2022 12th International Congress on Advanced Applied Informatics (IIAI-AAI) | 978-1-6654-9755-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/IIAIAAI55812.2022.00091

Evaluating Contrastive Learning for Fine-grained Reading Detection

Md. Rabiul Islam
Graduate School of Engineering
Osaka Prefecture University
Sakai, Japan
rabiul.apece@bsmrstu.edu.bd

Andrew W. Vargo
Graduate School of Informatics
Osaka Metropolitan University
Sakai, Japan
awv@m.cs.osakafu-u.ac.jp

Motoi Iwata
Graduate School of Informatics
Osaka Metropolitan University
Sakai, Japan
imotoi@omu.ac.jp

Masakazu Iwamura
Graduate School of Informatics
Osaka Metropolitan University
Sakai, Japan
masa.i@omu.ac.jp

Koichi Kise
Graduate School of Informatics
Osaka Metropolitan University
Sakai, Japan
kise@omu.ac.jp

***Abstract*—The major drawback of deep learning (DL) algorithms is the necessity of large and labeled data sets in order to achieve peak performance. A DL technique that can overcome this constraint is self-supervised learning, which is applied as non-contrastive self-supervised learning (SSL) and contrastive self-supervised learning (contrastive learning). This paper evaluates a contrastive learning method, which is called simple framework for contrastive learning of visual representations (SimCLR), for the task of fine-grained reading detection. We employ in-the-wild electrooculography (EOG) data sets that describe the eye movement behaviors to evaluate the SimCLR method and compare it against the SSL and pure supervised methods. The results show a maximum performance gain of 3.02 and 3.96 percentage points compared to the SSL and pure supervised methods, respectively, over an equal amount of training data. In addition, the SimCLR method shows a data efficiency of about 80%. The obtained results show a direction for system designers and researchers to handle the lack of large-sized labeled data issues in developing DL models that help to improve user reading habits through eye movement behaviors.**

***Index Terms*—self-supervised learning, contrastive learning, SimCLR, reading detection**

I. Introduction

Classical machine learning algorithms suffer from poor performance in handling high-dimensional data sets. In addition, noisy data, including that which is collected in-the-wild, make the applicability of these algorithms inappropriate. An increasingly popular algorithm is deep learning (DL), which enables simultaneous feature extraction and model creation. It has solved many challenges in various fields [1], [2]. Among all these successes, the major drawback is that large-sized labeled data sets are needed for pure supervised DL algorithms to attain performance at the level of service that is desirable. A large-sized labeled data set is always difficult to produce because of labor, cost, and similar issues. Large-sized labeled data sets are therefore scarce, and it is necessary to devise alternative strategies to handle such issues.

Researchers have employed different techniques, including self-supervised learning [3], [4], to tackle the lack of large-sized labeled data issues. Self-supervised learning works by pre-training the model followed by supervised training for the downstream task. Self-supervised learning is adopted in two different ways, non-contrastive self-supervised learning (SSL) and contrastive self-supervised learning (contrastive learning) [5]. In contrastive learning, data augmentation is used to pre-train the model by employing unlabeled data by calculating contrastive loss [6]. The simple framework for contrastive learning of visual representations (SimCLR) [7] is a contrastive learning method proposed in the computer vision domain. In activity recognition, researchers have also adopted the SimCLR method with effective performance for tackling the lack of large-sized labeled data issues by employing simple signal transformations for data augmentation [8], [9].

Previous work of SimCLR on activity recognition is within the category of "physical activity recognition", which means recognizing human physical activities by employing sensors that are worn on the body and producing data from physical movement. Another activity recognition is "cognitive activity recognition", which is recognizing activities of a person that are relating to the mental processes by capturing biological signals [10]. Reading is one such cognitive activity.

In the cognitive activity recognition tasks which center around reading analysis, such as reading detection, reading quality classification, and read word count, the lack of large-sized labeled data is also problematic because of privacy and other issues in data recording. Researchers also made steps to tackle such issues in reading analysis, including the adoption of SSL [10]. However, we do not know the usefulness of contrastive learning in this field and if different signal transformations for data augmentation will be more effective.

