# Summary

# Combining TopN (N-Best)

## Binary Switch Model

In this approach, the classifier will make a binary decision about whether the corresponding NLU is correct or not.

Therefore, the binary choice can serve as a binary switch to control whether to pick up the NLU or not.

In this method, all of NLU used the top ASR [It is not straightforward to get which ASR the NLU comes from. For example, the second NLU is not necessarily coming from the second ASR.]

### Result

First Correct

Trained on Top3 Model

Trained on Top1

## Bayes Rules Model

* Error Correction

HMM Model

# Method Classifier

* Add slot-name into feature
* Differentiate System and User Acts

# Build my Own SLU: From ASR directly to Goals

First Observations:

* All the slot-value pairs appear in the given ontology, except “dontcare”.
* Only 4 slots are in the goals: area, food, name, pricerange
* Number of Possible slot values are limited

|  |  |  |  |
| --- | --- | --- | --- |
| slot | | number of values | value examples |
| Goals | area | 5 | centre,north,west,south,east |
| food | 91 | catalan,chinese,christmas,corsica,creative,crossover,cuban |
| name | 113 | ali baba,anatolia,ask,backstreet bistro,bangkok city,bedouin |
| pricerange | 3 | cheap,moderate,expensive |

Thus, we can build classifiers to test whether there is such slot in this turn; Then, for “area” and “pricerange”, just build 3-way classifier and 3-way classifier; For “food” and “name”, we can we can use un-supervised method to find the possible values according to the ontology [like edit distance] (where ASR error correction applies).

This can be formulized as an example for multi-label classifiers. Take area as an example, it is a 7-way classifier: ***Area.No***, ***Area.Yes.Dontcare***, Area.Yes.Centre, Area.Yes.North, Area.Yes.West, Area.Yes.South, Area.Yes.East.

For food and name, only the first step will perform: binary classification.

# MindChang Model for Goals: make it independent

However, it has the similar issue as “method”: depending on the previous turn.

The MindChange model will make the goals independent with the previous one. It relies on the fact that only the changed goals matter. If the annotated goals don’t change, we can assume the user didn’t say new information in the current turn; thus, we can assume there are no goals in the turn.

Following this idea, one rule is used to recovery the current method label from the annotation:

* If the goal is only in the current turn and not in the previous one, use the labelled one as golden standard; Or else, assume the goal is not in the current turn.

## Results: