Speech Situation Frames Evaluation Specification v1.5

Nikolaos Malandrakis
Signal Analysis and Interpretation Laboratory (SAIL), USC, Los Angeles, CA 90089, USA
malandra@usc.edu

July 7, 2017

1 Task Definition

Given an audio segment in the incident language an SF system is expected to automatically identify any situation frames covered in the segment. A complete SF includes a document id, situation type, localization (optional) and a confidence score.

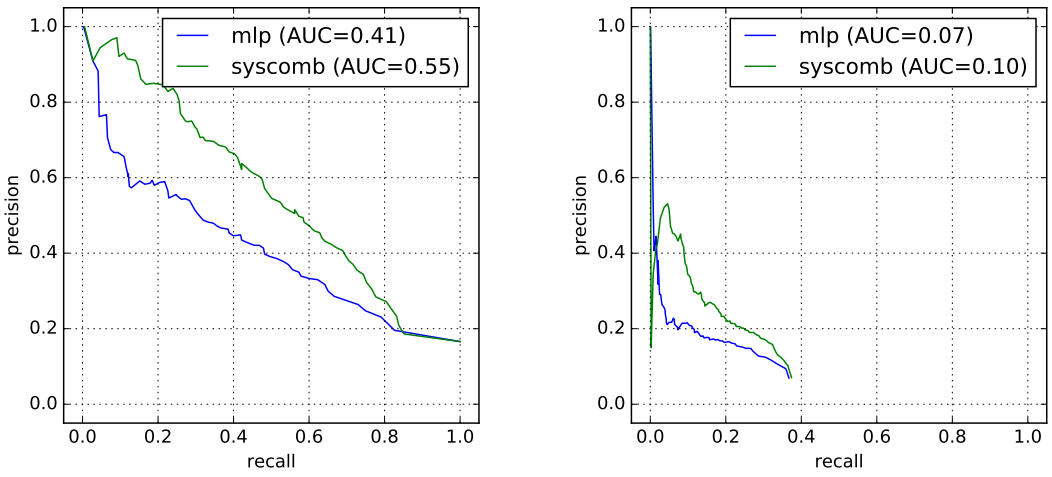
- Document ID: the file name of the corresponding audio segment (without extension)
- Situation Type: is a string corresponding to one of the pre-defined types, as defined in the Appen annotations.
- PlaceMention (optional): is a string - in the incident language script - indicating the physical place where the situation occurs.
- TypeConfidence: a number in $[0, 1]$ indicating the system's confidence that the frame exists. This is mandatory to allow for a curve-based evaluation.

Each system is expected to process all audio segments in a set and produce the corresponding frames.

2 Performance Measurement

In order to facilitate the creation of systems that can perform at various operating points, we will be performing a curve based evaluation. We will be using Precision-Recall (PR) curves, which allow the approach to generalize to the localization level (ROC and DET curves can not, due to the requirement for a True Negative estimate). For each system submission & for each layer of the evaluation a PR curve will be generated, with each point of the curve corresponding to a combination of micro-averaged recall and precision.
The curve will be produced by sweeping across the confidence values in the system output, using 100 percentiles at exponentially increasing intervals of the system output cardinality (higher resolution at low recall). Additionally, as an aggregate metric we will report the Area Under the Curve (AUC).
The process to estimate a single point on the PR curve is as follows:

1. Remove all frames below the current confidence threshold
2. Transform the remaining frames to the current evaluation layer, by removing extraneous attributes and merging duplicates.
3. Align the ground truth and output frames via maximum similarity



(a) (b)

Figure 1: PR curve examples, for (a) Type and (b) Type+Place, for 2 systems

4. Calculate True Positives, False Positives and False Negatives
5. Calculate Precision and Recall

Figure 1 shows two examples of PR curves at the Type and Type+Place layers. Note that the Type+Place curve never reaches 1 recall; that is expected and part of why we will be conducting visual comparisons of these curves rather than depending solely on AUC.
To allow for the creation of these curves, we encourage the submission of low confidence results. For "Type", participants are advised to produce all possible Types for every segment, even if they have a confidence score of zero.

