LOME: Large Ontology Multilingual Extraction

Patrick Xia[1]*, Guanghui Qin[1]*, Siddharth Vashishtha[2]
Yunmo Chen[1], Tongfei Chen[1], Chandler May[1], Craig Harman[1]
Kyle Rawlins[1], Aaron Steven White[2], Benjamin Van Durme[1]
[1] Johns Hopkins University, [2] University of Rochester
{paxia,qin,vandurme}@jhu.edu

Abstract

We present LOME, a system for performing multilingual information extraction. Given a text document as input, our core system identifies spans of textual entity and event mentions with a FrameNet (Baker et al., 1998) parser. It subsequently performs coreference resolution, fine-grained entity typing, and temporal relation prediction between events. By doing so, the system constructs an event and entity focused knowledge graph. We can further apply third-party modules for other types of annotation, like relation extraction. Our (multilingual) first-party modules either outperform or are competitive with the (monolingual) state-of-the-art. We achieve this through the use of multilingual encoders like *XLM-R* (Conneau et al., 2020) and leveraging multilingual training data. LOME is available as a Docker container on Docker Hub. In addition, a lightweight version of the system is accessible as a web demo.

1 Introduction

As information extraction capabilities continue to improve due to advances in modeling, encoders, and data collection, we can now look (back) toward making richer predictions at the document-level, with a large ontology, and across multiple languages. Recently, Li et al. (2020) noted that despite a growth of open-source NLP software in general, there is still a lack of available software for knowledge extraction. We wish to provide a starting point that allows others to build increasingly comprehensive document-level knowledge graphs of events and entities from text in many languages.[1]

Therefore, we demonstrate LOME, a system for multilingual information extraction with large ontologies. Figure 1 shows the high-level pipeline by following a multilingual input example. A sentence-level parser identifies both INGESTION events and their arguments. To connect these events cross-sententially, the system clusters coreferent mentions and predicts the temporal relations between the events. LOME, which supports fine-grained entity types, additionally labels entities like **the rabbit** with LIVING_THING/ANIMAL.

Several prior packages have also used advances in state-of-the-art models to build comprehensive information extraction systems. Li et al. (2019) present an event, relation, and entity extraction and coreference system for three languages: English, Russian, and Ukrainian. Li et al. (2020, GAIA) extend that work to support cross-media documents. However, both of these systems consist of language-specific models that operate on monolingual documents after first identifying the language. On the other hand, work prioritizing coverage across tens or hundreds of languages is limited in their scope in extraction (Akbik and Li, 2016; Pan et al., 2017).

Like prior work, LOME is focused on extracting entities and events from raw text documents. However, LOME is language-agnostic; all components prioritize multilinguality. Using *XLM-R* (Conneau et al., 2020) as the underlying encoder paves the way for both training on multilingual data (where it exists) and inference in many languages.[2] Our pipeline includes a full FrameNet parser for events and their arguments, neural coreference resolution, an entity typing model over large ontologies, and temporal resolution between events.

Our system is designed to be modular: each component is trained independently and tuned on task-specific data. To communicate between modules, we use CONCRETE (Ferraro et al., 2014), a data schema used in other text processing systems (Peng et al., 2015). One advantage of using a stan-

[*] Equal Contribution

[1] Information on using the Docker container, web demo, and demo video at https://nlp.jhu.edu/demos.

[2] *XLM-R* itself is trained on CommonCrawl data spanning one hundred languages.

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations, pages 149–159
April 19 - 23, 2021. ©2021 Association for Computational Linguistics

