Computers in Biology and Medicine 144 (2022) 105314

Contents lists available at ScienceDirect

Computers in Biology and Medicine

journal homepage: www.elsevier.com/locate/compbiomed

# Automated identification of cell populations in flow cytometry data with transformers

Matthias Wödlinger [a,b,*], Michael Reiter [a,b], Lisa Weijler [a], Margarita Maurer-Granofszky [b,c], Angela Schumich [b,c], Elisa O. Sajaroff [d], Stefanie Groeneveld-Krentz [e], Jorge G. Rossi [d], Leonid Karawajew [e], Richard Ratei [f], Michael N. Dworzak [b]

[a] TU Wien, Vienna, Austria
[b] St Anna Children's Cancer Research Institute, Vienna, Austria
[c] Labdia Labordiagnostik GmbH
[d] Cellular Immunology Laboratory, Hospital de Pediatria "Dr. Juan P. Garrahan", Buenos Aires, Argentina
[e] Department of Pediatric Oncology/Hematology, Charité Universitätsmedizin Berlin, Berlin, Germany
[f] Department of Hematology, Oncology and Tumor Immunology, HELIOS Klinikum Berlin-Buch, Berlin, Germany

ARTICLE INFO

*Keywords:*
Multiparameter flow cytometry
Automated gating
Deep learning
Self-attention

ABSTRACT

Acute Lymphoblastic Leukemia (ALL) is the most frequent hematologic malignancy in children and adolescents. A strong prognostic factor in ALL is given by the Minimal Residual Disease (MRD), which is a measure for the number of leukemic cells persistent in a patient. Manual MRD assessment from Multiparameter Flow Cytometry (FCM) data after treatment is time-consuming and subjective. In this work, we present an automated method to compute the MRD value directly from FCM data. We present a novel neural network approach based on the transformer architecture that learns to directly identify blast cells in a sample. We train our method in a supervised manner and evaluate it on publicly available ALL FCM data from three different clinical centers. Our method reaches a median $F_1$ score of $\approx 0.94$ when evaluated on 519 B-ALL samples and shows better results than existing methods on 4 different datasets.

## 1. Introduction

Acute Lymphoblastic Leukemia (ALL) is a malignant disorder of lymphoid progenitor cells. It is the most frequent hematologic malignancy in children and adolescents and treated patients show relapse rates of 15–20% [1]. A means of tracking the progress of treatment is provided by the Minimal Residual Disease (MRD), which is the fraction of remaining leukemic cells (*blast* cells) after therapy. Low MRD values in early stages of treatment have been shown to be powerful predictors for better outcomes [2]. For this reason, the correct assessment of MRD values is an important part of modern treatment methods.

### 1.1. Flow cytometry

Multiparameter Flow Cytometry (FCM) provides a reliable way to obtain MRD values during treatment [3]. In this process, a blood or bone marrow sample of a patient is stained with a specific combination of fluorescence-labelled antibodies that bind to their respective cell antigens. In the Flow cytometer machine, the cells are then illuminated by a selection of lasers that allow the detection and measurement of physical properties (granularity, size) as well as biological properties through detection of the antibodies if attached to respective antigens. The resulting data for a single cell (an *event*), is a collection of measurements of cell surface marker concentrations (see Fig. 1 for an example of FCM data as seen during clinical routine). However, manual analysis of FCM data is time-consuming, subjective and dependent on the operator's experience.

### 1.2. Contribution

To tackle the shortcomings of manual gating several methods have been proposed that allow automated FCM analysis. The structure of FCM data samples however proves to be challenging for neural network-based approaches as these often require data points on a grid (e.g.

* Corresponding author. TU Wien, Vienna, Austria.
*E-mail address:* mwoedlinger@cvl.tuwien.ac.at (M. Wödlinger).

https://doi.org/10.1016/j.compbiomed.2022.105314
Received 14 December 2021; Received in revised form 3 February 2022; Accepted 8 February 2022
Available online 16 February 2022

