# Hierarchical Event Grounding

**Jiefu Ou, Adithya Pratapa, Rishubh Gupta, Teruko Mitamura**

Language Technologies Institute, Carnegie Mellon University
{jiefuo, vpratapa, rishubhg, teruko}@andrew.cmu.edu

## Abstract

Event grounding aims at linking mention references in text corpora to events from a knowledge base (KB). Previous work on this task focused primarily on linking to a single KB event, thereby overlooking the hierarchical aspects of events. Events in documents are typically described at various levels of spatio-temporal granularity. These hierarchical relations are utilized in downstream tasks of narrative understanding and schema construction. In this work, we present an extension to the event grounding task that requires tackling hierarchical event structures from the KB. Our proposed task involves linking a mention reference to a set of event labels from a subevent hierarchy in the KB. We propose a retrieval methodology that leverages event hierarchy through an auxiliary hierarchical loss. On an automatically created multilingual dataset from Wikipedia and Wikidata, our experiments demonstrate the effectiveness of the hierarchical loss against retrieve and re-rank baselines. Furthermore, we demonstrate the systems' ability to aid hierarchical discovery among unseen events. Code is available at https://github.com/JefferyO/Hierarchical-Event-Grounding

## Introduction

Grounding entity and event references from documents to a large-scale knowledge base (KB) is an important component in the information extraction stack. While entity linking has been extensively explored in the literature (Ji and Grishman 2011), event linking is relatively unexplored.[1] Recently, Pratapa, Gupta, and Mitamura (2022) presented a dataset for linking event references from Wikipedia and Wikinews articles to Wikidata KB. However, this work limited the grounding task to a subset of events from Wikidata, missing out on hierarchical event structures available in Wikidata.

Text documents often describe events at varying levels of spatio-temporal granularity. Figure 1 illustrates this through three text snippets from English Wikipedia. The mentions 'Falaise Gap', 'Normandy campaign', and 'western campaign' refer to three separate events from Wikidata.[2] These three events (Q602744, Q8641370, Q216184) themselves constitute a hierarchical event structure.

Prior work studied hierarchical relations between events as a part of datasets and systems proposed for narrative understanding (Glavaš et al. 2014), event sequencing (Mitamura, Liu, and Hovy 2017) and schema construction (Du et al. 2022). These works focused on hierarchical relations between mentions (e.g., subevent). In this work, we instead focus on *hierarchical relations between grounded mentions* (i.e., events in a KB). This allows for studying hierarchy at a coarser level and leverages information across numerous mentions. To this end, we extend Pratapa, Gupta, and Mitamura (2022) to include hierarchical event structures from Wikidata (Figure 1). In contrast to prior work, our formulation captures the hierarchical aspects by including non-leaf events such as 'Operation Overlord' and 'Western Front (World War II)'. Our formulation presents a challenging variant of the event linking task by requiring systems to differentiate between mentions of child and parent events.

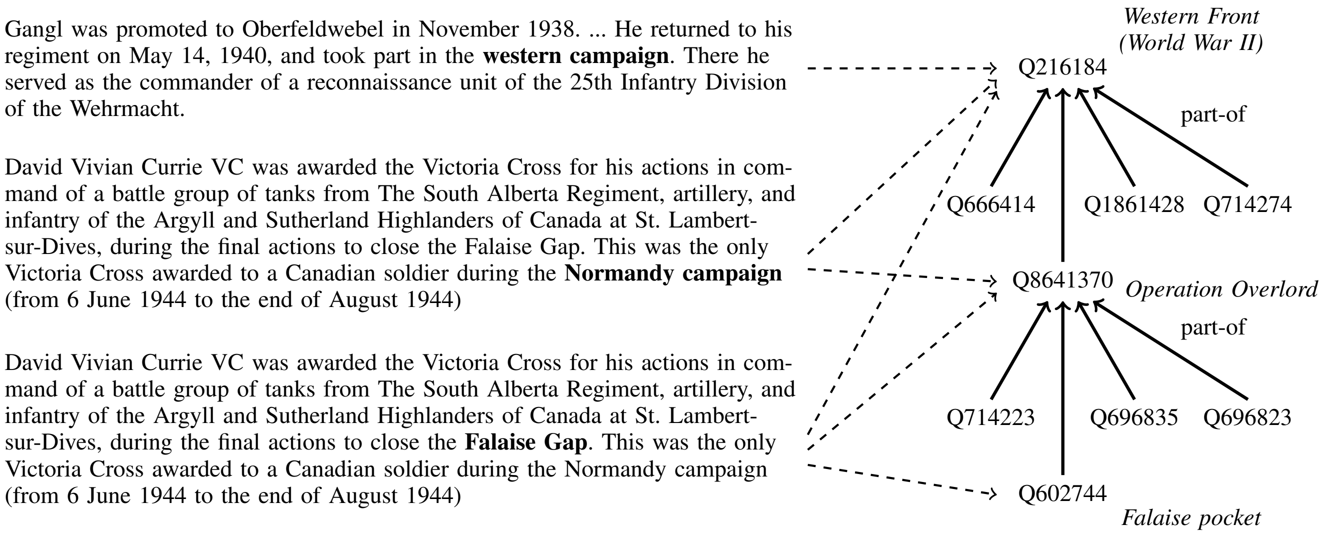
For the proposed hierarchical linking task, we present a baseline that adopts the retrieve & re-rank paradigm, which has been previously developed for entity and event linking tasks (Wu et al. 2020; Pratapa, Gupta, and Mitamura 2022). To enhance the system with hierarchy information, we present a methodology to incorporate such information via a hierarchy-aware loss (Murty et al. 2018) during the retrieval training. We experiment with the proposed systems on a multilingual dataset. The dataset is constructed by collecting mentions from Wikipedia and Wikinews articles that link to a set of events in Wikidata. Experiments on the collected dataset show the effectiveness of explicitly modeling event hierarchies.