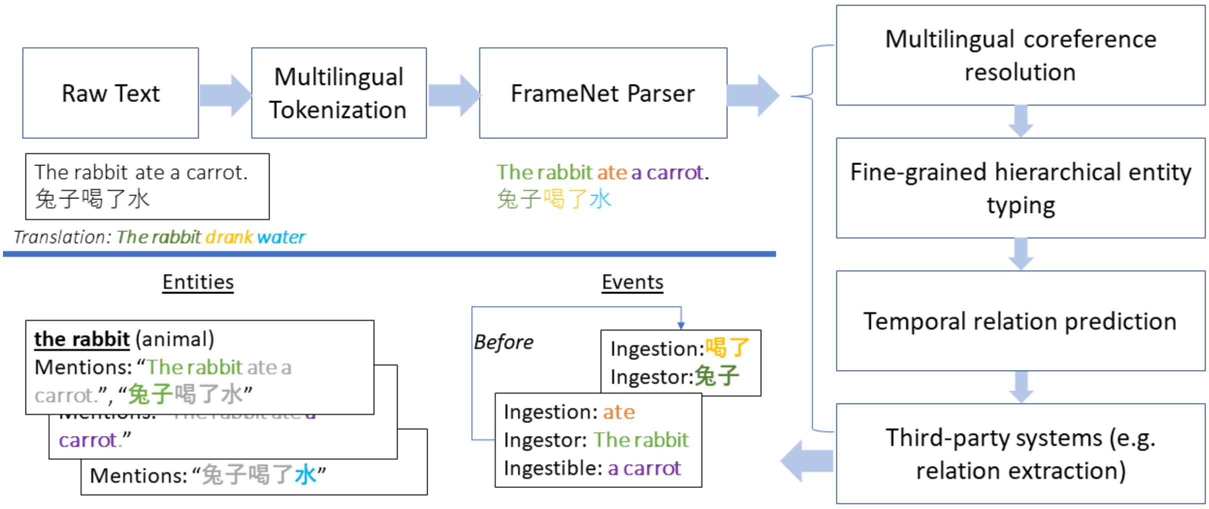
Raw Text
Multilingual Tokenization
FrameNet Parser
The rabbit ate a carrot.
兔子喝了水
Translation: The rabbit drank water
The rabbit ate a carrot.
兔子喝了水
Multilingual coreference resolution
Fine-grained hierarchical entity typing
Temporal relation prediction
Third-party systems (e.g. relation extraction)
Entities
Events
the rabbit (animal)
Mentions: "The rabbit ate a carrot.", "兔子喝了水"
carrot."
Mentions: "兔子喝了水"
Before
Ingestion:喝了
Ingestor:兔子
Ingestion: ate
Ingestor: The rabbit
Ingestible: a carrot

Figure 1: Architecture of LOME. The system processes text documents as input and first uses a FrameNet parser to detect entities and events. Then, a suite of models enrich the entities and events with additional predictions. Each individual model can be trained and tuned independently, ensuring modularity of the pipeline. Annotations between models are transferred using CONCRETE, a data schema for NLP.

dardized data schema is that it enables modularization and extension. Unless there are annotation dependencies, individual modules can be inserted, replaced, merged, or bypassed depending on the application. We discuss two example applications of our CONCRETE-based modules, one of which further extracts relations and the other performs cross-sentence argument linking for events.

2 Tasks

The overarching application of LOME is to extract an entity- and event-centric knowledge graph from a textual document. In particular, we are interested in using these graphs to support a multilingual schema learning task (KAIROS[3]) for which data has been annotated by the LDC (Cieri et al., 2020). As a result, some parts of LOME are designed for compatibility with the KAIROS event and entity ontology. Nonetheless, there is significant overlap with publicly available datasets, which we describe for those tasks.

Figure 1 presents the architecture of our pipeline. Besides the FrameNet parser, which is run first, the remaining modules can be run in any order, if at all. In addition, our use of a standardized data schema for communication allows for the integration of third-party systems. In this section, we will go into further detail for each task.

2.1 FrameNet Parsing

FrameNet parsing is a semantic role labeling style task. The goal is to find all the frames and their roles, as well as the trigger spans associated with them in a sentence. Frames are concepts, such as events or entities, in a sentences. Every frame is associated with some roles, and both of them are triggered by spans in the sentence.

Unlike most previous work (Yang and Mitchell, 2017; Peng et al., 2018; Swayamdipta et al., 2018), our system is not conditioned on the trigger spans or frames. We perform "full parsing" (Das et al., 2014), where the input is a raw sentence, and the output is the complete structure predictions.

As the first model in the whole pipeline system, the trigger spans found by the FrameNet parser will be used as candidate spans for all other tasks.

2.2 Entity Coreference Resolution

In coreference resolution, the goal is to cluster spans in the text that refer to the same entity. Neural models for doing so typically encode the text first before identifying possible mentions (Lee et al., 2017; Joshi et al., 2019, 2020). These spans are scored pairwise to determine whether two spans refer to each other. These scores then determine coreference clusters by decoding under a variety of strategies (Lee et al., 2018; Xu and Choi, 2020).

In this work, we choose a constant-memory variant of that model which also achieves high per-

[3] This goal is to develop a system that identifies, links, and temporally sequences complex events. More information at https://www.darpa.mil/program/knowledge-directed-artificial-intelligence-reasoning-over-schemas.

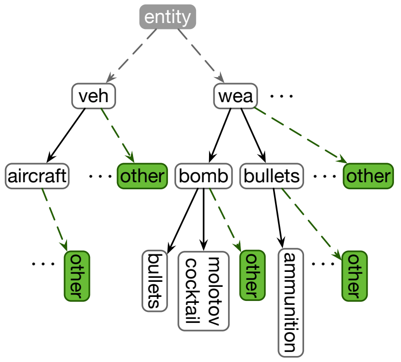

Figure 2: A portion of the AIDA entity type ontology.

formance (Xia et al., 2020). The motivation here is robustness: we prioritize the ability to soundly run on all document lengths over slightly better performing but fragile systems. In addition, because this coreference resolution model is part of a broader entity-centric system, the module used in this system does not perform the mention detection step (which is left to the FrameNet parser). Instead, both training and inference assumes given mentions, and the task we are concerned about in this paper is mention *linking*.

