A simple but effective model for attachment in discourse parsing with multi-task learning for relation labeling

Zineb Bennis
IRIT
zinebennisa@gmail.com

Julie Hunter
Linagora
jhunter@linagora.com

Nicholas Asher
IRIT
Nicholas.Asher@irit.fr

Abstract

We present a discourse parsing model for conversation trained on the STAC corpus (Asher et al., 2016). We fine-tune a BERT-based model to encode pairs of discourse units and use a simple linear layer to predict discourse attachments. We then exploit a multi-task setting to predict relation labels, which effectively aids in the difficult task of relation type prediction; our F1-score equals or surpasses the state of the art in the approaches we have reimplemented using code from the authors with no loss in performance for attachment, confirming the intuitive interdependence of these two tasks. Our method also improves over other discourse parsing models in the literature in permitting attachments in which one node has multiple parents, an important feature of multiparty conversation.

1 Introduction

Discourse parsing, the task of predicting graphs that represent semantic relations (arcs) between *elementary discourse units* or EDUs (nodes), is a hard problem in NLP due to the complexity of discourse graphs and the frequent lack of surface cues provided by EDUs, which forces parsers to rely on deep, semantic information. Multiparty, spontaneous conversation is especially tricky as the structure often meanders and a single participant can respond to multiple discourse moves at once, yielding non-tree-like structures that challenge existent parsing techniques (Afantenos et al., 2015).

While it is tempting to attack a complex problem with complex machinery, as recent research on discourse parsing has done (see Section 6), we show that a simple model can achieve or surpass state of the art results for discourse relation labeling with a little inspiration from human discourse processing.

First, because a decision about whether two EDUs are attached by a semantic relation generally requires reasoning about their contents together, our transformer-based model encodes embeddings of EDU *pairs*, exploiting a form of message passing simpler than graph neural net models (Wang et al., 2021) while achieving better results. Next, we draw on the fact that for human annotators, the tasks of discourse attachment and relation labeling are often interdependent: sometimes one sees how to attach two EDUs but only later how to determine the relation that links them; sometimes, a clue, e.g. an explicit marker like *because*, makes the relation clear, and the task is to find the second argument to the relation. Our model exploits this interdependence with a multi-task architecture for attachment prediction and relation labeling. Finally, our model allows for the non-tree-like structures.

In Section 2, we provide background on discourse structure and data sets. In Sections 3 and 4, we describe our model and results. Section 6 presents related work in discourse parsing.

2 Discourse parsing theories and data sets

Just as a sentence is not a bag of words but comes with a structure that serves to compute its meaning from that of its constituent words, so too a discourse or conversation is not a bag of dialogue moves but comes with a structure that enables an interpreter to compute an overall meaning from its constituents. EDUs are clauses or subclausal units that serve as the minimal, linguistic constituents upon which discourse structures are built (Marcu, 1999), and discourse parsing involves finding the recursive structure over EDUs that exploits their semantic content together with various contextual features.

There are two main theories that have investigated complete discourse structures for texts: RST (Mann and Thompson, 1987) and SDRT (Asher, 1993; Asher and Lascarides, 2003). Only SDRT has been applied to multi party conversation as in the STAC and Molweni datasets (Asher et al., 2016;

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 3412–3417
May 2-6, 2023 ©2023 Association for Computational Linguistics

Li et al., 2020). Given our interest in multiparty conversation, we use SDRT and three versions of the STAC corpus, a set of multi-party chats from an online version of the game *Settlers of Catan*, in which players trade or otherwise acquire resources in order to build roads and settlements and thereby score victory points. The standard version, *S*, contains only linguistic (chat) moves made by players; the situated version, *S-Sit*, integrates descriptions of nonlinguistic events (game moves), represented as *elementary event units* (EEUs). In *S-Sit*, both EEUs and EDUs are integrated into discourse structure. We present the third version in Section 2.1.