Finally, we present a method for zero-shot hierarchy discovery in Wikidata (§). We obtain a score for potential child-parent relations between two Wikidata events by computing the fraction of overlapping mentions from text documents. Results on an unseen subset of event hierarchies illustrate the effectiveness of our linking system in discovering hierarchical structures. Our key contributions are,

- We propose the hierarchical event grounding task which requires linking mentions from documents to a set of hierarchically related events in a KB.
- We collect a large-scale multilingual dataset for this task that consists of mentions from Wikipedia and Wikinews articles linking to a set of events from Wikidata that are organized into hierarchies.

[1] We interchangeably use the terms, grounding, and linking.

[2] Mention is a text span that refers to an underlying event.



Gangl was promoted to Oberfeldwebel in November 1938. ... He returned to his regiment on May 14, 1940, and took part in the **western campaign**. There he served as the commander of a reconnaissance unit of the 25th Infantry Division of the Wehrmacht.

David Vivian Currie VC was awarded the Victoria Cross for his actions in command of a battle group of tanks from The South Alberta Regiment, artillery, and infantry of the Argyll and Sutherland Highlanders of Canada at St. Lambert-sur-Dives, during the final actions to close the Falaise Gap. This was the only Victoria Cross awarded to a Canadian soldier during the **Normandy campaign** (from 6 June 1944 to the end of August 1944)

David Vivian Currie VC was awarded the Victoria Cross for his actions in command of a battle group of tanks from The South Alberta Regiment, artillery, and infantry of the Argyll and Sutherland Highlanders of Canada at St. Lambert-sur-Dives, during the final actions to close the **Falaise Gap**. This was the only Victoria Cross awarded to a Canadian soldier during the Normandy campaign (from 6 June 1944 to the end of August 1944)

Figure 1: An illustration of mention linking to hierarchical event structures in Wikidata. The left column shows three mentions (highlighted) with their contexts, and the right column presents a hierarchy of Q-nodes from Wikidata. Each mention is linked to a set of events from a hierarchy path from Wikidata (e.g., mention 'Normandy campaign' is linked to a set of two events, {Q8641370, Q216184}).

- We present a methodology that incorporates hierarchy-based loss for the grounding task. We show improvements over competitive retrieve-and-rerank baselines.
- We demonstrate an application of our linking systems for zero-shot hierarchical relation extraction.

## Related Work

**Event Linking:** Nothman et al. (2012) proposed linking event references in newswire articles to an archive of first reports of the events. Recent work on this task focused on linking mentions to knowledge bases like Wikipedia (Yu et al. 2021) and Wikidata (Pratapa, Gupta, and Mitamura 2022). Our work is built upon the latter with a specific focus on hierarchical event structures in Wikidata.

**Event Typing:** Given an event mention span and its context, typing aims at classifying the mention into one or more types from a pre-defined ontology. Commonly used ontologies include ACE 2005 (Walker et al. 2006), Rich-ERE (Song et al. 2015), TAC-KBP (Mitamura, Liu, and Hovy 2017). In contrast, event linking grounds mentions to one or more events from a KB (e.g., World War II in Wikidata vs Conflict type in ACE 2005).

**Relation Extraction:** Extracting temporal, causal, and sub-event relations has been an integral part of event extraction pipelines. Glavaš et al. (2014) presented a dataset for studying hierarchical relations among events in news articles. Ning, Wu, and Roth (2018) studied event temporal relations, and Han et al. (2021) proposed extracting event relations as a question-answering task.

**Hierarchy Modeling:** Chen, Chen, and Van Durme (2020) presented a rank-based loss that utilizes entity ontology for hierarchical entity typing task. Murty et al. (2018) explored a complex structured loss and Onoe et al. (2021) utilized box embeddings (Vilnis et al. 2018) to model hierarchical relations between entity types.

## Hierarchical Event Grounding

The task of grounding involves linking event references in text documents to the corresponding entries in a knowledge base (Chandu, Bisk, and Black 2021). Pratapa, Gupta, and Mitamura (2022) studied linking event references from Wikipedia and Wikinews articles to Wikidata items. However, they restrict the dictionary of Wikidata items to leaf events. This ignores important parent events such as 'Operation Overlord' and 'Western Front (World War II)' from Figure 1. In our preliminary analysis, we observed a significant number of mentions that refer to the parent events, motivating us to expand the dictionary to include all events.

Modeling hierarchy relations between events has been extensively studied (Glavaš et al. 2014; Mitamura, Liu, and Hovy 2017; Du et al. 2022). These works typically focus on hierarchical relations among mentions in a document. In contrast, we focus on hierarchical relations among events in a KB. At a high level, this can be viewed as a combination of coreference resolution and hierarchy relation extraction.

### Task Definition

Consider an event knowledge base ($K$) that constitutes a set of events ($E$) and their relations ($R$). Each event ($e_i \in E$) has an id, title, and description. The relation set ($R$) includes both temporal and hierarchical (parent-child) links between events. Given an input mention span $m$ from a text document, the task is to predict an *unordered* subset of events ($E_m \subset E$). This set $E_m$ constitutes events within a hierarchy tree, from the leaf to the root of the tree.[3] In Figure 1, the mention 'Normandy campaign' is linked to the set {Q8641370, Q216184}, whereas the mention 'Falaise Gap' is linked to {Q602744, Q8641370, Q216184}.