2.3 Entity Typing

Entity typing assigns a fine-grained semantic label to a span of text, where the span is a *mention* of some entity found by the FrameNet parser. Traditionally, labels include `PER`, `GPE`, `ORG`, etc., but recent work in *fine-grained* entity typing seek to classify spans into types defined by hierarchical type *ontologies* (e.g. BBN (Weischedel and Brunstein, 2005), FIGER (Ling and Weld, 2012), UltraFine[4] (Choi et al., 2018), COLLIE (Allen et al., 2020)). Such ontologies refine coarse types like `PER` to fine-grained types such as `/person/artist/singer` that sits on a type hierarchy. A portion of the AIDA ontology (LDC2019E07) is illustrated in Figure 2.

To support fine-grained ontologies, we employ a recent coarse-to-fine-decoding entity typing model (Chen et al., 2020a) that is specifically designed to assign types that are defined by hierarchical ontologies. The use of a coarse-to-fine model also allows users to select between coarse- and fine-grained types. We swap the underlying encoder from ELMo (Peters et al., 2018) to *XLM-R* to be able to assign types over mentions in different languages using a single multilingual model, and to enable transfer between languages.

The base typing model in Chen et al. (2020a) supports entity typing on entity *mentions*. We extend this model to gain the ability to perform entity typing on *entities*, i.e. clusters of entity mentions. Since our decoder is coarse-to-fine and predicts a type at each level of the type hierarchy, we employ Borda voting on each level. Specifically, given a coreference chain comprising mentions $m_{1,\cdots,n}$, and the score for mention m_i being typed as type t as $s_{i,t}$, we perform Borda counting to select the most confident type $t^* = \arg\max_t \sum_i r(i,t)$ over all t's in a specific type level, where $r(i,t) = 1/\mathrm{rank}_t(s_{i,t})$ is the ranking relevance score used in Borda counting.

2.4 Temporal Relation Extraction

The task of temporal relation extraction focuses on finding the chronology of events (e.g., *Before*, *After*, *Overlaps*) in text. Extracting temporal relation is useful for various downstream tasks – curating structured clinical data (Savova et al., 2010; Soysal et al., 2018), text summarization (Glavaš and Šnajder, 2014; Kedzie et al., 2015), question-answering (Llorens et al., 2015; Zhou et al., 2019), etc. The task is most commonly viewed as a classification task where given a pair of events and its textual context, the temporal relation between them needs to be identified.

The construction of the TimeBank corpus (Pustejovsky et al., 2003) largely spurred the research in temporal relation extraction. It included 14 temporal relation labels. Other corpora (Verhagen et al., 2007, 2010; Sun et al., 2013; Cassidy et al., 2014) reduced the number of labels to a smaller number owing to lower inter-annotator agreements and sparse annotations. Various types of models (Chambers et al., 2014; Cheng and Miyao, 2017; Leeuwenberg and Moens, 2017; Ning et al., 2017; Vashishtha et al., 2019; Zhou et al., 2021) have been used in the recent years to extract temporal relations from text.

In this work, we use Vashishtha et al. (2019)'s best model and retrain it using *XLM-R*. We evaluate their model using the transfer learning approach described in their work and retrain it on TimeBank-Dense (TBD) (Cassidy et al., 2014). TBD uses a reduced set of 5 temporal relation labels – *before*, *after*, *includes*, *is_included*, and *vague*.

[4] UltraFine is slightly different in that the types are bucketed into 3 categories of different granularity, but without explicit subtyping relations.

3 System Design

3.1 Modularization

Our system is modularized into separate models and libraries that communicate with each other using CONCRETE, a data format for richly annotating natural language documents (Ferraro et al., 2014). Each component is independent of each other, which allows for both inserting additional modules or deleting those provided in the default pipeline. We choose this loosely-affiliated design to enable both faster and independent prototyping of individual components, as well as better compartmentalization of our models.

We emphasize that the system is a pipeline: while individual modules can be further improved, the system is not designed to be trained end-to-end and benchmarking the richly-annotated output depends on the application and priorities. In this paper, we only benchmark individual components and describe a couple of applications.

