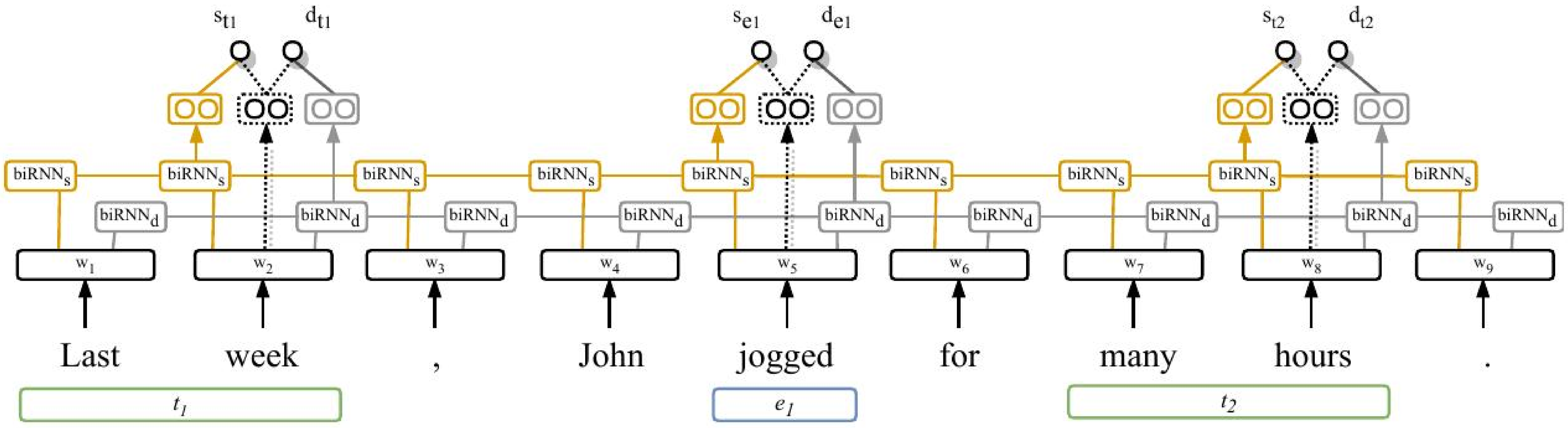

Figure 2: Schematic overview of our two time-line models: C-TLM (solid edges), exploiting entity context, and the simpler S-TLM (dotted edges), which is context independent. The models predict a starting point (s) and duration (d) for each given temporal entity (t_1, e_1, and t_2) in the input.

In contrast, DCT duration d_{DCT} is modeled as a single variable that is learned (initialized with 1). Since multiple entities may be included in the DCT, and entities have a minimum duration d_{min}, a constant d_{DCT} could possibly prevent the model from fitting all entities in the DCT. Modeling d_{DCT} as a variable allows growth of d_{DCT} and averts this issue.[3]

Training Losses

We propose three loss functions to train time-line models from TimeML-style annotations: a regular time-line loss L_τ, and two slightly expanded discriminative time-line losses, $L_{\tau ce}$ and $L_{\tau h}$.

Regular Time-line Loss (L_τ)

Ground-truth TLinks can be seen as constraints on correct positions of entities on a time-line. The regular time-line loss L_τ expresses the degree to which these constraints are met for a predicted time-line. If all TLinks are satisfied in the time-line for a certain text, L_τ will be 0 for that text.

As TLinks relate entities (intervals), we first convert the TLinks to expressions that relate the start and end points of entities. How each TLink is translated to its corresponding point-algebraic constraints is given in Table 1, following Allen (1990).

As can be seen in the last column there are only two point-wise operations in the point-algebraic constraints: an order operation ($<$), and an equality operation ($=$). To model to what degree each point-wise constraint is met, we employ hinge losses, with a margin m_τ, as shown in Eq. 2.

[3]Other combinations of modeling s_{DCT} and d_{DCT} as variable or constant decreased performance.

[4]No TLink for Allen's overlap relation is present in TimeML, also concluded by UzZaman and Allen (2011).

Table 1: Point algebraic interpretation (I_{PA}) of temporal links used to construct the loss function. The start and end points of event X are indicated by s_x and e_x respectively.

Allen Algebra	Temporal Links	Point Algebra
X precedes Y Y preceded by X	X before Y Y after X	$e_x < s_y$
X starts Y Y started by X	X begins Y Y begun by X	$s_x = s_y$ $e_x < e_y$
X finishes Y Y finished by X	X ends Y Y ended by X	$e_x = e_y$ $s_y < s_x$
X during Y Y includes X	X is included Y Y includes X	$s_y < s_x$ $e_x < e_y$
X meets Y Y met by X	X immediately before Y Y immediately after X	$e_x = s_y$
X overlaps Y Y overlapped by X	absent absent	$s_x < s_y$ $s_y < e_x$ $e_x < e_y$
X equals Y	X simultaneous Y X identity Y	$s_x = s_y$ $e_x = e_y$

To explain the intuition and notation: If we have a point-wise expression ξ of the form $x < y$ (first case of Eq. 2), then the predicted point $\hat{x}$ should be at least a distance m_τ smaller (or earlier on the time-line) than predicted point $\hat{y}$ in order for the loss to be 0. Otherwise, the loss represents the distance $\hat{x}$ or $\hat{y}$ still has to move to make $\hat{x}$ smaller than $\hat{y}$ (and satisfy the constraint). For the second case, if ξ is of the form $x = y$, then point $\hat{x}$ and $\hat{y}$ should lie very close to each other, i.e. at most a distance m_τ away from each other. Any distance further than the margin m_τ is counted as loss. Notice that if we set margin m_τ to 0, the second case becomes an L1 loss $|\hat{x} - \hat{y}|$. However, we use a small margin m_τ to promote some distance between ordered points and prevent con-