Treasure Hunting Data Documentation

Domain Knowledge Files

The following are user-independent files that are used to encode knowledge in the Treasure Hunting domain:

semanticDomainKnowledge.xml:

The domain model file defines (1) the semantic properties and terminology that can be used to refer to each object in the domain and (2) the semantic properties of object-independent terminology that can be used to refer to any object in the domain.

Each object must have a `type` property and may have additional properties (e.g., color, size, shape, material). Each object property is annotated with a concept (WordNet synset in the format of "`<word>#<pos-tag>#<sense-id>`". The `type` property has a list of singular and plural *gold standard* nouns that can be used to refer to the object. Other properties each have a list of *gold standard* attributes that correspond to the object. The *gold standard* terminology is manually compiled from all users' speech transcripts. Additional words are added for terms that can be spelled multiple ways (e.g. bathtub and bath_tub), with tokens in compound terms separated by an underscore. An example object is shown in Figure 1, where:

- Object `apple` has two properties: `type` and `color`.
- The concept of property `type` is "`apple#n#1`"; the concept of property `color` is "`color#n#1`".
- The *gold standard* singular nouns are "apple" and "fruit".
- The *gold standard* plural nouns are "apples", "fruit", and "fruits".
- The *gold standard* word for property `color` is "red".

```
<object name="apple">
  <properties>
    <type text="apple#n#1">
      <noun_singular>apple fruit</noun_singular>
      <noun_plural>apples fruit fruits</noun_plural>
    </type>
    <color text="color#n#1">red</color>
  </properties>
</object>
```

Figure 1: Example `object` in `semanticDomainKnowledge.xml` domain model file

The object-independent terminology includes adjectives, prepositions, definite and indefinite articles, demonstratives, locatives, pronouns, and numeric expressions. Each term may include zero or more properties with each property having a single value. Figure 2 shows an example demonstrative with three properties: `category_anaphora`, `anaphora_manner`, and `topic` having the respective values "yes", "demonstrative", and "collection".

```
<dem name="these">
  <properties>
    <category_anaphora value="yes"/>
    <anaphora_manner value="demonstrative"/>
    <topic value="collection"/>
  </properties>
</dem>
```

Figure 2: Example demonstrative (`dem`) in `semanticDomainKnowledge.xml` domain model file

objects.csv:

The comma-separated values file is composed of a list of objects in the Treasure Hunting virtual world along with their properties. Very similar objects are grouped into one semantic name. For example, each unique door object in the virtual world is semantically named a `door` object. Each row in the file contains an object ID, the room in the virtual world that contains this object, a list of constituent objects (delimited by a space character), and the semantic name. Three sample rows are shown in Table 1. Here, two specific door objects (`door_bathroom1` and `door_bathroom2`) compose the `door_bathroom` object. Each of these objects are in the bathroom and each of them have the semantic name `door`.

ID	ROOM	CONSTITUENTS	NAME
door_bathroom	bathroom	door_bathroom1 door_bathroom2	door
door_bathroom1	bathroom		door
door_bathroom2	bathroom		door

Table 1: Example row in `objects.csv` file

cnfParserules.csv:

The comma-separated values file defines the parsing grammar, which is shown in its entirety in Appendix B. The file is composed of a list of production rules, with one rule per row. In each row, the first column denotes a list of non-terminal symbols that can be generated from the non-terminal symbol in the second column.

User Data Files

The following are data files that were constructed via user studies in the Treasure Hunting domain:

*_annotated.xml

This log file contains a list of dialogue turns with zero or more `<user_input/>` tags and exactly one `<system_response/>` tag, denoting user and system dialogue turns, respectively. Each user input consists of recognized, transcribed, and annotated speech (with reference resolution tags) along with gaze fixation information. A sample user input is shown in Figure 3, where:

- The user's speech starts at "12868124101485" (system time in ms) and lasts for 8380 ms.
- The `<transcript/>` tag contains manually transcribed (gold standard) speech.
- The `<rr_annotation/>` tag contains reference resolution annotations of the speech transcript using the following XML-like markup—with angle brackets (`<`) replaced by square brackets (`[`):
    - The text in each `[ref/]` tag denotes a referring expression.
    - The `object` attribute denotes a set of objects referred to by the referring expression.
- The `<wavefile/>` tag contains the relative path to the audio file that was used to generate the recognized speech. Additionally, the wavefile name is used to uniquely identify the utterance (e.g. 20081010-105510-601).
- Each `<phrase/>` tag contains a recognition hypothesis (Microsoft Speech Recognizer).
- The `<gaze/>` tag contains all of the gaze fixations occurring concurrently with the speech input.
- Each gaze fixation (specified by `<gaze_fixation/>` tags) starts at a particular time (system time in ms) and is positioned on particular screen coordinates (e.g. `pos="568,364"`).
- Each `<gaze_fixation/>` tag contains a list of potentially fixated objects (specified by `<mesh/>` tags) along with their fixation probabilities. Multiple visually overlapping objects may be fixated simultaneously. The fixation probabilities are calculated based on the distance between the user's eye camera and an object: the i-th closest object is given the probability

$$p = \frac{1}{i}\left(\sum_{i=1}^{n}\frac{1}{i}\right)^{-1}$$

```
<user_input>
  <speech>
    <transcript>it’s a wooden desk with three drawers on the
        right side</transcript>
    <rr_annotation>[ref object="{desk_75}"]it’s[/ref]
        [ref object="{desk_75}"]a wooden desk[/ref] with
        [ref object="{desk_drawer1, desk_drawer2, desk_drawer3}"]
        three drawers[/ref] on the right side</rr_annotation>
    <wavefile>log\user2001\audio\20081010-105510-601.wav</wavefile>
    <phrase start="12868124101485" length="8380" rank="0">
      <token start="4290" length="390">and</token>
      <token start="4680" length="190">say</token>
      <token start="4870" length="360">wooden</token>
      <token start="5230" length="370">desk</token>
      <token start="5600" length="440">with</token>
      <token start="6070" length="380">three</token>
      <token start="6450" length="630">drawers</token>
      <token start="7170" length="260">from</token>
      <token start="7430" length="130">the</token>
      <token start="7560" length="300">right</token>
      <token start="7860" length="470">side</token>
    </phrase>
    <phrase start="12868124101485" length="8380" rank="1">
      ...
    </phrase>
    ...
  </speech>
  <gaze>
    <gaze_fixation start="12868124100940" length="40" pos="568,364">
      <mesh prob="0.55">computer_monitor</mesh>
      <mesh prob="0.27">desk_75</mesh>
      <mesh prob="0.18">castle</mesh>
    </gaze_fixation>
    <gaze_fixation start="12868124101080" length="20" pos="771,375">
      <mesh prob="0.67">computer_body</mesh>
      <mesh prob="0.33">castle</mesh>
    </gaze_fixation>
    ...
  </gaze>
<user_input>
```

Figure 3: Example user input in `*_annotated.xml` file

*_gsTurns.xml:

This log file contains a list of dialogue turns with zero or more `<user_input/>` tags and exactly one `<system_response/>` tag, denoting user and system dialogue turns, respectively. Each user input consists of speech (gold standard, i.e. transcribed and timestamped) and gaze fixation information, exemplified in Figure 4. In this example:

- The `<speech/>` tag contains two attributes: (1) a unique utterance id and (2) and a flag specifying whether the user's speech matches its accompanying gaze fixation. In this example, `matched="1"` because the user mentions an object (`desk_75`) that has been captured by a gaze fixation.
- The user's speech starts at "12868124101485" (system time in ms) and lasts for 8380 ms.
- Each `<phrase/>` tag contains the timestamped (in ms) tokens of the speech transcript.
- Each `<gaze/>` tag contains all of the gaze fixations occurring concurrently with the speech input.
- Each `<gaze_fixation/>` tag contains a list of potentially fixated objects (specified by `<mesh/>` tags) along with their fixation probabilities.

