11/16: running on test and ablation experiments

daily plan

this is probably the last entry before the writeup.

- ☑ find best number of epochs for best hyperparameter set via early stopping
- ☑ run on train+val and test on test
- ☑ ablation: without distances
- ☑ ablation: without player counts
- ☑ player order permutation

early stopping for number of epochs

basically just run the best hyperparameter set with early stopping, and when it finishes we can look at the results.

well, i had confused which one was the best one from my experiments yesterday. but after a very confusing 20 minutes, i've fixed it (and updated the excel sheet), and am now running the right model with learning rate 1e-6.

number of epochs is 22. let's go train the final model!

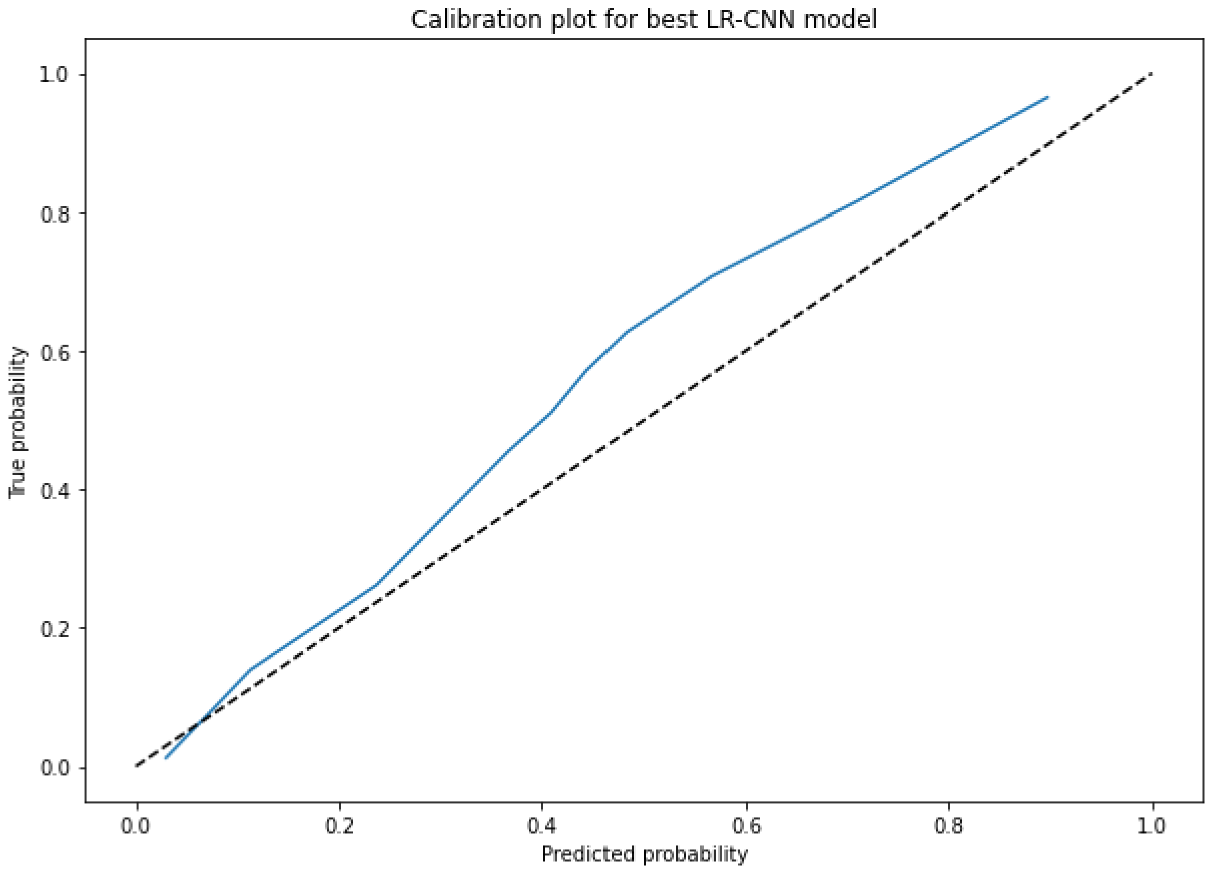
final model

results:

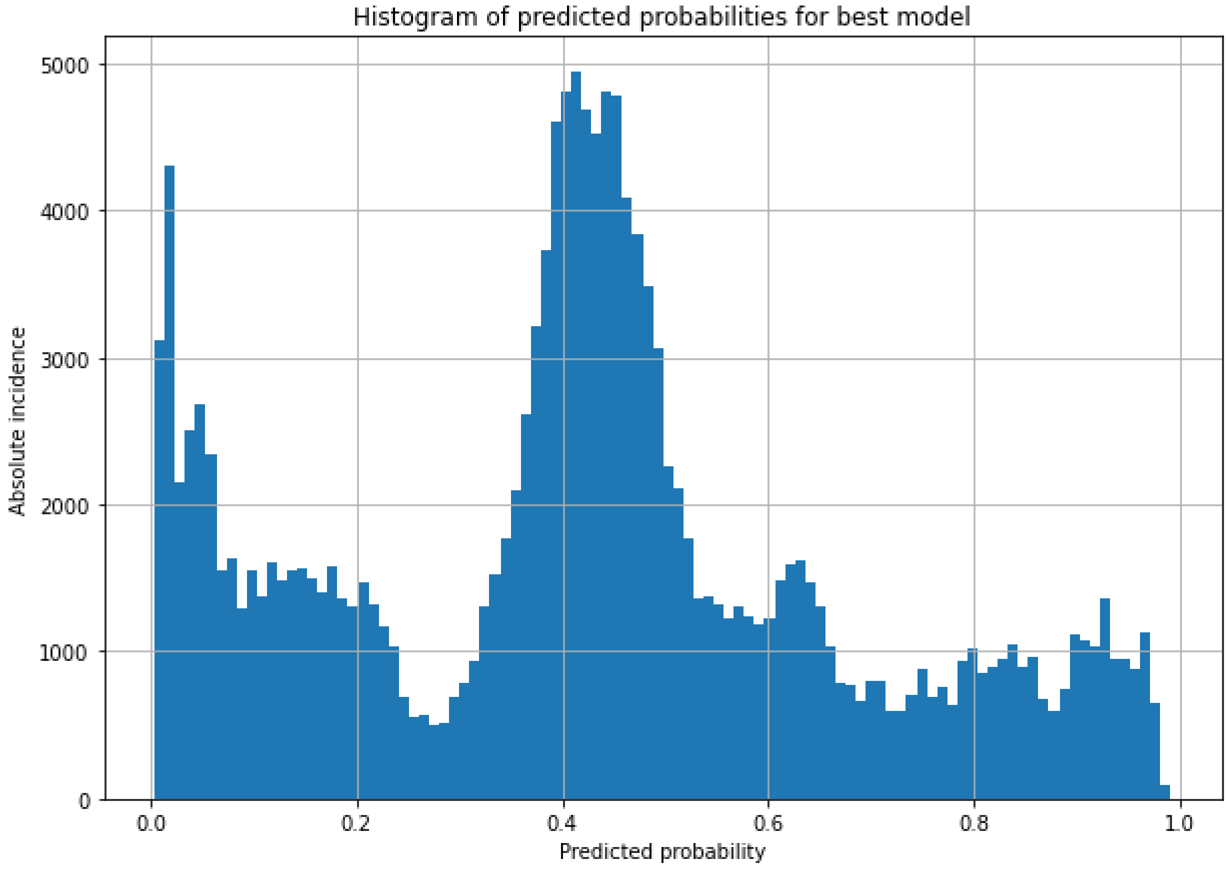
accuracy: 0.6951
auc: 0.8268
log loss: 0.5164

that's actually pretty decent AUC (which was what i was optimizing). the log loss isn't great and the accuracy is actually lower than baseline.

here's a calibration plot with 10 bins:



it kind of underestimates the probabilities for the most part, which is a little unfortunate. here's the histogram of model predictions:



clearly the model has a pretty good indication of when CTs are losing, but not a very good one for when they're obviously winning. on instances when CTs are not obviously losing, then the mean is probably tending to guess the CTs are actually more likely to win.

i suspect this is because there isn't any "bomb related" information in the dataset i'm passing in. even when all T's are dead, they can still win the round by exploding the bomb; whereas if all CTs are dead there's no way to win the round anymore. this probably gives the T's an edge when looking strictly at player counts. on the other hand, T's have the added difficulty of having to "storm" the bomb sites in order to plant a bomb, whereas CT's can mostly just wait until the T's arrive - this probably gives the CTs a small advantage.

interesting next experiment would be adding in not only whether a player is alive but bomb-related information such as the bomb being planted, distance to bomb sites, etc. this would probably give a much better model (though the baseline would also change).

anyway - time for ablation experiments.

ablation

in general i'm not sure whether the ablation experiments should have the final layer or not. it feels like it should (to have the same architecture), but it also feels weird to map from 5 cells to 5 cells. i think i'm going to run with the final layer because the final layer can always map identity if it's not required (right?).

this involves a little bit of changing the LR-CNN code but it should be fine as long as i keep the defaults to what the original code was doing.

no distances

as expected:

accuracy: 0.6852
auc: 0.8136
log loss: 0.5093

overall it's worse, but somehow better calibrated. the distances add in more information but apparently not enough to significantly influence the probabilities.

no player counts

also as expected:

accuracy: 0.5407
auc: 0.5662
log loss: 0.6878

these results are actually surprisingly good for me. i was expecting a lot worse. it seems like just the distances by themselves are enough for the model to have some inclination of who might win, but obviously it doesn't provide enough information by itself.

permutation of players

finally, one last experiment: permuting the players around. since there isn't a definite player 1/2/3..., and they're all interchangeable, i expect the network to have pretty much the exact same results (barring any differences due to randomness).

if we do this for both train *and* test and train the model anew - it doesn't really make sense. the model should still learn from the right inputs. but if we do it just for test then we can check if the model is independent of player order (which is what it should be).

accuracy: 0.6963
auc: 0.8290
log loss: 0.5143

wow! it... somehow figured it out. i'm guessing it's because i was always putting in the order alphabetically (which, without knowing each player's name, makes it probably seem random)? i'm just amazed that the numbers are that close. it actually improved the model a bit overall but this is probably just aleatoric uncertainty at work.

---

summary

things done

- final model run
- ablation experiments
- test data permutation experiment

questions for peterx

- n/a

next steps/remaining action items

- writeup!