Matrix-Level Documentation of Processing Steps

WORK IN PROGRESS

Pre-Processing (Common)

Text Input

This is the "user" input for the whole processing.

Source Code

- Related struct: `gliner::model::input::text::TextInput`

Format

- n: number of input texts
- k: number of entity class labels
- I: sequence of input texts matrix of type `string` and size n
- E: entity class labels matrix, of type `string` and size k

$$I = \begin{bmatrix} \text{text}_1 \\ \text{text}_2 \\ \vdots \\ \text{text}_n \end{bmatrix}$$

$$E = \begin{bmatrix} \text{label}_1 \\ \text{label}_2 \\ \vdots \\ \text{label}_k \end{bmatrix}$$

Example

$$I = \begin{bmatrix} \text{"My name is James Bond"} \\ \text{"I like to drive my Aston Martin"} \end{bmatrix}$$

$$E = \begin{bmatrix} \text{"movie character"} \\ \text{"vehicle"} \end{bmatrix}$$

Word-Level Tokenization

Transformation

$(I, E) \to (T, E)$

Source Code

- Struct: `gliner::model::input::tokenized::TokenizedInput`
- Transformation: `gliner::model::input::prompt::RawToTokenized`

Format

- n, k: same as before
- T: sequence of sequence of tokenized input texts, of type `string` and size n
- E: same as before

$$T = \begin{bmatrix} [\text{token}_{1,1} & \text{token}_{1,2} & ...] \\ [\text{token}_{2,1} & \text{token}_{2,2} & ...] \\ & \vdots & \\ [\text{token}_{n,1} & \text{token}_{n,2} & ...] \end{bmatrix}$$

Example

$$T = \begin{bmatrix} [\text{"My"} & \text{"name"} & \text{"is"} & \text{"James"} & \text{"Bond"}] & & \\ [\text{"I"} & \text{"like"} & \text{"to"} & \text{"drive"} & \text{"my"} & \text{"Aston"} & \text{"Martin"}] \end{bmatrix}$$

Prompt Preparation

Prepared prompts, appending entity and text tokens.

Transformation

$(T, E) \to P$

Source Code

- Struct: `gliner::model::input::prompt::PromptInput`
- Transformation from `TokenizedInput`: `gliner::model::input::prompt::TokenizedToPrompt`

Format

$$P = \begin{bmatrix} [\text{<<ENT>>} \ \text{label}_{1,1} \ \text{<<ENT>>} \ \text{label}_{1,2} \ \ldots \ \text{<<SEP>>} \ \text{token}_{1,1} \ \text{token}_{1,2} \ \ldots] \\ [\text{<<ENT>>} \ \text{label}_{2,1} \ \text{<<ENT>>} \ \text{label}_{2,2} \ \ldots \ \text{<<SEP>>} \ \text{token}_{2,1} \ \text{token}_{2,2} \ \ldots] \\ \vdots \\ [\text{<<ENT>>} \ \text{label}_{n,1} \ \text{<<ENT>>} \ \text{label}_{n,2} \ \ldots \ \text{<<SEP>>} \ \text{token}_{n,1} \ \text{token}_{n,2} \ \ldots] \end{bmatrix}$$

Example

$$P = \begin{bmatrix} [\text{<<ENT>>} \ \text{"movie character"} \ \text{<<ENT>>} \ \text{"vehicle"} \ \ldots \ \text{<<SEP>>} \ \text{"My"} \ \text{"name"} \ \text{"is"} \ \text{"James"} \ \text{"Bond"}] \\ [\text{<<ENT>>} \ \text{"movie character"} \ \text{<<ENT>>} \ \text{"vehicle"} \ \ldots \ \text{<<SEP>>} \ \text{"I"} \ \text{"like"} \ \text{"to"} \ \text{"drive"} \ \text{"my"} \ \text{"Austin"} \ \text{"Martin"}] \end{bmatrix}$$

Prompt Encoding (Sub-Word Tokenization)

Transformation

$P \rightarrow (I, M, W, L)$

Source Code

- Struct: `gliner::model::input::encoded::EncodedPrompt`
- Transformation: `gliner::model::input::encoded::PromptsToEncoded`

Format

- k: maximum number of sub-word tokens within a sequence, adding start (1) and end (2) tokens
- I: encoded prompts of type `i64` and shape $(n * k)$
- A: attention masks of type `i64` and shape $(n * k)$
- W: word masks of type `i64` and shape $(n * k)$
- L: text lengths of type `i64` and shape $(n * 1)$

$$I = \begin{pmatrix} \text{token\_id}_{1,1} & \text{token\_id}_{1,2} & \dots & \text{token\_id}_{1,k} \\ \text{token\_id}_{2,1} & \text{token\_id}_{2,2} & \dots & \text{token\_id}_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ \text{token\_id}_{n,1} & \text{token\_id}_{n,2} & \dots & \text{token\_id}_{n,k} \end{pmatrix}$$

