# Summery

## First Model: NoHistory Model

This is the first model that beats the baseline.

## Baseline:

It is given by the organizer. It scans all dialog acts for each slot observed so far in the dialog, and outputs the one hypothesis which has the highest score.

## NoHistory:

Different from the baseline, it only scans the dialog acts in the current turn and ignores all history hypotheses.

MajorityBaseline:

It always predicts “none” for each slot.

The results are shown in below.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| team | entry | slot | schedule | metric | test1 | test2 | test3 | test4 |
| baseline | entry0 | all | schedule1 | accuracy | 0.7748 | 0.7928 | 0.9178 | 0.8233 |
| baseline | entry0 | all | schedule2 | accuracy | 0.6020 | 0.4905 | 0.6202 | 0.5841 |
| baseline | entry0 | all | schedule3 | accuracy | 0.5982 | 0.4869 | 0.7033 | 0.6396 |
| baseline | entry0 | joint | schedule1 | accuracy | 0.1455 | 0.1394 | 0.4732 | 0.1775 |
| baseline | entry0 | joint | schedule2 | accuracy | 0.1467 | 0.1399 | 0.4475 | 0.1572 |
| baseline | entry0 | joint | schedule3 | accuracy | 0.1043 | 0.1409 | 0.4923 | 0.1162 |
| nohistory | entry0 | all | schedule1 | accuracy | 0.7878 | 0.7921 | 0.9156 | 0.8270 |
| nohistory | entry0 | all | schedule2 | accuracy | 0.6372 | 0.4967 | 0.6065 | 0.5906 |
| nohistory | entry0 | all | schedule3 | accuracy | 0.6108 | 0.4790 | 0.6898 | 0.6432 |
| nohistory | entry0 | joint | schedule1 | accuracy | 0.1640 | 0.1508 | 0.4572 | 0.1845 |
| nohistory | entry0 | joint | schedule2 | accuracy | 0.1655 | 0.1511 | 0.4281 | 0.1629 |
| nohistory | entry0 | joint | schedule3 | accuracy | 0.1176 | 0.1450 | 0.4675 | 0.1294 |
| majority | entry0 | all | schedule1 | accuracy | 0.8162 | 0.7971 | 0.7720 | 0.8434 |
| majority | entry0 | all | schedule2 | accuracy | 0.7056 | 0.5267 | 0.2590 | 0.6585 |
| majority | entry0 | all | schedule3 | accuracy | 0.6255 | 0.4790 | 0.1200 | 0.6234 |
| majority | entry0 | joint | schedule1 | accuracy | 0.2122 | 0.1723 | 0.0815 | 0.2406 |
| majority | entry0 | joint | schedule2 | accuracy | 0.2143 | 0.1729 | 0.1084 | 0.2485 |
| majority | entry0 | joint | schedule3 | accuracy | 0.1376 | 0.1680 | 0.0616 | 0.1162 |

The “all” in the slot column means the average accuracy of all 9 slots.

The “joint” is the combination of the 9 slots. If one of them is wrong, the result is wrong.

The table above shows that the “nohistory” model is better than the baseline except the test3.

Here are some possible reasons:

The “nohistory” model assumes that the latest SLU result is the best. It makes sense. If the slot appear again in the current turn, the same slot in the previous turns might be wrong. Or else, it will not show up.

“test1”, “test2” and “test4” are from Group A and C. They both used a mixed-initiative design, where “test3” from Group B used a directed design. (Question: why nohistory is better for a mixed-initiative design?)

Inspired Feature to try in the future:

Whether the slot appeared before

Is the old slot confirmed or denied?

The distance between the two slots

Whether there is a conflict between the two slots

If the slot values are different, at least one of them must be wrong

If the slot values are the same, the label for them must be the same too.

Majority Baseline works pretty well except for “test3”. It is because that most of time, there is no value for many slots.

## Topline

For the topline, I consider the top 10 SLU hypothesis for each slot.

The results are shown below.

## Observation

* TopK != Baseline

For the topline, I didn’t take the 10 SLU hypotheses first, but assign all SLU hypotheses to each slot and then take the top 10 candidates in each slot.

That’s why the Top1 is not equal to the baseline.

* TopK > Baseline for test3 and test4

It might be because NLU in these two data are better.

* TopK < Baseline for test1 and test2

It might be because that the candidates for test1 and test2 are generated offline and the numbers of candidates are much more than the number in test3 and test4. (>40)

We might need to remove noise for test1 and test2.

## Future Direction

### BinaryDecision Model

The task of dialog state tracking is to decide whether the NLU result is correct or not. It means that what we need to do is just to check a decision is right or wrong. Therefore, we can assume it is right and check whether there is a conflict between them using some AI logic derivation.

Possible rules:

* Only one value of the slot can be right

route=71A and route=71C are conflict

* For each turn, only one user slot can be identified. (Assume, only one information could be handled)

The topline model doesn’t follow this rule.

### Ranking Model

We can also treat the task as a ranking problem: rank the candidates and select the best one. Therefore, we can use some ranking model instead of classification model for this problem.

### Hierarchical Model

For this problem, if we identify a slot that should not be there, it will be wrong forever until it is been denied by the user. Therefore, it might be better if we can identify the slot exist or not first, and then assign the value to that slot with another model. I think, it will also improve the joint score.

### Matching Different Dialog System (Graph Matching Problem?)

The dialog manager, SLU, ASR are all different from different dialog system. Thus, if we can identify the relationship between different dialog systems, we can adapt it.

The thing we can match could be the dialog acts and the relationship between the acts.

### Add Bus Schedule Facts

We can filter out the names that don’t appear in the bus schedule database.

We can give more scores to specific street name if we know the route number. For example, University of Pittsburgh can get more scores for 71A; airport gets more scores for 28X.

### Clustering ASR Score

The assumption here is that if two ASR or NLU are similar with each other, they might be correct or they might be close to the correct one. For example, “from fifth avenue”, “town fifth avenue”, “ov fifth avenue”, “of fifth avenue”, etc. probably mean “fifth avenue” might be the correct one.

Thus, we group the ASR by word edit distance or phone edit distance. Then, infer the correct one by the groups.

# Open Question

## How to generate NBest lists for the joint?

The joint score for all systems are very low. That would be an interesting problem to solve.

However, I found I have no good idea to generate the NBest lists for the joint and rescore them.

### No Correct label?

The annotation is to just check whether each hypothesis is correct or not. But if all of them are wrong, it doesn’t provide the correct NLU. It doesn’t make sense.