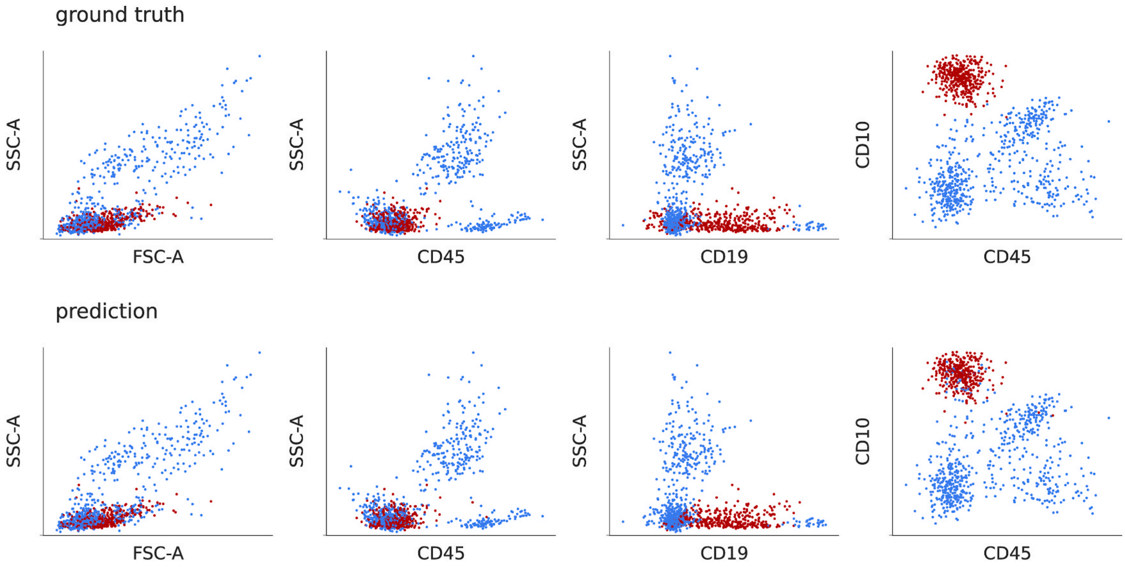

**Fig. 1.** An example prediction (bottom row) of our method and the corresponding manual labeling (top row). Red dots denote blast and blue dots non-blast cells. Every plot shows a different 2-dimensional projection of the same underlying FCM data sample. For this visualization we randomly sampled 5000 cells from a sample from the *bue* dataset. The prediction is from a model trained on *vie14* (see Table 1 for a description of the datasets).

convolutional neural networks for a 2d grid or recurrent neural networks for sequences). Some methods [4,5] circumvent this problem by applying neural networks on single cells instead of samples, however, these approaches can only learn static decision boundaries and are not able to capture global sample information. In this work, we present a novel method for the detection of blast cells and MRD quantification that is capable of capturing long-range information in the full data space by attending to all events in a sample at once[1]. Our method consists of a neural network based on the transformer [6,7] architecture, that learns gating directly from FCM data. This allows fast inference and easy adaption to new data. The remaining paper is structured as follows: After a discussion of the related work in the next section 2, we present our approach in section 3 and show the results in section 4.

## 2. Related work

In manual gating methods, cells are identified based on 2-dimensional projections of the (higher-dimensional) FCM data. Automated methods, on the other hand, can utilize the full parameter space. Typically, these methods aim to assign the correct population to every single cell. This produces an output similar to manual gating. This output can then be used directly in clinical routine (for example for MRD quantification) or as a starting point for further data analysis. We present a selection of related methods for automated FCM analysis and then discuss recent progress in Transformer [6] related neural networks. In particular, the progress with respect to complexity and memory footprint reduction for long sequences is discussed. Being able to process longer sequences is essential for FCM data where samples typically contain $10^5$ to $10^6$ cells.

### *2.1. Automated FCM analysis*

Several works formulate automated FCM analysis as an unsupervised learning problem by adopting non-parametric density estimation or clustering methods [8,9]. A line of research that recently showed good results in both the unsupervised [10–12] and supervised [13,14] setting are Gaussian Mixture Models (GMM). In SWIFT [10] the conventional GMM algorithm is adapted to better detect rare sub-populations; BayesFlow [12] employs a hierarchical Bayesian model were expert knowledge can be incorporated through informative priors [13]. accounts for inter-sample variation with a supervised approach where GMMs are matched to GMMs of a labelled reference dataset. The method is advanced in Ref. [14] where a closed form optimization in the fitting process is introduced. Deep learning has been successfully applied to automated processing of image cell data [15,15], however, apart from imaging FCM applications [16–18], few examples of successful application of deep neural networks to FCM data exist. In Refs. [4,5,19] neural networks based on fully connected layers are presented that work on single events. These methods can only learn fixed decision boundaries to separate biologically meaningful sub-populations. Only recently in Ref. [20] a method has been proposed to circumvent this problem by transforming FCM data to image space and processing it with a learned CNN.

### *2.2. Transformers*

The original Transformer paper [6] introduced a neural network layer that allows capturing of long-range information. In theory, the layer is capable of capturing global information, however, due the complexity of both memory and time being quadratic in the input length $\mathcal{O}(N^2)$ the authors restrict input sequences to 2048 tokens. One solution to this is provided by the Reformer [21]. Here the authors use locality-sensitive hashing to restrict the attention to nearby positions which reduces the time complexity to $\mathcal{O}(N\log N)$. While this results in a similar performance to the original transformer for many tasks, it restricts the attention to the local neighbourhood. Another line of research is given by models [22,23] that aim to achieve linear complexity $\mathcal{O}(N)$ by approximating the Softmax function in the self-attention layer with a kernel which allows factorizing the computation of the attention matrix. Most related to our application are Set Transformers [7], a type of transformer architecture specifically designed for set inputs where the order of inputs is not relevant. These networks achieve linear complexity with the sequence length $\mathcal{O}(N)$ by applying the idea of inducing points from the theory of Gaussian processes.

## 3. Methods

We start with a brief discussion of the structure of FCM data and then

[1] Our code is available on Github: https://github.com/mwoedlinger/flowformer.