3.2 System Inputs and Outputs

The system can consume, as input, either tokenized or untokenized text, which is first tokenized either by whitespace or with a multilingual tokenizer, PolyGlot.[5] However, this tokenization is not necessarily used by all modules, which may choose to either operate on the raw text itself or on a SentencePiece (Kudo and Richardson, 2018) retokenization.

The system outputs a CONCRETE communication file for each input document. This output file contains annotations including entities, events, coreference, entity types, and temporal relations. This schema used is entirely self-contained and the well-documented library also contains tools for visualizing and inspecting CONCRETE files.[6] For the web demo, the output is displayed in the browser.

4 Evaluation Benchmarks

4.1 FrameNet Span Finding

The FrameNet parser is comprised of an *XLM-R* encoder, a BIO tagger, and a typing module. It encodes the input sentences into a list of vectors, used by both the BIO tagger and the typing module. The goal of BIO tagger is to find trigger spans, which are then labeled by the typing module. To parse a sentence, we run the model to find all frames, and then find their roles conditioned on the frames.

We train the FrameNet parser on the FrameNet v1.7 corpus following Das et al. (2014), with statistics in Table 1. We evaluate the results with exact matching as our metric,[7] and get 56.34 labeled F1 or 66.41 unlabeled F1. Since we are not aware of previous work on both full parsing and a metric for its evaluation, we do not have a baseline. However, we can force the model to perform frame identification given the trigger span, like prior work. These results are shown in Table 2.

	# Sentences	# Frames	# Roles
train	3120	18604	32419
dev	311	2209	3853
test	1333	6687	11277

Table 1: Statistics of FrameNet v1.7

Model	Accuracy
Yang and Mitchell (2017)	88.2
Hermann et al. (2014)	88.4
Peng et al. (2018)	90.0
This work	**91.3**

Table 2: Result on frame identification

4.2 Coreference Resolution

We retrain the model by Xia et al. (2020) with *XLM-R* (large) as the underlying encoder and with additional multilingual data. The model is a constant-memory variant of neural coreference resolution models. We refer the reader to Xia et al. (2020) for model and training details.

Unlike that work, we operate under the assumption that we are provided gold spans. This is motivated by the location of coreference in LOME. In addition, while they use a frozen encoder, we found that finetuning improves performance.[8] Finally, we train on the full OntoNotes 5.0 (Weischedel et al., 2013; Pradhan et al., 2013), a subset of SemEval 2010 Task 1 (Recasens et al., 2010), and two additional sources of Russian data, RuCor (Toldova et al., 2014) and AnCor (Budnikov et al., 2019).

We benchmark the performance of our model on each language. We report the average F1 of MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), and CEAF$_{\phi_4}$ (Luo, 2005) by language in Table 3. We can compare the model's performance to monolingual gold-only baselines, where they exist. For

[5] https://github.com/aboSamoor/polyglot

[6] http://hltcoe.github.io/concrete/

[7] A role is considered to be correctly predicted only when its frame is precisely predicted.

[8] We use AdamW and a learning rate of 5×10^{-6}.

English, we trained an identical model but instead use SpanBERT (Joshi et al., 2020), an English-only encoder finetuned for English OntoNotes coreference. That model achieves 92.2 average (dev.) F1, compared to our 92.7. There is also a comparable system for Russian AnCor from Le et al. (2019), which achieves 79.9 F1 using the model from Lee et al. (2018) and RuBERT (Kuratov and Arkhipov, 2019). This shows that our single, multilingual model, can perform similarly to monolingual models, with the advantage that our model does not need to perform language ID. This finding mirrors prior findings showing multilingual encoders are strong cross-lingually (Wu and Dredze, 2019).

Language	# Training	# Eval Docs	Avg. F1
Arabic^{O}	359	44	71.3
Catalan^{S}	829	142	58.7
Chinese^{O}	1810	252	90.8
Dutch^{S}	145	23	63.5
English^{O}	2802	343	92.7
Italian^{S}	80	17	47.2
Russian^{A}	573	127	77.3
Spanish^{S}	875	140	63.5

Table 3: Average F1 scores by language with gold mentions. The superscripts O indicates data from OntoNotes 5.0 (dev), S indicates data from SemEval 2010 Task 1 (dev), and A is the AnCor data (test).

4.3 Entity Typing

We retrain the coarse-to-fine entity typer by Chen et al. (2020a) with *XLM-R* as the underlying encoder, and using the AIDA ontology as the type label inventory. The dataset annotated from AIDA is relatively small. To make the model more robust, we pre-train the model using extra training data from GAIA (Li et al., 2020), where they obtained YAGO fine-grained types (Suchanek et al., 2008) from the results of Freebase entity linking, and mapped these types to the AIDA ontology. After pre-training, we fine-tune the model using the AIDA M18 and M36 data with 3-fold cross-validation, where each fold is distinct in the topics of these documents. The sizes of these datasets are shown in Table 4.

