MSc Project - Reflective Essay

Project Title:	A Convolutional Approach to Action Anticipation
Student Name:	Shelly Srivastava
Student Number:	190385633
Supervisor Name:	Dr Lorenzo Jamone
Programme of Study:	MSc in Artificial Intelligence with Industrial Experience

1. Introduction

Action anticipation is quite an interesting area of research as it can help in human-robot interaction and collaboration. There are a lot of intricate movements which we as humans do. Anticipating such actions are quite intuitive in the case of human-human collaboration. When the second person in the interaction is replaced by a robot, this collaboration becomes difficult.

As a society, we are driving towards a future in which we work in close collaboration with robots. Such collaboration is only possible when both parties can communicate even without using verbal information. Non-verbal signals such as gestures, movements, facial expressions are quite important for a robot to understand, for successful collaboration (Hoffman 2010).

2. Motivation

There has been a lot of development in the field of action anticipation. Two of the work, from which I drew most of my inspiration from, are Santos et al. (2019) and Schydlo et al. (2018). Both the works are based on the Acticipate dataset (Duarte et al., 2018). Both, Santos at al. (2019) and Schydlo et al. (2018), focus on improving the anticipation accuracy by providing additional information to the neural network model. Their methods prove to be efficient as the contextual information increases the anticipation accuracy very early on in the action sequence.

Both the works use LSTMs (Long Short Term Memory) as the basic building block of the architecture. Schydlo et al. (2018) uses an encoder-decoder architecture whereas Santos et al. (2019) uses two LSTMs stacked together which produce an output at each time step. LSTMs are great at handling sequence information but it is difficult to train. Recently, many domains which used LSTMs have started shifting to CNNs. CNNs can capture the sequential relationship and offer better results in some cases. My main objective was to implement an architecture using CNNs which can at least replicate the same results as seen in Santos et al. (2019). They also proposed a stochastic model for action anticipation. I was planning to use a library (Esposito, 2020) for the implementation of Bayesian neural networks.

Santos et al. (2019) also proposed a technique for feature extraction, selection and embedding. My initial thought was to research alternatives which can help in providing an end to end approach with minimal pre-processing steps.

3. Implementation

The deep learning framework used for the project was Pytorch. Having only a brief knowledge about the framework, I decided to start learning more about Pytorch and the encoder-decoder architecture. One of the sources that I referred to was Trevett (2020). I started working on implementing feature extraction by using only the RGB images. I used a pre-trained Alexnet to extract the features of the frames as suggested by

Venugopalan et al. (2015). This method did not prove to be effective as the dataset is quite small (with only 19000 frames) and the frames were very similar to each other. To even fine-tune a pre-trained model, a lot more data are required. The actions in the frame where ambiguous as well (the main scene of the video did not change and only hand positions changed). This presented to be a challenge.

Due to this, I focused on feature extraction as suggested in by Santos et al. (2019). The pre-processing suggested by them use OpenPose and OpenCV. I used OpenPose demo to extract JSON files related to each frames containing joint position information of the actor. I also used OpenCV for segmentation as suggested by Santos et al. (2019). The segmentation algorithm segments out the red colour ball. The scene also contains a red sofa, therefore the input to the algorithm is a cropped picture. The output of the algorithm is the approximate x and y coordinated of the ball.

After the pre-processing, I started implementing the CNN model. Initially, I was focusing on the encoder-decoder architecture provided by Schydlo et al. (2018). I found a convolutional encoder-decoder model (Gehring et al., 2017). This model has a teacher forcing ratio of 1 at the time of decoding. This proved to be a challenge as the model didn't learn anything for the task of anticipation. Therefore I changed the model to a convolutional encoder with an LSTM decoder as proposed by Gehring et al. (2016).

I also implemented the Bayesian version of the CNN-LSTM model using Esposito (2020). The deterministic and stochastic model's accuracy was good for action recognition but for action anticipation, it didn't work quite well. Also, it was difficult to evaluate the model as per the algorithm used by Santos et al. (2019). I had to change the architecture of the model again to a convolution version of stacked LSTMs as suggested by Santos et al. (2019).

The implemented model uses convolutional block which is inspired by Gehring et al., (2017). The stacked convolutions replace the LSTM cells in the model proposed by Santos et al. (2019). The model works in the same manner as the LSTM version. The input to the model is the embedding at each time step and the output of the model is the probability distribution over the action classes for each time step. The model can learn sequential information, just like the LSTM version.

I also implemented an LSTM model based on the works of Santos et al. (2019) to compare my implementation with. At this point, there was not enough time for me to implement the Bayesian version of the new CNN model.

