XLTime: A Cross-Lingual Knowledge Transfer Framework for Temporal Expression Extraction

Yuwei Cao[1], William Groves[2], Tanay Kumar Saha[3], Joel R. Tetreault[2], Alex Jaimes[2], Hao Peng[4], Philip S. Yu[1]
[1]Computer Science Department, University of Illinois Chicago
[2]Dataminr Inc. [3]Walmart Global Tech
[4]BDBC, Beihang University
{ycao43,psyu}@uic.edu, {wgroves,jtetreault,ajaimes}@dataminr.com
tanaykumar.saha@walmart.com, penghao@buaa.edu.cn

Abstract

Temporal Expression Extraction (TEE) is essential for understanding time in natural language. It has applications in Natural Language Processing (NLP) tasks such as question answering, information retrieval, and causal inference. To date, work in this area has mostly focused on English as there is a scarcity of labeled data for other languages. We propose XLTime, a novel framework for multilingual TEE. XLTime works on top of pre-trained language models and leverages multi-task learning to prompt cross-language knowledge transfer both from English and within the non-English languages. XLTime alleviates problems caused by a shortage of data in the target language. We apply XLTime with different language models and show that it outperforms the previous automatic SOTA methods on French, Spanish, Portuguese, and Basque, by large margins. XLTime also closes the gap considerably on the handcrafted HeidelTime method.

1 Introduction

Temporal Expression Extraction (TEE) refers to the detection of *temporal expressions* (such as dates, durations, etc., as shown in Table 1). It is an important NLP task (UzZaman et al., 2013) and has downstream applications in question answering (Choi et al., 2018), information retrieval (Mitra et al., 2018), and causal inference (Feder et al., 2021). Most TEE methods work on English and are rule-based (Strötgen and Gertz, 2013; Zhong et al., 2017). Deep learning-based methods (Chen et al., 2019; Lange et al., 2020) are less common and report results on par with or inferior to the rule-based SOTAs.

Moreover, methods that work on other languages are rare, because of the scarcity of annotated data. We find that that there is considerable room for improving TEE, especially for low-resource languages. For example, the previous SOTA performance on the English TE3 dataset (UzZaman et al., 2013) is around 0.90 in F1, while that on the Basque TEE benchmark (Altuna et al., 2016) is merely 0.47. Recent deep learning methods, which have shown gains for many tasks, are underexplored for this important area of NLP.

In the last three months (**Duration**), net revenue rose 4.3% to $525.8 million from $504.2 million last year (**Date**).
The official news agency, which gives the daily (**Set**) tally of inspections, updated on Friday evening (**Time**).

Table 1: Temporal expressions of different types (See Appendix A for the definitions of the types).

Developing an approach that can learn using the existing limited amount of training data is crucial for this field because of the effort required to develop high-quality rules for each language. Thus we propose a cross-lingual knowledge transfer framework for multilingual TEE, namely, XLTime. We base our framework on pre-trained multilingual models (Devlin et al., 2019; Conneau et al., 2020). We then use Multi-Task Learning (MTL) (Liu et al., 2019a) to prompt knowledge transfer both from English and within the low-resource languages. For this, we design primary and secondary tasks. The primary task leverages the existing, annotated TEE data of the other languages. It transfers *explicit knowledge* that tells the forms of the temporal expressions in a *source language*. The secondary task maps the annotated source language TEE data samples to the target language using machine-translation tools, such as Google Translate, and acquires sentence-level labels (of the presence of one or more time expressions) from the original token-level labels. It constructs training data in a weakly-supervised manner. The secondary task transfers *implicit knowledge* by teaching the model to detect the presence of temporal

Findings of the Association for Computational Linguistics: NAACL 2022, pages 1931 - 1942

expressions in text from the target language.

Contributions. **1)** We propose XLTime, which prompts cross-lingual knowledge transfer using MTL to address multilingual TEE. **2)** We show that XLTime outperforms the previous automatic SOTA methods by large margins on four languages including French (FR), Spanish (ES), Portuguese (PT), and Basque (EU), which are "low-resource" for the TEE task. **3)** We show that XLTime also approaches the performance of the heavily handcrafted HeidelTime (Strötgen and Gertz, 2013), and XLTime even outperforms it on two languages (Portuguese and Basque). We make our code and data publicly available.[1]

2 Related Work

While TEE is an important problem in NLP, there is relatively little work in the area, and most of this work focuses on English. Prior art can be divided into two classes: rule/pattern-based and deep learning approaches. In the first class, HeidelTime (Strötgen and Gertz, 2013) is the top performing approach to date, and covers over a dozen languages. It is driven by a collection of finely-tuned rules. The approach was later extended to more languages with HeidelTime-auto (Strötgen and Gertz, 2015), which leverages language-independent processing and rules. Other approaches include SynTime (Zhong et al., 2017), which is based on heuristic rules, and SUTIME (Chang and Manning, 2012) and PTime (Ding et al., 2019), which leverages pattern learning.

For the second class, Laparra et al. (2018) proposes a model based on RNNs. Chen et al. (2019) uses BERT with a linear classifier. Lange et al. (2020) inputs mBERT embeddings to a BiLSTM with a CRF layer and outperforms HeidelTime-auto on four languages. However, the reported performances of the deep learning-based methods are inferior to the rule-based ones, which is, in part, due to the complexity of the problem and training data paucity. In our work, we propose a new model which outperforms prior deep learning methods but also closes the gap considerably on HeidelTime, despite the data issues.

In addition, we are aware that applying label projection methods (Jain et al., 2019) can be a straightforward way to address the data scarcity in non-English TEE. TMP (Jain et al., 2019), originally proposed for cross-lingual named entity recognition (NER) (Lample et al., 2016), projects English data in IOB (Inside Outside Beginning) tagging format (Ramshaw and Marcus, 1999) to that of the other languages using machine translation, orthographic, and phonetic similarity packages. We show that the proposed XLTime, specifically designed to transfer temporal knowledge between languages, outperforms TMP by large margins.

