Deep Learning Project Report

529005673 Jyun-Sheng Fu

Research Topic

In this project, an effective and interpretable network module, the ***Temporal Relation Network (TRN)*** *[1]*, is implemented and utilized to perform human motion detection. The model takes in videos as inputs (RGB image streams), and outputs the prediction probabilities for the 339 classes. Particularly, I focus on the class “slipping” as my aim of detection in this project. The model is pre-trained on the Moments-in-Time dataset *[2]*, which contains the target motion — “slipping”.

Dataset

The dataset I use the the Moments-in-Time dataset *[2]*, a collection of one million short videos each with a label corresponding to an event unfolding in 3 seconds, involving human, animals, and natural phenomenon.

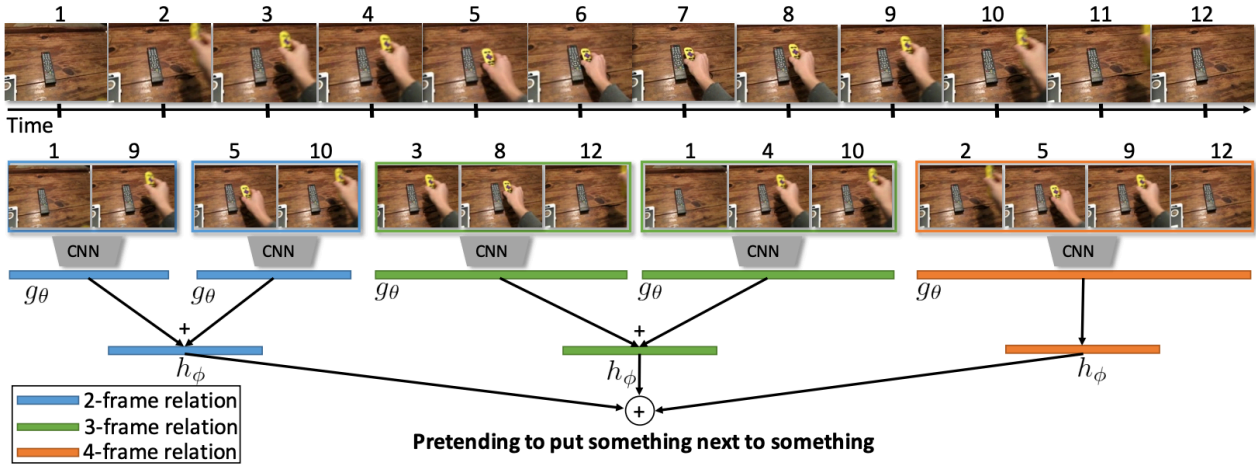
Moments-in-Time dataset *[2]*

The Moments in Time dataset consists of videos corresponding to 399 verbs (labels) :