We follow prior work on linking (Logeswaran et al. 2019) to formulate the task in a zero-shot fashion. Specifically, the set of event hierarchies for evaluation is completely unseen during training. Following Pratapa, Gupta, and Mitamura (2022), we present two task variants, 1. *Multilingual*, where the event title and description are given in the same language as the mention and its context, and 2. *Crosslingual*, where the event title and description are in English.

An alternate task formulation involves traditional event linking followed by hierarchy propagation in the KB. However, such a formulation requires access to gold hierarchy relations at test time. In contrast, we present a task that facilitates hierarchy relation extraction among unseen events.

### Hierarchical Relation Extraction

In addition to our key focus task of event linking, we explore hierarchical relation extraction for events. Similar to standard KB relation extraction (Trouillon et al. 2016), this involves predicting parent-child relationships in the KB. Specifically, given a hierarchical triple $(e_c, r, e_p)$ in $K$, where $e_c$ is the child of $e_p$ and $r$ is the child $\rightarrow$ parent relation, we mask $e_p$ and task models to retrieve it from the pool of events in $K$. To this end, we present a methodology to utilize our trained event-linking system for hierarchical relation extraction in Wikidata (§).

## Dataset

To the best of our knowledge, there are no existing datasets for the task of hierarchical event linking. Therefore, we expand the XLEL-WD dataset (Pratapa, Gupta, and Mitamura 2022) to include hierarchical event structures.

### Event and Mention Collection

Following prior work, we use Wikidata as our KB and follow a three-step process to collect events and their mentions. First, events are identified from Wikidata items by determining whether they have caused a change of state and possess spatio-temporal attributes. Then each event is associated with a set of language Wikipedia articles following the pointer contained in the event Wikidata item page. The title and description for each event in different languages are therefore obtained by taking the title and first paragraph of the associated language Wikipedia article. Finally, mentions linked to each event are collected by iterating over the language Wikipedia and identifying hyperlinks to the event-associated Wikipedia articles (obtained in the previous step). The anchor text of hyperlinks is taken as mentions and the surrounding paragraph of each mention is extracted as context.

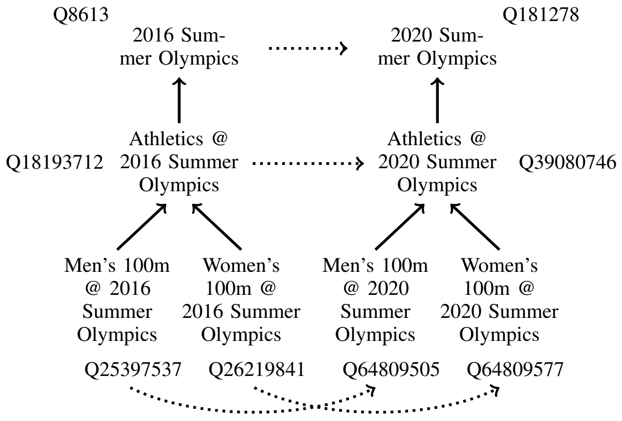

Q8613
2016 Summer Olympics
2020 Summer Olympics
Q181278
Athletics @ 2016 Summer Olympics
Q18193712
Athletics @ 2020 Summer Olympics
Q39080746
Men's 100m @ 2016 Summer Olympics
Women's 100m @ 2016 Summer Olympics
Men's 100m @ 2020 Summer Olympics
Women's 100m @ 2020 Summer Olympics
Q25397537
Q26219841
Q64809505
Q64809577

Figure 2: An illustration of hierarchical event structures in Wikidata. Each node represents an event from Wikidata. Solid arrows (⟶) and dotted arrows (···›) denote hierarchical and temporal relations respectively.

### Hierarchy Construction

We further organize the collected pool of events into hierarchical trees by exploring property nodes from Wikidata. Property nodes act as edges between Q-nodes in Wikidata. We utilize two asymmetric and transitive properties, *has-part* (P527) and *part-of* (P361). Given two events, $e_1$ and $e_2$, if there exist edges such that $e_1$ has-part $e_2$ or $e2$ part-of $e_1$, we mark $e_1$ as the parent event $e_2$ and add the edge ($e2$, part-of, $e_1$) into the hierarchies. The full hierarchies are yielded by following such procedure and iterating over the full set of candidate events collected in §.

### Zero-shot Setup

For zero-shot evaluation, the train and evaluation splits should use disjoint hierarchical event trees from Wikidata. However, just isolating trees might not be sufficient. As shown in Figure 2, event trees can be a part of a larger temporal sequence. For instance, the events '2016 Summer Olympics', and '2020 Summer Olympics' share very similar hierarchical structures. To overcome this issue, we instead split events based on connected components of both hierarchical and temporal relations. In particular, candidate events are first organized into connected components that are grown by the two hierarchical properties: *has-part* and *part of* and two temporal properties: *follows* (P155) and *followed-by* (P156). The connected components are then assigned to disjoint splits.

After building the hierarchies among events, the final dictionary includes only events that are part of a hierarchy (event tree) of height $\geq 1$. However, to conduct realistic evaluations that do not assume any prior knowledge on whether events belong to any hierarchy, all the events collected in §, no matter whether they are part of a hierarchy or not, are presented to models as candidates at inference time.

