Discourse Information for Document-Level Temporal Dependency Parsing

Jingcheng Niu[123], **Victoria Ng**[2], **Erin E. Rees**[2], **Simon de Montigny**[24], **Gerald Penn**[13]
{niu,gpenn}@cs.toronto.edu {victoria.ng,erin.rees}@phac-aspc.gc.ca
simon.de.montigny@umontreal.ca
University of Toronto[1], Public Health Agency of Canada[2], Vector Institute[3], University of Montreal[4]

Abstract

In this study, we examine the benefits of incorporating discourse information into document-level temporal dependency parsing. Specifically, we evaluate the effectiveness of integrating both high-level discourse profiling information, which describes the discourse function of sentences, and surface-level sentence position information into temporal dependency graph (TDG) parsing. Unexpectedly, our results suggest that simple sentence position information, particularly when encoded using our novel sentence-position embedding method, performs the best, perhaps because it does not rely on noisy model-generated feature inputs. Our proposed system surpasses the current state-of-the-art TDG parsing systems in performance.

Furthermore, we aim to broaden the discussion on the relationship between temporal dependency parsing and discourse analysis, given the substantial similarities shared between the two tasks. We argue that discourse analysis results should not be merely regarded as an additional input feature for temporal dependency parsing. Instead, adopting advanced discourse analysis techniques and research insights can lead to more effective and comprehensive approaches to temporal information extraction tasks.

1 Introduction

Temporal Information Extraction (TIE) is the task of modelling the relative and/or absolute temporal relations between all the temporal nodes in an article. A temporal node can be either an event or a time expression (timex). TIE is a core component task of text comprehension. Despite its importance, TIE remains one of the lowest performing natural language understanding tasks. It is a difficult task, and the challenge is further compounded when expanding it to the document level, as the number of temporal relations scales quadratically with the number of temporal nodes, and the requisite amount of reasoning must incorporate longer spans of text.

To address these challenges, Kolomiyets et al. (2012); Zhang and Xue (2018b); Yao et al. (2020) proposed the use of temporal dependency structures to represent the overall temporal relational structure within an article. This approach is based on the phenomenon of *temporal anaphora*, where the interpretation of the occurring time of one temporal node depends on knowing the occurring time of another temporal node. By modelling these temporal dependency relations, the overall temporal structure of an article can be obtained without the need for exhaustively labelling every pair of temporal nodes.

As a result, temporal dependency parsing not only models the temporal relations between events but also captures narrative and discourse structure. There are striking similarities between temporal dependency structures and the constituency discourse tree structures (Guz and Carenini, 2020) used for discourse parsing in the context of Rhetorical Structure Theory (RST; Mann and Thompson, 1988), and not just in their use of trees or graphs. More importantly, temporal dependency relations can be viewed as a specific type of anaphoric relation that discourse analysis models attempt to capture. This observation suggests a potential connection between dependency parsing and discourse analysis, warranting further investigation into their relationship and potential synergies.

This connection between document-level temporal structure and discourse structure was corroborated by Choubey and Huang (2022), who discovered that incorporating discourse profiling (DP) information, specifically the functional role of each sentence, could enhance the overall performance of temporal dependency graph parsing (TDG; Yao et al., 2020). Their evaluation may not have been sufficiently comprehensive, however. TDG parsing encompasses three distinct types of relation parsing:

Proceedings of the 4th Workshop on Computational Approaches to Discourse (CODI 2023), pages 82–88

timex to timex (t2t), event to timex (e2t), and event to event (e2e), each requiring a different prediction mechanism. Upon a more detailed reexamination of Choubey and Huang's (2022) findings, DP information in fact does not consistently improve performance across all three relation types; it reliably enhances e2e, but may lead to a decline in performance for the other two.

We believe this is caused by two major limitations of Choubey and Huang's (2022) approach. First, DP is a hard problem in its own right. The state-of-the-art DP system (Choubey and Huang, 2021) only yields a 59.21% F1 performance. This means TIE systems following Choubey and Huang's (2022) guidance will only have access to noisy and inaccurate DP features. Second, sentence function is a relatively high-level, descriptive type of discourse structure. Temporal dependency structure, on the other hand, can also benefit from a lot of simple surface-level discourse information, such as precedence (Zhang and Xue, 2018a).

To address these issues, we have experimented with incorporating surface-level sentence-position information into a TIE system, and in two ways: encoding absolute sentence-position by appending the sentence number directly onto the context sentences, following Choubey and Huang (2022), and proposing a novel Sentence Position Embedding (SPE) using a sinusoid. Our experiments demonstrate that SPE could significantly enhance temporal dependency graph parsing performance across all relation types, with the performance increase being mostly greater or at least comparable to that provided by DP information. The resulting TDG parsing system[1] with SPE obtains the state-of-the art performance.