To answer these questions, we take a fine-grained reading

This work was supported in part by JST Trilateral AI Research (JPMJCR20G3), JSPS Grant-in-Aid for Scientific Research (20H04213, 20KK0235).

978-1-6654-9755-8/22/$31.00 ©2022 IEEE
DOI 10.1109/IIAIAAI55812.2022.00091

detection task that is the classification of the following four classes: reading Japanese horizontal (JH), Japanese vertical (JV), English (ENG) texts, and not reading (NR) anything, as a surrogate task of reading analysis and evaluate the SimCLR method employing seven signal transformations with combinations of twos for data augmentation. The Japanese writing system follows two conventions, horizontal and vertical. The horizontal writing moves from left to right with multiple downward rows, but there have no spaces between words. On the other hand, the vertical writing moves from top to bottom with multiple columns from right to left [11]. This fine-grained reading detection is very useful because the classes JV, ENG, and JH suggest reading newspapers or novels, technical materials, and other types of materials, respectively. We evaluate the SimCLR method by employing in-the-wild electrooculography (EOG) data sets that describe the user eye movement behaviors and compare it against the SSL and pure supervised (supervised) baseline methods.

The results show that the SimCLR method outperforms the SSL and supervised baselines for a significant number of signal transformation pairs for a wide range of 100% to 20% of available labeled data. Moreover, the SimCLR method shows a maximum performance gain of 3.02 and 3.96 percentage points for an equal number of labeled data used compared to the SSL and supervised baselines, respectively. The SimCLR method also shows a data efficiency of about 80% against both baselines. It means that the same performance was obtained for the SimCLR method with 20% of the data and baselines with 100% of the data. The results will show directions to apply contrastive learning to pursue the best performance based on the amount of available labeled data that will make it practical to get accurate user reading behaviors for giving feedback to motivate and improve reading habits.

II. Related Work

The self-supervised learning, devised to handle the lack of large-sized labeled data issues, is an intermediate between supervised and unsupervised learning that eliminates the need for human interaction in generating labels [12], [13]. It draws labels itself to formulate a pseudo task in non-contrastive and contrastive ways [5]. In both techniques, the aim is to pre-train the model by solving the pseudo task to learn data representations. The non-contrastive technique arranges a pseudo task such as classification by generating pseudo labels [4], solving jigsaw puzzles [13], repairing images [14], and patch alignment of images [15]. Lately, the contrastive learning [16], [17] technique has attracted much attention by enhanced performances. This technique teaches the model to identify positive data samples, coming from similar distribution and negative data samples, coming from dissimilar distribution. The model, therefore, learns the features of the input data by solving a contrastive pseudo task of maximizing agreement, that is, minimizing the distance between positive data samples and maximizing it for negative data samples [6].

Contrastive learning has been performed in several ways [16], and recently, Chen et al. [7], [18] proposed a contrastive learning framework called SimCLR in the computer vision domain, and the authors reported a significant improvement against SSL and supervised pipelines. The SimCLR method generates positive pairs from the same data samples by applying data augmentation and then calculates contrastive loss [6] in latent space by contrasting these augmented views against other data samples. It trains the feature extractor to be agnostic against the data augmentation. The SimCLR method has also been explored in the physical activity recognition and health by recent work [8], [9], [19], [20] using sensory data to tackle the lack of large-sized labeled data issues where authors employed signal transformations for data augmentation.

Researchers conducted reading analysis that gives us useful information on reading behaviors for various purposes to assist readers [21]–[23]. The lack of large-sized labeled data issues in this field forces researchers to carry out most of the studies by employing classical machine learning, except for a tiny portion employing DL algorithms [24]. The application of SSL for reading detection has shown ways to tackle the lack of large-sized labeled data issues [10]. Contrastive learning shows a potential solution to adopt DL in reading detection using small-sized labeled data, but it has been as of yet unexplored. This study, therefore, aimed to evaluate the SimCLR method for reading detection.