*_scene.xml:

The scene log file contains a record of visible entities on the screen each time the user's gaze is captured. An example record is shown in Figure 5, where:

- A user's gaze fixation lasting 80 ms is captured at time `"12868123694708"` (system time in ms) at position `"605,558"` (screen coordinates in pixels). This gaze fixation falls on two objects, specified by `<mesh/>` tags.
- The visible objects on the screen are specified in the `<scene/>` tag. Here, each `<mesh/>` tag contains the center position (screen coordinates: `pos="x,y"`) and the surrounding rectangle (screen coordinates: `rect="left,top,bottom,right"`) of the object's visible region.

```
<user_input>
  <speech utt_id="20081010-105510-601" matched="1">
    <transcript>it’s a wooden desk with three drawers on the right
    side</transcript>
    <phrase start="12868124101485" length="8380">
      <token start="4290" length="390">it’s</token>
      <token start="4680" length="190">a</token>
      <token start="4870" length="360">wooden</token>
      <token start="5230" length="370">desk</token>
      <token start="5600" length="440">with</token>
      <token start="6070" length="380">three</token>
      <token start="6450" length="630">drawers</token>
      <token start="7190" length="240">on</token>
      <token start="7430" length="130">the</token>
      <token start="7560" length="300">right</token>
    <token start="7860" length="390">side</token>
    </phrase>
  </speech>
  <gaze>
    <gaze_fixation start="12868124100940" length="40">
      <mesh prob="0.550000">computer_monitor</mesh>
      <mesh prob="0.270000">desk_75</mesh>
      <mesh prob="0.180000">castle</mesh>
    </gaze_fixation>
    <gaze_fixation start="12868124101080" length="20">
      <mesh prob="0.670000">computer_body</mesh>
      <mesh prob="0.330000">castle</mesh>
    </gaze_fixation>
    ...
  </gaze>
</user_input>
```

Figure 4: Example user input in `*_gsTurns.xml` file

nbest*.nbest:

This directory contains files generated from the `*_annotated.xml` log specifying an n-best list for each user utterance, identified by the utterance ID. The files use the SRI Decipher(TM) `NBestList2.0` format. Each recognition hypothesis in the file has the following format:

```
<record>
  <gaze_fixation start="12868123694708" length="80" pos="605,558">
    <mesh>door_dining</mesh>
    <mesh>castle</mesh>
  </gaze_fixation>
  <scene>
    <mesh pos="565,529" rect="487,552,573,578">sword_long</mesh>
    <mesh pos="599,502" rect="496,595,545,604">sword_short</mesh>
    <mesh pos="564,788" rect="244,0,968,1135">door_dining</mesh>
  </scene>
</record>
```

Figure 5: Example `record` in `*_scene.xml` file

```
(score) w1 ( st:  st1 et:  et1 g:  g1 a:  a1 ) w2 ...
```

Here, a word is followed by a start and end time, language model and acoustic score. The scores are in bytelog scale; a bytelog is a logarithm to base 1.0001, divided by 1024 and rounded to an integer. More information about this file format can be found in the SRILM manual:

`http://www-speech.sri.com/projects/srilm/manpages/nbest-format.5.html`

confusionNetwork*.cn:

This directory contains files specifying a confusion network (also called word mesh) for each user utterance, identified by the utterance ID. These files use the `wlat-format` (a file format for SRILM word posterior lattice) and are generated from the `nbest\*.nbest` files with the SRILM toolkit. The file is formatted as follows:

name s
numaligns N
posterior P
align $a\ w_1\ p_1\ w_2\ p_2\ ...$
info *a w start dur ascore gscore phones phonedurs*

$\ldots$

The file format specifies the name of the confusion network s, the number of alignment positions A and the total posterior probability mass P contained in the confusion network, followed by one or more confusion set specifications. For each alignment position a, the hypothesized words w_i and their posterior probabilities p_i are listed in alternation. The ***DELETE*** tag represents an empty hypothesis word. Following the **info** tag, word-level information is specified for alignment position a and hypothesized word w. For the purpose of this work, the word start time $start$ and duration dur (in ms) is considered. The remaining acoustic information is ignored. More information about this file format can be found in the SRILM manual:
`http://www-speech.sri.com/projects/srilm/manpages/wlat-format.5.html`