2 Temporal Dependency Parsing

TIE is the task of classifying the temporal relation between two temporal nodes. A temporal node can be either an event trigger (a.k.a. event mention) that represents an event that exists in the narrative of an article, or a timex that is a nominal description of a date or time. When treating a pair of temporal nodes as either intervals or points on the timeline, the temporal relation between temporal nodes can be described by Allen's (1983) temporal calculus. There are some variations between different TIE annotation standards, but generally speaking, temporal relations include links such as BEFORE, AFTER and OVERLAP.

[1]The code and data are publicly available online: https://github.com/frankniujc/tdg-discourse.

"A 26 years [sic] old woman **died** **early this week**. She **fell** roughly 30m down the Bergisel mountain in Tyrol on **Friday**. Remaining conscious after the **fall**, she had **alerted** her family via telephone who in turn **contacted** emergency services."

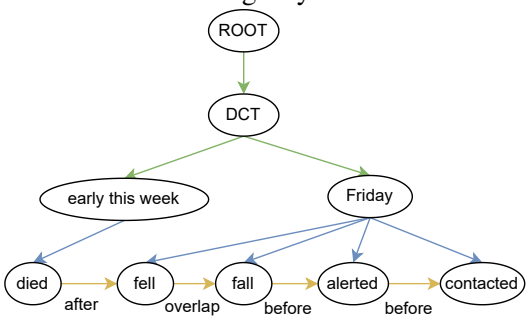

Figure 1: An example of a TDG from (Yao et al., 2020). In the example text (upper), event triggers are highlighted in green and timexes are highlighted in orange. In the TDG (lower), different types of dependency relations have different edge colours (t2t, e2t and e2e). Each arrow points from the parent node to the child node.

	Docs	Timex	Event	t2t	e2t	e2e
Train	400	1,952	12,047	2,352	15,369	8,725
Dev	50	325	1,717	375	2,136	1,298
Test	50	209	1,015	259	1,324	706
Total	500	2,486	14,779	2,986	18,829	10,729

Table 1: TDG corpus statistics.

This pairwise annotation scheme, however, fails to generalize to the document level. The number of temporal node pairs is quadratic in the number of temporal nodes (n), i.e., $\binom{n}{2} \in O(n^2)$. Yao et al. (2020) pointed out that this quadratic increase, together with the increase in the complexity and number of vague relation links for annotators to consider will, in practice, inevitably cause errors to annotation.

To address this issue, Kolomiyets et al. (2012); Zhang and Xue (2018b); Yao et al. (2020) have advocated for using dependency structures to represent document-level temporal relations. Kolomiyets et al. (2012) annotated a children's story with temporal dependency trees. Each event u only depends on one other event v iff the interpretation of when u occurred requires knowing when v occurred. Kolomiyets et al.'s (2012) temporal dependency tree structure only includes events, but this standard may yield disconnected structures. Zhang and Xue (2018b) refined temporal dependency tree structure to allow the inclusion of timex

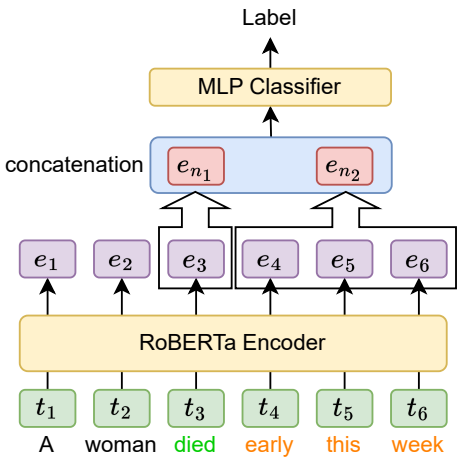

Figure 2: An overview of the pairwise classification model architecture.

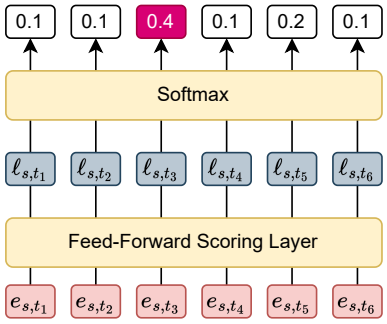

Figure 3: An overview of the joint ranking model architecture. Given a temporal node s in the article, the model predicts a scalar reference score for every candidate node $(t_1, \ldots, t_6)$. This reference score can be considered as classification logits and later trained using the cross-entropy loss.

vertices as well as two special vertices: a document creation time (DCT) vertex and a ROOT vertex. The inclusion of timex vertices allows for capturing the missing events in timex (e2t) temporal dependencies and timex to timex (t2t) temporal dependencies. The addition of the DCT and ROOT vertices ensures each document is always parsed into a valid TDT.