clapping, praying, dropping, burying, covering, flooding, leaping, drinking, slapping, cuddling, sleeping, preaching, raining, stitching, spraying, twisting, coaching, submerging, breaking, tuning, boarding, running, destroying, competing, giggling, shoveling, chasing, flicking, pouring, buttoning, hammering, carrying, surfing, pulling, squatting, aiming, crouching, tapping, skipping, washing, winking, queuing, locking, stopping, sneezing, flipping, sewing, clipping, working, rocking, asking, playing+fun, camping, plugging, pedaling, constructing, slipping, sweeping, screwing, shrugging, hitchhiking, cracking, scratching, trimming, selling, marching, stirring, kissing, jumping, starting, clinging, socializing, picking, splashing, licking, kicking, sliding, filming, driving, handwriting, steering, filling, crashing, stealing, pressing, shouting, hiking, vacuuming, pointing, giving, diving, hugging, building, swerving, dining, floating, cheerleading, leaning, sailing, singing, playing, hitting, bubbling, joining, bathing, raising, sitting, drawing, protesting, rinsing, coughing, smashing, slicing, balancing, rafting, kneeling, dunking, brushing, crushing, rubbing, punting, watering, playing+music, removing, tearing, imitating, teaching, cooking, reaching, studying, serving, bulldozing, shaking, discussing, dragging, gardening, performing, officiating, photographing, sowing, dripping, writing, clawing, bending, boxing, mopping, gripping, flowing, digging, tripping, cheering, buying, bicycling, feeding, emptying, unpacking, sketching, standing, weeding, stacking, drying, crying, spinning, frying, cutting, paying, eating, lecturing, dancing, adult+female+speaking, boiling, peeling, wrapping, wetting, attacking, welding, putting, swinging, carving, walking, dressing, inflating, climbing, shredding, reading, sanding, frowning, closing, hunting, clearing, launching, packaging, fishing, spilling, leaking, knitting, boating, sprinkling, baptizing, playing+sports, rolling, spitting, dipping, riding, chopping, extinguishing, applauding, calling, talking, adult+male+speaking, snowing, shaving, marrying, rising, laughing, crawling, flying, assembling, injecting, landing, operating, packing, descending, falling, entering, pushing, sawing, smelling, overflowing, fighting, waking, barbecuing, skating, painting, drilling, punching, tying, manicuring, plunging, grilling, pitching, towing, telephoning, crafting, knocking, playing+videogames, storming, placing, turning, barking, child+singing, opening, waxing, juggling, mowing, shooting, sniffing, interviewing, stomping, chewing, arresting, grooming, rowing, bowing, gambling, saluting, fueling, autographing, throwing, drenching, waving, signing, repairing, baking, smoking, skiing, drumming, child+speaking, blowing, cleaning, combing, spreading, racing, combusting, adult+female+singing, fencing, swimming, adult+male+singing, snuggling, shopping, bouncing, dusting, stroking, snapping, biting, roaring, guarding, unloading, lifting, instructing, folding, measuring, whistling, exiting, stretching, taping, squinting, catching, draining, massaging, scrubbing, handcuffing, celebrating, jogging, colliding, bowling, resting, blocking, smiling, tattooing, erupting, howling, parading, grinning, sprinting, hanging, planting, speaking, ascending, yawning, cramming, burning, wrestling, poking, tickling, exercising, loading, piloting, typing.

Architecture

The model inherits the framework of Temporal Relation Networks. An illustration of the model architecture is provided in the figure below. It captures multi-scale temporal relations, i.e., 2-frame, 3-frame, and 4-frame relations. Take the 2-frame temporal relation for example. Features of each frame are first extracted with a series of CNN layers respectively, then the pairwise relation is then modeled with the function $g_\theta(*)$, and composited with the function $h_\theta(*)$.



Temporal relational reasoning in videos *[1]*

- Input

  The input shape of each image is 256x256x3.
  In practice, the input fed to this model is a sequence of RGB images (256x256x3) of at least 8 frames. 12 images will be uniformly sampled for each CNN feature extractor.

- Output

  The output tensor shape is (339,).
  Representing the probabilities corresponding to the 339 classes.

- Model parameters

  The parameters of the actual model is provided in the link below:
  https://github.com/b01901143/TRN-pytorch/blob/master/model_structure.pdf

