ISSN:2790-167X Volume-5-(2023)

Pairwise Cross-Document Event Coreference Resolution with Contrastive Learning

Jie Li [1, a], Yanchang Lu [2, b] and Zhenxuan Liang [2, c]

[1] Faculty of Science and Technology, BNU-HKBU United International College, Zhuhai, China;

[2] School of Computer Science and Engineering, University of New South Wales,

Sydney, NSW 2052, Australia.

[a] lijie201718@163.com, [b] luyanchang0125@gmail.com, [c] zhenxuanliang1999@gmail.com

Abstract. Cross-Document (CD) Event Coreference Resolution (ECR) is a fundamental task in Natural Language processing (NLP) and Knowledge Base Population (KBP). Resolving the event coreference relationship is a challenging task, which necessitates a thorough semantic comprehension of the events. Existing methods have spontaneously formulated this problem as a binary classification task based on sentence segments. However, the information distributed in the longer context is ignored by most prior works. Recent event coreference works focus on adding auxiliary information to improve the performance. The information such as involved entity information, structural information or database queried information is extracted from related documents as long context. Whereas extraction error, matching error and other types of error are introduced in the process of obtaining related information when the context is incomplete. In this work, we present a Contrastive Learning based Pairwise Event Coreference (CLPEC) framework to complete and optimize the CD-ECR task utilizing contextual information with contrastive learning technology. We also adopt augmentation technology to preprocess event representations and improve our performance. Experiment results show that we achieve competitive results on a number of key metrics on the ECB+ corpus.

Keywords: Event Coreference; Contrastive Learning; Data Augmentation.

1. Introduction

The Cross-Document (CD) event coreference task is to find event mentions that refer to the same real-world event in any document context. Growing attention moves to event coreference resolution (ECR) for its applications in Knowledge Base Population (KBP), Natural Language Processing (NLP) and other related fields. ECR is an important step for higher-level tasks, e.g. text summarization [1], information extraction [2], question answering [3], etc.

Due to semantic diversity, even with the same words, event mentions may represent different real-world occurrence in different contexts. Therefore, most prior works address the importance of embedding types and components. Word embeddings for event mentions can be mainly divided into three types: character embedding, static embedding, and contextual embedding [4, 5]. Recent works represent events in four main components: action component which describes what happens (e.g. killing, earthquake), time slot component which records when the event exists, location component which records where the event took place, and participant component describes who or what is involved in (e.g. students, China, pandas) [6]. One potential challenge is the error involved in when extracting these components from contexts. The other question is the potential scale of context for obtaining related information. Mainstream researches tend to reduce the scale of event context into only one sentence where the mention is in [7]. However, in certain real scenarios, event information could be distributed in long textual description even in different corpus. And event related documents increase in both size and length nowadays [8]. Recent works make attempts to involve the whole document as the context [9, 10]. Therefore, how to utilize the contextual information in the long descriptive text effectively is a crucial point in CD event coreference task.

Majority previous researches in CD event coreference resolution formulate the problem in a pairwise Event Coreference (PEC) framework which works as follows: given pairs of mentions, a

cross-encoder to obtain their embeddings, a scorer to output a similarity score and classify the pairs as referring to the same event or not, then mentions referring to the same event are clustering according to certain rules [7, 11, 12]. To address the aforementioned problems, we propose a novel framework Contrastive Learning-based PEC (CLPEC) framework, which consists of three core components. Firstly, in order to involve document context, we propose to apply Longformer [13] as the encoder to learn document-level textual information; Moreover, we design the contrastive learning-based model CLPEC to complete ECR task which introduces contrastive learning process to learn crucial information and features of events and differentiate them. Besides, we apply data augmentation technologies to the original event data and increase the number of coreference pairs.

The main contributions of our work are summarized as follows:

- Utilize Longformer as the encoder to capture document-level textual information.
- Develop the CLPEC framework which performs contrastive learning technology.
- Apply data augmentation skills to balance event data.

Our model has optimized result in related researches and achieves competitive performance to state-of-the-art methods.