Both Kolomiyets et al. (2012) and Zhang and Xue (2018b) assumed that each event or timex had exactly one reference temporal node (to which the dependency edge points), resulting in a tree structure. Yao et al. (2020), on the other hand, argued that this assumption is overly stringent, and that it is possible for an event to have both a reference timex and an reference event. They therefore proposed to characterise temporal structure with temporal dependency graphs (TDG), in which each event can have a timex parent, an event parent, or both. As depicted in Figure 1, the event *alerted* depends on both the timex *Friday* and the event *fall*. As a result, TDG is more expressive than the earlier TDTs. In this work, we used the TDG corpus released by Yao et al. (2020). Table 1 shows the statistics of this corpus.[2]

3 Model Architectures

3.1 Pairwise Classification Model

Typically, TIE is formulated as a classification task. Given a pair of temporal nodes (n_1, n_2), the sentences containing the nodes $([t_{11}, \ldots, t_{1m}], [t_{21}, \ldots, t_{2n}])$ are encoded into a context vector $\mathbf{e} = [e_{11}, \ldots, e_{1m}, e_{21}, \ldots, e_{2n}]$. Next, the event embedding pair $[e_{n_1}; e_{n_2}]$ are concatenated and the classification task is performed using a multilayer perceptron (MLP) layer. Where a temporal node spans multiple tokens, we utilize Lee et al.'s (2017) method for obtaining an attentive span representation. Figure 2 depicts an overview of this architecture. In this model, we deliberately avoid jointly learning the pairwise model to observe the effects of different discourse information on various relation types.

3.2 Joint Ranking Model

Neural ranking models (Zhang and Xue, 2018a; Ross et al., 2020; Choubey and Huang, 2022) formulate the task as a regression problem. For each temporal node, the model predicts a scalar reference score for every potential parent node and selects the edge with the highest reference score. Therefore, this edge selection process can be formulated as a classification task — the reference scores can be considered as classification logits, and the cross-entropy loss of the edge prediction can be calculated. The three relation types (t2t, e2e, and e2e) are trained jointly. Unlike the pairwise model that uses the concatenation of the two event embeddings, we follow Choubey and Huang (2022), who enclose both triggers in special symbols ($\$n_1\$$ and $\#n_2\#$) and use the embedding of the `[CLS]` token as the pair embedding $e_{n_1,n_2} = e_{\texttt{[CLS]}}$.

[2]There are some minor discrepancies between the statistics reported by Yao et al. (2020) and the final released corpus. We used the final version of the TDG corpus released at https://github.com/Jryao/temporal_dependency_graphs_crowdsourcing.

4 Discourse Analysis for TDG Parsing

Based on Dijk's (1986) schemata of news content, Choubey et al. (2020) proposed the task of discourse profiling (DP). The task is to classify each sentence into one of eight content types (see appendix B). There are two ways of encoding DP information, as proposed by Choubey and Huang (2022). The first (**DP Feature**) appends the content type label directly, marked with a special token #. For instance, if the sentence represents a *main event*, the label `#MAIN#` is appended to the sentence. We obtained the same model-generated content type labels from Choubey and Huang (2022). The second (**DP Distillation**) involves using model distillation. In this approach, the model is equipped with two decoders: one predicts the reference score, while the other performs DP classification. The training of both tasks occurs simultaneously, distilling the DP information into the underlying language model.

4.1 Sentence Position Information

Sentence position information has proven valuable in various tasks. For instance, the next sentence prediction (NSP) task played a crucial role in training BERT (Devlin et al., 2019), and similar techniques have been shown to be effective for discourse analysis (Yu et al., 2022). In temporal dependency parsing, previous work (Zhang and Xue, 2018a) employed hand-crafted precedence features to enhance performance. In this study, we also present two methods for encoding sentence position:

Sentence Position Feature (SPF) We experimented with directly incorporating sentence position information into the context sentence, in a manner similar to the DP feature. For each sentence, we prepend the context sentences with "`Sentence X:`," where `X` represents the sentence number.