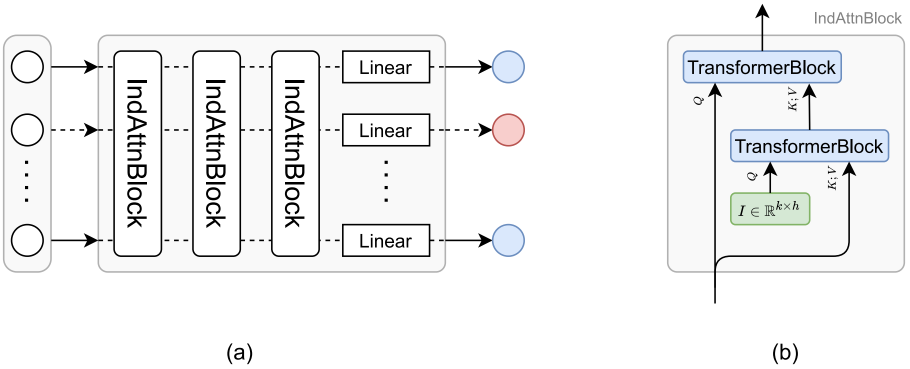

**Fig. 2.** (a) Our network architecture. The input consists of a sample, represented by the event matrix. For every input cell we predict a binary classification label (indicated with colors blue and red). (b) The induced attention block from eq. (4) as introduced in Ref. [7] with the learnable parameters $I$ in green and the TransformerBlock from (3) in blue.

give a detailed description of the network architecture.

### 3.1. FCM data

A single sample is represented by a matrix $E \in \mathbb{R}^{N \times m}$ (the *event matrix*), where $N$ denotes the number of cells in the sample (typically $10^5 - 10^6$, the exact value for $N$ is different for separate samples) and $m$ denotes the number of markers (typically 10–20 in our case, the exact number depends on the antibodies used). While both the number of cells $N$ and the number of markers $m$ can vary between different samples, there is a set of markers present in every sample (the *base panel*). We restrict our method to the markers in this base panel and ignore measurements for other markers, i.e. we keep $m$ fixed and discard measurements of non-base-panel markers. For every index $n \in \{1, \dots, N\}$, $E_n \in \mathbb{R}^m$ is a quantitative representation of the surface markers present on the cell $n$. Ignoring the ordering of cells induced by the FCM machine (i.e. we represent a sample as a set of vectors instead of a sequence) a sample can also be viewed as a bag of features (where a feature is the marker measurement vector for a single cell).

### 3.2. Network architecture

The absence of a low dimensional grid structure (as is typically the case for domains where neural networks excel, like text, where the data is structured on a one-dimensional grid of images that form a two-dimensional grid) makes a direct application of typical neural networks difficult. Self-attention based networks that recently have dominated Natural Language Processing (NLP) related tasks can learn features from sets of embedding vectors (when one ignores the positional embedding that is typically used in NLP problems). However, the memory requirements of such models grow quadratically in the set size [23] which prevents a direct application to FCM data. To see this, consider the multi-head attention block

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}(Q^T K)V. \tag{1}$$

If $Q$ and $K$ derive from the same set of inputs, which is the case for self-attention, the $Q^T K$ multiplication is quadratic in the size set. However, recently an adaption of the self-attention layer has been proposed that reduce the memory requirements from a quadratic growth to linear growth in the set size. In Lee et al. [7] the standard multi-head self--attention block is replaced by a two-step procedure. For a given input set $X \in \mathbb{R}^{N \times m}$ and $k \in \mathbb{N}$ we initialize a set of parameters $I \in \mathbb{R}^{k \times h}$. Then

1. latent features $h \in \mathbb{R}^{k \times h}$ are extracted by performing an attention operation between the set of learnable parameters $I \in \mathbb{R}^{k \times h}$ as query and the input set $X$ as key and value input.
2. The resulting hidden features $h$ are used as key and value input for a second attention computation with the input $X$ acting as the query.

We denote this block from now on as Induced Attention Block (IndAttnBlock). This breaks the original $\mathcal{O}(N^2)$ operation into two $\mathcal{O}(N \cdot k)$ operations which circumvents the problem of quadratic complexity (with $k \ll N$ held constant). The latent features are capable of capturing global sample information and the full operation has been proven to be permutation invariant [7] which justifies the application to set data. We want to point out, that, while the network as a whole is permutation invariant, the order of samples in a single forward pass is not mixed up. This is allows us to identify binary classifications in the output with cells in the input. With the multihead attention block from [6].

$$\mathrm{AttnBlock}(X, Y) = \mathrm{LayerNorm}(X + \mathrm{Attn}(X, Y, Y)) \tag{2}$$

where

$$\mathrm{TransfBlock}(X, Y) = \mathrm{LayerNorm}(\mathrm{AttnBlock}(X, Y) + FF(\mathrm{AttnBlock}(X, Y))) \tag{3}$$

and the Layernorm from Ref. [24], the induced attention block (see Fig. 2, b) can be written as[2]

$$\mathrm{IndAttnBlock}(X) = \mathrm{TransfBlock}(X, \mathrm{TransfBlock}(I, X)).^2 \tag{4}$$

Using the induced attention block as a building block, we propose a novel neural network that processes a single sample of FCM data in a single forward pass. Our network (see Fig. 2, a) is defined as a sequence of three IndAttnBlocks with a row-wise linear layer on top, trained with binary cross-entropy loss. We do not apply a separate embedding step as in other transformer-based methods but apply our model directly to FCM features (in particular without any positional embedding). We set the number of induced points to $m = 16$, the latent embedding dimension to $d = 32$ and the number of attention heads to 4 for all three layers. The resulting model is comparatively lightweight with only 27 657 parameters and can process ≈ 150 samples/s on an NVIDIA GeForce Titan X[3].

## 4. Experiments

We start this section with a brief discussion of the data in subsection

[2] The first Transformer block can be understood as a *HopfieldPooling* layer from Ref. [25] while the second block performs the computation of the Layer *Hopfield*.

[3] Only counting the model forward pass, i.e. ignoring time needed for data loading.

**Table 1**
Description of the FCM datasets used for experiments.

| Name | City | Years | # |
|---|---|---|---|
| vie14 | Vienna | 2009–2014 | 200 |
| vie20 | Vienna | 2015–2020 | 319 |
| vie | Vienna | 2009–2020 | 519 |
| bln | Berlin | 2015 | 72 |
| bue | Buenos Aires | 2016–2017 | 65 |

4.1 and the training in subsection 4.2, followed by the evaluation in subsection 4.3.

### 4.1. Data

We evaluate our method on publicly available data[4] from three different clinical centers. The data consists of bone marrow samples of pediatric patients with B-ALL on day 15 after induction therapy. For all samples ground truth information acquired by manual gating is available for blast and non-blast cells. Table 1 contains an overview of the datasets.

#### 4.1.1. Vienna

The Vienna dataset has been collected at the St. Anna Children's Cancer Research Institute (CCRI) from 2009 to 2020 with a LSR II flow cytometer (Becton Dickinson, San Jose, CA) and FACSDiva v6.2. We denote this dataset with *vie*, it contains 519 samples. We extract two disjunct datasets from these samples:

- *vie14:* This dataset contains 200 samples collected between 2009 and 2014. It is identical to the *vie* dataset in Ref. [14]. The samples were stained using a conventional seven-colour drop-in panel ("B7") consisting of the liquid fluorescent reagents: CD20- FITC/ CD10-PE/ CD45-PerCP/ CD34-PE-Cy7/ CD19-APC/ CD38-Alexa-Fluor700 and SYTO 41.
- *vie20:* This dataset contains 319 samples collected between 2016 and 2020. The samples were stained using dried format tubes ("ReALB", DuraClone$^{TM}$, Beckman Coulter, Brea, CA ) consisting of the fluorochrome-conjugated antibodies CD58-FITC/ CD34-ECD/ CD10-PC5.5/ CD19-PC7/ CD38-APC-Alexa700/ CD20-APC-Alexa750/ CD45-Krome Orange plus drop-in SYTO 41.

#### 4.1.2. Berlin

The bln Dura [14] (from now on referred to as *bln*) dataset contains 72 samples collected in 2016 at Charité Berlin. These samples were recorded with a Navios flow cytometer (Beckmann Coulter, Brea, CA) and assessed by 8-colour multiparameter FCM ("B8") using a customized dried format tube (DuraClone™, Beckman Coulter, Brea, CA) consisting of the seven fluorochrome-conjugated antibodies CD58-FITC/ CD10-PE/ CD34-PerCPCy5.5/ CD19-PC7/ CD38-APC/ CD20-APC-Alexa750/ CD45-Krome Orange plus drop-in SYTO 41.

#### 4.1.3. Buenos Aires

The bue Dura [14] (from now on referred to as *bue*) dataset consists of 65 samples collected between 2016 and 2017 at the Garrahan Hospital in Buenos Aires. The staining panel is identical to the bln Dura set (based on DuraClone$^{TM}$ cocktail tube; "B8", Beckman Coulter, Brea, CA). The data has been acquired on a FACSCanto II (Becton Dickinson, San Jose, CA) with FACSDiva v8.0.1.

### 4.2. Training

We conduct a thorough investigation into cross platform compatibility of our method. For this we train separate models for all four datasets discussed in the subsection above and test the models on every other dataset from Table 1. The validation datasets for a specific experiment consist of all other datasets (example: for vie14 train and bue test sets we use vie20 and bln as validation data). Additionally we show that our method can be trained on as little as 10 samples while still reaching competitive results. For these experiments we only use 10 samples for validation. For all experiments we use the Adam optimizer [26] with an initial learning rate of $1e-3$ and a Cosine Annealing scheduler [27] with 10 iterations and a minimal learning rate of $2e-4$. We train for 100 epochs with a batch size of 1 and evaluate on the test with the best checkpoint as measured by the average $F_1$-score on the validation set. We implement our method in Pytorch 1.7.1 [28] and use the pre-implemented optimizer and scheduler. Due to the small model size, a training run until convergence on a single NVIDIA GeForce Titan X takes only $\approx 2-8$ hours, depending on the dataset size.