III. Method

We evaluate the SimCLR method for reading detection, with necessary changes for adoption, which consists of two steps: SimCLR pre-training and target task training, as shown in Fig. 1. Reading detection differentiates the periods of reading and not reading. We implement the convectional reading detection as a classification task where at first, we divide the user activities into short data segments and then classify them into pre-defined fine-grained classes. We design the fine-grained classes as reading JH, JV, ENG texts, and NR.

The SimCLR pre-training, as shown in the upper section of Fig. 1, is conducted using the unlabeled EOG data. SimCLR introduces a constraint based on data augmentation applied to unlabeled data; the training of SimCLR is based on the constraint that the features extracted from two augmented data originating from the same data should be similar, whereas those originating from different data should be dissimilar. Therefore, the unlabeled time-series EOG data is divided into short data segments. The data segments are then grouped into 1024 sized batches. To each batch (b), we apply a pair of signal transformations twice with different random parameters to generate two slightly dissimilar copies ($\dot{b}$ and $\ddot{b}$) of the original batch. The augmented batches are fed to an encoder that generates two output vectors ($\dot{h}$ and $\ddot{h}$) which are then fed to a projection head that generates two feature vectors ($\dot{z}$ and $\ddot{z}$). From these two feature vectors, we calculate NT-Xent contrastive loss [7] that maximizes the agreement, maximizing the similarity between augmented data segments originating from the same data segment and minimizing the similarity between augmented data segments originating from the different data segments.

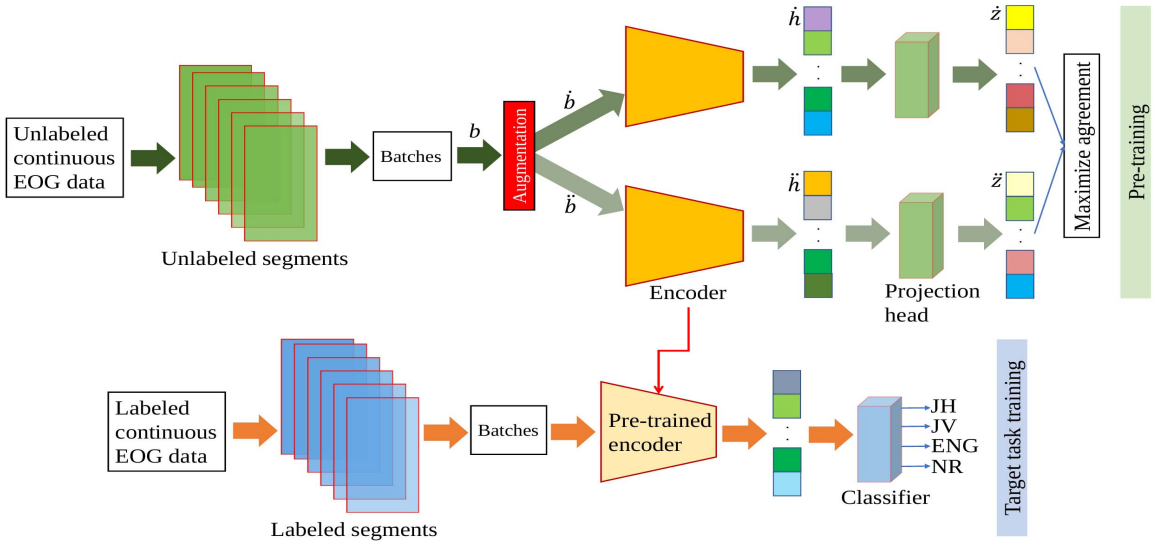

Fig. 1: The SimCLR method, which consists of the pre-training, comprises four basic components of data augmentation, encoder, projection head, and maximizing agreement based on contrastive loss, followed by a target task training.

We use seven signal transformations [8], [10] to create signal transformation pairs that are scaling: multiplied the data segment values by a random number generated from a normal distribution with mean 1 and standard deviation 0.1, noise addition: added random Gaussian noise with mean 0 and standard deviation -0.05 to the data segment, negate: inverted the data segment values by multiplying with -1, time-flip: reversed the time direction of the entire data segment, channel shuffle: randomly permuted the data segment channels, permutation: applied random permutation along the time axis, and time-warp: stretched and warped the data segment.