Our models perform well in these datasets. Using one third of the AIDA M36 data as dev, our method obtained 60.1% micro-F_1 score;[9] with pre-training using GAIA extra data, we get 76.5%.

Data source	Language	# of entities
AIDA M18	English	4,433
	Russian	4,826
`LDC2019E07`	Ukrainian	4,261
AIDA M36	English	703
	Spanish	557
`LDC2020E29`	Russian	729
GAIA	English	42.8M
	Spanish	11.1M
	Russian	2.4M

Table 4: Statistics of the datasets used for training our entity typing model.

Our system can also be extended to support other commonly used fine-grained entity type ontologies. We report the results in micro-F_1 in Table 5.

Ontology	Prior state-of-the-art	Ours
BBN	78.1 (Lin and Ji, 2019)	**80.5**
FIGER	79.8 (Lin and Ji, 2019)	**80.8**
UltraFine	40.1 (Onoe and Durrett, 2019)	**41.5**

Table 5: Performance of our hierarchical entity typing model across several typing ontologies.

4.4 Temporal Relation Extraction

We retrain Vashishtha et al. (2019)'s best fine-grained temporal relation model on UDS-T (Vashishtha et al., 2019) using *XLM-R* (large). We then use their transfer learning approach and train an SVM model on event-event relations in TimeBank-Dense (TBD) to predict categorical temporal relation labels. With this approach, we see a micro-F1 score of 56 on the test set of TBD.[10]

For better performance, we train the same model on additional TempEval3 (TE3) dataset (UzZaman et al., 2013). Since TE3 and TBD use a different set of temporal relations, we consider only those instances that are labeled with 4 temporal relations from both TE3 and TBD for joint training – *before*, *after*, *includes* (*container*), and *is_included* (*contained*). We retrain Vashishtha et al. (2019)'s transfer learning model on the combined TE3 and TBD dataset considering only these 4 relations and evaluate on their combined test set.[11] Results on the combined test set are reported in Table 6. We use this model as the default temporal relation extraction model in LOME.

[9] Please refer to Chen et al. (2020a) for the exact definitions of the evaluation metric.

[10] The train and dev set of TBD has a total of 4,590 instances and the test set has 1,405 instances of event-event relations.

[11] We consider only event-event relations and the combined dataset has 5,987 (1,249) instances in the train (test) set.

We also test our default model on a Chinese temporal relation extraction dataset (Li et al., 2016).[12] In the zero-shot setting, we get a micro F1 score of 52.6 on the provided dataset, as compared to a majority baseline of 37.5.[13] Similar to the default temporal system in LOME, we use the *XLM-R* version of Vashishtha et al. (2019)'s model obtaining relation embeddings for the Chinese dataset and train an SVM model using the transfer learning approach to get a micro F1 score of 64.4.[14]

Relation	Precision	Recall	F1
before	68	89	77
after	74	69	71
includes	83	5	10
is_included	44	15	22

Table 6: Result on the combined test set of TempEval3 and TimeBank-Dense when trained with just 4 temporal relation labels

5 Extensions

5.1 Incorporating third-party systems

Besides the core components described above, we also discuss the viability of including additional modules that may not fit directly in the core pipeline but can be included depending on the downstream application. For example, the system described above does not predict any relation information, which is needed for the motivating application of downstream schema inference. To do so, we wrote a CONCRETE and Docker wrapper around OneIE (Lin et al., 2020) and attached it at the end of the pipeline. With our CONCRETE based design, the integration of any third-party module can be done via implementing the *AnnotateCommunicationService* service interface, which can ensure compatibility between LOME and external modules. The OneIE wrapper is one example of an external module.

5.2 Mix and Match Modules: SM-KBP

As another example application, we reconfigured our pipeline for the NIST SM-KBP 2020 Task 1 evaluation, which aims to produce document-level knowledge graphs.[15] Each given document may be in English, Russian, or Spanish. On a development set consisting solely of text-only documents,[16] we started with initial predictions made by GAIA (Li et al., 2020), for entity clusters, entity types, events and relations. Our goal was to recluster and relabel the a dataset for knowledge extraction.

Our pipeline consisted of the multilingual coreference resolution (using the predetermined mention from GAIA) and hierarchical entity typing models discussed in this paper, followed by a separate state-of-the-art argument linking model (Chen et al., 2020b). We found improved performance[17] with entity coreference (from 29.1 F1 to 33.3 F1), especially in Russian (from 26.2 F1 to 33.3 F1), likely due to our use of multilingual data and contextualized encoders. The improved entity clusters also led to downstream improvements in entity typing and argument linking. This example highlights the ability to pick out subcomponents of LOME and customize according to the downstream task.