**Table 2**
The experimental results evaluated with precision (p), recall (r), average $F_1$-score (avg $F_1$) and median $F_1$-score (med $F_1$). For every experiments we train on all samples from a certain dataset and test on all samples from a different dataset. We compare our method to Reiter et al. [14]. Boldface values indicate the best performing method for a specific train/test dataset combination.

| train | test | p | r | avg $F_1$ | med $F_1$ | med $F_1$ [14] |
|---|---|---|---|---|---|---|
| vie | vie | 0.81 | 0.83 | 0.81 | 0.94 | - |
| bln | bue | 0.63 | 0.84 | 0.66 | **0.87** | *0.68* |
| bln | vie14 | 0.77 | 0.83 | 0.77 | **0.90** | *0.35* |
| bln | vie20 | 0.79 | 0.77 | 0.74 | **0.87** | *0.48* |
| bue | bln | 0.56 | 0.92 | 0.62 | **0.77** | *0.5* |
| bue | vie14 | 0.76 | 0.88 | 0.79 | **0.90** | *0.84* |
| bue | vie20 | 0.79 | 0.74 | 0.72 | **0.88** | *0.86* |
| vie14 | bln | 0.78 | 0.82 | 0.75 | **0.9** | *0.81* |
| vie14 | bue | 0.82 | 0.81 | 0.78 | **0.95** | *0.84* |
| vie14 | vie20 | 0.81 | 0.74 | 0.73 | **0.89** | *0.86* |
| vie20 | bln | 0.64 | 0.87 | 0.66 | **0.81** | *0.25* |
| vie20 | bue | 0.82 | 0.69 | 0.71 | **0.86** | *0.81* |
| vie20 | vie14 | 0.82 | 0.69 | 0.71 | 0.86 | ***0.89*** |

### 4.3. Results

For the first set of experiments we train on data from a specific laboratory and test on data from a different laboratory. Exceptions are made for the data from Vienna that we split in 2 sets of data: *vie14* (collected between 2009 and 2014) and *vie20* (collected between 2015 and 2020). We denote experiments with *train/test*, where *train* stands for the training set and *test* stands for the test set (for example, vie14/bln means we train on vie14 and test on bln). An exception being the *vie* experiment where we combine the vie14 and vie20 sets to a single dataset which we randomly split into train, validation and test set. The results of our experiments are listed in Table 2. To assess the quality of the results we compute average precision (p), average recall (r), average $F_1$ scores (avg $F_1$) and median $F_1$ scores (med $F_1$):

$$p = \frac{\mathrm{tp}}{\mathrm{tp}+\mathrm{fp}}, \quad r = \frac{\mathrm{tp}}{\mathrm{tp}+\mathrm{fn}}, \quad F_1 = 2\frac{p \cdot r}{p+r} \tag{5}$$

where blast cells are "positive" and non-blasts "negative". For samples without blasts or only very few blast cells (see the left-most region in Fig. 4, In particular the first 10 samples, where no blasts are present) the $F_1$-score is not a good measure of performance because classification mistakes of single cells can have a significant effect on the $F_1$-score (in particular, for samples with zero blasts, wrongly classifying a cell as a blast cell results in a $F_1$-score of 0), that is not reflected in clinical significance. We find that because of these reasons, the median $F_1$-score is a better measure for model performance than the average $F_1$-score.

We compare our method to the GMM based model described in Reiter et al. [14] which we evaluated on the vie14, vie20, bln and bue datasets. The complete set of results for the conducted experiments can

[4] flowrepository.org.

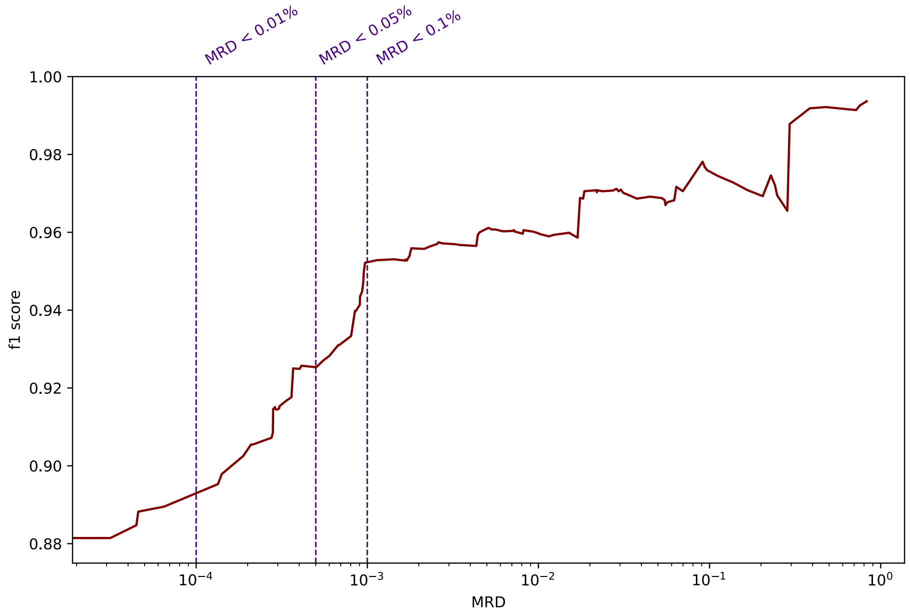

**Fig. 3.** Running average of $F_1$ scores against the ground truth MRD values, i.e. for a given MRD value x the line shows the average of $F_1$ scores of all samples within the vie test set with an MRD value larger or equal to x. Due to the logarithmic scale, the 11 samples in the vie test set with MRD values of 0 are not shown which explains the missmatch between the lowest running average of 0.88 and the mean over the whole test set of 0.81.