The encoder consists of four 1D CNN layers where the number of filters in each CNN layer is 16, 32, 64, and 96, respectively, with a kernel size of 32, 24, 16, and 8, respectively. Each CNN layer is followed by a dropout layer. After the last dropout layer, we add a Global max-pooling layer. The projection head consists of three dense layers with 128, 64, and 32 units, respectively. We use relu and Adam as the activation function and optimizer, respectively. In Fig. 1, we repeat the same encoder and projection head for better understanding, although both share the same parameters.

After SimCLR pre-training, the next step is the target task training, as shown in the lower section of Fig. 1. The pre-trained encoder is fine-tuned and then re-trained by removing the projection head and adding a classifier consisting of a dense layer of 4 units with linear activation. We use segmented labeled EOG data for the target task training. We use hyper-parameters the same as the SimCLR pre-training.

IV. Data Sets

We use eye movements data recorded employing an eye-wear JINS-MEME glasses, as shown in Fig. 2. Although JINS-MEME carries EOG, accelerometer, and gyroscope sensors, we use data only for the EOG sensor that records data as

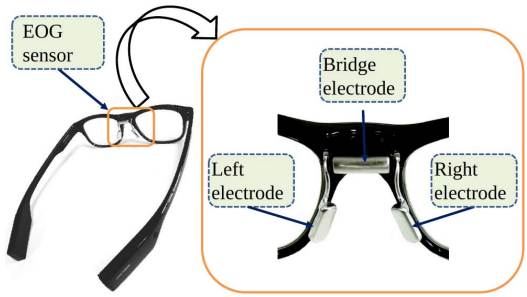

Fig. 2: JINS-MEME glasses, used for data recording, carry an EOG sensor consisting of a bridge, left, and right electrodes.

two channels corresponding to the horizontal and vertical eye movements. This is because preliminary experiments show that using only eye movements data is sufficient to describe user reading behaviors. We employ two data sets, namely the labeled EOG data set and the unlabeled EOG data set.

The labeled EOG data set was prepared by Ishimaru et al. [25] and also employed by Islam et al. [10]. This data set consists of data for ten native Japanese university students who wore the JINS-MEME glasses for two days (12 hours per day). They read JH, JV, and ENG texts each for one hour per day, and the remaining period did not read anything. Participants also wore a front camera called the Narrative Clip to take frontal images that were used for data labeling purposes. The recorded EOG data were then pre-processed and divided into data segments of 30 seconds (have 15 seconds overlap). The dimension of each data segment is 3000×2. The number of data segments for JH, JV, ENG, and NR classes is 5,792, 5,798, 5,340, and 32,708, respectively.

The unlabeled EOG data set, introduced by Islam et al. [10],

was also recorded in the same way except for the collection of the labels. This data set was recorded for a total of 52 participants. After pre-possessing the data in the above-mentioned way, the total number of unlabeled EOG data segments is 177,921. The labeled and unlabeled EOG data sets were recorded under the in-the-wild condition.

V. Evaluation and Results

A. Training and Evaluation Protocols

The performance evaluation includes the SimCLR pre-training followed by target task training. In SimCLR pre-training, we made a pool of seven signal transformations as described in Section III and created a matrix of 7×7 possible signal transformation pairs. For each pair, we applied two signal transformations sequentially, one after another, to a data segment except for the case where the signal transformation pair is created by repeating the same signal transformation (diagonal) and, here, we applied it to a data segment only once. We evaluated the SimCLR method for all signal transformation pairs by employing each pair for pre-training using the unlabeled EOG data and then generating the target task model by re-training it using labeled EOG data.

We employ two baselines, SSL and supervised methods, to measure the competence of the SimCLR method. For SSL, we reproduced the results for the method described in [10] using the same EOG data sets (unlabeled and labeled) and signal transformations employed for the SimCLR method. We generated a pseudo task (classification) and corresponding labels by applying the signal transformations, where pre-training was done by predicting the transformation applied to the data segment instead of optimizing the contrastive objective as did for the SimCLR method. We trained the model described for SSL for the target task using the same labeled EOG data set without any pre-training for the supervised baseline.

We removed the class imbalance in the labeled EOG data set by downsampling the majority classes, with random selection, to 5,340 samples (the smallest number) and the chance rate in prediction being 25%. To evaluate the performance for a wide range of available data, we changed the labeled training data segments as 100% (all available), 50%, 20%, and 0.1% training samples per class for the target task training. We evaluated methods in a person independent way where the model was trained using data from nine out of ten participants and tested on the data from the remaining one. We employed prediction accuracy as the evaluation metric.

B. SimCLR Outcomes

Table I reports the results obtained in accuracy (in percentage) for the SimCLR method. Table I(a)-(d) report the results

TABLE I: Prediction accuracy (in percentage) for 7×7 signal transformation pairs for SimCLR method. The diagonal entries correspond to using only a single transformation. The red (bold) and blue (italic) texts outperform both SSL and supervised baselines, and only supervised baseline, respectively.

(a) 100% training samples per class

1st transformation \ 2nd transformation	scale	noise	negate	time-flip	channel shuffle	permutation	time-warp
scale	**57.95**	**59.45**	*56.47*	*56.43*	**60.38**	**58.24**	**58.30**
noise	**58.07**	**58.23**	**59.76**	**58.49**	**58.82**	**57.71**	**58.41**
negate	**59.70**	**59.01**	*57.16*	*57.14*	**57.43**	**57.69**	**58.69**
time-flip	**58.37**	**59.44**	55.83	**58.24**	*56.68*	**57.73**	**59.23**
channel shuffle	**57.85**	*56.86*	56.01	**59.66**	**57.57**	**58.64**	**59.48**
permutation	**58.59**	**57.67**	**57.81**	**57.47**	**58.73**	**58.56**	**57.45**
time-warp	**59.02**	**57.84**	**58.11**	**57.47**	*57.24*	**58.13**	**58.43**

(b) 50% training samples per class

1st transformation \ 2nd transformation	scale	noise	negate	time-flip	channel shuffle	permutation	time-warp
scale	*56.51*	**57.82**	*56.26*	**57.54**	**58.14**	**57.72**	**59.10**
noise	**57.97**	**58.26**	**57.77**	**57.65**	**57.45**	**58.48**	**58.70**
negate	**58.18**	**58.22**	*56.76*	*56.63*	**57.74**	**58.22**	**58.48**
time-flip	**58.64**	**57.99**	**57.57**	**58.41**	**58.30**	**57.97**	**58.25**
channel shuffle	*57.00*	**57.32**	*56.60*	**57.69**	**58.33**	**58.47**	**58.45**
permutation	**58.22**	**57.58**	**58.59**	**57.91**	**58.57**	**58.10**	**57.76**
time-warp	**57.79**	**58.85**	**57.73**	**57.74**	**58.88**	**57.97**	**57.52**

(c) 20% training samples per class

1st transformation \ 2nd transformation	scale	noise	negate	time-flip	channel shuffle	permutation	time-warp
scale	*54.45*	**56.14**	*54.94*	*54.91*	*55.15*	*55.13*	**57.03**
noise	**55.62**	**55.64**	**56.00**	*55.16*	**55.99**	**56.83**	**56.07**
negate	*55.03*	*54.67*	*54.34*	*54.69*	*54.64*	**55.46**	**56.28**
time-flip	54.44	**55.96**	53.08	**55.71**	*53.80*	**56.03**	**55.75**
channel shuffle	**55.35**	*55.00*	*55.10*	*54.84*	*55.22*	**56.82**	**55.36**
permutation	**57.29**	**55.72**	**56.07**	**56.42**	**56.89**	**55.99**	*55.26*
time-warp	**55.68**	**55.71**	**56.73**	**55.53**	**56.68**	**56.55**	**56.24**