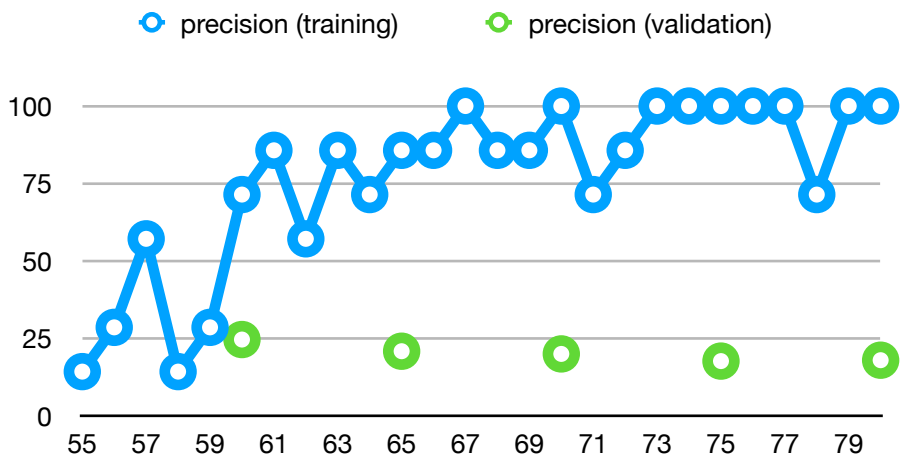
Training and Testing Performance

Navigating through the Moments in Time Dataset, I noticed that the videos under the class "slipping" contains a lot of human sliding, animal sliding, objects slipping away from hand, etc. Since we are focusing on "human slipping", I re-selected the video data under the "slipping class" so that in contains only human slipping (instead of sliding) and form a new dataset. The method afterward was to start from the pre-trained weighs, and train our model on this new, human-slipping-specific dataset.
The biggest problem encountered here is the resource. Among all the resource I can seek for, I trained my model on the TAMU High Performance Computing Research Computing. The file usage on this platform is limited to 250000 files. Training with videos, it is almost required that the training data be parsed into single frames and stored respectively. However, due to the file usage limit,

eventually I am only able to train with 1772 videos and their retracted frames. This includes 32 videos of human slipping and 5 videos each for the rest of the 338 classes.

```
Your current disk quotas are:
Disk       Disk Usage      Limit    File Usage      Limit
/home          51.25M        10G           75      10000
/scratch       12.63G         1T       223204     250000
```

The result is as expected — significantly overfitting. With this little amount of training data, the model easily overfitted within 10 epochs. The training and testing precision is provided in the figure below.



- Training Hyper-Parameters

Batch Size	8
(Additional) Epoch	20
Dropout	0.8
Learning Rate	0.0003

Code

```
> git clone https://github.com/b01901143/TRN-pytorch.git
```

Training and Testing Instructions

- Dependancies
  - Anaconda

```
> conda create -n pytorch python=3.6 numpy scipy
> source activate pytorch
> conda install pytorch torchvision -c pytorch
> conda install -c anaconda pyyaml
> conda install -c menpo opencv=3
> conda install -c conda-forge moviepy
```

- Execution
  - Training

```
> cd TRN-pytorch/
> python parse_moments.py                # parse the Moments in Time Dataset
> ./train.sh                              # run the training script
```

```
## train.sh
python main.py moments RGB \
--arch InceptionV3 --num_segments 8 \
--consensus_type TRNmultiscale --batch-size 64 \
--resume pretrain/
TRN_moments_RGB_InceptionV3_TRNmultiscale_segment8_best.pth.tar \
--eval-freq 5
```

  - Testing on dataset

```
> cd TRN-pytorch/
> ./test_moment.sh                    # run the script for testing on validation set
```

```
## test_moment.sh
python test_moment.py \
--data_folder data/Moments_in_Time_256x256_30fps/validation/ \
--weight pretrain/trained_model.pth.tar \
--arch InceptionV3 --dataset moments
```

  - Testing on video (ex. sample_data/test2.mp4)

```
> cd TRN-pytorch/
> ./test_video.sh                    # run the script for testing on video
```

```
## test_moment.sh
python test_video.py --video_file sample_data/test2.mp4 \
--weight pretrain/trained_model.tar \
--arch InceptionV3 --dataset moments
```

  - Prediction timeline on video (ex. sample_data/sample1.mp4)

```
> cd TRN-pytorch/
> ./test_segment.sh                    # run the script for testing on video segments
```

```
## test_segment.sh
python test_segment.py --output_file sample1.json \
--video_file sample_data/sample1.mp4 \
```