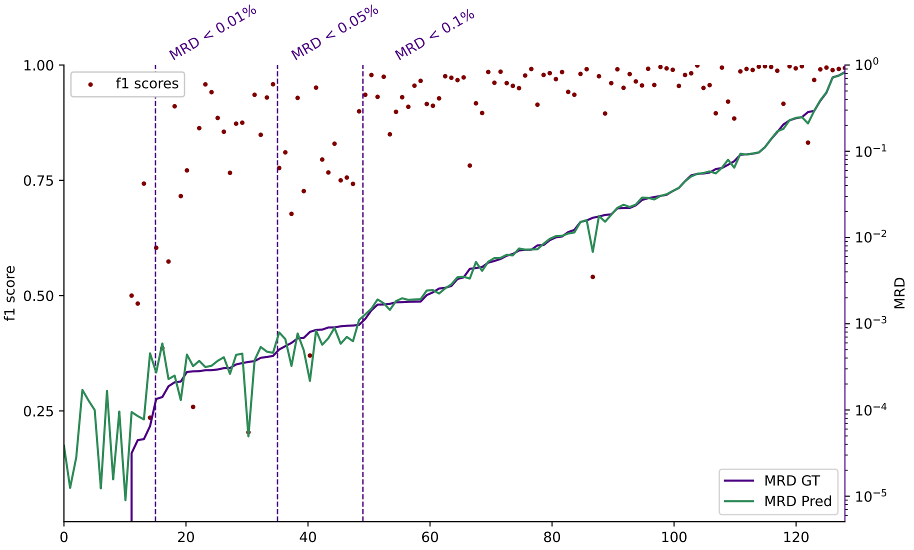

**Fig. 4.** $F_1$ scores (red dots), ground truth MRD values (purple lines) and predicted MRD values (green line) for the vie test set. The x-axis shows the sample index when sorted by GT MRD.

be seen in Table 2. We outperform the existing approach [14] in all train/test combinations except for vie20/vie14 where we reach comparable results. Our method is significantly faster with inference times of 5 ms for our method vs 3000 ms for the GMM based approach [14]. We consistently reach median $F_1$ scores $\geq 0.86$ with the only exceptions being bue/bln with 0.77 and vie20/bln with 0.81. For bue/bln in particular our method only reaches a median $F_1$ score of 0.77 with a precision of 0.56 and recall of 0.92 indicating that for sufficiently different data sources the performance can degrade. However, adding 5 random samples from the test set to the training set and testing on the remaining samples improves median $F_1$ score, precision and recall to 0.87, 0.7 and 0.91 indicating that if a small number of labelled data is available cross-laboratory performance of our method can be improved significantly. In general we find that when measured with respect to the $F_1$ score, our method performs better for samples with larger MRD values. Fig. 3 shows the average $F_1$ score for all samples with an MRD value above the threshold given by the value on the x axis for the vie test set. Samples with low $F_1$ score are predominantly those with a smaller MRD i.e. lower count of blast cells. For low MRD values our method tends to overestimate the true value more often than it underestimates it. This can be seen in Fig. 5 where the ground truth MRD is plotted against the predicted MRD. A different visualization is given by Fig. 4 where for every sample the true MRD, the predicted MRD and the $F_1$ score are given.

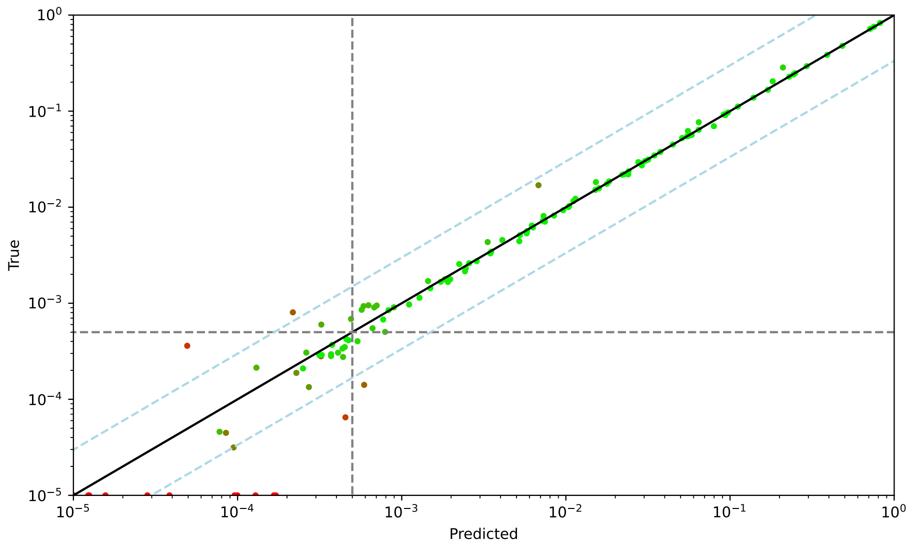

**Fig. 5.** $F_1$ scores and predicted MRD values for the test set of the vie experiments. Every dot represents a single sample, with the colour donating the $F_1$ score (colors going from red = 0.0 to green = 1.0. The dashed lines correspond to MRD values of $5e-4$ which is the lower necessary resolution for patient stratification according to the current international therapy trials of the allied study groups of the iBFM consortium. Predictions that are within the range of either less than 3 times or more than 1/3 of the true MRD are considered as acceptable (correct) predictions [29].