In SDRT a conversation provides a number of EDUs linked together to form a weakly connected, Directed Acyclic Graph (DAG). Each EDU apart from the head has at least one incoming link. Backwards links (where an EDU attaches to another EDU that comes after it in the dialogue) are prohibited if the EDUs are produced by different speakers (Perret et al., 2016). Each edge of the DAG is labeled with one of 16 different types of discourse relation, such as Explanation (Exp), Question-Answer-Pair (QAP), or Acknowledgement (Ack).

The DAGs postulated by SDRT allow one child to have multiple parents in the structure. Figure 1 provides a representative example from the STAC corpus: with his 'kk' William acknowledges *both* refusals to his offer and signals that he is moving on before ending his turn.

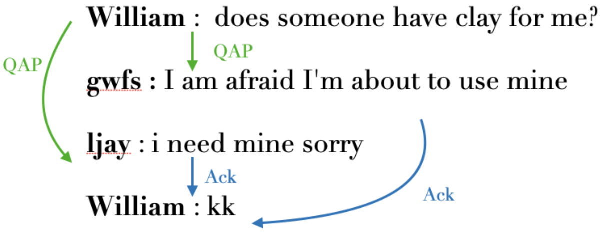

William : does someone have clay for me?
QAP
QAP
gwfs : I am afraid I'm about to use mine
ljay : i need mine sorry
Ack
Ack
William : kk

Figure 1: Example from the STAC corpus *S* illustrating an EDU ('kk') with multiple parents

In multiparty dialogue, coordination and negotiation over content is key. EDUs with multiple parents typically mark moves that signal such a coordination or end of negotiation. Multi-parent links can thus have a major impact on conversational structure and its effects on downstream tasks, despite their statistical infrequency. An important advantage of our model over alternative neural parsers is that it is the only one to take these links into account in both model design and evaluation.

2.1 A new STAC dataset: STAC-Squished

Like other recent work in discourse parsing, we develop a model trained on *S* and detail our results on *S* in Section 4. But *S* is not ideal for two reasons.

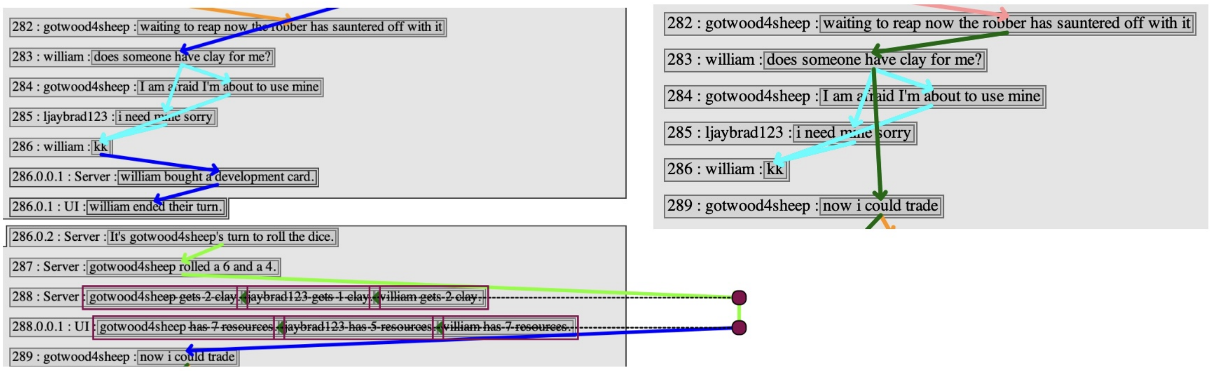
First, we discovered that it contains duplicated data points due to a mistake in the data extraction script. There are 488 duplicated EDUs, corresponding to 60 duplicated dialogues.

Second, and more seriously, there are rhetorical incoherences in *S*. When we examined relation labels, we found, with Badene et al. (2019), that some were not coherently used, because *S* lacks essential nonlinguistic (game) information (Asher et al., 2016). Figure 2 provides an example. The exchange on the right, taken from *S*, is somewhat incoherent; the Continuations (dark green) from 282-283 and 283-289 make little sense, but were included in order to make a complete DAG. On the left, we see the full exchange, from *S-Sit*, with a far more intuitive discourse structure lacking the Continuations.