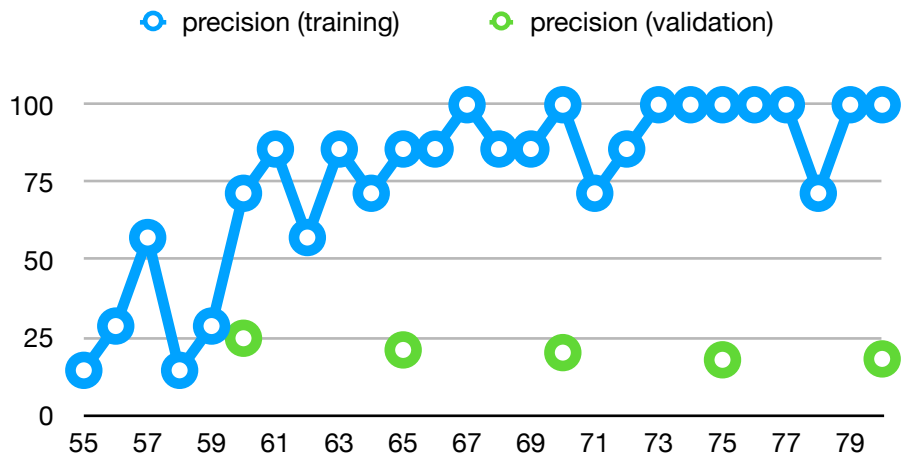
```
> python plot.py
```

- Video Links
  - Training: https://youtu.be/Da65N1ryxmE
  - Testing on dataset: https://youtu.be/GED-pgWEXXw
  - Testing on video: https://youtu.be/nX_CCET4xQ8
  - Prediction timeline on video: https://youtu.be/LgY-nbwLD0s

Part 5

Previous problem

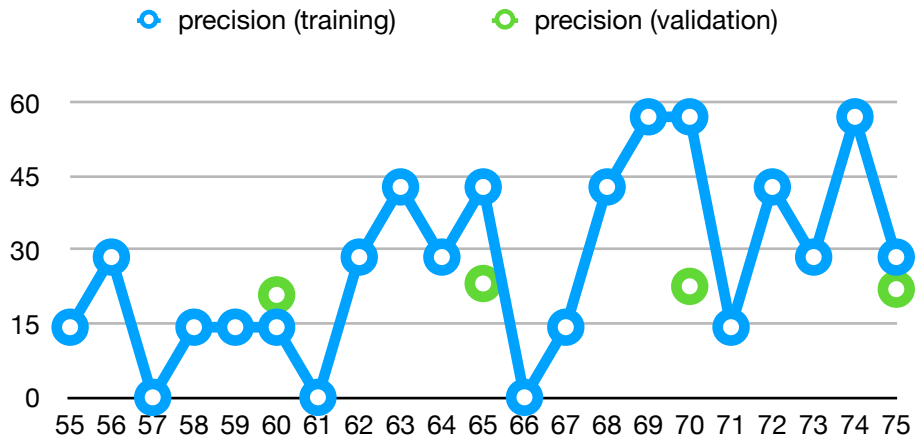
- Overfitted on the small training dataset.



- Undesired rate of false positive.

Solution

- ***Freeze the early layers (basically the convolutional layers), and train only the final fully connected layers.***
- The resulting trining curve is provided below.
- It can be seen the the overfitting problem has been mitigated.



- The learning rate is reduced to 0.0003, as compared to the previous 0.001 learning rate.

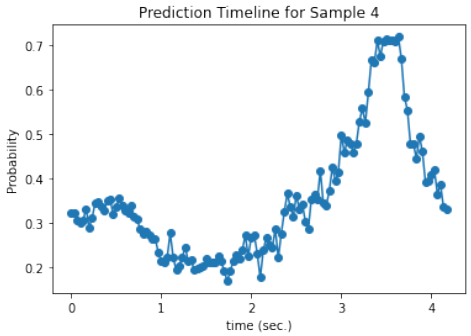
Result Demonstration

- For the prediction timeline, the output value at time ***t*** indicates the probability of "slipping" happening from the t^{th} second to the $(t+1)^{th}$ second.

- Positive sample

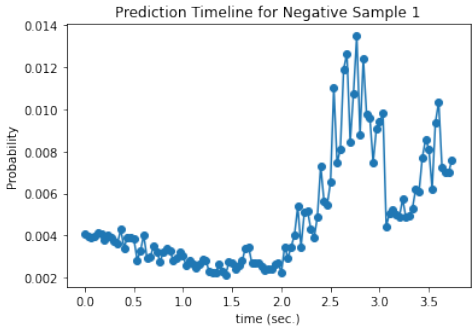
Frame at 0 sec. for sample 4

Frame at 3.5 sec. for sample 4



- Negative sample

Frame at 0 sec. for negative sample 1



Future Work

- The training dataset is way too limited.
- Need to find solutions to either
  - avoid extracting every frame of the training dataset while training, or
  - access an unlimited resource to perform training on the entire or larger portion of the training dataset.

Reference

[1] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. “Temporal relational reasoning in videos.” In ECCV, 2018.
[2] Monfort, Mathew, et al. "Moments in time dataset: one million videos for event understanding." IEEE transactions on pattern analysis and machine intelligence 42.2 (2019): 502-508.