6 Usage

We present two methods to interact with the pipeline. The first is a Docker container which contains the libraries, code, and trained models of our pipeline. This is intended to run on batches of documents. As a lighter demo of some of the system capabilities, we also have a web demo intended to interactively run on shorter documents.

Docker Our Docker image[18] consists of the four core modules: FrameNet parser, coreference resolution, entity typing, and temporal resolution. Furthermore, there are two options for entity typing: a fine-grained hierarchical model (with the AIDA typing ontology) and a coarse-grained model (with the KAIROS typing ontology). The container and documentation is available on Docker Hub.

As some modules depend on GPU libraries, the image also requires NVIDIA-Docker support. Since there is a high start-up (time) cost for using Docker and loading models, we recommend using this container for batch processing of documents. Further instructions for running can be found on the LOME Docker Hub page.

[12] We remove the instances with *unknown* relation from the dataset and convert the predictions with *includes* and *is_included* relations to the *overlaps* relation to match the label set of their dataset with our system.

[13] The authors were able to provide only half of the dataset with 10,476 event-event pairs, from which we ignore instances with *unknown* relation, resulting into 9,362 instances.

[14] The results are the average of the 5-fold cross validation splits provided by Li et al. (2016).

[15] `https://tac.nist.gov/2020/KBP/SM-KBP/index.html`

[16] AIDA M36, `LDC2020E29`.

[17] This evaluation metric is specific to the NIST SM-KBP 2020 task. It takes entity types into account.

[18] `https://hub.docker.com/r/hltcoe/lome`

Web Demo We make a few changes for the web demo.[19] To reduce latency, we preload the models into memory and we do not write the CONCRETE communications to disk. At the cost of modularity, this makes the demo lightweight and fast, allowing us to run it on a single 16GB CPU-only server. To present the predictions, our front-end uses `AllenNLP-demo`.[20]

In addition, the web demo is currently limited to FrameNet parsing and coreference resolution, as other models will increase latency and may impede usability. The web demo is intended to highlight only some of the system's capabilities, like its ability to process multilingual documents.

7 Conclusions

To facilitate increased interest in multilingual document-level knowledge extraction with large ontologies, we create and demonstrate LOME, a system for event and entity knowledge graph creation. Given input text documents, LOME runs a full FrameNet parser, coreference resolution, fine-grained entity typing, and temporal relation prediction. Furthermore, each component uses *XLM-R*, allowing our system to support a broader set of languages than previous systems. The pipeline uses a standardized data schema, which invites extending the pipeline with additional modules. By releasing both a Docker image and presenting a lightweight web demo, we hope to enable the community to build on top of LOME for even more comprehensive information extraction.

Acknowledgments

We thank Anton Belyy, Kenton Murray, Manling Li, Varun Iyer, and Zhuowan Li for helpful discussions and feedback. This work was supported in part by DARPA AIDA (FA8750-18-2-0015) and KAIROS (FA8750-19-2-0034). The views and conclusions contained in this work are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, or endorsements of DARPA or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Alan Akbik and Yunyao Li. 2016. POLYGLOT: Multilingual semantic role labeling with unified labels. In *Proceedings of ACL-2016 System Demonstrations*, pages 1–6, Berlin, Germany. Association for Computational Linguistics.

James Allen, Hannah An, Ritwik Bose, Will de Beaumont, and Choh Man Teng. 2020. A broad-coverage deep semantic lexicon for verbs. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3243–3251, Marseille, France. European Language Resources Association.

Amit Bagga and Breck Baldwin. 1998. Algorithms for scoring coreference chains. In *The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference*, pages 563–566.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Quebec, Canada. Association for Computational Linguistics.

A. E. Budnikov, S Yu Toldova, D. S. Zvereva, D. M. Maximova, and M. I. Ionov. 2019. Ru-eval-2019: Evaluating anaphora and coreference resolution for russian. In *Computational Linguistics and Intellectual Technologies - Supplementary Volume*.

Taylor Cassidy, Bill McDowell, Nathanael Chambers, and Steven Bethard. 2014. An annotation framework for dense event ordering. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 501–506, Baltimore, Maryland. Association for Computational Linguistics.

Nathanael Chambers, Taylor Cassidy, Bill McDowell, and Steven Bethard. 2014. Dense event ordering with a multi-pass architecture. *Transactions of the Association for Computational Linguistics*, 2:273–284.

Tongfei Chen, Yunmo Chen, and Benjamin Van Durme. 2020a. Hierarchical entity typing via multi-level learning to rank. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8465–8475, Online. Association for Computational Linguistics.