2. Related Work

2.1 Event Representation

Generally, event is defined as something that happens to a state or participants, and modeled into four components (action, time, location, human/non-human participant) [6]. The action component is usually extracted as "event mention" to denote different event instances. Lexical features like head lemma and surrounding words (context), class features including type information and topic information, Wordnet features from the database, have been utilized to generalize event representations in existing event coreference research [14]. Word-level, sentence-level, document-level and topic-level features all have been utilized as event representations [10]. Due to the complicated information contained in event mentions, above features are typically grouped to produce event representations. Recent works also extract and introduce auxiliary information to supplement event representation. Component information is extracted and integrated into event representations using textual feature embedding [15], semantic role labelling (SRL) mapped arguments [12] or joint training process [7]. However, not all event mentions have textual context containing complete structural information of all the components in a given context [16]. Extraction and matching processes for aforementioned event features are certain to introduce errors. In contrast, our study attempts to avoid these processes and utilize the document context to capture the implicated information, which is unstructured to have a robust model performance.

2.2 Event Coreference Resolution

Event Coreference Resolution (ECR) is defined as the task to determine whether two event mentions refer to the same occurrence in real world. Recent event coreference approaches convey the problem as a pairwise binary classification task and follow the PEC framework [7, 11, 12]. Transformer-based models like BERT [17], SpanBERT [18], RoBERTa [19] and Longformer [13] have been applied as effective encoders to learn event information instead of traditional neural networks. A pairwise scorer usually follows the encoder and generates a similarity score for each pair of events. Most existing works build the scorer on neural networks [15]. In addition to common MLP scorer with Binary Cross Entropy (BCE) loss, Siamese network with circle loss is applied and achieves fair performance [20]. Clustering-oriented regularization terms are also used in the loss function in the training process [21]. For the clustering stage, most of previous studies apply agglomerative clustering to compute event clusters.

2.3 Data Augumentation in NLP

In real corpus (e.g. ECB+, KBP.), the amount of texts describing different events is significantly greater than that describing the same event. To obtain more positive instances and improve the robustness of representations, augmentation method is introduced in our research. Data augmentation methods could be mainly divided into three categories: sampling-based, noising-based and paraphrasing-based method in NLP fields [22]. The sampling-based methods learn and model the feature of representations and capture the distributions and different dimensions of information in texts. The paraphrasing-based methods generate diverse representations with same or similar information for the instance. The noising-based methods add feasible noise (words, sentences, etc.) to original representations. Recent text augmentation researches focus on adding continuous noise on the embeddings and have made great progress, such as ConSERT [23] and SimCSE [24].

3. Methodology

3.1 Model

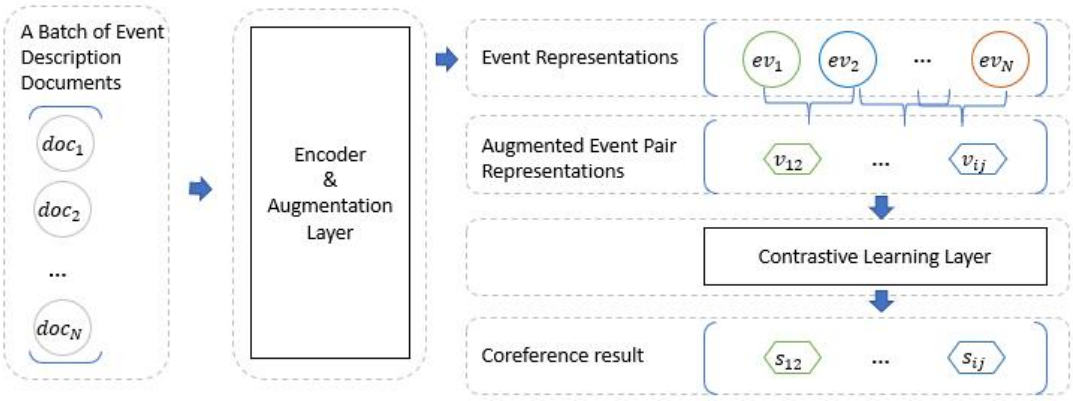

Fig. 1 Model Framework Diagram

We develop our model following the typical end-to-end structure [11] and the framework is shown in Fig. 1. Description documents are put into encoders to get embeddings for each event mention. Then the embeddings are paired to get the representation of the corresponding event mention pairs. In these pairs, positive event pair represents coreferent mentions and negative event pair includes two events which are not coreferential. The positive pair is augmented by putting the event mentions into the encoder twice and constructing more pairs. Put these event pair representations into the Contrastive Learning Layer, then output the similarity score for the input event pairs. At last, agglomerative clustering method is applied to put possible coreferential event mentions with similarity score higher than certain threshold into the same cluster.

