precision and recall.

Pre-training the models supervised with the CONTROL and DIA dataset improves results for all models. The biggest increase can be seen for our FATE model, indicating that the proposed architecture can successfully learn the relation between features across different samples. Surprisingly, when adding the ALL-MRD dataset to pre-training, the performance of the ST drops. One explanation is the increased difference in features and in cancer cells to be

<table>
<thead>
<tr>
<th>Name</th>
<th>City</th>
<th>Years</th>
<th>Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>AML-MRD</td>
<td>Vie</td>
<td>2021-22</td>
<td>71</td>
</tr>
<tr>
<td>CONTROL</td>
<td>Vie, Pad, Ess</td>
<td>2016-21</td>
<td>308</td>
</tr>
<tr>
<td>DIA</td>
<td>Vie, Pad, Ess</td>
<td>2016-22</td>
<td>110</td>
</tr>
<tr>
<td>ALL-MRD</td>
<td>Vie, Bln, Bue</td>
<td>2009-14</td>
<td>338</td>
</tr>
</tbody>
</table>

3 https://github.com/mwoedlinger/flowformer
Table 2. Results of our main experiments. We compare our model trained from scratch and pre-trained with different strategies to the current state-of-the-art Set-Transformer model [35]. The results show that our model performs similarly to ST on a supervised setup, but, when pre-trained with additional data, is able to outperform this model by using more data during pre-training.

<table>
<thead>
<tr>
<th>Pre-train</th>
<th>Dataset</th>
<th>p</th>
<th>r</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST [35]</td>
<td>-</td>
<td>0.385</td>
<td>0.366</td>
<td>0.314 ±0.015</td>
</tr>
<tr>
<td>Sup. CON, DIA</td>
<td>0.446</td>
<td>0.356</td>
<td>0.324 ±0.029</td>
<td></td>
</tr>
<tr>
<td>ST [35] (All Features)</td>
<td>-</td>
<td>0.328</td>
<td>0.456</td>
<td>0.328 ±0.011</td>
</tr>
<tr>
<td>Sup. CON, DIA</td>
<td>0.493</td>
<td>0.436</td>
<td>0.332 ±0.017</td>
<td></td>
</tr>
<tr>
<td>Sup. CON, DIA, ALL-MRD</td>
<td>0.457</td>
<td>0.385</td>
<td>0.297 ±0.001</td>
<td></td>
</tr>
<tr>
<td>FATE (ours)</td>
<td>-</td>
<td>0.445</td>
<td>0.435</td>
<td>0.315 ±0.004</td>
</tr>
<tr>
<td>Sup. CON, DIA</td>
<td>0.6175</td>
<td>0.401</td>
<td>0.389 ±0.058</td>
<td></td>
</tr>
<tr>
<td>Sup. CON, DIA, ALL-MRD</td>
<td>0.531</td>
<td>0.444</td>
<td>0.413 ±0.053</td>
<td></td>
</tr>
<tr>
<td>Sup.+MAE CON, DIA, ALL-MRD</td>
<td>0.611</td>
<td>0.550</td>
<td>0.496 ±0.045</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Results for different masking ratios of the input features. We can see that a strong masking ratio decreases performance due to the small number of features of each event.

<table>
<thead>
<tr>
<th>Masking ratio</th>
<th>p</th>
<th>r</th>
<th>$F_1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.75%</td>
<td>0.574</td>
<td>0.389</td>
<td>0.363 ±0.004</td>
</tr>
<tr>
<td>0.5%</td>
<td>0.578</td>
<td>0.472</td>
<td>0.459 ±0.016</td>
</tr>
<tr>
<td>0.25%</td>
<td>0.611</td>
<td>0.550</td>
<td>0.496 ±0.045</td>
</tr>
</tbody>
</table>

6. Conclusion

In this work, we introduce a novel architecture, FATE, that is agnostic to the number and order of features of the input data by learning a generalized feature space using only one feature-agnostic encoder. Further, we propose a pre-training strategy based on MAE which, contrary to standard practices where complete feature vectors are masked during pre-training, masks part of the input features and tries to reconstruct them from the remaining ones. We show that, with this model and pre-training strategy, we are able to leverage new datasets during pre-training and improve current state-of-the-art on FCM data by a large margin.

In the future, we would like to investigate which features are more relevant for feature reconstruction and downstream tasks by analyzing the attention weights of our model, in order to improve future data acquisition. Moreover, we would like to investigate how the quality of the representations is affected by the number of common features during pre-training.

References


7962


[12] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 16000–16009, 2022. 1, 2


[26] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervi-