[3] In Figure 1, the leaf (or atomic) events for the mentions 'Normandy campaign' and 'Falaise Gap' are *Operation Overlord* and *Falaise pocket* respectively.

|  | Train | Dev | Test | Wikinews |
|---|---|---|---|---|
| # mentions | 751550 | 93047 | 91928 | 258 |
| # events | 2288 | 216 | 273 | 64 |
| # trees | 262 | 64 | 68 | 51 |
| # children (avg.) | 4.37 | 2.30 | 2.81 | 1.18 |
| tree depth (avg.) | 1.23 | 1.03 | 1.06 | 0.22 |

Table 1: Dataset statistics on train/dev/test splits from Wikipedia and Wikinews evaluation set. # children (avg.) refers to the average number of children per non-terminal node. Due to its limited scale, there are only 200+ mentions in Wikinews articles that are linked to events within hierarchies. Therefore, for some of the event trees, there only exists mentions linking to the root node, which results in the average effective tree depth $< 1$.

### Wikinews Evaluation

In addition to Wikipedia, assessing whether a system can generalize to new domains (e.g., news reports) has been regarded as a vital evaluation for entity (Botha, Shan, and Gillick 2020) and event linking systems (Pratapa, Gupta, and Mitamura 2022). We follow the same procedure described above to construct a test set based on Wikinews articles with hyperlinks to Wikidata.

### Dataset Statistics

To this end, we automatically compile a dataset with event hierarchies of maximum height=3 that consists of 2K events and 937K across 42 languages. The detailed statistics of the train-dev-test split as well as the Wikinews evaluation set (WN) are presented in Table 1. Other than the events within hierarchies, we use the full set of events (including singletons) as the pool of candidates, this expands the effective size of our event dictionary to nearly 13k.

We perform a human evaluation of the parent-child relations in the development split of our dataset.[4] We find the accuracy to be 96.7%, highlighting the quality of our hierarchical event structures.

## Methodology

### Baseline

We use the standard retrieve and re-rank approach (Wu et al. 2020) as our baseline,

**Retrieve:** We use two multilingual bi-encoders to independently encode mention (+context) and event candidates. We use the dot product between the two embeddings as the mention-event pair similarity score. To adapt this to our set-of-labels prediction task, for each mention $m$ and its target events set $E_m$, we pair $m$ with every event in $E_m$ to obtain $|E_m|$ mention-event pairs as positive samples. During each training step, the bi-encoder is optimized via the binary cross entropy loss (BCE) with in-batch negatives. At inference, for each mention, the bi-encoder retrieves the top-$k$ (e.g., $k = 8$) events via nearest neighbor search.

**Rerank:** We use a single multilingual cross-encoder to encode a concatenation of mention (+context) and event candidate. We follow prior work to pass the last-layer `[CLS]` through a prediction head to obtain scores for retrieved mention-event pairs. Due to computational constraints, the cross-encoder is trained only on the top-$k$ candidates retrieved by the bi-encoder. Cross-encoder is also optimized using a BCE loss that maximizes the score of gold events against other retrieved negatives for every mention. At inference, we output all the retrieved candidates with a score higher than a threshold ($\tau_c$; a hyperparameter) as the set of predicted events.

For both the bi-encoder and cross-encoder, we use XLM-RoBERTa (Conneau et al. 2020) as the multilingual transformer encoder.

### Encoding Hierarchy

The baselines described above enable linking mentions to multiple KB events. However, they predict a flat set of events, overlooking any hierarchical relationships among them. To explicitly incorporate this hierarchy, we add a hierarchy aware loss in the bi-encoder training (Murty et al. 2018). In addition to scoring mention-event pairs, the bi-encoder is also optimized to learn a scoring function $s(e_p, e_c)$ with the parent-child event pair $e_p, e_c \in \mathcal{K}$.

We parameterize $s$ based on the ComplEx embedding (Trouillon et al. 2016). It has been shown to be effective in scoring asymmetric, transitive relations such as hypernym in WordNet and hierarchical entity typing (Murty et al. 2018; Chen, Chen, and Van Durme 2020). ComplEx transforms type and relation embeddings into a complex space and scores the tuple with the real part of the Hermitian inner product. In our implementation, we use only the asymmetric portion of the product.

In particular, given the embeddings of parent-child event pair $e_p, e_c \in \mathcal{K}$ as $\mathbf{e_p}, \mathbf{e_c} \in \mathbb{R}^d$ respectively, the score $s(e_p, e_c)$ is obtained by:

$$\begin{aligned} s(e_p, e_c) =& \langle \mathrm{Im}(\mathbf{r}), \mathrm{Re}(\mathbf{e_c}), \mathrm{Im}(\mathbf{e_p}) \rangle \\ &- \langle \mathrm{Im}(\mathbf{r}), \mathrm{IM}(\mathbf{e_c}), \mathrm{Re}(\mathbf{e_p}) \rangle \qquad (1) \\ =& \mathrm{Im}(\mathbf{e_p}) \cdot (\mathrm{Re}(\mathbf{e_c}) \odot \mathrm{Im}(\mathbf{r})) \\ &- \mathrm{Re}(\mathbf{e_p}) \cdot (\mathrm{IM}(\mathbf{e_c}) \odot \mathrm{Im}(\mathbf{r})) \qquad (2) \end{aligned}$$

Where $\odot$ is the element-wise product, $\mathbf{r} \in \mathbb{R}^d$ is a learnable relation embedding.

$$\mathrm{Re}(\mathbf{e}) = W_{\mathrm{Re}} \cdot \mathbf{e} + b_{\mathrm{Re}},\ \mathrm{Im}(\mathbf{e}) = W_{\mathrm{Im}} \cdot \mathbf{e} + b_{\mathrm{Im}} \qquad (3)$$

$\mathrm{Re}(\mathbf{e})$ and $\mathrm{Im}(\mathbf{e})$ are the biased linear projections of event embedding into real and imaginary parts of complex space respectively. $W_{\mathrm{Re}}, W_{\mathrm{Im}} \in \mathbb{R}^{d \times d}$ and $b_{\mathrm{Re}}, b_{\mathrm{Im}} \in \mathbb{R}^d$ are learnable weights. During training, a batch of $N_h$ parent-child event pairs is independently sampled and the bi-encoder is

[4] Two authors on this paper independently annotated the relations, followed by an adjudication phase to resolve disagreements.

trained to minimize the in-batch BCE loss:

$$L_h = \frac{1}{N_h} \sum_{i=1}^{N_h} \Big( - \sum_{e_{cj} \in \mathcal{C}_i} \log(\sigma(s(e_{pi}, e_{cj}))) + \sum_{e_{ck} \notin \mathcal{C}_i} \log(\sigma(s(e_{pi}, e_{ck}))) \Big) \tag{4}$$

Where $\mathcal{C}_i$ denotes the set of children events in the batch that are the children of $e_{pi}$. We further explore three strategies for incorporating the hierarchy prediction in learning the bi-encoder:

- **Pretraining**: the bi-encoder is pre-trained with the hierarchy-aware loss, followed by training with the mention linking loss.
- **Joint Learning**: in each epoch, the bi-encoder is jointly optimized with both the hierarchy-aware and mention linking loss.
- **Pretraining + Joint Learning**: bi-encoder is pretrained with the hierarchy-aware loss, followed by joint training with the hierarchy-aware and mention linking loss.

For each of the above bi-encoder configurations, we train a cross-encoder using the same training recipe as the baseline. We leave the development of hierarchy-aware cross-encoder models to future work.

### Hierarchical Relation Extraction

In addition to the mention-linking, we propose a methodology to leverage the trained bi-encoders for hierarchical relation extraction. For each mention, we first retrieve the top-$k$ event candidates. We then construct a list of mentions ($M_e$) for each event $e$ in the dictionary. Finally, given a pair of events $(e_i, e_j)$, we compute a score for potential child-parent ($e_i$–$e_j$) relation as follows,

$$h(e_i, e_j) = \frac{|M_{e_i} \cap M_{e_j}|}{|M_{e_i}|} \tag{5}$$

The scoring function $h$ is derived based on the intuition that if $e_j$ is the parent event of $e_i$, then all the mentions which are linked to $e_i$ should also be linked to $e_j$ by our linker. In such case, $M_{e_i}$ would be the subset of $M_{e_j}$ which indicates that $h(e_i, e_j) = 1$ approaches its maximum.

For each event, we iteratively calculate the child-parent score with every other event and rank them in descending order of the $h$ score. With this process, we could obtain a ranking of all other events as its candidate parents.

## Experiments

We experiment with the proposed configurations of bi-encoders and corresponding cross-encoders on the Wikipedia dataset for both multilingual and crosslingual tasks. To further assess out-of-domain generalization performance, we conduct the same experiments for baseline and the best-performing hierarchy-aware system on the Wikinews evaluation set. For the hierarchy relation extraction task, we evaluate the approach proposed in § based on the retrieval results of the best-performing bi-encoder.

### Metrics

**Bi-encoder:** For bi-encoder, we follow prior work to report Recall@$k$ and extend it to a multi-label version: measuring the fraction of mentions where all the gold events contained in the top-$k$ retrieved candidates. Since the longest path of hierarchies in the collected dataset consists of 4 events, we only evaluate with $k \geq 4$. And for $k < 4$, we instead report Recall@*min*: the fraction of mentions where all the gold events are contained in the top-$x$ retrieved candidates, where $x$ is the number of gold events for that mention. Recall@*min* measures whether the bi-encoder could predict all and only the gold events with the minimal number of retrievals. For cases with the single gold event, Recall@*min* = Recall@1 which falls back to single event linking.

Our task requires the model to predict all the relevant events, from the atomic event to the root of the hierarchy tree. As an upper bound for model performance, we also report scores for predicting the most atomic event. In particular, the set of gold events is considered fully contained in the top-$k$ retrieval of a mention if the most atomic gold event in the hierarchy is contained in the top-$k$ candidates. Our original task reduces to this atomic-only prediction task if the event hierarchy is known at test time. However, such an assumption might not be true in real-world settings.

While Recall@$k$ is a strict and binary metric, i.e. the retrieval is counted as successful if and only if all the gold events are predicted, we further introduce a fraction version of it, denoted as Recall@$k$ (fraction), that allows for partial retrieval, with details in Appendix **??**.

**Cross-encoder:** Similar to bi-encoder, we also follow previous work on entity linking to evaluate *strict accuracy*, *macro F1*, and *micro F1* on the performance of cross-encoder. For a mention $m_i$, denote its gold events set as $E_i$, predicted events set as $\hat{E}_i$, with $N$ mentions:

$$\text{Strict Accuracy} = \frac{\sum_{i=1}^{N} \mathbf{1}_{E_i = \hat{E}_i}}{N} \tag{6}$$

$$\text{MaP} = \frac{1}{N} \sum_{i=1}^{N} \frac{|E_i \cap \hat{E}_i|}{|\hat{E}_i|}, \text{MaR} = \frac{1}{N} \sum_{i=1}^{N} \frac{|E_i \cap \hat{E}_i|}{|E_i|} \tag{7}$$

$$\text{Macro F1} = \frac{2\text{MaP} \cdot \text{MaR}}{\text{MaP} + \text{MaR}} \tag{8}$$

$$\text{MiP} = \frac{\sum_{i=1}^{N} |E_i \cap \hat{E}_i|}{\sum_{i=1}^{N} |\hat{E}_i|}, \text{MiR} = \frac{\sum_{i=1}^{N} |E_i \cap \hat{E}_i|}{\sum_{i=1}^{N} |E_i|} \tag{9}$$

$$\text{Micro F1} = \frac{2\text{MiP} \cdot \text{MiR}}{\text{MiP} + \text{MiR}} \tag{10}$$

We additionally evaluate the strict accuracy of cross-encoders where we report the top-$x$ reranked candidates as predictions with $x$ being the number of gold events linked with the mention. This is the same condition as evaluating the bi-encoder on Recall@*min*. It enables a direct comparison of strict accuracy to Recall@*min* such that to assess if cross-encoders make improvements on bi-encoders. We denote the strict accuracy calculated under this condition as *strict accuracy (top min)*.