[1] https://github.com/YuweiCao-UIC/XLTime

3 Proposed Method

We formalize TEE as a sequence labeling task, similar to NER (Lample et al., 2016). The architecture is shown in Figure 1.

3.1 Pre-trained Multilingual Backbone

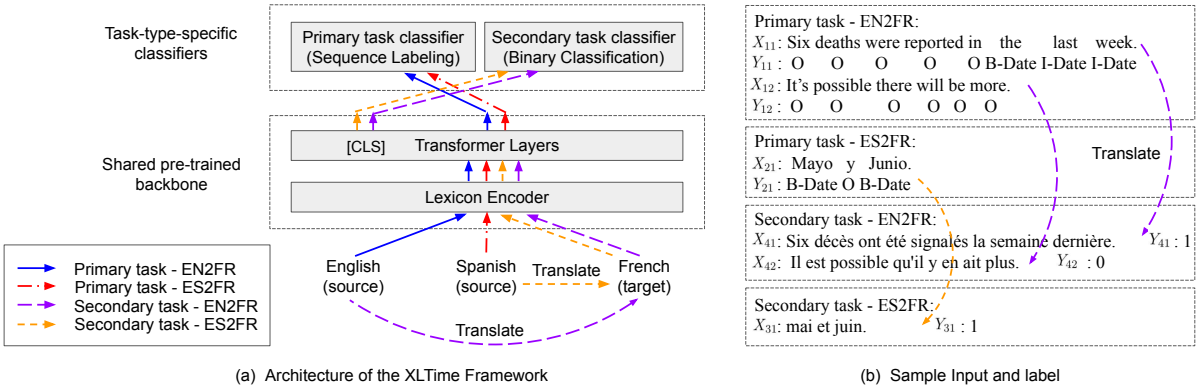
XLTime adopts SOTA multilingual models, i.e., mBERT (Devlin et al., 2019) and XLMR (Conneau et al., 2020) as the backbone. The pre-trained backbone contains lexicon and Transformer encoder layers as shown in Figure 1(a). The backbone allows XLTime to acquire semantic and syntactic knowledge of various languages. The backbone is shared by the MTL tasks introduced in Section 3.2.

3.2 MTL-based Cross-Lingual Knowledge Transfer

XLTime transfers knowledge from multiple *source languages* to the low-resource *target language*. The source languages include English and others for which TEE training data is available. We design *primary* and *secondary* tasks on top of the backbone to prompt *explicit* and *implicit* knowledge transfer. The primary task transfers knowledge that explicitly encodes the forms of the temporal expressions in a source language. It is formalized as sequence labeling and directly leverages the training data of the source language to train the backbone along with the primary task classifier, shown in Figure 1 (a). The primary task minimizes $\mathcal{L}_{sl}$:

$$\mathcal{L}_{sl} = -\sum_{i=1}^{b}\sum_{j=1}^{m_i} \mathbb{1}(y_{ij}, c) log(softmax(\mathbf{W} \cdot \mathbf{x})), \quad (1)$$

where b is the total number of input sequences and m_i is the length of the ith sequence. $\mathbf{x} \in \mathbb{R}^d$, output by the backbone, is the embedding of the jth token in the ith sequence. d is its dimension. $c = argmax(\mathbf{W} \cdot \mathbf{x})$ and y_{ij} are the predicted and ground-truth labels of the token. $\mathbf{W} \in \mathbb{R}^{|c| \times d}$ is the parameter of the primary task classifier. $|c|$ is the total number of unique ground-truth labels. $\mathbb{1}(,)$ is 1 if its two arguments are equal and 0 otherwise.



(a) Architecture of the XLTime Framework (b) Sample Input and label

Figure 1: The architecture and sample training input of the proposed XLTime framework (*best viewed in color)*. (a) shows how XLTime transfers knowledge from English (EN) and Spanish (ES) to French (FR) through the primary and the secondary tasks. (b) presents sample input of the tasks.

The secondary task implicitly reveals how the temporal expressions would be expressed in the target language. We translate the sequences in the source language training data into the target language using Google Translate (we observe similar results with AWS Translate). The secondary task is formalized as binary classification, where the input samples are the translated sequences and the labels are sentence-level indicators of whether or not the sequences contain temporal expressions (which can be easily inferred from the original labels). This task tunes the model to learn the characteristics of temporal expressions in the target language in an implicit manner. It is weakly-supervised and requires no token-level labeling. It trains the backbone and the secondary task classifier by minimizing $\mathcal{L}_{bc}$:

$$\mathcal{L}_{bc} = -\sum_{i=1}^{b} \mathbb{1}(y_i', c') log(softmax(\mathbf{W}' \cdot \mathbf{x}')), \quad (2)$$

where $\mathbf{x}' \in \mathbb{R}^d$ is the sequence embedding output by the [CLS] of the backbone. $\mathbf{W}' \in \mathbb{R}^{2 \times d}$ is the parameter matrix of the secondary task classifier. $c' = argmax(\mathbf{W}' \cdot \mathbf{x}')$ and y_i' are the predicted and true sequence labels of the ith sequence. We train XLTime concurrently on the primary and secondary tasks (further details found in Appendix B).

An Illustrative Example. In Figure 1, *Primary task - EN2FR* and *Secondary task - EN2FR* transfer knowledge from *English* to *French*. *Primary task - EN2FR* reveals the exact forms of English temporal expressions using token-level labels (Y_{11} and Y_{12}). *Secondary task - EN2FR* takes the French translations (X_{41} and X_{42}) of X_{11} and X_{12} as input. Y_{41} and Y_{42} indicate whether the sequences contain temporal expressions or not (can be inferred from Y_{11} and Y_{12}). *Secondary task - EN2FR* provides indirect knowledge about French temporal expressions. Similarly, *Primary task - ES2FR* and *Secondary task - ES2FR* transfer from *Spanish* to *French*.