$$M = \begin{pmatrix} \text{mask}_{1,1} & \text{mask}_{1,2} & \dots & \text{mask}_{1,k} \\ \text{mask}_{2,1} & \text{mask}_{2,2} & \dots & \text{mask}_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ \text{mask}_{n,1} & \text{mask}_{n,2} & \dots & \text{mask}_{n,k} \end{pmatrix}$$

$$W = \begin{pmatrix} \text{word\_mask}_{1,1} & \text{word\_mask}_{1,2} & \dots & \text{word\_mask}_{1,k} \\ \text{word\_mask}_{2,1} & \text{word\_mask}_{2,2} & \dots & \text{word\_mask}_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ \text{word\_mask}_{n,1} & \text{word\_mask}_{n,2} & \dots & \text{word\_mask}_{n,k} \end{pmatrix}$$

$$L = \begin{pmatrix} l_1 \\ \vdots \\ l_n \end{pmatrix}$$

Example

$$I = \begin{pmatrix} 1 & 128002 & 1421 & 1470 & 128002 & 1508 & 128003 & 573 & 601 & 269 & 1749 & 8728 & 2 & 0 & 0 \\ 1 & 128002 & 1421 & 1470 & 128002 & 1508 & 128003 & 273 & 334 & 264 & 1168 & 312 & 20844 & 2963 & 2 \end{pmatrix}$$

$$M = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}$$

$$W = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 4 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 0 \end{pmatrix}$$

$$L = \begin{pmatrix} 5 \\ 7 \end{pmatrix}$$

Pre-Processing (Span Mode)

Downstream of the aforementioned steps.

Span Preparation

Transformation

$(I, M, W, L) \rightarrow (I, M, W, L, S_I, S_M)$

Format

- n, k, I, A, W, L: same as before.
- s: maximum possible number of spans for one sequence
- S_I: span offsets, of type `i64` and shape $(n * s * 2)$
- S_M: span masks, of type `bool` and shape $(n * s)$

$$S_I = \begin{pmatrix} (\text{start}_{1,1} \;\; \text{end}_{1,1}) & (\text{start}_{1,2} \;\; \text{end}_{1,2}) & \dots & (\text{start}_{1,s} \;\; \text{end}_{1,s}) \\ (\text{start}_{2,1} \;\; \text{end}_{2,1}) & (\text{start}_{2,2} \;\; \text{end}_{2,2}) & \dots & (\text{start}_{2,s} \;\; \text{end}_{2,s}) \\ \vdots & \vdots & \ddots & \vdots \\ (\text{start}_{n,1} \;\; \text{end}_{n,1}) & (\text{start}_{n,2} \;\; \text{end}_{n,2}) & \dots & (\text{start}_{n,s} \;\; \text{end}_{n,s}) \end{pmatrix}$$

$$S_M = \begin{pmatrix} \text{span\_mask}_{1,1} & \text{span\_mask}_{1,2} & \dots & \text{span\_mask}_{1,s} \\ \text{span\_mask}_{2,1} & \text{span\_mask}_{2,2} & \dots & \text{span\_mask}_{2,s} \\ \vdots & \vdots & \ddots & \vdots \\ \text{span\_mask}_{n,1} & \text{span\_mask}_{n,2} & \dots & \text{span\_mask}_{n,s} \end{pmatrix}$$

Example

Note: for readability purposes, inside matrices are split into rows (one per token) but they are actually in one dimension s (see format above).

$$S_I = \begin{pmatrix} \begin{pmatrix} (0\;0) & (0\;1) & (0\;2) & (0\;3) & (0\;4) & (0\;0) & (0\;0) & (0\;0) & (0\;0) & (0\;0) & (0\;0) & (0\;0) & \hookleftarrow \\ (1\;1) & (1\;2) & (1\;3) & (1\;4) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (2\;2) & (2\;3) & (2\;4) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (3\;3) & (3\;4) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (4\;4) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \end{pmatrix} \\ \begin{pmatrix} (0\;0) & (0\;1) & (0\;2) & (0\;3) & (0\;4) & (0\;5) & (0\;6) & (0\;0) & (0\;0) & (0\;0) & (0\;0) & (0\;0) & \hookleftarrow \\ (1\;1) & (1\;2) & (1\;3) & (1\;4) & (1\;5) & (1\;6) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (2\;2) & (2\;3) & (2\;4) & (2\;5) & (2\;6) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (3\;3) & (3\;4) & (3\;5) & (3\;6) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (4\;4) & (4\;5) & (4\;6) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (5\;5) & (5\;6) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \hookleftarrow \\ (6\;6) & (0\;0) & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \end{pmatrix} \end{pmatrix}$$

$$S_M = \left(\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \end{pmatrix} \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \leftarrow \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \end{pmatrix} \right)$$

Pre-Processing (Token Mode)

Nothing more to be done beside the common steps.

Post-Processing (Span Mode)

TODO

Post-Processing (Token Mode)

TODO