We augment positive pairs before the training process. Our augmentation strategy is illustrated in Fig. 2. Referring to SimCSE [24] who augments sentences, our augmentation target is the description document. We put tokenized document sequence of two event mentions in positive pairs into encoder twice to naturally add dropout as noising-based method and augment documents. Each coreferent event pair are augmented into 4 pieces. Notice that we only do this augmentation operation in training stage and only on positive event pairs.

We train our model using augmented positive mention pairs and original negative pairs as the input. At inference stage, our model generates prediction and similarity scores for the mention pairs. And at clustering stage, mentions are clustering with a similarity threshold using agglomerative clustering algorithm, as was done previously [7, 12].

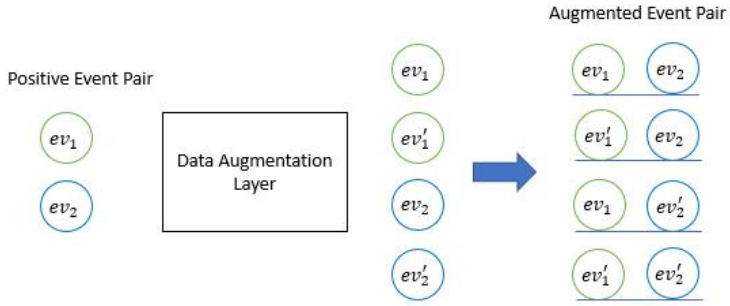

Fig. 2 Encoder Augmentation

In most cases, words and sentences closer to the event mention head are more relevant to its content. Following Cross-Document Language Modeling (CDLM) [25], we use this Longformer-based encoder to learn the information of the document context for represent the event mentions. We put importance on the location the mention appears in and annotate the corresponding location in the sentence using [E] and [/E] tokens. And the context sequence for every event mention are surrounded by [doc-s] and [/doc-s] tokens. We truncate and encode a certain length (L) tokens around the mention as the context and input the context sequence into the encoder. The encoder will produce the original representation vector v_r for each token of the input sequence e.

$$v_r = Longformer(e) \#(1)$$

The mention vector is also represented as the concatenation of four components as previous relevant experiments, which can be divided into two parts. One part consists of the first and the last contextualized representation of event mentions and the weighted sum of token vectors in the selected span using head-finding attention mechanism following previous systems [11].

$$\begin{aligned} \alpha_t &= \boldsymbol{w}_\alpha \cdot (\boldsymbol{v}_r) \\ a_{i,t} &= \frac{\exp(\alpha_t)}{\sum_{k=\text{START}(i)}^{\text{END}(i)} \exp(\alpha_k)} \#(2) \\ \hat{\boldsymbol{v}}_i &= \sum_{t=\text{START}(i)}^{\text{END}(i)} a_{i,t} \cdot \boldsymbol{v}_r \end{aligned}$$

We add CLS embedding v_{cls} as the other part of mention embedding. It is introduced as the information for the entire document context. We concat these components together as the whole representation for event mentions, indicated by v_i. And the final representation v_{ij} for the mention i and j is given by the concatenation of the two vectors and their product:

$$v_i = [v_{cls}; v_{START}; v_{END}; \hat{\boldsymbol{v}}_i] \#(3)$$

$$v_{ij} = [v_i; v_j; v_i * v_j] \#(4)$$

Each of these paired representations will then pass through a contrastive learning based multilayer perceptron (MLP) and a Softmax layer to get the final similarity score s_{ij}.

$$s_{ij} = Softmax\left(MLP(v_{ij})\right) \#(5)$$

In order to compare our language model with earlier approaches, we follow earlier works and use agglomerative clustering over these scores s_{ij} to find coreference event clusters. We use $d_{ij} = 1 - s_{ij}$ as the precomputed distance and cluster mention representations according to this metric. Representations within an average threshold distance are considered to be in the same cluster.