4 Experiments

This section evaluates the proposed XLTime framework. Section 4.1 introduces the datasets, models evaluated, metrics, and experimental settings. Section 4.2 quantitatively shows how XLTime alleviates data scarcity and prompts TEE performances. Section 4.3 studies the effect of transferring knowledge from other languages in addition to English. We also qualitatively show how XLTime transfers knowledge to the target languages in an error analysis in Appendix E.

4.1 Experimental Setup

Datasets. We use the English (EN), French (FR), Spanish (ES), Portuguese (PT), and Basque (EU) TEE benchmark datasets. Table 2 shows dataset statistics. For each target language, we split its dataset with 10% for validation and 90% for test. For each source language (applicable to XLTime), we use the whole dataset for training.

Baselines. We evaluate against rule-based, deep learning-based, and entity projection-based methods. We compare to the handcrafted HeidelTime (Strötgen and Gertz, 2013) and its automatically extended version, HeidelTime-auto (Strötgen and Gertz, 2015). We also compare to deep learning methods: BiLSTM+CRF (Lange et al., 2020), mBERT, base and large versions of XLMR. In addition, we compare to TMP (Jain et al., 2019), a cross-lingual label projection method which relies

Table 2: The statistics of the datasets.

Lang	Dataset	Domain	#Docs	#Exprs	#Dates	#Times	#Durations	#Sets
FR	Bittar et al. (2011)	News	108	425	227	130	52	16
ES	UzZaman et al. (2013)	News	175	1,094	749	57	251	37
PT	Costa and Branco (2012)	News	182	1,227	998	41	176	12
EU	Altuna et al. (2016)	News	91	847	662	22	151	12
EN	TE3 (UzZaman et al., 2013)	News	276	1,830	1,471	34	291	34
	Wikiwars (Mazur and Dale, 2010)	Narrative	22	2,634	2,634	0	0	0
	Tweets (Zhong et al., 2017)	Utterance	942	1,128	717	173	200	38

Model	FR	ES	PT	EU
Automatic Baseline Models				
HeidelTime-auto	0.55	0.42	0.50	0.17
BiLSTM+CRF	0.64	0.62	0.64	0.47
mBERT	0.63	0.62	0.66	0.65
XLMR-base	0.69	0.54	0.63	0.46
XLMR-large	0.75	0.72	0.75	0.70
Projection Method				
TMP-mBERT	0.56	0.23	0.66	/
TMP-XLMRbase	0.55	0.23	0.64	/
TMP-XLMRlarge	0.56	0.24	0.65	/
Transfer from EN (Ours)				
XLTime-mBERT	0.73	0.71	0.67	0.76
XLTime-XLMRbase	0.78	0.66	0.68	0.71
XLTime-XLMRlarge	0.76	0.72	0.77	*0.78*
Transfer from EN and others (Ours)				
XLTime-mBERT	0.80	**0.77**	*0.80*	0.77
XLTime-XLMRbase	*0.82*	0.72	0.73	**0.79**
XLTime-XLMRlarge	**0.84**	*0.75*	**0.84**	**0.79**
Handcrafted Method				
HeidelTime	0.86	0.86	0.60	/

Table 3: Results for Multilingual TEE (Metric: F1).

on machine translation as well as orthographic and phonetic similarity packages (unavailable for EU).

Our Approaches. We test several variants of our proposed model, which can be broken into two classes: 1) Cross-lingual transfer from EN. We apply XLTime on mBERT, base and large versions of XLMR and use EN as the only source language. 2) Cross-lingual transfer from EN and others. We transfer from other languages in addition to EN.

Evaluation Metrics. We report F1in *strict match* (UzZaman et al., 2013), i.e., all its tokens must be correctly recognized for an expression to be counted as correctly extracted.

We follow the setting in prior work of evaluating "without type" and report the results without considering the types of the temporal expressions (e.g., for 'see you tomorrow', a prediction such as 'O O B-Duration' would be counted as correct, though the proper labeling would be 'O O B-Date').[2]

Experimental Setting. We set d, the embedding dimension, to be consistent with the pre-trained multilingual backbone's dimension (768 for the base version language models and 1024 for large versions). We use AdamW (Loshchilov and Hutter, 2019) with a learning rate of $7e^{-6}$ and warm-up proportion of 0.1. We train the models for 50 epochs and use the best model as indicated by the validation set for prediction. All datasets are transformed into IOB2 format to fit the sequence labeling setting. All the deep learning methods are trained on English TEE datasets, validated and evaluated on low-resource languages. For BiLSTM+CRF, we use the hyperparameters as suggested in the original paper (Lange et al., 2020). For TMP, we use it to project the English dataset to the target languages, take the projected data to train the language models, then validate and evaluate on the target languages. We perform a grid search over $\{0.05, 0.1, 0.15, 0.25, 0.5\}$ to tune δ, the similarity score threshold of TMP, and present the best performance. We repeat all experiments for 5 times and report the mean result. All experiments are conducted on a 64 core Intel Xeon CPU E5-2680 v4@2.40GHz with 512GB RAM and 1×NVIDIA Tesla P100-PICE GPU.

