Social Determinants of Health Extraction Challenge - Evaluation Criteria

December 22, 2021

Annotation structure

Social determinants of health (SDOH) are annotated as events using the BRAT rapid annotation tool (Stenetorp et al., 2012). Figure 1 is a BRAT annotation example, describing a patient's employment and substance use. Each event includes exactly one trigger (shown in white) and one or more arguments that characterize the event. There are two categories of arguments: *span-only* (shown in green) and *span-with-value* (shown in blue). The trigger anchors and disambiguate events and indicates the event type (e.g. *Employment* or *Tobacco*). *Span-only arguments* include an annotated span and argument type (e.g. *Duration* or *History*). *Span-with-value arguments* include an annotated span, argument type (e.g. *StatusTime* or *StatusEmploy*), and argument subtype (e.g. *past* or *unemployed*). The triggers connect to arguments through argument roles. The argument roles that connect triggers and arguments can be interpreted as binary connectors because there is only one valid argument role for each argument type. For example, all *StatusTime* arguments connect to triggers through a *Status* argument role, so the label associated with the argument role does not add information.

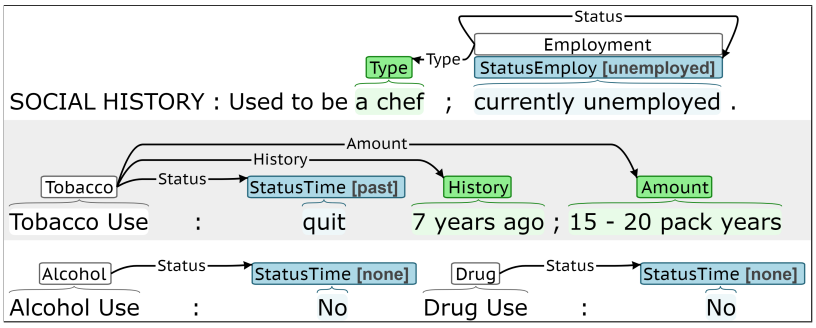

Figure 1: BRAT annotation example

Evaluation criteria

The evaluation criteria interprets the SDOH event extraction task as a slot filling task, as this is most relevant to secondary use applications. As such, there can be multiple equivalent span annotations. Figure 2 presents the same sentence with two sets of annotations, A and B, along with the populated slots.

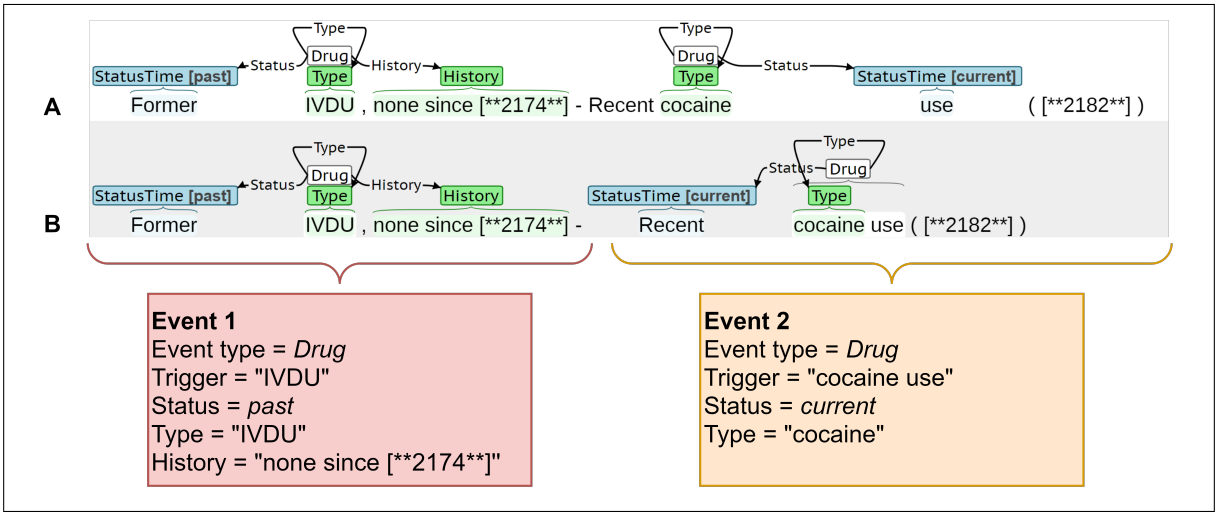

Figure 2: Annotation examples describing event extraction as a slot filling task

Both annotations identify two *Drug* events: *Event 1* and *Event 2*. Event 1 describes past intravenous drug use (IVDU), and Event 2 describes current cocaine use. Event 1 is annotated identically by both annotators. However, there are differences in the annotated spans of Event 2, specifically for the *Trigger* ("cocaine" versus "cocaine use") and *StatusTime* ("use" vs. "Recent"). From a slot perspective, the annotations for Event 2 could be considered equivalent. The scoring criteria include relaxed match criteria that reflect the clinical meaning of the extracted phenomena. Performance is evaluated using precision (P), recall (R), and F1, micro averaged over the event types, argument types, and/or argument subtypes.

Trigger: The i^{th} trigger, T_i, is defined by the event type, e_i, and character indices, x_i. Trigger equivalence is defined as

$$T_i \equiv T_j \text{ if } (e_i \equiv e_j) \wedge (x_i \equiv x_j). \tag{1}$$

The equivalence of the triggers spans, x, can be assessed using the following criteria:

- *exact*: $x_i \equiv x_j$ if x_i matches x_j exactly
- *overlap*: $x_i \equiv x_j$ if x_i overlaps x_j by at least one character
- *min distance*: Triggers of the same event type are aligned by minimizing the distance between the span centers of the characters indices. Triggers that are aligned using this distance criterion are considered equivalent.