**Hierarchical Relation Extraction:** As defined in §, we evaluate the proposed hierarchical relation extraction method on whether it could identify the parent for a given child event. In particular, given an event with a ranked list of candidate parents (generated by the proposed method), we measure Recall@$k$ for the gold parent in the list. Since Recall@$k$ is ill-defined for events without a parent, we only calculate it for the non-root events within hierarchies of the dev and test sets. For those events that have parents but are not linked to any mentions by the bi-encoder, they are added as miss at every $k$. Such evaluation with Recall@$k$ measure is similar to the HIT@$k$ evaluation in the KB link prediction literature (Bordes et al. 2011).

## Bi-encoder Models

As discussed in §, we evaluate the baseline bi-encoder (*Baseline*) and three hierarchy-aware configurations: Hierarchy Pretraining (*Baseline + HP*); Hierarchy Joint Learning (*Baseline + HJL*); Hierarchy Pretraining & Hierarchy Joint Learning (*Baseline + HP + HJL*).

## Cross-encoder Models

Given the top-$k$ retrieval results from each of the aforementioned bi-encoders, we train and evaluate a unique crossencoder respectively. The value of $k$ used for all cross-encoder experiments is selected to balance retrieval qualities (i.e. bi-encoder Recall@$k$ on dev set) and computation throughput (§). In case some of the gold events are not retrieved among top-$k$ candidates for the corresponding mention in the training set, we substitute each missing gold event for the negative candidates with the current lowest probability and repeat this process until all the missing gold events are added. At inference time, we apply a threshold $\tau_c$ to the reranked event candidates and emit those with score $\geq \tau_c$ as final predictions. If there is no event yielded, we add a `NULL` event to the prediction.

**Hierarchical Relation Extraction**: We apply the proposed method to top-4 retrieval results from the best-performing bi-encoder to perform hierarchical relation extraction.

# Result and Analysis

## Bi-encoder

Bi-encoder retrieval results on the dev split for both multi- and cross-lingual tasks are illustrated in Figure 3 and Figure 4 respectively. Since the gain in Recall@$k$ is relatively minor when doubling $k$ from 8 to 16 across all configurations and tasks, the cross-encoder is trained with the top 8 retrieved candidates, with the consideration of computation efficiency. It is also shown that all configurations attain better performance when evaluated by retrieving the most atomic event only (set of dense dots vs line plots), which reflects the benefits of following the gold hierarchies and indicates the performance upper-bound for current models that try to learn these hierarchies.

We further report the quantitative results of bi-encoder Recall@*min* on dev and test set of both tasks in Table 2. Among all the hierarchy-integration strategies, hierarchical pretraining offers consistent improvements on both tasks compared with the baseline. On the other hand, hierarchical joint learning presents a mixture of effects. In particular, it attains the best performance on the crosslingual test set when applied in conjunction with hierarchical pretraining while contributing negatively in all other scenarios.

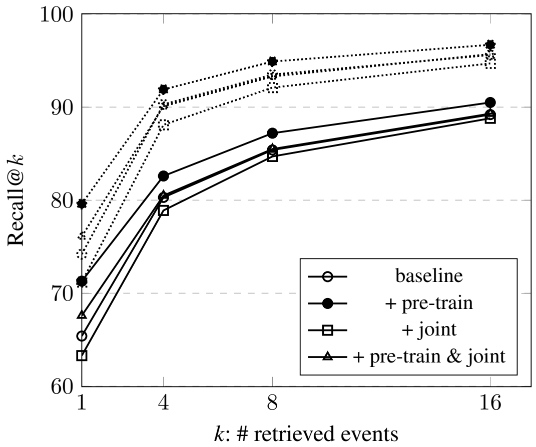


Figure 3: Multilingual bi-encoder Recall@$k$ on the dev set. The densely dotted plots (········) denote the prediction scores for the atomic label, an upper bound for model performance.

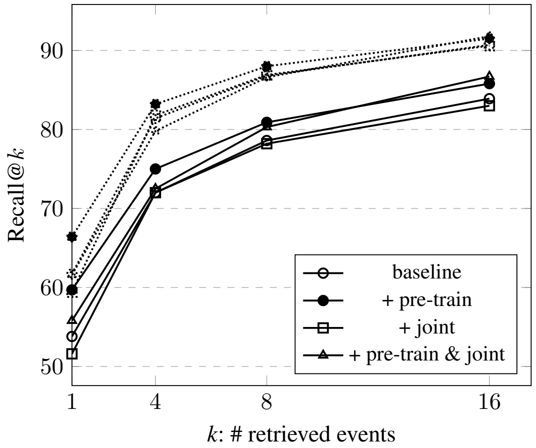


Figure 4: Crosslingual bi-encoder Recall@$k$ on the dev set. The densely dotted plots (········) denote the prediction scores for the atomic label, an upper bound for model performance.

In terms of task languages, all the multilingual configurations attain higher performance than their crosslingual counterparts, indicating that in general crosslingual task is more challenging than the multilingual task, which is similar to the single event linking scenario.

As described in Section , we further report bi-encoder results under Recall@$K$ (fraction) in Appendix **??**.

