Domain-Intent-Slot model for task-based dialogue

A **domain** refers to what topic or subcategory of tasks the user is referring to.

An **intent** represents some discrete goal that a user is trying to accomplish.

Intents require varying amounts of information in order for a particular action to be taken. These pieces of information are referred to as **slots**.

It is the job of the natural language understanding engine to extract the relevant **entities** from the utterance in order to fill each of the slots necessary to complete the task. This is referred to as **slot filling**.

```
{
    "fetchBalance": {
        "accountType": "credit",
        "accountNumber": "0000-0000-0000-0000"
    }
}
```

Example of the intent **fetchBalance** and slots {**accountType**, **accountNumber**}

Entities

Intents are actually implicit entities

Not all implicit entities are intents

Two types of entities:
        * Explicit
        * Implicit

Four common scenarios:
        * All slot entities present
        * One+ slots unfilled
        * One+ slots implicit
        * Intent not recognized

Implicit entities/intents fall in 2 categories:
        * Structured (i.e., parsable to Struct.Q)
        * Unstructured (not parsable)

```
{
    "examples": [
        {
            "utterance": "check money in 0000 acct pls",
            "Intent": "fetchBalance",
            "entities": [
                {
                    "name": "accountNumber",
                    "value": "0000"
                }
            ]
        },
        {
            "utterance": "the last transaction on credit c is fraudulent",
            "Intent": "reportFraud",
            "entities": [
                {
                    "name": "Ordinal",
                    "value": "last"
                },
                {
                    "name": "account",
                    "value": "credit"
                },
                {
                    "name": "merchant",
                    "value": "???"
                }
            ]
        }
    ]
}
```