Project review

First of all, I thank the authors for their work. A lot has already been done, and I hope that the work will find its place!

Now I will comment on the work point by point.:

- The project report includes: 1. problem statement, 2. the main idea is described, 3. A comparison with relevant methods is present.

The paper contains a statement of the problem, the main idea is described and understandable, and the novelty and architectural comparison with other models are also shown. The authors want to learn from data that has spatial and temporal components using convlstm, namely on the task of contrastive self-supervised learning. Unlike current approaches, the authors want to use their approach to better account for temporal dependencies (current approaches look more at spatial dependencies).

- Describe if each part is described clearly and all the necessary for understanding information is provided. What is missing? What is left uncovered? Can it be done better?

The current approaches are described in some detail, but it would be great to add a mathematical description of the current approaches.
There is a lack of comparison with audio and video processing approaches. It seems to me that there should be similar models that take time dependencies well into account.
https://arxiv.org/pdf/2207.00419

- Provide recommendations on styling, quality and structure.

The article is designed pretty well, but here's what I would suggest adding. It's fun to create a pseudocode and show the dimensions of the model, its parameters, and training parameters separately in the plate.

Unfortunately, dimensions and terms are being introduced that were not introduced earlier and it is not clear what they refer to if you do not understand the code.

It would be great to add pictures with the current architecture (a description of the current approach and how it differs from others). Currently, the previous approaches are described in some detail, but the current one is almost not described.

It is unclear how much time/memory/resources it took to train models.

- The experiment protocol is reasonable.

As an experiment, we took the data, trained self-supervised on it, and then applied it downstream to check the quality. Everything is pretty clear and seems reasonable.

- The presented results are reasonable.

It's great that we managed to train the model and present the results.
But there is no comparison with other approaches in the results, it is not very clear what to compare with and how good the resulting quality is.

There is a lack of understanding why you use such augmentations. If the whole joke is that you take both dimensions into account well at once, then maybe the augmentation needs to be changed. It is also unclear why convlstm, maybe you can do with just convolutions or use something smarter.

The current setup and hyperparameters are poorly explained.

- Github is clear and consequent.

There is a lot of good code in the repository, there are configs and a structure. But I would add.gitignore to remove unnecessary folders (including those from python) and the tmp file, if necessary. You can try to make a different folder architecture to make the repository easier to navigate (for example, move ipynb to a separate location and put all the code in /src).
I also think that comments should be written in the same language, preferably English.
I think it would be a plus to add the weights of the already trained models so that it would be easier for the layman to run the code and play with the solution.

The downstream task is present in a separate laptop in colab, but it is not in the repository, while there is a laptop with a loss, which is not very well described and nothing is said about it anywhere.

- README.md includes all the necessary steps and is easy to execute.

The readme contains the necessary launch information. But I think the readme should be made more beautiful. The Requirements are already present in the repository, they don't need to be duplicated. I would add some of the information from the article, as well as new pictures to understand and attract attention.

- All intermediate results are reproducible.

No fixed seeds

- Provided code is running without errors.

At first I couldn't install torch, but it's good that the authors added docker

- What other improvements can be made?

I would think towards transformer models. It's not clear why you're using ConvLSTM.
It would also be good to visualize the received embeddings.