## Cross-encoder

Cross-encoder reranking results on both tasks are also shown in Table 2. On the multilingual task, all the cross-encoders

|  |  | Bi-encoder | Cross-encoder |  |  |  |
|---|---|---|---|---|---|---|
|  | Methods | Recall@*min* | Strict Acc | Strict Acc (Top Min) | Macro F1 | Micro F1 |
| *Multilingual* |  |  |  |  |  |  |
| (a) | Baseline | 65.4 / 54.8 | 34.4 / 37.6 | 57.8 / 59.5 | 56.4 / **62.8** | 53.0 / 58.3 |
| (b) | + HP | **71.3 / 58.1** | 40.4 / 39.2 | 61.7 / 60.3 | 60.8 / 62.0 | 57.5 / **58.7** |
| (c) | + HJL | 63.3 / 51.4 | **43.6 / 40.2** | **63.6 / 60.8** | **62.1** / 60.1 | **59.2** / 57.5 |
| (d) | + HP + HJL | 67.6 / 55.2 | 38.2 / 39.9 | 60.4 / 60.6 | 57.6 / 61.4 | 54.7 / 57.8 |
| *Crosslingual* |  |  |  |  |  |  |
| (a) | Baseline | 53.8 / 32.8 | 8.5 / 11.9 | 21.2 / 27.5 | 22.1 / 28.8 | 23.6 / 29.2 |
| (b) | + HP | **59.7** / 37.0 | 8.6 / 10.9 | 18.6 / 25.7 | 22.3 / 28.0 | 24.0 / 28.9 |
| (c) | + HJL | 51.6 / 33.3 | **9.7** / 12.0 | 22.3 / 26.1 | 21.6 / 28.1 | 23.4 / 29.4 |
| (d) | + HP + HJL | 55.8 / **38.8** | 9.6 / **13.1** | **23.0 / 28.0** | **25.7 / 34.3** | **27.3 / 33.1** |

Table 2: Bi-encoder and Cross-encoder performance on multilingual and crosslingual event linking (dev/test). Strict Acc (Top Min) refers to the cross-encoder strict accuracy under Top Min, which is directly comparable to the bi-encoder Recall@*min*.

|  | R@1 | R@4 | R@8 | R@16 |
|---|---|---|---|---|
| Multilingual | 46.0 / 45.1 | 69.0 / 72.6 | 76.6 / 81.3 | 79.6 / 84.6 |
| Crosslingual | 52.0 / 37.4 | 78.3 / 60.5 | 86.2 / 75.6 | 90.8 / 83.9 |

Table 3: Hierarchical relation extraction results (dev/test) with the top-4 retrieval predictions by the best performing bi-encoder.

that are paired with hierarchy-aware bi-encoders outperform the baseline on strict accuracy and attain better or comparable performance on macro/micro F1. On the crosslingual task, (d) is the only hierarchy-aware system that outperforms the baseline across all metrics. All the models attain better results with Top Min accuracy and the relative performance differences between them remain similar to that of normal accuracy. Similar to the bi-encoder, the large performance gap of cross-encoders between the two tasks confirms that the crosslingual setting is more challenging.

## Bi-encoder vs. Cross-encoder

We further investigate whether the cross-encoder could make improvements on its bi-encoder across all configurations. As discussed in §, by comparing the strict accuracy of cross-encoders under the Top Min condition with the Recall@*min* of associated bi-encoders, we find that cross-encoders further enhance bi-encoder performance on the test set in multilingual tasks while underperforms in other cases. For closer inspection into the performance of systems on each language, we report the per-language bi- and cross-encoder results in Table **??** and Table **??** in Appendix **??**.

## Hierarchy Discovery

Table 3 presents the hierarchical relation extraction results of our proposed set-based approach using the retrieved candidates by the best performance bi-encoder ((b) in Table 2). On both tasks, the proposed method is able to assign high rankings to true parents for events within hierarchies, demonstrating its capability in aiding humans to discover new hierarchical relations on a set of previously-unseen events.

|  |  | Bi-encoder | Cross-encoder |  |  |
|---|---|---|---|---|---|
|  | Methods | R@*min* | Strict Acc | Macro F1 | Micro F1 |
| *Multilingual* |  |  |  |  |  |
| (a) | Baseline | **68.6** | 51.7 | **67.2** | 62.0 |
| (c) | + HJL | 67.4 | **55.8** | 65.3 | **62.7** |
| *Crosslingual* |  |  |  |  |  |
| (a) | Baseline | 51.2 | 15.3 | 29.8 | 30.0 |
| (d) | + HP + HJL | **53.7** | **21.1** | **37.6** | **35.7** |

Table 4: Bi-encoder and cross-encoder performance on multilingual & crosslingual event linking on Wikinews Dataset

## Wikinews

As shown in Table 4, applying our baseline and two of the hierarchy-aware linking systems ((c) in multilingual and (d) in crosslingual) on the Wikinews dataset results in a similar performance to that on Wikipedia mentions, which demonstrates that our methods could generalize well on the news domain.

# Conclusion & Future Work

In this paper, we present the task of hierarchical event grounding, for which we compile a multilingual dataset with Wikipedia and Wikidata. We propose a hierarchy-loss-based methodology that improves upon a standard retrieve and rerank baseline. Our experiments demonstrate the effectiveness of our approaches to model hierarchies among events in both multilingual and crosslingual settings. Additionally, we show promising results for zero-shot hierarchical relation extraction using the trained event linker. Some potential directions for future work include adapting encoders to directly include hierarchy and further exploring hierarchical relation extraction on standard datasets.