(d) 0.1% training samples per class

1st transformation \ 2nd transformation	scale	noise	negate	time-flip	channel shuffle	permutation	time-warp
scale	27.60	*28.67*	*29.44*	27.82	*29.57*	*28.49*	*29.62*
noise	*30.56*	*29.21*	28.19	*29.56*	27.48	*30.16*	28.23
negate	*29.00*	*28.50*	*28.92*	27.59	28.32	*30.43*	*28.73*
time-flip	28.25	27.90	*28.89*	27.60	27.92	*28.90*	28.39
channel shuffle	*30.97*	27.86	27.72	*28.82*	27.78	*29.47*	*31.34*
permutation	28.09	*28.83*	*32.35*	*29.34*	*29.89*	*29.68*	*29.44*
time-warp	*30.07*	*29.58*	*29.52*	28.04	*30.74*	*29.16*	28.43

TABLE II: Prediction accuracy (in percentage) for SSL and supervised baselines with SimCLR (max accuracy).

Method	Training samples per class			
	100%	50%	20%	0.1%
SimCLR (max accuracy)	60.38	59.10	57.29	32.35
SSL	57.36	57.06	55.28	40.57
Supervised	56.42	55.77	53.51	28.46
Chance rate	25.00	25.00	25.00	25.00

for 100%, 50%, 20%, and 0.1% training samples per class, respectively, for a batch size of 32 and a learning rate of 0.003. On the other hand, Table II reports the results of the SSL and supervised baselines along with maximum SimCLR outputs obtained. In Table I, the red (bold) and blue (italic) texts mean that the SimCLR method outperforms both SSL and supervised baselines, and only supervised baseline, respectively. The results show that the SimCLR method outperforms for a vast majority of signal transformation pairs employed for data augmentation in SimCLR pre-training for a wide range of 100%, 50%, and 20% training samples per class. On the other hand, for a short-range around 0.1% training samples per class, the SSL performed well, although the SimCLR method outperformed the supervised baseline for a significant number of signal transformation pairs.

The performance of the SimCLR method has a dependency on the signal transformation pairs employed for data augmentation in SimCLR pre-training and the number of training samples per class in target task training. A closer look shows that noise addition, permutation, and time-warp signal transformations alongside other signal transformations perform best for 100%, 50%, and 20% training samples per class in terms of the number of outperforms. On the other hand, only permutation alongside other signal transformations performs best for 0.1% training samples per class. Another key point is that, although the best performance is achieved for the pair of different signal transformations, even the pair of same signal transformation (diagonal) performed quite well. The results show a path for system designers to select the best model depending on the signal transformation pairs and available training samples.

C. Study of Performance Gain and Data Efficiency

We also analyzed the performance gain and data efficiency of the SimCLR method. The performance gain is calculated as the difference of the outputs (accuracy) for the SimCLR method and baselines. On the other hand, the data efficiency means whether the SimCLR method performs well with fewer training samples per class or not compared to the baselines that we measured by setting performance gain as a parameter.

We calculated performance gain based on Table II and reported in Table III. Table III(a) reports the performance gain of the SimCLR method compared to the baselines when an equal amount of data were employed for training both methods; SimCLR and baseline. The results show a maximum performance gain of 3.02 and 3.96 percentage points compared to the SSL and supervised baselines, respectively. Table III(b) reports the performance gain of the SimCLR method for 100%, 50%, 20%, and 0.1% training samples per class cases compared to the baselines when the baseline method is trained for 100% training samples per class. The results show an almost equal performance when the SSL baseline and SimCLR method are trained by employing 100% and 20% training samples per class, respectively. This shows that the SimCLR method is about 80% data efficient. On the other hand, a performance gain of 0.87 is obtained when the supervised baseline and SimCLR method are trained by employing 100% and 20% training samples per class, respectively. This shows that the SimCLR method is more than 80% data efficient.

This excellent performance gain and data efficiency demonstrate that the SimCLR pre-training help in capturing discriminative features that, in turn, help to achieve the best performance in the target task by improving class-level recognition.

TABLE III: Performance gain of the SimCLR method calculated based on Table II.