Sentence Position Embedding Vaswani et al. (2017) utilized sine and cosine functions with varying frequencies for token position encoding. We extend this idea by proposing a sentence position encoding (SPE; Equation 1), where pos denotes the sentence number, i is the dimension, and d_{model} is the model's dimension.

$$\begin{aligned} SPE_{(pos,2i)} &= \sin(pos/10000^{2i/d_{\text{model}}}) \\ SPE_{(pos,2i+1)} &= \cos(pos/10000^{2i/d_{\text{model}}}) \end{aligned} \quad (1)$$

Since the SPE shares the same dimension as RoBERTa's word embeddings, they can be summed. For the pairwise model, we add the SPE of the event's sentence to its event embedding. For the joint ranking model, we directly add both SPEs of both sentences to the pooler's output. A *post-hoc* classifier on RoBERTa itself serves as our baseline.

5 Experimental Results

5.1 Pairwise Prediction Results

The left side of Table 2 displays the performance of the models with various types of discourse information. Among the results, we can emphasize two key comparisons. First, **the addition of all kinds of discourse information leads to a substantial performance increase in the e2e parsing task**; however, it may result in a decline in performance for the other two types. A contributing factor is that the e2e task not only models temporal dependency structure but also requires the model to learn a shortcut heuristic that takes sequence length into account. Upon closer examination, we discovered that Yao et al.'s (2020) assumption that each event can depend on at most one other event is not always valid. It is common for an event to have multiple parents. In such cases, the TDG annotation standard instructs the annotator to choose the event that is closer in time. If this is not feasible, the annotator should select the event that is closer in textual order. Therefore, discourse information offers extra benefits for e2e parsing, regardless of the DP encoding.

Second, **SPE is the only information that leads to performance improvements across all three relation types, and it also yields the most significant performance increase**. As previously discussed, DP information that is model-generated is noisy. Moreover, the discourse structure of TDG news articles is relatively simple. Surface-level sentence position can be considered a reliable proxy for the article's discourse structure. For instance, every news article in the TDG corpus begins with a timex indicating the publication date of the article. Additionally, the majority of the articles follow the publication time with the lead sentence of the article. Directly incorporating the sentence number into the article, however, does not produce the same level of performance improvement. This outcome is also expected, as a BERT-based language model struggles with representing numbers (Wallace et al., 2019).

Model	Pairwise Model			Joint Ranking Model			
Relation	t2t	e2t	e2e	t2t	e2t	e2e	overall
Baseline	94.72	74.07	60.59	93.82	78.72	70.37	77.94
DP-F	94.55	76.64	70.79	**94.59**	76.91	70.99	77.15
DP-D	92.87	73.71	67.72	91.12	77.97	**73.20**	78.07
SPF	94.53	71.41	70.74	92.66	76.83	71.78	77.05
SPE	**95.37**	**77.69**	**72.19**	91.12	**79.10**	72.73	**78.64**

Table 2: Performance on different settings. Top performance of each segment is highlighted in bold.

5.2 Ranking Model Results

The right side of Table 2 presents the performance of the ranking models. Once again, SPE achieves the highest overall performance, showcasing the effectiveness of this approach. Similar to the pairwise results, all models surpass the baseline for the e2e task. Interestingly, with only a few exceptions, the e2t and t2t performance of each model declines. In addition to the previously mentioned reasons, one contributing factor is the imbalanced distribution of the three relation types. The TDG corpus contains 2,486 timexes and 14,779 events, resulting in 20,862 t2t, 63,065 e2t, and 233,065 e2e potential pairs in the training set. When all three types are trained jointly, the model overfits on the e2t and e2e relations, leading to performance disparities across the three relation types.

Despite the issue of data imbalance, the benefits of joint learning are substantial. All models exhibit better performance on the e2t and e2e tasks compared to their pairwise counterparts. The three relation types are not disconnected; for instance, events that depend on the same timex are likely to depend on each other. Without joint learning, this valuable TDG structural information is lost. There are moreover several ways to better model structural information, such as the application of GNNs (Ji et al., 2019), as well as methods to address the data imbalance issue. We leave these topics for future research.

6 Discussion

Before Choubey and Huang (2022), the relationship between discourse and TIE had not been explored, and indeed our own experiments corroborate the value of their insight to incorporate discourse information into constructing document-level temporal structures. Merely using the output of a discourse system as an additional input feature for document-level TIE may not be the most effective strategy, however. A very superficial, but novel sentence position embedding effectively encodes surface-level sentence-order information, and seems to be more reliable as a proxy for the discourse structure of news articles. Incorporating this information leads to state-of-the-art performance in TDG parsing.

The success of sentence-position embedding offers a significant opportunity to bridge discourse analysis and document-level temporal dependency parsing. It suggests that we should not naïvely rely on discourse information as a separate, modular input source. Instead, the similarities between the two tasks indicate that various techniques and insights can be transferred and applied across both domains, leading to more effective models and a deeper understanding of the relationship between discourse analysis and temporal dependency parsing.