4.2 Multilingual TEE

We evaluate XLTime on multilingual TEE (see Table 3 and Appendix D). We observe: **1)** XLTime-XLMRlarge outperforms the strongest automatic baseline by up to 9% in F1 on all languages. It even outperforms the handcrafted HeidelTime method by a sizable margin (24% in F1) in PT. **2)** Applying XLTime improves upon the vanilla language models, even when transferring knowledge only from EN. E.g., XLTime-XLMRbase outperforms XLMR-base by 13%, 22%, 8%, and 54% in F1 on FR, ES, PT, and EU. **3)** Introducing ad-

[2]We do note that the temporal expression field should ultimately evaluate on the more complex task of identifying temporal expressions as well as their types. This is in the spirit of the annotations and is in line with other sequence labeling tasks, such as NER. Therefore, we also experiment with the "with type" setting and show results in Appendix C. In both settings, the observations made in Sections 4.2 and 4.3 hold and XLTime outperforms the previous SOTAs by large margins.

Target Language	FR				ES			
Source Language(s)	EN	EN, EU	EN, PT	EN, ES	EN	EN, EU	EN, PT	EN, FR
XLTime-mBERT	0.73	0.76	0.72	0.80	0.71	0.72	0.72	0.77
XLTime-XLMRbase	0.78	0.76	0.78	0.82	0.66	0.68	0.71	0.72
XLTime-XLMRlarge	0.76	0.81	0.80	0.84	0.72	0.72	0.75	0.73
Target Language	PT				EU			
Source Language(s)	EN	EN, FR	EN, ES	EN, EU	EN	EN, PT	EN, ES	EN, FR
XLTime-mBERT	0.67	0.80	0.70	0.80	0.76	0.73	0.75	0.77
XLTime-XLMRbase	0.68	0.73	0.63	0.56	0.71	0.74	0.75	0.79
XLTime-XLMRlarge	0.77	0.82	0.84	0.74	0.78	0.79	0.79	0.77

Table 4: Low-resource language TEE with additional source languages (F1 scores). The blue cells are expected to, while the underlined cells actually outperform (by $\geq 4\%$) using EN as the only source language.

ditional source languages to XLTime further improves the performance: the F1 improves by up to 19%, 11%, and 11% for XLTime-mBERT, XLTime-XLMRbase, and XLTime-XLMRlarge. **4)** HeidelTime is a very hard baseline to beat given the time and care that went into developing language-specific rules. However, XLTime approaches its performance for FR and ES, outperforms it for PT, and makes predictions for EU (where HeidelTime has no rules). Note the previous automatic SOTA, XLMR-large, also outperforms HeidelTime for PT, but not as significantly. This shows that the automatic methods are increasingly promising for the non-English TEE task. **5)** XLTime-XLMRlarge improves upon XLMR-large by a large margin (11% in F1) in EU. For FR, ES, and PT, the improvements are smaller. This may because XLMR-large, compared to mBERT and XLMR-base, is already very knowledgeable (especially in FR, ES, and PT, which are more common than EU). Therefore, applying XLTime may not provide much improvement (in contrast, applying XLTime on mBERT and XLMR-base dramatically boosts F1 by 8-54%). **6)** TMP performs poorly probably because the falsely projected entities can mislead the language models. Specifically, the token-by-token machine translation and matching process of TMP does not work well for temporal entities, especially when the target language TEs contain definite articles, prepositions, etc., that do not have explicit matches in the source language. E.g., EN TE 'yesterday morning' can be correctly map to FR TE 'hier matin' ('yesterday' to 'hier' and 'morning' to 'Matin') but not to EU TE 'ayer por la mañana' ('yesterday' to 'ayer' and 'morning' to 'Mañana', leaving 'por' and 'la' unmatched).

4.3 Transfer Knowledge from Additional Languages

We also study the effect of transferring additional knowledge from a low-resource language in addition to English, see Table 4 and Appendix D. Our assumption is that similar languages (FR, ES, and PT) would help each other (one exception is PT, as the published dataset is EN text translated to PT and we, therefore, don't expect machine translation to provide additional knowledge). We observe: **1)** In most cases, transferring additional knowledge from similar languages (blue cells) does dramatically improve performance (underlined cells), with F1 increasing by up to 13%. **2)** In some rare cases, negative knowlege transfer (Wu et al., 2020) occurs as adding source languages hurts performance (e.g., EN, ES $\rightarrow$ PT scores lower than EN $\rightarrow$ PT for XLTime-XLMRbase). We hypothesize this is related to the quality of the datasets and plan to address this in the future.

5 Conclusion

We propose XLTime for multilingual language TEE in low-resource scenarios. It is based on language models and leverages MTL to prompt cross-language knowledge transfer. It greatly alleviates the problems caused by the shortage in training data and shows results superior to the previous automatic SOTA methods on four languages. It also approaches the performance of a highly engineered rule-based system.

Acknowledgements

This work is supported in part by NSF under grants III-1763325, III-1909323, III-2106758, SaTC-1930941 and the S&T Program of Hebei through grant 21340301D. For any correspondence, please refer to Hao Peng.

References

Begoña Altuna, María Jesús Aranzabe, and Arantza Díaz de Ilarraza. 2016. Adapting timeml to basque: Event annotation. In *Proceedings of CICLing 2016*, pages 565–577. Springer.

André Bittar, Pascal Amsili, Pascal Denis, and Laurence Danlos. 2011. French timebank: an iso-timeml annotated reference corpus. In *Proceedings of ACL-HLT 2011*, pages 130–134.

Angel X Chang and Christopher D Manning. 2012. Sutime: A library for recognizing and normalizing time expressions. In *Lrec*, volume 3735, page 3740.

Sanxing Chen, Guoxin Wang, and Börje Karlsson. 2019. Exploring word representations on time expression recognition. Technical report, Tech. rep., Microsoft Research Asia.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. In *Proceedings of EMNLP 2021*.

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 ACL 2020*, pages 8440–8451.

Francisco Costa and António Branco. 2012. Timebankpt: A timeml annotated corpus of portuguese. In *LREC*, volume 12, pages 3727–3734.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*, pages 4171–4186.