# Acknowledgments

This material is based on research sponsored by the Air Force Research Laboratory under agreement number

FA8750-19-2-0200. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the U.S. Government.

## References

Bordes, A.; Weston, J.; Collobert, R.; and Bengio, Y. 2011. Learning Structured Embeddings of Knowledge Bases. In *Proceedings of the Twenty-Fifth National Conference on Artificial Intelligence*, 301–306. Menlo Park, Calif.: AAAI Press.

Botha, J. A.; Shan, Z.; and Gillick, D. 2020. Entity Linking in 100 Languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7833–7845. Online: Association for Computational Linguistics.

Chandu, K. R.; Bisk, Y.; and Black, A. W. 2021. Grounding 'Grounding' in NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 4283–4305. Online: Association for Computational Linguistics.

Chen, T.; Chen, Y.; and Van Durme, B. 2020. Hierarchical Entity Typing via Multi-level Learning to Rank. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8465–8475. Online: Association for Computational Linguistics.

Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451. Online: Association for Computational Linguistics.

Du, X.; Zhang, Z.; Li, S.; Yu, P.; Wang, H.; Lai, T.; Lin, X.; Wang, Z.; Liu, I.; Zhou, B.; Wen, H.; Li, M.; Hannan, D.; Lei, J.; Kim, H.; Dror, R.; Wang, H.; Regan, M.; Zeng, Q.; Lyu, Q.; Yu, C.; Edwards, C.; Jin, X.; Jiao, Y.; Kazeminejad, G.; Wang, Z.; Callison-Burch, C.; Bansal, M.; Vondrick, C.; Han, J.; Roth, D.; Chang, S.-F.; Palmer, M.; and Ji, H. 2022. RESIN-11: Schema-guided Event Prediction for 11 Newsworthy Scenarios. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations*, 54–63. Hybrid: Seattle, Washington + Online: Association for Computational Linguistics.

Glavaš, G.; Šnajder, J.; Moens, M.-F.; and Kordjamshidi, P. 2014. HiEve: A Corpus for Extracting Event Hierarchies from News Stories. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, 3678–3683. Reykjavik, Iceland: European Language Resources Association (ELRA).

Han, R.; Hsu, I.-H.; Sun, J.; Baylon, J.; Ning, Q.; Roth, D.; and Peng, N. 2021. ESTER: A Machine Reading Comprehension Dataset for Reasoning about Event Semantic Relations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 7543–7559. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Ji, H.; and Grishman, R. 2011. Knowledge Base Population: Successful Approaches and Challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 1148–1158. Portland, Oregon, USA: Association for Computational Linguistics.

Logeswaran, L.; Chang, M.-W.; Lee, K.; Toutanova, K.; Devlin, J.; and Lee, H. 2019. Zero-Shot Entity Linking by Reading Entity Descriptions. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 3449–3460. Florence, Italy: Association for Computational Linguistics.

Mitamura, T.; Liu, Z.; and Hovy, E. H. 2017. Events Detection, Coreference and Sequencing: What's next? Overview of the TAC KBP 2017 Event Track. In *Proceedings of the 2017 Text Analysis Conference, TAC 2017*. Gaithersburg, Maryland: NIST.

Murty, S.; Verga, P.; Vilnis, L.; Radovanovic, I.; and McCallum, A. 2018. Hierarchical Losses and New Resources for Fine-grained Entity Typing and Linking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 97–109. Melbourne, Australia: Association for Computational Linguistics.

Ning, Q.; Wu, H.; and Roth, D. 2018. A Multi-Axis Annotation Scheme for Event Temporal Relations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1318–1328. Melbourne, Australia: Association for Computational Linguistics.

Nothman, J.; Honnibal, M.; Hachey, B.; and Curran, J. R. 2012. Event Linking: Grounding Event Reference in a News Archive. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 228–232. Jeju Island, Korea: Association for Computational Linguistics.

Onoe, Y.; Boratko, M.; McCallum, A.; and Durrett, G. 2021. Modeling Fine-Grained Entity Types with Box Embeddings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2051–2064. Online: Association for Computational Linguistics.

Pratapa, A.; Gupta, R.; and Mitamura, T. 2022. Multilingual Event Linking to Wikidata. In *Proceedings of the Workshop on Multilingual Information Access (MIA)*, 37–58. Seattle, USA: Association for Computational Linguistics.

Song, Z.; Bies, A.; Strassel, S.; Riese, T.; Mott, J.; Ellis, J.; Wright, J.; Kulick, S.; Ryant, N.; and Ma, X. 2015. From Light to Rich ERE: Annotation of Entities, Relations, and Events. In *Proceedings of the The 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Represen-*

*tation*, 89–98. Denver, Colorado: Association for Computational Linguistics.

Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, E.; and Bouchard, G. 2016. Complex Embeddings for Simple Link Prediction. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, 2071–2080. JMLR.org.

Vilnis, L.; Li, X.; Murty, S.; and McCallum, A. 2018. Probabilistic Embedding of Knowledge Graphs with Box Lattice Measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 263–272. Melbourne, Australia: Association for Computational Linguistics.

Walker, C.; Strassel, S.; Medero, J.; and Maeda, K. 2006. ACE 2005 Multilingual Training Corpus. *Linguistic Data Consortium, Philadelphia*, 57.

Wu, L.; Petroni, F.; Josifoski, M.; Riedel, S.; and Zettlemoyer, L. 2020. Scalable Zero-shot Entity Linking with Dense Entity Retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6397–6407. Online: Association for Computational Linguistics.

Yu, X.; Yin, W.; Gupta, N.; and Roth, D. 2021. Event Linking: Grounding Event Mentions to Wikipedia. arXiv:2112.07888.