# Summary Report

# The Bottleneck of Goal Classifier

Now, the goal classifier model has 4 classifiers: area, food, pricerange and name.

For the food and name, if the classifier’s prediction is “Yes”, I will just search the first slot as the goal value.

Therefore, I want to test what’s the bottleneck of this approach.

I can have several different toplines.

* When area classifier is perfect
* When food classifier is perfect
* When priecerange classifier is perfect
* When name is perfect
* When slot of “Yes” for food and name is perfect
* When the four classifiers are perfect
* When the four classifiers and the slot of “Yes” is correct

The results are shown in next page.

* “2waymodel\_enrich\_more\_goals” is the current model
* “2waymodel\_goals\_topline\_slot” is the model When slot of “Yes” for food and name is perfect

## Result:

|  |  |  |  |
| --- | --- | --- | --- |
| method | test | Joint\_Goals\_accuracy | Joint\_Goals\_l2 |
| HWUbaseline | dstc2\_train | 0.732 | 0.451 |
| HWUbaseline | dstc2\_dev | 0.623 | 0.601 |
|  |  |  |  |
| 2waymodel\_enrich\_more\_goals | dstc2\_train | 0.711 | 0.577 |
| 2waymodel\_enrich\_more\_goals | dstc2\_dev | 0.561 | 0.878 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_area | dstc2\_train | 0.775 | 0.450 |
| 2waymodel\_goals\_topline\_area | dstc2\_dev | 0.654 | 0.692 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_food | dstc2\_train | 0.716 | 0.568 |
| 2waymodel\_goals\_topline\_food | dstc2\_dev | 0.584 | 0.832 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_name | dstc2\_train | 0.711 | 0.577 |
| 2waymodel\_goals\_topline\_name | dstc2\_dev | 0.562 | 0.875 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_pricerange | dstc2\_train | 0.747 | 0.507 |
| 2waymodel\_goals\_topline\_pricerange | dstc2\_dev | 0.621 | 0.757 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_slot | dstc2\_train | 0.725 | 0.550 |
| 2waymodel\_goals\_topline\_slot | dstc2\_dev | 0.578 | 0.843 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_4classifier | dstc2\_train | 0.817 | 0.366 |
| 2waymodel\_goals\_topline\_4classifier | dstc2\_dev | 0.770 | 0.459 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_all | dstc2\_train | 0.823 | 0.353 |
| 2waymodel\_goals\_topline\_all | dstc2\_dev | 0.779 | 0.442 |
|  |  |  |  |
| 2waymodel\_goals\_topline\_area\_pricerange | dstc2\_train | 0.815 | 0.369 |
| 2waymodel\_goals\_topline\_area\_pricerange | dstc2\_dev | 0.734 | 0.531 |

## Observations:

* It surprised me because I thought the main issue is from when predicting “Yes” for the food and name, because I just used the first available SLU to set the actual value.
* It turns out that the classifiers for “area” and “pricerange” are the bottleneck. If they are classified perfectly (shown in the last two rows in the table above), the performance is much better. It is even very close to the topline.

Then, the next step will improve these two classifiers.

# Whether a user answered the system?

When the system requests information from the user, I want to check whether the user gives the requested answer or just talks about others.

Here are the numbers I got.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | train | |  | test | |
|  |  |  | Yes | No |  | Yes | No |
| area | Requested | Yes | 445 | 242 |  | 574 | 351 |
|  |  | No | 898 | 10092 |  | 1172 | 13514 |
|  |  |  |  |  |  |  |  |
|  |  |  | Yes | No |  | Yes | No |
| food | Requested | Yes | 1192 | 617 |  | 1379 | 706 |
|  |  | No | 2553 | 22926 |  | 2978 | 26159 |
|  |  |  |  |  |  |  |  |
|  |  |  | Yes | No |  | Yes | No |
| name | Requested | Yes | 1379 | 706 |  | 1379 | 706 |
|  |  | No | 3029 | 37785 |  | 3054 | 41694 |
|  |  |  |  |  |  |  |  |
|  |  |  | Yes | No |  | Yes | No |
| pricerange | Requested | Yes | 1616 | 808 |  | 1694 | 847 |
|  |  | No | 3996 | 52090 |  | 4281 | 55622 |

It is like confusion matrix: 2\*2 matrix. For example, in the training data set, when the area is requested by the system, the user answered the area in 445 cases (system-initiative), but didn’t answer it in 242 cases; In addition, even though the system doesn’t ask about “area”, the user can give his constraints on “area”, this happens in 898 cases (user initiative)

The dialog manager seems very flexible. But, if the system does request something, the user has a bigger chance to answer it. However, simple rule-based method might not work.

# TODO:

Sequence Labeling of Goals, probably only for “area” and “pricerange”.