(a) Performance gain when both SimCLR and baseline methods are trained using an equal number of training samples per class.

		training samples per class used for both SimCLR and baseline			
		100%	50%	20%	0.1%
Baselines	SSL	3.02	2.04	2.01	-8.22
	Supervised	3.96	3.33	3.78	3.89

(b) Performance gain when the SimCLR and baseline methods are trained using different numbers of training samples per class and only 100% training samples per class, respectively.

		SimCLR (training samples per class used)			
		100%	50%	20%	0.1%
Baselines (training samples per class used)	SSL 100%	3.02	1.74	–0.07	–25.01
	Supervised 100%	3.96	2.68	0.87	–24.07

TABLE IV: Effect of batch size and learning rate, where #Outperform represents the number of signal transformation pairs for which the SimCLR method outperforms the SSL and supervised baselines.

(a) Effect of batch size

Data size	Parameter	Batch size				
		16	24	32	64	128
100%	#Outperform	27	34	40	37	29
	Max accuracy	58.88	59.90	60.38	59.73	59.27
50%	#Outperform	22	39	43	40	17
	Max accuracy	58.48	58.91	59.10	58.77	58.15
20%	#Outperform	25	38	30	28	11
	Max accuracy	57.00	56.76	57.29	57.16	56.09
0.1%	#Outperform	0	0	0	0	0
	Max accuracy	33.08	32.74	32.35	30.78	30.78

(b) Effect of learning rate

Data size	Parameter	Learning rate				
		0.0001	0.0005	0.001	0.002	0.003
100%	#Outperform	0	3	18	45	40
	Max accuracy	57.13	58.09	58.42	59.85	60.38
50%	#Outperform	0	4	10	40	43
	Max accuracy	55.91	57.42	57.49	58.83	59.10
20%	#Outperform	0	3	8	23	30
	Max accuracy	53.22	56.27	56.19	56.65	57.29
0.1%	#Outperform	0	0	0	0	0
	Max accuracy	27.25	28.57	29.42	31.15	32.35

D. Study of Batch Size and Learning Rate Dependency

We also studied the effect of two hyperparameters of batch size and learning rate for the SimCLR method. To explore the sensitivity, we trained the SimCLR method for target task with different batch sizes (for a learning rate of 0.003) and learning rates (for a batch size of 32) with 100%, 50%, 20%, and 0.1% training samples per class. We evaluated the SimCLR method against the following criteria; the number of signal transformation pairs employed for pre-training for which the SimCLR method outperformed the SSL and supervised baselines, and the maximum accuracy achieved among all signal transformation pairs. Tables IV(a) and IV(b) report the obtained results. The results show that the SimCLR method is robust for a wide range of smaller batch sizes, although it produces the best results with a batch size of 32. On the other hand, the SimCLR method performs worst for low learning rate and starts to improve by increasing it and produces the best results with a learning rate of 0.003. Therefore, in general, the SimCLR method performs well for smaller batch sizes and larger learning rates.

VI. Conclusion

The SimCLR method, in recent studies, shows superior performance in tackling the lack of large-sized labeled data issues. This method is explored in various domains, including physical activity recognition. This study is one of the first in exploring the SimCLR method for a cognitive activity like fine-grained reading detection for a large number of signal transformation pairs compared to the SSL and supervised baselines. The results show that the SimCLR method outperforms baselines for a vast majority of signal transformation pairs for a wide range of available labeled training data. The analysis also shows that the SimCLR method is able to produce a similar performance to the baselines using only 20% of labeled training data that demonstrate an excellent data efficiency and performance gain. In addition to these results, the additional analysis carried out in this study shows a direction to achieve the best performance by applying the SimCLR method regardless of the available labeled training data that constitute useful feedback for reading analysis.

Future work includes an investigation to check the effectiveness and suitability of signal transformation in pre-training using other combinations such as three transformations.

References

[1] N. Y. Hammerla, S. Halloran, and T. Plötz, "Deep, convolutional, and recurrent models for human activity recognition using wearables," Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, AAAI Press, pp. 1533-1540, 2016.

[2] A. Y. Hannun et al., "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," Nature Medicine, vol. 25, no. 1, pp. 65–69, 2019.

[3] H. Haresamudram et al., "Masked reconstruction based self-supervision for human activity recognition," Proceedings of the International Symposium on Wearable Computers, Mexico (Virtual Event), ACM, pp. 45–49, 2020.

[4] A. Saeed, T. Ozcelebi, and J. Lukkien, "Multi-task self-supervised learning for human activity detection," Proceedings of the ACM on IMWUT, vol. 3, no. 2, article no. 61, pp. 1-30, 2019.

[5] S. Albelwi, "Survey on self-supervised learning: auxiliary pretext tasks and contrastive learning methods in imaging," Entropy, vol. 24, no. 551, pp. 1-22, 2022.

[6] F. Wang and H. Liu, "Understanding the behaviour of contrastive loss," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2495-2504, 2021.

[7] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," Proceedings of the 37th International Conference on Machine Learning, PMLR, vol. 119, pp. 1597-1607, 2020.

[8] C. I. Tang, I. Perez-Pozuelo, D. Spathis, and C. Mascolo, "Exploring contrastive learning in human activity recognition for healthcare," ArXiv:2011.11542, 2020.

[9] J. Wang, T. Zhu, J. Gan, L. Chen, H. Ning, and Y. Wan, "Sensor data augmentation by resampling for contrastive learning in human activity recognition," ArXiv:2109.02054, 2021.

[10] M. R. Islam et al., "Self-supervised learning for reading activity classification," Proceedings of the ACM on IMWUT, vol. 5, no. 3, article no. 105, pp. 1-22, 2021.

[11] Wikipedia, "Horizontal and vertical writing in east asian scripts," https://en.wikipedia.org/w/index.php?title=Horizontal_and_vertical_writing_in_East_Asian_scripts&oldid=984358336, Accessed: April 2, 2022.

[12] H. Haresamudram, I. Essa, and T. Plötz, "Assessing the state of self-supervised human activity recognition using wearables," ArXiv, 2022.

[13] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," Proceedings of the European Conference on Computer vision. Springer, Cham, pp. 69–84, 2016.

[14] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: feature learning by inpainting," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA. IEEE, pp. 2536–2544, 2016.

[15] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, IEEE, pp. 1422–1430, 2015.

[16] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," Technologies, vol. 9, no. 1, 2021.

[17] P. Khosla et al., "Supervised contrastive learning," Advances in Neural Information Processing Systems, vol. 33, pp. 18661-18673, 2020.

[18] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. Hinton, "Big self-supervised models are strong semi-supervised learners," ArXiv:2006.10029, 2020.

[19] B. Khaertdinov, E. Ghaleb, and S. Asteriadis, "Contrastive self-supervised learning for sensor-based human activity recognition," 2021 IEEE International Joint Conference on Biometrics, pp. 1-8, 2021.

[20] K. Shah, D. Spathis, C. I. Tang, and C. Mascolo, "Evaluating contrastive learning on wearable timeseries for downstream clinical outcomes," ArXiv:2111.07089, 2021.

[21] A. Bulling, J. A. Ward, and H. Gellersen, "Multimodal recognition of reading activity in transit using body-worn sensors," ACM Transactions on Applied Perception, vol. 9, no. 1, article no. 2, pp. 1–21, 2012.

[22] A. Strukelj and D. C. Niehorster, "One page of text: eye movements during regular and thorough reading, skimming, and spell checking," Journal of Eye Movement Research, vol. 11, no. 1, pp. 1–22, 2018.

[23] C. Kelton et al., "Reading detection in real-time," Proceedings of the ACM Symposium on Eye Tracking Research and Applications, ACM, article no. 43, pp. 1–5, 2019.

[24] L. Copeland, T. Gedeon, and S. Mendis, "Predicting reading comprehension scores from eye movements using artificial neural networks and fuzzy output error," Artificial Intelligence Research, vol. 3, no. 3, pp. 35–48, 2014.

[25] S. Ishimaru, T. Maruichi, M. Landsmann, K. Kise, and A. Dengel, "Electrooculography dataset for reading detection in the wild," Adjunct Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers, London, UK, pp. 85–88, 2019.