3.2 Training

To train the model, we construct pairs of event mentions: positive pairs indicate the mentions are coreferential while negative pairs indicate the mentions are not coreferential within gold topics. Negative pairs were chosen from within gold topics and were constructed by coreference label

checking. While positive pairs were from both coreference and augmentation. For a given pair of event context documents, gold label $y = 1\ or\ 0$ indicates that $y = 1$ if the instances are coreferent and $y = 0$ otherwise. Each pair is encoded using our trained model and get a similarity score s_i. Distance of event pairs is denoted as d_i, which is subtracted s_i from 1.

To maximum the difference of different events and minimize the difference of coreferent events, we build a contrastive learning process and train the model with the weighted $(1{:}\,\lambda)$ sum of general Binary Cross Entropy (BCE) loss ℓ_b and Contrastive loss ℓ_c [26], where m is the margin. This combined loss ℓ significantly pushes our positive pairs closer and negative pairs more distant.

$$\ell = \ell_b + \lambda \ell_c \#(6)$$

$$\ell_b = -\frac{1}{N}\sum_{i=1}^{N} \left[y_i \log(s_i) + (1 - y_i)\log(1 - s_i)\right] \#(7)$$

$$\ell_c = \sum_{i=1}^{N} \left[y_i * (d_i)^2 + (1 - y_i) * max(m - (d_i)), 0)^2\right] \#(8)$$

4. Result and Discussion

4.1 Dataset

We follow recent works and use the ECB+ dataset [6] as the benchmark dataset, the detailed dataset statistics shown in Table 1. Singletons are included in event cluster counting.

Table 1. ECB+ Statistics

	Train Set	Dev Set	Test Set
Topics	25	8	10
Documents	574	196	206
Event Mentions	3808	1245	1780
Event Clusters	1527	409	805

As shown in Table 2, only considering event mention pairs within same topic, original negative pair number significantly exceeds the number of positive pairs. This unbalanced data structure has a bad effect on learning event coreferential relations. Therefore, the preprocessing process (i.e., up sampling of positive pairs and down sampling of negative pairs) is necessary.

Table 2. Within-Topic Event Pair Statistics (In ECB+ Dataset)

	Positive Pair Number	Negative Pair Number	Total Number
Train Set	14944	170549	185493
Validation Set	5881	50653	56534
Test Set	6889	87053	93942

4.2 Experiment Settings

In encoding stage, we keep 200 tokens as the document context in front and after the event mention, respectively. The embedding dimension of every token is set as 768. In the training process, the Adam optimizer is used. We choose the learning rate as 10^{-5}, batch size as 128, maximum iteration time for model as 50, margin of contrastive loss as 1.6 and ratio for contrastive loss and BCE loss as 9:1. Fixed random seed is given for each experiment. In agglomerative clustering stage, the optimum clustering threshold is setting by grid search in every individual experiment.

4.3 Results

We use the commonly adopted measures for model evaluation: MUC [27], B3 [28], CEAF-e [29] and CoNLL (the average of the MUC, B3, and CEAF-e F1 scores) [30]. For comparation, we report the results of CD models with similar topic or structure within topic as baseline models in Table 3.

We compare our model performance with following models: *Lemma Baseline* represents basic event clustering work within the same document, which share the same head lemma; *CV* (*Cybulska and Vossen, 2015a*) [6] encodes events with participants, time and location related information; *KCP (Kenyon-Dean et al., 2018*) [21] encodes the document as part of the mention representation for event coreference resolution; *DisJoint (Barhom et al. ,2019*) [7] represent the result of its variant model using only the span and context vectors as event pair representations.

Table 3. Cross-Document Coreference Performance on ECB+ (F1)

	MUC	B3	CEAF-e	CONLL
Lemma Baseline	78.1	77.8	73.6	76.5
CV (Cybulska and Vossen, 2015a)	73.0	74.0	64.0	73.0
KCP (Kenyon-Dean et al., 2018)	69.0	69.0	69.0	69.0
DisJoint (Barhom et al. ,2019)	79.4	80.4	75.9	78.5
Our Model	81.0	81.2	77.4	79.8

4.4 Ablations

To validate the effectiveness of different components, i.e., augmentation strategy, event representation and contrastive learning process in our proposed framework, we conduct the following ablation study. Specifically, we removed our CLS embedding which represents the whole document context (denoted as *cls embedding*) and the contrastive loss to test the effect of contrastive learning module (denoted as *contrastive loss*). The ablation experiments for augmentation are done in two sections. One is to remove positive pair augmentations for positive event pairs (denoted as *positive pair augmentation*) and the other is to add augmentations for negative event pairs (denoted as *negative pair augmentation*). The results are reported in Table 4, where "-" means removing the content from the whole framework and "+" means adding the following module to the original framework.

Table 4. Ablation Experiments (F1)

	MUC	B3	CEAF-e	CONLL
Our Model	80.95	81.15	77.40	79.83
- positive pair augmentation	79.91	80.19	76.28	78.79
+negative pair augmentation	80.12	79.89	76.28	78.76
- cls embedding	79.62	79.89	75.60	78.37
- contrastive loss	76.52	75.91	73.9	75.44

To assess the effect of our augmenting strategy, we trained our model with original event pairs in dataset and augmenting all pairs (including negative pairs). Experiment results going down 0.94 and 1.07 percent, respectively, which proves the positive effect of our augmentation strategies. The effect of document information is examined by ablating CLS embedding. We found that the performance became worse (-1.46 percent). Therefore, the CLS embedding makes contribution to event representations. As for the training loss, we compare the performance of current model with the model training with only BCE loss. We can see a biggest drop of the event performance (-4.39 percent) in all the ablation tests. The combined loss type reaches the best performance means that the model learns more features about events, which is inline of our research.

5. Conclusion

In this paper, we propose a CLPCR framework for cross-document ECR task. We use augmentation technology to preprocess event data, deploy contrastive learning to distinguish events and utilize document-level textual information as context for representing events. We evaluate our model on the topic level of the ECB+ corpus with golden topics and find that our approach gets comparative result with state-of-the-art methods. The records and results of our ablation experiments methods. We demonstrate that contrastive learning approaches are effective at learning

significant features of event mentions and differentiating them. Besides, data augmentation technologies and document-level information are significant for optimizing event representation and event coreference resolutions.

References

[1] Xiuying Chen, Zhangming Chan, Shen Gao, Meng-Hsuan Yu, Dongyan Zhao, and Rui Yan. Learning towards abstractive timeline summarization. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, pages 4939–4945. International Joint Conferences on Artificial Intelligence Organization, 2019.

[2] He Zhao, Longtao Huang, Rong Zhang, Quan Lu, and Hui Xue. Span-Mlt: A span-based multi-task learning framework for pair-wise aspect and opinion terms extraction. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3239–3248, Online, 2020.

[3] Jiexin Wang, Adam Jatowt, Michael Färber, and Masatoshi Yoshikawa. Improving question answering for event-focused questions in temporal collections of news articles. Inf. Retr. 24, 1 (Feb 2021), 29–54.

[4] Felipe Almeida and Geraldo Xexéo. Word embeddings: A survey. arXiv:1901.09069, 2019.

[5] Qi Liu, Matt J. Kusner, and Phil Blunsom. A survey on contextual embeddings. ArXiv, abs/2003.07278, 2020.

[6] Agata Cybulska and Piek Vossen. Using a Sledgehammer to Crack a Nut? Lexical Diversity and Event Coreference Resolution. In: Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14). Reykjavik, Iceland: European Language Resources Association (ELRA), 2014.

[7] Shany Barhom, Vered Shwartz, Alon Eirew, Michael Bugert, Nils Reimers, and Ido Dagan. Revisiting Joint Modeling of Cross-document Entity and Event Coreference Resolution. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 4179–4189, Florence, Italy, 2019.

[8] Michael Bugert, Nils Reimers, Iryna Gurevych: Generalizing Cross-Document Event Coreference Resolution Across Multiple Corpora. Comput. Linguistics 2021, 47(3): 575-614.