Wentao Ding, Guanji Gao, Linfeng Shi, and Yuzhong Qu. 2019. A pattern-based approach to recognizing time expressions. In *Proceedings of AAAI 2019*, volume 33, pages 6335–6342.

Amir Feder, Katherine A. Keith, Emaad Manzoor, Reid Pryzant, Dhanya Sridhar, Zach Wood-Doughty, Jacob Eisenstein, Justin Grimmer, Roi Reichart, Margaret E. Roberts, Brandon M. Stewart, Victor Veitch, and Diyi Yang. 2021. Causal inference in natural language processing: Estimation, prediction, interpretation and beyond.

Alankar Jain, Bhargavi Paranjape, and Zachary C. Lipton. 2019. Entity projection via machine translation for cross-lingual NER. 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 1083–1092, Hong Kong, China. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of NAACL-HLT 2016*, pages 260–270.

Lukas Lange, Anastasiia Iurshina, Heike Adel, and Jannik Strötgen. 2020. Adversarial alignment of multilingual models for extracting temporal expressions from text. In *Proceedings of Workshop on Representation Learning for NLP at ACL 2020*, pages 103–109.

Egoitz Laparra, Dongfang Xu, and Steven Bethard. 2018. From characters to time intervals: New paradigms for evaluation and neural parsing of time normalizations. *Transactions of the Association for Computational Linguistics*, 6:343–356.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. In *Proceedings of ACL 2019*, pages 4487–4496.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of ICLR 2019*.

Pawel Mazur and Robert Dale. 2010. Wikiwars: A new corpus for research on temporal expressions. In *Proceedings of EMNLP 2010*, pages 913–922.

Bhaskar Mitra, Nick Craswell, et al. 2018. *An introduction to neural information retrieval*. Now Foundations and Trends.

James Pustejovsky, Kiyong Lee, Harry Bunt, and Laurent Romary. 2010. Iso-timeml: An international standard for semantic annotation. In *LREC*, volume 10, pages 394–397.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

Lance A Ramshaw and Mitchell P Marcus. 1999. Text chunking using transformation-based learning. In *Natural language processing using very large corpora*, pages 157–176. Springer.

Jannik Strötgen and Michael Gertz. 2013. Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2):269–298.

Jannik Strötgen and Michael Gertz. 2015. A baseline temporal tagger for all languages. In *Proceedings of EMNLP 2015*, pages 541–547.

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, Volume 2: Proceedings of SemEval 2013*, pages 1–9.

Sen Wu, Hongyang R Zhang, and Christopher Ré. 2020. Understanding and improving information transfer in multi-task learning. In *Proceedings of ICLR 2020*.

Xiaoshi Zhong, Aixin Sun, and Erik Cambria. 2017. Time expression analysis and recognition using syntactic token types and general heuristic rules. In *Proceedings of ACL 2017*, pages 420–429.

A Types of the Temporal Expressions

According to ISO-TimeML (Pustejovsky et al., 2010), the TEE dataset annotation guideline, there are four types of temporal expressions, i.e., *Date*, *Time*, *Duration*, and *Set*. *Date* refers to a calendar date, generally of a day or a larger temporal unit; *Time* refers to a time of the day and the granularity of which is smaller than a day; *Duration* refers to the expressions that explicitly describe some period of time; *Set* refers to a set of regularly recurring times (Pustejovsky et al., 2010).

Algorithm 1: Training XLTime

```
// Initialize model.
Load the parameters from a pre-trained
  multilingual model.
Initialize W and W' randomly.
// Prepare task data.
for t in {primary, secondary} do
    Split the data of task t into
      mini-batches B_t
B = B_primary ∪ B_secondary
for e in 1, ..., epoch do
    Randomly shuffle B
    // b_t is a mini-batch of task t
    for b_t in B do
        if t is a primary task then
            L_sl = Equation 1
        else
            L_bc = Equation 2
        Compute gradient and update
          model parameters
```

B The Training Procedure

We adopt mini-batch-based stochastic gradient descent (SGD) to train XLTime, as shown in Algorithm 1. To concurrently train on the primary and secondary tasks, we split the training data of both tasks into mini-batches and randomly take one mini-batch at each step. We then calculate loss using that mini-batch and update the parameters of the shared backbone as well as the task-type-specific classifier. The classifier of the other task type is unaffected.

C Full Results for Low-resource Language TEE

Table 7 shows the full results for low-resource language TEE with/without considering the types of the temporal expressions. Note that the superiority of our proposed XLTime over the previous automatic SOTA still holds.

D Full Results for Low-resource Language TEE with Additional Source Languages

Tables 8 and 9 show the full results for low-resource language TEE with additional source languages.

E Comparative Error Analysis

This section qualitatively shows how the proposed XLTime framework transfers knowledge to the target language. Specifically, we show how the errors made by the vanilla multilingual models can be fixed by applying XLTime. We also show how applying XLTime on other languages in addition to English would help fix more errors.

We compare mBERT and XLTime-mBERT (transfer from EN) on FR TEE. Table 5 summarizes cases where mBERT fails while XLTime-mBERT gives correct predictions. We can tell that XLTime-mBERT learns 'hier (yesterday)', which is not understood by the mBERT model. XLTime-mBERT also learns to recognize vague time spans such as 'désormais (from now on)' and 'longtemps (long time)', which are missed by the mBERT model. Moreover, compared to mBERT, XLTime-mBERT understands FR grammar better, as it recognizes the roles of definite articles and adjectives, such as 'le (the)' and 'prochain (next)', in TEs. In a word, the proposed XLTime framework helps connect the concepts in EN to the corresponding ones in FR.