Yunmo Chen, Tongfei Chen, and Benjamin Van Durme. 2020b. Joint modeling of arguments for event understanding. In *Proceedings of the First Workshop on Computational Approaches to Discourse*, pages 96–101, Online. Association for Computational Linguistics.

Fei Cheng and Yusuke Miyao. 2017. Classifying temporal relations by bidirectional LSTM over dependency paths. In *Proceedings of the 55th Annual*

[19] `https://nlp.jhu.edu/demos/lome/`

[20] `https://github.com/allenai/allennlp-demo`.

Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 1–6, Vancouver, Canada. Association for Computational Linguistics.

Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. 2018. Ultra-fine entity typing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 87–96, Melbourne, Australia. Association for Computational Linguistics.

Christopher Cieri, James Fiumara, Stephanie Strassel, Jonathan Wright, Denise DiPersio, and Mark Liberman. 2020. A progress report on activities at the Linguistic Data Consortium benefitting the LREC community. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 3449–3456, Marseille, France. European Language Resources Association.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Dipanjan Das, Desai Chen, André F. T. Martins, Nathan Schneider, and Noah A. Smith. 2014. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56.

Francis Ferraro, Max Thomas, Matthew R. Gormley, Travis Wolfe, Craig Harman, and Benjamin Van Durme. 2014. Concretely annotated corpora. In *4th Workshop on Automated Knowledge Base Construction (AKBC)*.

Goran Glavaš and Jan Šnajder. 2014. Event graphs for information retrieval and multi-document summarization. *Expert Systems with Applications*, 41(15):6904–6916.

Karl Moritz Hermann, Dipanjan Das, Jason Weston, and Kuzman Ganchev. 2014. Semantic frame identification with distributed word representations. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1448–1458, Baltimore, Maryland. Association for Computational Linguistics.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. BERT for coreference resolution: Baselines and analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5803–5808, Hong Kong, China. Association for Computational Linguistics.

Chris Kedzie, Kathleen McKeown, and Fernando Diaz. 2015. Predicting salient updates for disaster summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1608–1617, Beijing, China. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Yuri Kuratov and Mikhail Arkhipov. 2019. Adaptation of deep bidirectional multilingual transformers for russian language. In *Computational Linguistics and Intellectual Technologies*, pages 333–339.

T. A. Le, M. A. Petrov, Y. M. Kuratov, and M. S. Burtsev. 2019. Sentence level representation and language models in the task of coreference resolution for russian. In *Computational Linguistics and Intellectual Technologies*, pages 364–373.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, New Orleans, Louisiana. Association for Computational Linguistics.

Artuur Leeuwenberg and Marie-Francine Moens. 2017. Structured learning for temporal relation extraction from clinical records. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1150–1158, Valencia, Spain. Association for Computational Linguistics.

Manling Li, Ying Lin, Joseph Hoover, Spencer Whitehead, Clare Voss, Morteza Dehghani, and Heng Ji. 2019. Multilingual entity, relation, event and human value extraction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 110–115, Minneapolis, Minnesota. Association for Computational Linguistics.

Manling Li, Alireza Zareian, Ying Lin, Xiaoman Pan, Spencer Whitehead, Brian Chen, Bo Wu, Heng Ji, Shih-Fu Chang, Clare Voss, Daniel Napierski, and Marjorie Freedman. 2020. GAIA: A fine-grained multimedia knowledge extraction system. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 77–86, Online. Association for Computational Linguistics.

Peifeng Li, Qiaoming Zhu, Guodong Zhou, and Hongling Wang. 2016. Global inference to Chinese temporal relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1451–1460, Osaka, Japan. The COLING 2016 Organizing Committee.

Ying Lin and Heng Ji. 2019. An attentive fine-grained entity typing model with latent type representation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6197–6202, Hong Kong, China. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, pages 94–100.

Hector Llorens, Nathanael Chambers, Naushad UzZaman, Nasrin Mostafazadeh, James Allen, and James Pustejovsky. 2015. SemEval-2015 task 5: QA TempEval - evaluating temporal information understanding with question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 792–800, Denver, Colorado. Association for Computational Linguistics.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 25–32, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Qiang Ning, Zhili Feng, and Dan Roth. 2017. A structured learning approach to temporal relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1027–1037, Copenhagen, Denmark. Association for Computational Linguistics.

Yasumasa Onoe and Greg Durrett. 2019. Learning to denoise distantly-labeled data for entity typing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2407–2417, Minneapolis, Minnesota. Association for Computational Linguistics.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958, Vancouver, Canada. Association for Computational Linguistics.

Hao Peng, Sam Thomson, Swabha Swayamdipta, and Noah A. Smith. 2018. Learning joint semantic parsers from disjoint data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1492–1502, New Orleans, Louisiana. Association for Computational Linguistics.