For *Event 2* in Figure 2, let the trigger annotation in A be $T_i = (e_i = Drug; x_i = [45, 52])$ and the trigger annotation in B be $T_j = (e_j = Drug; x_j = [45, 56])$. T_i is not equivalent to T_j under the *exact* criterion; however, T_i is equivalent to T_j under the *overlap* and *min distance* criteria.

Arguments: Events are aligned based on trigger equivalence, and the arguments of aligned events are compared using different criteria for *span-only arguments* and *span-with-value arguments*.

Span-only arguments: $S_{i,k}$ is k^{th} argument connected to the i^{th} trigger. The span-only argument, $S_{i,k}$, is defined by the argument type, $a_{i,k}$, character indices, $x_{i,k}$, and connection to T_i. Span-only argument equivalence is defined as

$$S_{i,k} \equiv S_{j,l} \text{ if } (T_i \equiv T_j) \wedge (a_{i,k} \equiv a_{j,l}) \wedge (x_{i,k} \equiv x_{j,l}). \tag{2}$$

The equivalence of the span-only argument spans, x, can be assessed using the following criteria:

- *exact*: $x_{i,k} \equiv x_{j,l}$ if $x_{i,k}$ matches $x_{j,l}$ exactly
- *overlap*: $x_{i,k} \equiv x_{j,l}$ if $x_{i,k}$ overlaps $x_{j,l}$

Span-only arguments can also be compared at the token-level when the arguments have equivalent triggers and argument types, $(T_i \equiv T_j) \wedge (a_{i,k} \equiv a_{j,l})$. This token-level assessment is referred to as *partial*. Partial match scoring is relevant because partial matches can still contain useful information. Note that the *exact* and *overlap* criteria count equivalent spans, and the *partial* criterion counts equivalent tokens. For *Event 2* in Figure 2, let the *Type* annotation in B for "cocaine" can be represented as $S_{i,k} = (a_{i,k} = Type; x_{i,k} = [45, 52])$.

Span-with-value arguments: $L_{i,k}$ is k^{th} argument connected to the i^{th} trigger. The span-with-value argument, $L_{i,k}$, is defined by the argument type, $a_{i,k}$, character indices, $x_{i,k}$, subtype, $s_{i,k}$, and connection to T_i. Span-with-value equivalence is defined as

$$L_{i,k} \equiv L_{j,l} \text{ if } (T_i \equiv T_j) \wedge (a_{i,k} \equiv a_{j,l}) \wedge (x_{i,k} \equiv x_{j,l}) \wedge (s_{i,k} \equiv s_{j,l}). \tag{3}$$

For span-with-value arguments, the argument type, a, and subtype, s, capture the salient information. The equivalence of the span-with-value argument spans, x, can be assessed using the following criteria:

- *exact*: $x_{i,k} \equiv x_{j,l}$ if $x_{i,k}$ matches $x_{j,l}$ exactly
- *overlap*: $x_{i,k} \equiv x_{j,l}$ if $x_{i,k}$ overlaps $x_{j,l}$
- *label*: span not considered, such that $x_{i,k}$ always consider equivalent to $x_{j,l}$

For *Event 2* in Figure 2, let the *StatusTime* annotation in A be $L_{i,k} = (a_{i,k} = Status; x_{i,k} = [53, 56], s_{i,k} = current)$ and the *StatusTime* annotation in B be $L_{j,l} = (a_{j,l} = Status; x_{j,l} = [38, 44], s_{j,l} = current)$. $L_{i,k}$ is not equivalent to $L_{j,l}$ under the *exact* criterion, but $L_{i,k}$ is equivalent to $L_{j,l}$ under the *overlap* and *label* criteria.

Evaluation script

The scoring script, `score_brat.py`, implements the aforementioned evaluation by comparing two directories with BRAT-style annotations (*.txt and *.ann files). The scoring routine identifies all the *.ann files in both directories, finds matching filenames in the directories, and then compares the annotations defined in the *.ann files.

`score_brat.py` can be called from command-line. The required arguments that define the input and output paths include:

- `gold_dir`: *str*, path to the input directory with gold annotations in BRAT format, e.g. '/home/gold/'
- `predict_dir`: *str*, path to the input directory with predicted annotations in BRAT format, e.g. '/home/predict/'
- `output`: *str*, path for the output CSV file that will contain the evaluation results, e.g. , '/home/scoring.csv'

The optional arguments define the evaluation criteria:

- `score_trig`: *str*, trigger scoring criterion, options include {'exact', 'overlap', 'min_dist'}
- `score_span`: *str*, span-only argument scoring criterion, options include {'exact', 'overlap', 'partial'}
- `score_labeled`: *str*, span-with-value argument scoring criterion, options include {'exact', 'overlap', 'label'}

Below is an example usage:

```
python3 score_brat.py /home/gold/ /home/predict/ /home/scoring.csv
--score_trig min_dist --score_span exact --score_labeled label
```

References

P. Stenetorp, S. Pyysalo, G. Topić, T. Ohta, S. Ananiadou, and J. Tsujii. BRAT: a web-based tool for NLP-assisted text annotation. In *Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, 2012. URL `https://www.aclweb.org/anthology/E12-2021`.