[9] Arie Cattan, Alon Eirew, Gabriel Stanovsky, Mandar Joshi, Ido Dagan: Streamlining Cross-Document Coreference Resolution: Evaluation and Modeling. CoRR abs/2009.11032, 2020.

[10] Sheng Xu, Peifeng Li, and Qiaoming Zhu. Improving Event Coreference Resolution Using Document-level and Topic-level Information. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 6765–6775, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics, 2022.

[11] Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 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, 2017.

[12] Xiaodong Yu, Wenpeng Yin, and Dan Roth. Pairwise Representation Learning for Event Coreference. In Proceedings of the 11th Joint Conference on Lexical and Computational Semantics, pages 69–78, Seattle, Washington. Association for Computational Linguistics, 2022.

[13] Iz Beltagy, Matthew E. Peters, Arman Cohan: Longformer: The Long-Document Transformer. CoRR abs/2004.05150, 2020.

[14] Cosmin Bejan and Sanda Harabagiu. Unsupervised Event Coreference Resolution with Rich Linguistic Features. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 1412–1422, Uppsala, Sweden. Association for Computational Linguistics, 2010.

[15] Prafulla Kumar Choubey and Ruihong Huang. Event coreference resolution by iteratively unfolding inter-dependencies among events. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2124–2133, Copenhagen, Denmark. Association for Computational Linguistics, 2017.

[16] Alon Eirew, Arie Cattan, and Ido Dagan. WEC: Deriving a Large-scale Cross-document Event Coreference dataset from Wikipedia. In Proceedings of the 2021 Conference of the North American

Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 2498–2510, Online. Association for Computational Linguistics, 2021.

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics. 2019.

[18] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. SpanBERT: Improving Pre-training by Representing and Predicting Spans. Transactions of the Association for Computational Linguistics, 8:64–77. 2020.

[19] Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A Robustly Optimized BERT Pre-training Approach with Post-training. In Proceedings of the 20th Chinese National Conference on Computational Linguistics, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China. 2021.

[20] Beiya Dai, Jiangang Qian, Shiqing Cheng, Linbo Qiao, Dongsheng Li: Event Coreference Resolution based on Convolutional Siamese network and Circle Loss. 2022 International Joint Conference on Neural Networks (IJCNN), Padua, Italy, pages 1–7, 2022.

[21] Kian Kenyon-Dean, Jackie Chi Kit Cheung, and Doina Precup. Resolving event coreference with supervised representation learning and clusteringoriented regularization. In Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics, pages 1–10, New Orleans, Louisiana. Association for Computational Linguistics. 2018.

[22] Bohan Li, Yutai Hou, Wanxiang Che: Data augmentation approaches in natural language processing: A survey. AI Open 3: 71-90, 2022.

[23] Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, Weiran Xu: ConSERT: A Contrastive Framework for Self-Supervised Sentence Representation Transfer. ACL/IJCNLP (1) 2021: 5065-5075

[24] Tianyu Gao, Xingcheng Yao, Danqi Chen: SimCSE: Simple Contrastive Learning of Sentence Embeddings. EMNLP (1) 2021: 6894-6910

[25] Avi Caciularu, Arman Cohan, Iz Beltagy, Matthew Peters, Arie Cattan and Ido Dagan. CDLM: Cross-document language modeling. In Findings of the Association for Computational Linguistics: EMNLP 2021, pages 2648–2662, Punta Cana, Dominican Republic. 2021.

[26] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), volume 2, pages 1735–1742. 2006.

[27] Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. A modeltheoretic coreference scoring scheme. In Proceedings of the 6th Conference on Message Understanding, MUC6 '95, page 45–52, USA. Association for Computational Linguistics, 1995.

[28] Bagga, Amit and Breck Baldwin. Algorithms for scoring coreference chains. In The first international conference on language resources and evaluation workshop on linguistics coreference, volume 1, pages 563–566, 1998.

[29] Xiaoqiang Luo. 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, 2005.

[30] Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. CoNLL2012 shared task: Modeling multilingual unrestricted coreference in OntoNotes. In Joint Conference on EMNLP and CoNLL - Shared Task, pages 1–40, Jeju Island, Korea. Association for Computational Linguistics, 2012.