Acknowledgement

This study is funded by the Canadian Safety and Security Program (CSSP) from Defence Research and Development Canada (DRDC) awarded to the Public Health Agency of Canada (CSSP-2018-CP-2334: Incorporating Advanced Data Analytics into a Health Intelligence Surveillance System). We thank the Global Public Health Intelligence Network (GPHIN) and Epidemic Intelligence from Open Sources (EIOS) teams for their support. We would also like to thank Prafulla Kumar Choubey for sharing the DP data.

Limitations

In accord with Choubey and Huang (2022), our study focuses solely on the *unlabelled* performance of TDG parsing. This implies that our evaluation is limited to identifying reference temporal relations without considering the classification of relation types. We plan to explore the labelled TDG parsing task in future research.

Owing to resource constraints, our experiments were conducted using only one type of language model, RoBERTa-base. However, other models such as BERT (Devlin et al., 2019), DeBERTa (He et al., 2021), and ERNIE (Zhang et al., 2019) have demonstrated impressive performance across various natural language understanding benchmarks. We aim to evaluate these models in future research, and we encourage other researchers to reproduce our work using these alternative models.

References

James F. Allen. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.

Prafulla Kumar Choubey and Ruihong Huang. 2021. Profiling News Discourse Structure Using Explicit Subtopic Structures Guided Critics. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1594–1605, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Prafulla Kumar Choubey and Ruihong Huang. 2022. Modeling Document-level Temporal Structures for Building Temporal Dependency Graphs. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 357–365, Online only. Association for Computational Linguistics.

Prafulla Kumar Choubey, Aaron Lee, Ruihong Huang, and Lu Wang. 2020. Discourse as a Function of Event: Profiling Discourse Structure in News Articles around the Main Event. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5374–5386, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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.

Teun A. Van Dijk. 1986. News Schemata. *Studying writing: linguistic approaches*, pages 155–186.

Grigorii Guz and Giuseppe Carenini. 2020. Coreference for Discourse Parsing: A Neural Approach. In *Proceedings of the First Workshop on Computational Approaches to Discourse*, pages 160–167, Online. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with Disentangled Attention. In *International Conference on Learning Representations*.

Tao Ji, Yuanbin Wu, and Man Lan. 2019. Graph-based Dependency Parsing with Graph Neural Networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2475–2485, Florence, Italy. Association for Computational Linguistics.

Oleksandr Kolomiyets, Steven Bethard, and Marie-Francine Moens. 2012. Extracting Narrative Timelines as Temporal Dependency Structures. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 88–97, Jeju Island, Korea. Association for Computational Linguistics.

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.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Hayley Ross, Jonathon Cai, and Bonan Min. 2020. Exploring Contextualized Neural Language Models for Temporal Dependency Parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8548–8553, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP Models Know Numbers? Probing Numeracy in Embeddings. 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 5307–5315, Hong Kong, China. Association for Computational Linguistics.

Jiarui Yao, Haoling Qiu, Bonan Min, and Nianwen Xue. 2020. Annotating Temporal Dependency Graphs via Crowdsourcing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5368–5380, Online. Association for Computational Linguistics.

Nan Yu, Meishan Zhang, Guohong Fu, and Min Zhang. 2022. RST Discourse Parsing with Second-Stage EDU-Level Pre-training. In *Proceedings of the 60th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 4269–4280, Dublin, Ireland. Association for Computational Linguistics.

Yuchen Zhang and Nianwen Xue. 2018a. Neural Ranking Models for Temporal Dependency Structure Parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3339–3349, Brussels, Belgium. Association for Computational Linguistics.

Yuchen Zhang and Nianwen Xue. 2018b. Structured Interpretation of Temporal Relations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced Language Representation with Informative Entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

A Training Details

We follow Choubey and Huang's (2022) experiment setup. We first conducted a hyperparamter search on learning rate using the baseline models. In particular, we used 1e-5 for the t2t pairwise models, 3e-5 for the e2t and e2e pairwise models, and 8e-5 for the joint ranking models. We train each model for 15 epochs, and report the test set performance on the model with the highest development set performance. RoBERTa-base is used as the encoder for all the experiments. For the pairwise model, we down sampled e2e labels by a factor of 10.

B DP Content Types

Choubey et al. (2020) specified eight DP content types: Main event (M1), Consequence (M2), Previous Event (C1), Current Context (C2), Historical Event (D1), Anecdotal Event (D2), Evaluation (D3) and Expectation (D4).