## 5. Conclusion

In this work, we proposed a novel method for automated identification of cell populations and used it for the detection of blast cells in B-ALL FCM data. Our method is based on a lightweight (27 657 parameters) neural network that allows fast (≈ 150 samples/s) processing of samples with $10^5 - 10^6$ cells on a NVIDIA GeForce GTX TITAN X. We trained the model in a supervised manner on as few as 65 samples of data from three different sources and showed that our method is capable of generalizing to unseen data. Our method is different from existing approaches that utilize neural networks for automated FCM analysis [4, 5] in that we make use of self-attention layers that allow the network to attend to all cells in the sample at once instead of processing every cell independently. For future work, we argue that data augmentation methods that capture device differences (for example as proposed in Ref. [30] for mass spectroscopy) would help improve generalization (like the performance drop for the vie20/bln experiment).

## Declarations of competing interest

Michael N. Dworzak received payments for travel, accommodation or other expenses from Beckman-Coulter. The other authors declare no competing financial interests.

## Acknowledgement

We thank Dieter Printz (FACS Core Unit, CCRI) for flow-cytometer maintenance and quality control, as well as Daniela Scharner and Susanne Suhendra-Chen (CCRI), Jana Hofmann (Charité), Marianne Dunken (HELIOS Klinikum), Marianela Sanz, Andrea Bernasconi, and Raquel Mitchell (Hospital Garrahan) for excellent technical assistance. We are indebted to Melanie Gau, Roxane Licandro, Florian Kleber, Paolo Rota and Guohui Qiao (all from TU Vienna) for valuable contributions to the AutoFLOW project. We thank Markus Kaymer and Michael Kapinsky (both from Beckman Coulter Inc.) for kindly assisting in the provision of customized DuraClone$^{Tm}$ tubes for this study as designed by the authors. Notably, Beckman Coulter Inc. did not have any influence on study design, data acquisition and interpretation, or manuscript writing. The study has received funding from the European Union's H2020 Research and Innovation Program through Grant number 825749 "CLOSER: Childhood Leukemia: Overcoming Distance between South America and Europe Regions", the Vienna Business Agency under grant agreement No 2841342 (Project MyeFlow) and by the Marie Curie Industry Academia Partnership & Pathways (FP7-MarieCurie-PEOPLE-2013-IAPP) under grant no. 610872 to project "AutoFLOW" to MND. The authors acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme.

## References

[1] C.-H. Pui, L.L. Robison, A.T. Look, Acute lymphoblastic leukaemia, Lancet 371 (9617) (2008) 1030–1043.

[2] D. Campana, Minimal residual disease in acute lymphoblastic leukemia, 2010, Hematology (1) (2010) 7–12.

[3] M.N. Dworzak, G. Fröschl, D. Printz, G. Mann, U. Pötschger, N. Mühlegger, G. Fritsch, H. Gadner, Prognostic significance and modalities of flow cytometric minimal residual disease detection in childhood acute lymphoblastic leukemia, Blood, J. Am. Soc. Hematol. 99 (6) (2002) 1952–1958.

[4] J. Scheithe, R. Licandro, P. Rota, M. Reiter, M. Diem, M. Kampel, Monitoring acute lymphoblastic leukemia therapy with stacked denoising autoencoders, in: Computer Aided Intervention and Diagnostics in Clinical and Medical Images, Springer, 2019, pp. 189–197.

[5] R. Licandro, T. Schlegl, M. Reiter, M. Diem, M. Dworzak, A. Schumich, G. Langs, M. Kampel, Wgan latent space embeddings for blast identification in childhood acute myeloid leukaemia, in: 24th International Conference on Pattern Recognition (ICPR), IEEE, 2018, pp. 3868–3873, 2018.

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 6000–6010.

[7] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, Y.W. Teh, Set transformer: a framework for attention-based permutation-invariant neural networks, in: International Conference on Machine Learning, PMLR, 2019, pp. 3744–3753.

[8] T. Sörensen, S. Baumgart, P. Durek, A. Grützkau, T. Häupl, immunoclust—an automated analysis pipeline for the identification of immunophenotypic signatures in high-dimensional cytometric datasets, Cytometry 87 (7) (2015) 603–615.

[9] N. Aghaeepour, G. Finak, H. Hoos, T.R. Mosmann, R. Brinkman, R. Gottardo, R. H. Scheuermann, Critical assessment of automated flow cytometry data analysis techniques, Nat. Methods 10 (3) (2013) 228–238.

[10] I. Naim, S. Datta, J. Rebhahn, J.S. Cavenaugh, T.R. Mosmann, G. Sharma, Swift—scalable clustering for automated identification of rare cell populations in large, high-dimensional flow cytometry datasets, part 1: Algorithm design, Cytometry 85 (5) (2014) 408–421.