4. Analysis

Santos et al. (2019) uses sophisticated methods to avoid noisy data. They use filters to remove false-positive users in the case of joint position extraction using OpenPose. They also use hand positions as a source of information. In my implementation, the filter is not applied. Also, the hand positions are not extracted. The limb movements only contain positions of shoulder, elbow and wrists. Even the results of the segmentation algorithm is a bit noisy as it fails to find the ball in each frame. This was verified by reviewing the ball key points file. This pre-processing differences played a significant role in the prediction capability of the models.

The LSTM model which was implement couldn't recognize all the actions. Santos et al. (2019) LSTM model could recognize 100% of the actions. There was a huge drop in the anticipation accuracy for the implemented LSTM model. The CNN model receives the same inputs but could recognise all the actions in the dataset. The anticipation accuracy was also comparable to the model implemented by Santos et al. (2019).

The training of the CNN models with l =1 took significantly lesser time than the LSTM model. This is definitely an advantage. The family of CNN models seems to outperform the base LSTM model. The CNN model could also learn the representations in the same manner as described by Santos et al. (2019).

A few drawbacks of the implementation is the lack of cross-validation. The model was trained on a training set and validated against a validation set. It is not always the best approach for a small dataset. The CNN model should be cross-validated. It was not performed due to the lack of time.

5. Future Work

There is still a lot of scope for research for the CNN model before stating its effectiveness for action anticipation. Effect of more hyperparameters like the size of feature maps should be considered and evaluated. The model itself should be evaluated against all the metrics provided by Santos et al. (2019). The performance of the model should be compared with different models and on different available datasets. Bayesian versions of the model should be implemented (Santos et al., 2019) in the future to realise the overall anticipation accuracy of the model.

6. Conclusion and Societal perspective

An LSTM model comes with a lot of constraints. To overcome this, a convolutional approach was taken for the task of action anticipation. The new model looks at the action sequence at each time step and produce a prediction at each time step with the information available till that time. The new model has a lot of potentials and should be evaluated further. My work only scratches the surface and opens the door for further research in this direction.

The successful implementation of an action anticipation model will enable humans to work closely with robots and other AI systems. The robots will be easily accepted in the society and the learning curve will not be steep. If robots can work alongside humans without changing most of the workflow, then we as a society are going to be more open to the change. A robot which can anticipate an action just like any other human can be accepted quickly in the society than a robot which doesn't have this capability.

An anticipation model needs to have good accuracy at small observation ratio. Sometimes, incorrect decisions can be made due to a false action classification. This can result in disasters. It feels like an impossible task to achieve, given that humans still make mistakes when trying to anticipate the immediate future. This uncertainty is always present for such tasks. Interestingly, it has been seen that models are capable of outperforming humans as well (He et al., 2015). With the trend in technology, even the task of anticipation can be solved with super-human capabilities.

References

Hoffman, G., 2010, March. Anticipation in human-robot interaction. In *2010 AAAI Spring Symposium Series*.

Santos, C.C.D., Moreno, P., Samatelo, J.L.A., Vassallo, R.F. and Santos-Victor, J., 2019. Action Anticipation for Collaborative Environments: The Impact of Contextual Information and Uncertainty-Based Prediction. *arXiv preprint arXiv:1910.00714*.

Schydlo, P., Rakovic, M., Jamone, L. and Santos-Victor, J., 2018, May. Anticipation in human-robot cooperation: A recurrent neural network approach for multiple action

sequences prediction. In *2018 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1-6). IEEE.

Duarte, N.F., Raković, M., Tasevski, J., Coco, M.I., Billard, A. and Santos-Victor, J., 2018. Action anticipation: Reading the intentions of humans and robots. *IEEE Robotics and Automation Letters*, *3*(4), pp.4132-4139.

Esposito, P. Blitz-bayesian layers in torch zoo (a Bayesian deep learning library for torch). https://github.com/piEsposito/blitz-bayesian-deep-learning/, 2020

Trevett, B. (2020). *bentrevett/pytorch-seq2seq*. [online] GitHub. Available at: https://github.com/bentrevett/pytorch-seq2seq [Accessed 1 Sep. 2020].

Gehring, J., Auli, M., Grangier, D., Yarats, D. and Dauphin, Y.N., 2017. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*.

Gehring, J., Auli, M., Grangier, D. and Dauphin, Y.N., 2016. A convolutional encoder model for neural machine translation. *arXiv preprint arXiv:1611.02344*.

Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T. and Saenko, K., 2015. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision* (pp. 4534-4542).

He, K., Zhang, X., Ren, S. and Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision* (pp. 1026-1034).