2.1 Evaluation Layers

For the purposes of this evaluation we consider the following layers.

1. Relevance: "does this segment contain at least 1 frame of any type?". For this class all attributes are discarded, except for the document id.
2. Type: "which (if any) types of frames are contained in the segment?". For a frame to be correct at this layer, it has to have the correct document id and type.
3. Type+Place: "which (if any) types of frames are contained in the segment and where are they localized?". For a frame to be correct at this layer it needs to have the correct document id, type and location. Note that non-localized frames are ignored at at this layer.

Each participant will only need to submit a single output to be evaluated on one or more of these layers in order.

- An output containing localized frames will be evaluated on all 3 layers.
- An output not containing any localized frames, but including actual Types will be evaluated for Type and Relevance.

2.2 Frame similarity

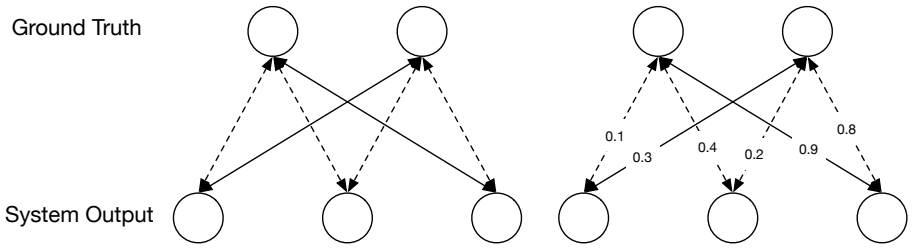
To allow for partial credit at the localization level, we are introducing the concept of frame similarity, indicated by a number in $[0, 1]$ with 1 indicating a perfect match.
For the Relevance and Type layers of the evaluation the calculation is trivial: the frames are either perfectly matched or not, giving the similarity metric values of 1 and 0 respectively. For the Type+Place layer, we will be using a soft matching of the PlaceMention strings and the similarity between two frames (if Type and Document ID match) will be equal to that string similarity measure.
String similarity is defined as the character-level edit distance between the two PlaceMentions, normalized by the sum of their string lengths:

$$Similarity = \frac{sum(length) - minimum\ edit\ distance}{sum(length)}. \quad (1)$$

This metric takes values in $[0, 1]$. The edit distance is calculated using costs of 1 for insertions and deletions and 2 for substitutions.

2.3 Frame alignment

The frames in the ground truth and system output are aligned using a maximum similarity criterion. All pair-wise similarities are calculated and, using a linear assignment algorithm, each frame in the output is mapped to 0 or 1 frames in the ground truth in such a way as to maximize the sum of similarities. The mappings are 1-to-1, no frame may be matched more than once.
An example is shown below, for the case of hard and soft matching.



The solid arrows represent the frame alignment and, in the case of soft matching, the arrows have similarity scores on them.
The scoring takes into account the similarity scores and gives partial credit, by using soft set cardinality. For the hard matching example, the scoring would be:

- True positive = 2
- False negative = 0
- False positive = 1

Whereas the soft matching example would yield:

- True positive = 0.9+0.3 = 1.2
- False negative = 2 (reference cardinality) - 1.2 = 0.8
- False positive = 3 (output cardinality) - 1.2 = 1.8

3 Output format

The system output is a single json file with a structure that adheres to the schema in the following page. Note that while the schema allows for the inclusion of the status variables “Need” and “Relief”, they will not be evaluated during the first year pilot of the task.
A complete frame would look like this:


```
{
"DocumentID": "CHN_EVAL_096_004",
"PlaceMention": "\u6c5f\u82cf",
"Type": "Medical Assistance",
"TypeConfidence": 0.5585732473158215
}
```


Note the unicode encoding of the “PlaceMention” string. A valid system output can use either proper Unicode characters in the native script or their u−code versions.
The complete system output contains a list of Situation Frames, separated by commas and enclosed in square brackets (also see the attached evaluation script & sample output.