To show how applying XLTime on extra source languages would help fix more errors, we compare XLTime-mBERT (transfer from EN) and XLTime-mBERT (transfer from EN and ES) on FR TEE. Table 6 summarizes the TEs that the former fails while the latter gives correct predictions. We can tell that by leveraging ES as an additional source language, XLTime-mBERT better masters FR grammar. Specifically, it learns to recognize definite articles and prepositions that share similar (e.g., 'le/los') or identical (e.g., 'de' and 'en') forms in ES and FR. It can also better distinguish TEs of different types (e.g., it learns that 'quelques jours

Table 5: mBERT vs. XLTime-mBERT (transfer from EN) frequent (count $\geq$ 10) errors.

Error Desc.	FR TEs	EN translations	mBERT results (wrong)	XLTime results (correct)	counts
fail to recognize 'hier (yesterday)'	hier soir hier	last night yesterday	O B-TIME O	B-TIME I-TIME B-DATE	30
fail to recognize vague time span	désormais longtemps toute l'année	from now on long time all year	O O O O	B-DATE B-DURATION B-SET I-SET	6
fail to recognize definite articles and adjectives	le 3 août la nuit lundi prochain	August 3 the night next Monday	O B-DATE I-DATE O O B-DATE O	B-DATE I-DATE I-DATE B-TIME I-TIME B-DATE I-DATE	10

Table 6: XLTime-mBERT (transfer from EN) vs. XLTime-mBERT (transfer from EN and ES) frequent (count $\geq$ 8) errors.

Error Desc.	FR TEs	EN translations	EN results (wrong)	EN and ES results (correct)	counts
fail to recognize definite articles and prepositions	en été le 13 février de dimanche	in summer February 13 of Sunday	B-DATE I-DATE O B-DATE I-DATE B-DATE I-DATE	O B-DATE B-DATE I-DATE I-DATE O B-DATE	20
wrong token types	mardi quelques jours	Tuesday A few days	B-TIME B-DATE I-DATE	B-DATE B-DURATION I-DURATION	18
recognized extra TEs	quotidiens la saison	daily the season	B-SET B-DATE I-DATE	O O O	8

(a few days)' is a *Duration*, instead of a *Date*). One interesting fact is, when transferring solely from EN, the model recognizes some extra TEs that are not in the ground truth of the FR dataset. This is because of an inconsistency in data labeling: 'daily' is considered as a *Set* in the EN dataset, while its counterpart, 'quotidiens' is overlooked in the FR dataset. The proposed XLTime framework eliminates the needs of manually labeling multiple datasets and therefore, can be applied to minimize data label inconsistency.

F Language Models on English TEE

In our early experiments, we reexamine the language models on English TEE. This section presents the results.

F.1 Experimental Setup

We study BERT (Devlin et al., 2019) and XLMR (Conneau et al., 2020) variants, RoBERTa (Liu et al., 2019b) and T5 Encoder (Raffel et al., 2019). We compare them to rule-based methods including HeidelTime (Strötgen and Gertz, 2013), SynTime (Zhong et al., 2017), and PTime (Ding et al., 2019), which report SOTA performances on Wikiwars, TE3, and Tweets, respectively. We experiment on both settings, i.e., "with type" and "without type", and report F1, precision, and recall in strict match (UzZaman et al., 2013). We use the data splits following Ding et al. (2019) and the experimental settings introduced in Section 4.1.

F.2 Evaluation Results

Tables 10, 11, and 12 show the results. We observe: **1)** When ignoring the types, the language models are inferior to SynTime on TE3, on par with or better than the rule-based methods on Wikiwars and Tweets. **2)** When considering the types, the language models outperform the previous SOTAs by 11-22%, 18-21%, and 30-41% in F1 on TE3, Wikiwars, and Tweets datasets.

Table 7: Multilingual TEE results (w/ type | w/o type).

w/ type	FR			ES			PT			EU		
	F1	Pr.	Re.	F1	Pr.	Re.	F1	Pr.	Re.	F1	Pr.	Re.
Automatic Baseline Models												
HeidelTime-auto	0.53	0.63	0.46	0.41	0.56	0.32	0.49	0.66	0.39	0.15	0.60	0.09
BiLSTM+CRF	0.58	0.64	0.51	0.56	0.61	0.51	0.58	0.59	0.58	0.44	0.54	0.37
mBERT	0.56	0.61	0.51	0.56	0.62	0.51	0.60	0.56	0.64	0.59	0.64	0.55
XLMR-base	0.64	0.69	0.59	0.51	0.58	0.46	0.59	0.59	0.59	0.43	0.60	0.34
XLMR-large	0.69	0.70	0.68	*0.68*	*0.71*	**0.66**	0.71	0.69	0.73	0.66	0.70	0.63
Projection Method												
TMP-mBERT	0.50	0.56	0.45	0.23	0.59	0.14	0.60	0.57	0.64	/	/	/
TMP-XLMRbase	0.50	0.60	0.43	0.23	0.57	0.14	0.61	0.58	0.64	/	/	/
TMP-XLMRlarge	0.52	0.61	0.46	0.24	0.59	0.15	0.61	0.58	0.63	/	/	/
Transfer from EN (Ours)												
XLTime-mBERT	0.62	0.62	0.62	0.65	0.70	0.61	0.61	0.58	0.66	0.68	0.72	0.65
XLTime-XLMRbase	0.67	0.67	0.68	0.60	0.63	0.58	0.64	0.62	0.66	0.64	0.68	0.60
XLTime-XLMRlarge	*0.71*	**0.74**	0.68	**0.70**	**0.76**	*0.65*	*0.74*	*0.71*	*0.78*	*0.72*	**0.79**	*0.66*
Transfer from EN and others (Ours)												
XLTime-mBERT	*0.71*	0.69	0.73	*0.68*	0.69	**0.66**	0.73	0.70	0.76	0.68	0.72	0.65
XLTime-XLMRbase	0.70	0.67	*0.74*	0.65	0.69	0.62	0.66	0.64	0.68	0.70	*0.76*	0.65
XLTime-XLMRlarge	**0.75**	*0.72*	**0.78**	**0.70**	**0.76**	*0.65*	**0.81**	**0.79**	**0.84**	**0.74**	**0.79**	**0.69**
Handcrafted Method												
HeidelTime	0.80	0.81	0.79	0.85	0.90	0.80	0.57	0.60	0.53	/	/	/