There are 300 such cases out of 1116 Continuation instances in the training set and 45 out of 113 such cases in the test set. Indeed, dialogue moves and speaker interactions in the conversations often depend on nonlinguistic actions taken by players, but we can only see this dependence once we move to *S-Sit*, where the EEUs provide such actions and other important information on game evolution. We must take account of those EEUs to accurately reflect the discourse structure.

S-Sit builds graphs over EDUs and EEUs and so includes information essential to properly understanding many linguistic interactions. For this reason, we consider *S-Sit* an overall better data source for correct discourse structures.

However, *S-Sit* also has certain drawbacks for training discourse parsing models. It contains multiple, long chains of EEUs connected to their immediate predecessors. The number of EEU-EEU adjacent attachments is very high and highly predictable, artificially inflating F-scores on attachment and relation labeling. Moreover, long sequences of EEUs often induce longer distances between EDUs that need to be attached. Our model, like most, is biased towards predicting closer attachments due to their abundance in the training set. *S-Sit* exacerbates this situation and makes our model actually perform worse on EDU-only attachments (see Section 4).



282 : gotwood4sheep : waiting to reap now the robber has sauntered off with it
283 : william : does someone have clay for me?
284 : gotwood4sheep : I am afraid I'm about to use mine
285 : ljaybrad123 : i need mine sorry
286 : william : kk
286.0.0.1 : Server : william bought a development card.
286.0.1 : UI : william ended their turn.
286.0.2 : Server : It's gotwood4sheep's turn to roll the dice.
287 : Server : gotwood4sheep rolled a 6 and a 4.
288 : Server : gotwood4sheep gets 2 clay. ljaybrad123 gets 1 clay. william gets 2 clay.
288.0.0.1 : UI : gotwood4sheep has 7 resources. ljaybrad123 has 5 resources. william has 7 resources.
289 : gotwood4sheep : now i could trade

282 : gotwood4sheep : waiting to reap now the robber has sauntered off with it
283 : william : does someone have clay for me?
284 : gotwood4sheep : I am afraid I'm about to use mine
285 : ljaybrad123 : i need mine sorry
286 : william : kk
289 : gotwood4sheep : now i could trade

Figure 2: An example from STAC. On the left, we see the full interaction, from *S-Sit*, that contains both chat and game moves. On the right, we see the version from *S* that contains only the chat moves.

Data	EDU	EEU	R-E	Mean
S	13054	0	0	2.04
S-Sit	12588	18576	16382	2.14
S-Sq	12588	12985	10790	2.04

Table 1: Data set stats

The shortcomings of *S-Sit* prompted us to modify it in order to ignore the highly predictable relations and attachments between adjacent EEUs that are not attached to any EDU. To do this, we treated each sequence of EEUs as one block without any internal structure. That is, we collapsed the sequence into a single EEU. Table 1 reflects differences between the various STAC corpora in numbers of EDUs, EEUs, Relation instances over only EEUs (R-E) and mean distances between attachments of linguistic units. We have made available the new *S-Sq* (squished) corpus at https://github.com/zineb198/LineBert.

3 Our Model

S contains a set of multi party dialogues, each consisting of a set of n EDUs $[e_1, e_2, .., e_n]$. Our model BERTLine, is a simple but efficient discourse parsing model with two components: one for inferring attachments and one for inferring relation labels. Like Shi and Huang (2019) and Wang et al. (2021), we compute a standard unlabeled attachment and labeled attachment score for evaluation.