The JSON schema.


```
{
   "$schema": "http://json-schema.org/draft-04/schema#",
   "$version": "1.0",
   "definitions": {
      "frame": {
         "type": "object",
         "properties": {
            "DocumentID": { "type": "string" },
            "Type": { "type": "string",
               "enum": [ "Civil Unrest or Wide-spread Crime",
               "Elections and Politics",
               "Evacuation",
               "Food Supply",
               "Infrastructure",
               "Medical Assistance",
               "Shelter",
               "Terrorism or other Extreme Violence",
               "Urgent Rescue",
               "Utilities, Energy, or Sanitation",
               "Water Supply" ] },
            "TypeConfidence": { "type": "number", "minimum": 0, "maximum": 1 },
            "PlaceMention": { "type": "string" },
            "Status": {
               "type": "object",
               "properties": {
                  "Need": {
                     "type": "string",
                     "enum": [ "Current",
                     "Future",
                     "Past Only" ] },
                  "Relief": {
                     "type": "string",
                     "enum": [ "Insufficient/Unknown",
                     "No_Known_Resolution",
                     "Sufficient" ] }
               },
               "required": [ "Need", "Relief" ]
            }
         },
         "required": ["DocumentID", "Type", "TypeConfidence"]
      }
   },
   "type": "array",
   "items": {
      "$ref": "#/definitions/frame"
   }
}
```

3.1 Frame examples - with layers

A complete frame, including status variables (which will be ignored during the evaluation)


```
{
"DocumentID": "CHN_EVAL_096_004",
"PlaceMention": "\u6c5f\u82cf",
"Status": {
    "Need": "Past Only",
    "Relief": "No_Known_Resolution"
},
"Type": "Medical Assistance",
"TypeConfidence": 0.5585732473158215
}
```


A localized frame.


```
{
"DocumentID": "CHN_EVAL_096_004",
"PlaceMention": "\u6c5f\u82cf",
"Type": "Medical Assistance",
"TypeConfidence": 0.5585732473158215
}
```


A non-localized frame. This is the minimum information required for a frame to be valid.


```
{
"DocumentID": "CHN_EVAL_096_004",
"Type": "Medical Assistance",
"TypeConfidence": 0.5585732473158215
}
```


4 The Appen annotations and Special Cases

The Appen annotations look like this:

```
TYPE: Type1
TIME: Past Only
Resolution: Sufficient
PLACE: Place1
```

Each annotation includes these 4 lines and each audio segment may correspond to multiple of these 4 line combinations. However, these lines may include multiple Types and locations. For example:

```
TYPE: Type1, Type2
TIME: Past Only
Resolution: Sufficient
PLACE: Place1
```

This, for the purposes of this evaluation, counts as two frames, both localized to Place1, with Types being Type1 and Type2. **In the cases where there is 1 Type & multiple locations or multiple Types & 0 or 1 locations we consider each possible combination of Type and location as a separate frame.**
A special case is when this structure contains multiple Types and multiple locations, like below:

```
TYPE: Type1, Type2
TIME: Past Only
Resolution: Sufficient
PLACE: Place1, Place2
```

This is meant be read as: "Type1 at Place1 or Place2 or both" and "Type2 at Place1 or Place2 or both". So each type may be connected to either or both types, it is ambiguous.
It is clear how to evaluate this at the "Type" layer: all types must be assigned to the segment. It is not clear how we may evaluate at the "Type+Place" layer, due to the ambiguity: if a system output contains "Type1 at Place2", we do not know if that is correct, since Type1 may only apply to Place1. Only a very small percentage of all annotations fall under this special case, so our current plan (unless there is a better suggestion) is to **ignore** these segments when evaluating at the Type+Place layer. They will be taken into account when evaluating at the Type and Relevance layers.