w/o type	FR			ES			PT			EU		
	F1	Pr.	Re.	F1	Pr.	Re.	F1	Pr.	Re.	F1	Pr.	Re.
Automatic Baseline Models												
HeidelTime-auto	0.55	0.65	0.47	0.42	0.58	0.33	0.50	0.67	0.39	0.17	0.66	0.10
BiLSTM+CRF	0.64	0.73	0.57	0.62	0.68	0.56	0.64	0.66	0.63	0.47	0.58	0.40
mBERT	0.63	0.70	0.58	0.62	0.69	0.56	0.66	0.63	0.69	0.65	0.71	0.60
XLMR-base	0.69	0.75	0.64	0.54	0.61	0.48	0.63	0.64	0.62	0.46	0.64	0.36
XLMR-large	0.75	0.78	0.73	0.72	0.75	0.69	0.75	0.74	0.76	0.70	0.74	0.67
Projection Method												
TMP-mBERT	0.56	0.63	0.50	0.23	0.59	0.14	0.66	0.64	0.69	/	/	/
TMP-XLMRbase	0.55	0.67	0.47	0.23	0.57	0.14	0.64	0.61	0.67	/	/	/
TMP-XLMRlarge	0.56	0.66	0.50	0.24	0.59	0.15	0.65	0.61	0.68	/	/	/
Transfer from EN (Ours)												
XLTime-mBERT	0.73	0.73	0.72	0.71	0.77	0.66	0.67	0.64	0.71	0.76	0.81	0.71
XLTime-XLMRbase	0.78	*0.79*	0.78	0.66	0.70	0.63	0.68	0.67	0.70	0.71	0.76	0.66
XLTime-XLMRlarge	0.76	*0.79*	0.73	0.72	**0.79**	0.67	0.77	0.74	0.81	*0.78*	*0.85*	0.71
Transfer from EN and others (Ours)												
XLTime-mBERT	0.80	0.77	*0.82*	**0.77**	**0.79**	**0.74**	*0.80*	*0.77*	*0.83*	0.77	0.82	0.72
XLTime-XLMRbase	*0.82*	*0.79*	**0.86**	0.72	*0.78*	0.68	0.73	0.72	0.75	**0.79**	**0.86**	*0.73*
XLTime-XLMRlarge	**0.84**	**0.82**	**0.86**	*0.75*	**0.79**	*0.71*	**0.84**	**0.82**	**0.87**	**0.79**	0.84	**0.74**
Handcrafted Method												
HeidelTime	0.86	0.87	0.85	0.86	0.91	0.81	0.60	0.64	0.57	/	/	/

Table 8: Low-resource language TEE with additional source languages (F1, precision, and recall scores w/ type). The blue cells are expected to, while the underlined cells actually outperform (by $\geq 3\%$) using EN as the only source language.

F1								
Target Language	FR				ES			
Source Language(s)	EN	EN, EU	EN, PT	EN, ES	EN	EN, EU	EN, PT	EN, FR
XLTime-mBERT	0.62	0.61	0.61	0.71	0.65	0.66	0.65	0.68
XLTime-XLMRbase	0.67	0.67	0.66	0.70	0.60	0.61	0.64	0.65
XLTime-XLMRlarge	0.71	0.73	0.73	0.75	0.70	0.68	0.69	0.68
Target Language	PT				EU			
Source Language(s)	EN	EN, FR	EN, ES	EN, EU	EN	EN, PT	EN, ES	EN, FR
XLTime-mBERT	0.61	0.72	0.59	0.73	0.68	0.66	0.66	0.68
XLTime-XLMRbase	0.64	0.66	0.55	0.52	0.64	0.66	0.66	0.70
XLTime-XLMRlarge	0.74	0.79	0.81	0.71	0.72	0.71	0.74	0.72
Precision								
Target Language	FR				ES			
Source Language(s)	EN	EN, EU	EN, PT	EN, ES	EN	EN, EU	EN, PT	EN, FR
XLTime-mBERT	0.62	0.59	0.62	0.69	0.70	0.69	0.71	0.69
XLTime-XLMRbase	0.67	0.66	0.67	0.67	0.63	0.64	0.67	0.69
XLTime-XLMRlarge	0.74	0.72	0.76	0.72	0.76	0.65	0.73	0.68
Target Language	PT				EU			
Source Language(s)	EN	EN, FR	EN, ES	EN, EU	EN	EN, PT	EN, ES	EN, FR
XLTime-mBERT	0.58	0.68	0.56	0.70	0.72	0.70	0.69	0.72
XLTime-XLMRbase	0.62	0.64	0.51	0.49	0.68	0.73	0.69	0.76
XLTime-XLMRlarge	0.71	0.75	0.79	0.68	0.79	0.75	0.79	0.79
Recall								
Target Language	FR				ES			
Source Language(s)	EN	EN, EU	EN, PT	EN, ES	EN	EN, EU	EN, PT	EN, FR
XLTime-mBERT	0.62	0.62	0.59	0.73	0.61	0.64	0.60	0.66
XLTime-XLMRbase	0.68	0.67	0.64	0.74	0.58	0.59	0.61	0.62
XLTime-XLMRlarge	0.68	0.73	0.71	0.78	0.65	0.71	0.65	0.67
Target Language	PT				EU			
Source Language(s)	EN	EN, FR	EN, ES	EN, EU	EN	EN, PT	EN, ES	EN, FR
XLTime-mBERT	0.66	0.75	0.62	0.76	0.65	0.63	0.64	0.64
XLTime-XLMRbase	0.66	0.68	0.60	0.55	0.60	0.60	0.63	0.65
XLTime-XLMRlarge	0.78	0.83	0.84	0.74	0.66	0.67	0.69	0.67

Table 9: Low-resource language TEE with additional source languages (precision and recall scores w/o type). The blue cells are expected to, while the underlined cells actually outperform (by $\geq 4\%$) using EN as the only source language.