Attachment. BERTLine's attachment module has an encoder-decoder architecture. Encoding for each EDU pair is obtained by finetuning a BERT model on the task of predicting if two EDUs are attached. We used BERT because it takes the position IDs of tokens as input, which is useful for the pair encodings. BERTLine has access only to the content and the speaker/emitter of the EDUs. $[CLS]$ vectors furnish embeddings h of EDU pairs.

$$h_{(i,j)} = BERT(e_i, e_j)$$

The encoder portion of BERTLine is finetuned using the loss L_{l_1} on S_l, the set of positive and negative attachments. Where θ is the parameters to be optimized, $s_{*i,j}$ refers to gold data and $s_{i,j}$ to predictions:

$$L_{l_1}(S_l, \theta) = -\Sigma_{S_l} logP(s_{*i,j} = s_{i,j}|h_{i,j}) \quad (1)$$

We concatenate h with a vector $struct_{(i,j)} = [t_{(i,j)}, d_{(i,j)}]$ that represents information useful for computing attachments (Perret et al., 2016), namely speaker change and EDU distance (i.e., how many EDUs occurred between i and j in the natural order of the dialogue). The function $t_{i,j}$ returns 1/0 if i and j have the same/different speaker; $d_{i,j}$ returns the distance. Our final pair embeddings are:

$$H_{(i,j)} = h_{(i,j)} \oplus struct_{(i,j)}$$

The decoder part of BERTLine's attachment model is a linear neural network. Like previous work, this layer predicts for each EDU e_j, which of the preceding EDUs e_i, with $j > i \geq j - 10$ (i.e., up to a certain limit) are likely parents of e_j. It is a binary classification on each pair embedding.

$$P_j = Linear(H_{(j-1,j)}, H_{(j-2,j)}, ..., H_{(j-10,j)})$$

Our loss function is a binary cross entropy over possible attachments for each dialogue d in training set D. Below $l_{*i,j}$ indicates if e_i and e_j are attached in the gold data, and $l_{i,j}$ is a predicted link:

$$L_{link}(d, \theta) = -\Sigma_{j=1}^{|d|} \Sigma_{i=0}^{j} logP(l_{*i,j} = l_{i,j}|H_{i,j}) \quad (2)$$

$$L_{link}(\theta) = \Sigma_{d \in D} L_{link}(d, \theta) \quad (3)$$

The set of predicted attachments $\mathcal{L}$ contains all pairs $l_{i,j}$ whose link probability exceeds a given threshold α, optimized at 0.8 through experiments on the validation set; i.e., $l_{i,j} \in \mathcal{L}$ *iff* $P(l_{i,j}) > \alpha = 0.8$. We trained the linear layer for 15 epochs.

This set up allows us to predict multiple parents for a given EDU. Of the 77 multi-parent structures (8 with more than 3 parents) in the test set, we captured about 20%.

Relation labeling. To test the interdependence of link and relation prediction, we built a BERT model with two classification heads. The first head learns to predict links between EDUs and the second head learns to classify EDU pairs in terms of SDRT's 16 relation labels. We compute the loss of each head independently, average the results, and back-propagate the average back to the BERT encoder. At inference time, only the relation classification head is used for predictions.

We fine-tuned the BERT model over S_l over 3 epochs, together with training over S_r, the set of relation label instances. $r_{(*i,j)}$ refers to gold relation data, $r_{(i,j)}$ to predictions.

$$L_{rel}(S_r, \theta) = -\Sigma_{S_r} logP(r_{*i,j} = r_{i,j} | h_{i,j}) \quad (4)$$

$$L_{multi}(S_l \cup S_r, \theta) = mean(L_{l_1} + L_{rel}) \quad (5)$$

All previous work apart from Wang et al. (2021) defines the task of relation prediction as following that of link prediction in a pipeline architecture. Our multitask setup allows us to have an embedding containing information about both tasks and improves the relation F1-score by one point over a pipeline relation prediction task.