[11] M. Dundar, F. Akova, H.Z. Yerebakan, B. Rajwa, A non-parametric bayesian model for joint cell clustering and cluster matching: identification of anomalous sample phenotypes with random effects, BMC Bioinf. 15 (1) (2014) 1–15.

[12] K. Johnsson, J. Wallin, M. Fontes, Bayesflow: latent modeling of flow cytometry cell populations, BMC Bioinf. 17 (1) (2016) 1–16.

[13] M. Reiter, P. Rota, F. Kleber, M. Diem, S. Groeneveld-Krentz, M. Dworzak, Clustering of cell populations in flow cytometry data using a combination of Gaussian mixtures, Pattern Recogn. 60 (2016) 1029–1040.

[14] M. Reiter, M. Diem, A. Schumich, M. Maurer-Granofszky, L. Karawajew, J.G. Rossi, R. Ratei, S. Groeneveld-Krentz, E.O. Sajaroff, S. Suhendra, et al., Automated flow cytometric mrd assessment in childhood acute b-lymphoblastic leukemia using supervised machine learning, Cytometry 95 (9) (2019) 966–975.

[15] M.S. Iqbal, I. Ahmad, L. Bin, S. Khan, J.J. Rodrigues, Deep learning recognition of diseased and normal cell representation, Trans. Eng. Telecommun. Technol. 32 (7) (2021), e4017.

[16] N. Nissim, M. Dudaie, I. Barnea, N.T. Shaked, Real-time stain-free classification of cancer cells and blood cells using interferometric phase microscopy and machine learning, Cytometry Part A 99 (2021) 511–523.

[17] P. Eulenberg, N. Köhler, T. Blasi, A. Filby, A.E. Carpenter, P. Rees, F.J. Theis, F. A. Wolf, Reconstructing cell cycle and disease progression using deep learning, Nat. Commun. 8 (1) (2017) 1–6.

[18] Y. Li, B. Cornelis, A. Dusa, G. Vanmeerbeeck, D. Vercruysse, E. Sohn, K. Blaszkiewicz, D. Prodanov, P. Schelkens, L. Lagae, Accurate label-free 3-part leukocyte recognition with single cell lens-free imaging flow cytometry, Comput. Biol. Med. 96 (2018) 147–156.

[19] H. Li, U. Shaham, K.P. Stanton, Y. Yao, R.R. Montgomery, Y. Kluger, Gating mass cytometry data by deep learning, Bioinformatics 33 (21) (2017) 3423–3430.

[20] M. Zhao, N. Mallesh, A. Höllein, R. Schabath, C. Haferlach, T. Haferlach, F. Elsner, H. Lüling, P. Krawitz, W. Kern, Hematologist-level classification of mature b-cell neoplasm using deep learning on multiparameter flow cytometry data, Cytometry 97 (10) (2020) 1073–1080.

[21] N. Kitaev, L. Kaiser, A. Levskaya, Reformer: the efficient transformer, in: International Conference on Learning Representations, 2019.

[22] K. Choromanski, V. Likhosherstov, D. Dohan, X. Song, A. Gane, T. Sarlos, P. Hawkins, J. Davis, A. Mohiuddin, L. Kaiser, et al., Rethinking Attention with Performers, 2020 arXiv preprint arXiv:2009.14794.

[23] A. Katharopoulos, A. Vyas, N. Pappas, F. Fleuret, Transformers are rnns: fast autoregressive transformers with linear attention, in: International Conference on Machine Learning, PMLR, 2020, pp. 5156–5165.

[24] J.L. Ba, J.R. Kiros, G.E. Hinton, Layer normalization, Stat 1050 (2016) 21.

[25] H. Ramsauer, B. Schäfl, J. Lehner, P. Seidl, M. Widrich, L. Gruber, M. Holzleitner, T. Adler, D. Kreil, M.K. Kopp, et al., Hopfield networks is all you need, in: International Conference on Learning Representations, 2020.

[26] D.P. Kingma, J. Ba, Adam: a method for stochastic optimization, 2014 arXiv preprint arXiv:1412.6980.

[27] I. Loshchilov, F. Hutter, Sgdr: Stochastic Gradient Descent with Warm Restarts, 2016 arXiv preprint arXiv:1608.03983.

[28] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, et al., Pytorch: an imperative style, high-performance deep learning library, Adv. Neural Inf. Process. Syst. 32 (2019) 8026–8037.

[29] M.N. Dworzak, G. Gaipa, R. Ratei, M. Veltroni, A. Schumich, O. Maglia, L. Karawajew, A. Benetello, U. Pötschger, Z. Husak, et al., Standardization of flow cytometric minimal residual disease evaluation in acute lymphoblastic leukemia: multicentric assessment is feasible, Cytometry Part B: Clinical Cytometry: The Journal of the International Society for Analytical Cytology 74 (6) (2008) 331–340.

[30] U. Shaham, K.P. Stanton, J. Zhao, H. Li, K. Raddassi, R. Montgomery, Y. Kluger, Removal of batch effects using distribution-matching residual networks, Bioinformatics 33 (16) (2017) 2539–2546.