Precision								
Target Language	FR				ES			
Source Language(s)	EN	EN, EU	EN, PT	EN, ES	EN	EN, EU	EN, PT	EN, FR
XLTime-mBERT	0.73	0.76	0.76	0.77	0.77	0.76	0.79	0.79
XLTime-XLMRbase	0.79	0.77	0.81	0.79	0.70	0.72	0.75	0.78
XLTime-XLMRlarge	0.79	0.81	0.84	0.82	0.79	0.70	0.79	0.74
Target Language	PT				EU			
Source Language(s)	EN	EN, FR	EN, ES	EN, EU	EN	EN, PT	EN, ES	EN, FR
XLTime-mBERT	0.64	0.77	0.67	0.77	0.81	0.78	0.79	0.82
XLTime-XLMRbase	0.67	0.72	0.60	0.54	0.76	0.82	0.79	0.86
XLTime-XLMRlarge	0.74	0.79	0.82	0.72	0.85	0.85	0.84	0.84
Recall								
Target Language	FR				ES			
Source Language(s)	EN	EN, EU	EN, PT	EN, ES	EN	EN, EU	EN, PT	EN, FR
XLTime-mBERT	0.72	0.77	0.69	0.82	0.66	0.69	0.66	0.74
XLTime-XLMRbase	0.78	0.76	0.75	0.86	0.63	0.64	0.68	0.68
XLTime-XLMRlarge	0.73	0.81	0.77	0.86	0.67	0.75	0.71	0.72
Target Language	PT				EU			
Source Language(s)	EN	EN, FR	EN, ES	EN, EU	EN	EN, PT	EN, ES	EN, FR
XLTime-mBERT	0.71	0.83	0.74	0.83	0.71	0.69	0.70	0.72
XLTime-XLMRbase	0.70	0.75	0.66	0.59	0.66	0.67	0.70	0.73
XLTime-XLMRlarge	0.81	0.87	0.87	0.77	0.71	0.74	0.74	0.71

Table 10: Supervised English TEE on TE3 (w/ type | w/o type).

	F1	Pr.	Re.
Rule-based Models			
HeidelTime	0.77| 0.81	*0.80*| 0.84	0.75| 0.79
SynTime	0.65| **0.92**	0.65| **0.91**	0.66| **0.93**
PTime	0.67| *0.85*	0.68| *0.88*	0.65| 0.83
Language Models			
BERT-base	0.76| 0.82	0.78| 0.85	0.74| 0.80
BERT-large	**0.79**| 0.83	0.77| 0.82	**0.80**| *0.84*
mBERT	**0.79**| 0.84	*0.80*| 0.86	0.77| 0.82
RoBERTa	*0.78*| 0.84	0.79| 0.86	0.77| 0.82
XLMR-base	**0.79**| 0.81	*0.80*| 0.82	0.77| 0.81
XLMR-large	*0.78*| 0.81	0.78| 0.82	*0.78*| 0.81
T5Encoder	**0.79**| 0.82	**0.82**| 0.85	*0.78*| 0.80

Table 11: Supervised English TEE on Wikiwars (w/ type | w/o type).

	F1	Pr.	Re.
Rule-based Models			
HeidelTime	0.80| 0.85	0.86| 0.92	0.75| 0.80
SynTime	0.79| 0.79	0.79| 0.79	0.79| 0.79
PTime	0.86| 0.86	0.87| 0.87	0.86| 0.86
Language Models			
BERT-base	0.94| 0.94	*0.95*| *0.95*	0.94| 0.94
BERT-large	0.95| 0.95	0.94| 0.94	0.96| 0.96
mBERT	**0.97**| **0.97**	**0.96**| **0.96**	*0.97*| *0.97*
RoBERTa	0.95| 0.95	0.94| 0.94	*0.97*| *0.97*
XLMR-base	**0.97**| **0.97**	*0.95*| *0.95*	**0.98**| **0.98**
XLMR-large	*0.96*| *0.96*	0.94| 0.94	*0.97*| *0.97*
T5Encoder	*0.96*| *0.96*	*0.95*| *0.95*	*0.97*| *0.97*

Table 12: Supervised English TEE on Tweets (w/ type | w/o type).

	F1	Pr.	Re.
Rule-based Models			
HeidelTime	0.80| 0.80	*0.90*| 0.90	0.72| 0.72
SynTime	0.63| 0.92	0.62| 0.91	0.65| 0.95
PTime	0.66| **0.95**	0.65| **0.94**	0.67| *0.96*
Language Models			
BERT-base	*0.92*| *0.94*	*0.90*| *0.93*	0.93| 0.95
BERT-large	0.86| 0.92	0.84| 0.92	0.88| 0.92
mBERT	0.87| 0.91	0.85| 0.88	0.90| 0.94
RoBERTa	0.91| **0.95**	0.89| *0.93*	*0.94*| **0.97**
XLMR-base	0.90| *0.94*	0.87| 0.92	0.93| **0.97**
XLMR-large	**0.93**| **0.95**	**0.91**| *0.93*	**0.95**| *0.96*
T5Encoder	0.87| 0.93	0.84| 0.91	0.91| 0.95