Nanyun Peng, Francis Ferraro, Mo Yu, Nicholas Andrews, Jay DeYoung, Max Thomas, Matthew R. Gormley, Travis Wolfe, Craig Harman, Benjamin Van Durme, and Mark Dredze. 2015. A concrete Chinese NLP pipeline. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 86–90, Denver, Colorado. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using OntoNotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 143–152, Sofia, Bulgaria. Association for Computational Linguistics.

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro, et al. 2003. The Timebank corpus. In *Corpus linguistics*, volume 2003, page 40. Lancaster, UK.

Marta Recasens, Lluís Màrquez, Emili Sapena, M. Antònia Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, and Yannick Versley. 2010. SemEval-2010 task 1: Coreference resolution in multiple languages. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 1–8, Uppsala, Sweden. Association for Computational Linguistics.

Guergana K Savova, James J Masanz, Philip V Ogren, Jiaping Zheng, Sunghwan Sohn, Karin C Kipper-Schuler, and Christopher G Chute. 2010. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association*, 17(5):507–513.

Ergin Soysal, Jingqi Wang, Min Jiang, Yonghui Wu, Serguei Pakhomov, Hongfang Liu, and Hua Xu. 2018. Clamp–a toolkit for efficiently building customized clinical natural language processing pipelines. *Journal of the American Medical Informatics Association*, 25(3):331–336.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2008. YAGO: A large ontology from wikipedia and wordnet. *Journal of Web Semantics*, 6(3):203–217.

Weiyi Sun, Anna Rumshisky, and Ozlem Uzuner. 2013. Evaluating temporal relations in clinical text: 2012 i2b2 challenge. *Journal of the American Medical Informatics Association*, 20(5):806–813.

Swabha Swayamdipta, Sam Thomson, Kenton Lee, Luke Zettlemoyer, Chris Dyer, and Noah A. Smith. 2018. Syntactic scaffolds for semantic structures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3772–3782, Brussels, Belgium. Association for Computational Linguistics.

S Toldova, A. Roytberg, Alina Ladygina, Maria Vasilyeva, Ilya Azerkovich, Matvei Kurzukov, G. Sim, D.V. Gorshkov, A. Ivanova, Anna Nedoluzhko, and Y. Grishina. 2014. Ru-eval-2014: Evaluating anaphora and coreference resolution for russian. *Computational Linguistics and Intellectual Technologies*, pages 681–694.

Naushad UzZaman, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, and James Pustejovsky. 2013. SemEval-2013 task 1: TempEval-3: Evaluating time expressions, events, and temporal relations. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 1–9, Atlanta, Georgia, USA. Association for Computational Linguistics.

Siddharth Vashishtha, Benjamin Van Durme, and Aaron Steven White. 2019. Fine-grained temporal relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2906–2919, Florence, Italy. Association for Computational Linguistics.

Marc Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz, and James Pustejovsky. 2007. SemEval-2007 task 15: TempEval temporal relation identification. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 75–80, Prague, Czech Republic. Association for Computational Linguistics.

Marc Verhagen, Roser Saurí, Tommaso Caselli, and James Pustejovsky. 2010. Semeval-2010 task 13: Tempeval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 57–62, Uppsala, Sweden. Association for Computational Linguistics.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Sixth Message Understanding Conference (MUC-6): Proceedings of a Conference Held in Columbia, Maryland, November 6-8, 1995.*

Ralph Weischedel and Ada Brunstein. 2005. BBN pronoun coreference and entity type corpus. *Philadelphia: Linguistic Data Consortium.*

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. OntoNotes release 5.0. *Linguistic Data Consortium, Philadelphia, PA.*

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Patrick Xia, João Sedoc, and Benjamin Van Durme. 2020. Incremental neural coreference resolution in constant memory. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8617–8624, Online. Association for Computational Linguistics.

Liyan Xu and Jinho D. Choi. 2020. Revealing the myth of higher-order inference in coreference resolution. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8527–8533, Online. Association for Computational Linguistics.

Bishan Yang and Tom Mitchell. 2017. A joint sequential and relational model for frame-semantic parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256, Copenhagen, Denmark. Association for Computational Linguistics.

Ben Zhou, Daniel Khashabi, Qiang Ning, and Dan Roth. 2019. "Going on a vacation" takes longer than "going for a walk": A study of temporal commonsense understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3363–3369, Hong Kong, China. Association for Computational Linguistics.

Yichao Zhou, Yu Yan, Rujun Han, J Harry Caufield, Kai-Wei Chang, Yizhou Sun, Peipei Ping, and Wei Wang. 2021. Clinical temporal relation extraction with probabilistic soft logic regularization and global inference. In *Proceedings of AAAI 2021*.