A Illustration of the metrics used in this paper

There are three widely-used evaluation metrics for the TempRel extraction task. The first standard metric is classification *accuracy* (Acc). The second metric is to view this task as a general relation extraction task, treat the label of *vague* for TempRel as *no relation*, and then compute the precision and recall. The third metric came into use since the TempEval3 workshop (UzZaman et al., 2013), which involves graph closure and reduction on top of the second metric, hoping to better capture how useful a TempRel system is.. Take the confusion matrix in Fig. 2 for example. The three metrics used in this paper are

1. Accuracy. $Acc = (C_{b,b} + C_{a,a} + C_{e,e} + C_{v,v})/S$.
2. Precision, recall, and F_1. $P = (C_{b,b} + C_{a,a} + C_{e,e})/S_1$, $R = (C_{b,b} + C_{a,a} + C_{e,e})/S_2$, and $F_1 = 2PR/(P + R)$.
3. Awareness score F_{aware}. Before calculating precision, perform a graph closure on the gold temporal graph and a graph reduction on the predicted temporal graph. Similarly, before calculating recall, perform a graph reduction on the gold temporal graph and a graph closure on the predicted temporal graph. Finally, compute the F_1 score based on this revised precision and recall. Since graph reduction and closure are involved in computing this metric, the temporal graphs all need to satisfy the global transitivity constraints of temporal relations (e.g., if A happened *before* B, and B happened *before* C, then C cannot be *before* A).

In this paper, we also report the average of the three metrics above, which we call *three-metric-average*.

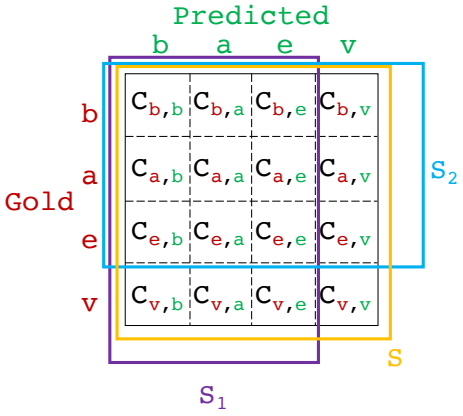

Figure 2: An example confusion matrix, where the four labels are *before* (b), *after* (a), *equal* (e), and *vague* (v), respectively. The variables, S, S_1, and S_2, are the summation of all the numbers in the corresponding area. (This figure is better viewed in color)

B Significance test for Tables 2-3

In Table 2, we mainly compared the performance of position indicator (P.I.) and simple concatenation (Concat), using 5 different word embeddings and 3 metrics, so there were 15 performances for both P.I. and Concat. Under the paired t-test, Concat is significantly better than P.I. with $p < 0.01$.

Another observation we had in Table 2 was that contextualized embeddings, i.e., ELMo and BERT, were much better than conventional ones, i.e., word2vec, GloVe and FastText. For both P.I. and Concat, we found that the difference between contextualized embeddings and conventional embeddings was significant with $p < 0.001$ under the McNemar's test (Everitt, 1992; Dietterich, 1998); however, between the two contextualized embeddings, ELMo and BERT, we did not see a significant difference, although it has been reported that in many *other* tasks, that BERT is better than ELMo.

In Table 3, we further improved Concat using the proposed common sense encoder (CSE). Under the McNemar's test, Concat+CSE was significantly better than Concat with $p < 0.001$, no matter either ELMo or BERT was used. Again, no significant difference was observed between ELMo and BERT.

Finally, since Concat+CSE improved over CogCompTime by a large margin either on MATRES or on TCR, it was not surprising to see that the proposed Concat+CSE is significantly better than CogCompTime with $p < 0.001$ as well.