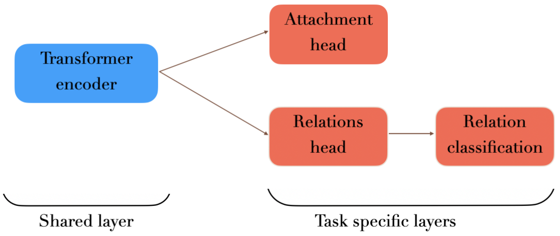


Figure 3: Structure of the multi-task model

Evaluation Metric: We used a micro averaged F1-metric over all gold data attachments and relation label instances for the scores in Table 2.

4 Results

Table 2 shows that BERTLine is competitive on S with the most advanced model of Wang et al. (2021) and even beats its relation labeling score. We provide both the reported scores of Shi and Huang (2019), Liu and Chen (2021) and Wang et al. (2021) and the scores that we got by rerunning their code using the same gold input from S with our evaluation metric, which explicitly considers EDUs with multiple parents.[1] Discrepancies in reported scores, also noted in Wang et al. (2021), may depend on choices of what and how to evaluate but also on machine hardware.

With regard to the different versions of STAC, BERTLine's overall scores improve between S and and *S-Sit* in Table 3, although there is a dramatic drop in linguistic only attachments (A-L) for *S-Sit*. This is due to the presence of many easy to predict attachments and relation instances between adjacent EEUs, which drown out performance on harder, longer distance EDU attachments. Interestingly, our score also improves considerably with *S-Sq*, and linguistic-only attachment suffers a much smaller drop. We hypothesize that this is because we have lowered the median distance between EDUs and rectified the imbalance between EDU and EEUs by deleting long series of EEUs.

BERTLine has a far simpler architecture than the neural models whose code we were able to rerun. It is essentially a local model that uses a minimal amount of message passing from nodes to potential neighbors, a technique from graph neural nets also used to encode entire graphs in Wang et al. (2021). BERTLine's main advantage comes from the explicit pair encoding of EDUs. While broader contextual information and structural constraints on DAGs like those in Perret et al. (2016) can undoubtedly improve scores for discourse attachment and relation labeling, we have sought to show that the efforts of complex architectures to harness this extra information have failed to lead to substantial gains. How to more successfully exploit it is our main research topic for the future.

5 Ablation study

We evaluated the efficiency of two key parts of our model, the multitask set up and the local structural information that we add to the EDU encodings. We

[1] We ran all experiments on a Dell T630 bi pro machine with Nvidia GTX1080Ti GPU cards

Model	DAG	DS-S	Hier-S	GNN-S	BL-S	BL-S-Sq
Attachments	69.0	[73.2/ 72.5]	[75.1/ 68.61]	[73.4/ 73.2]	73.1	79.49
Relation labeling	53.1	[55.7/ 54.8]	[57.1/50.48]	[57.3/ 55.5]	**56.25**	71.15

Table 2: We provide both the original, reported F1-scores for Perret et al. (2016) (DAG), Shi and Huang (2019) (DS), Liu and Chen (2021) (Hier) and Wang et al. (2021) (GNN) on the standard STAC dataset S, as well as the scores we were able to obtain using their models or retraining from their code using our evaluation metric (reported/recomputed). BERTLine (BL) beats the state of the art for relation label prediction on S with a score of 56.25. We also provide BL's results on the squished STAC dataset *S-Sq* with great performance on relation labeling.

F1	**Attach**	**Relns**	**A-L**
S	73.06	56.25	**73.06**
S-Sit	76.94	69.6	62
S-Sq	**79.93**	**71.64**	68.7

Table 3: F1 score for STAC data sets.

compared BERTLine with two baselines: (i) **BL-Simple**: simple BERT is finetuned to predict the relation of a predicted attachment; (ii) **BL-Noinfo**: $H_{i,i}$ without $struct_{i,j}$. We train each variation 10 times and compare the scores to the average over 10 runs for BERTLine. Table 4 shows how structural information and the multitask setup moderately improve the predictions of BERTLine on *S-Sq*.

Model	**BL**	**BL-Simple**
Attach	79.49	78.55
Model	**BL**	**BL-Noinfo**
Rel	71.41	71.15

Table 4: Scores on STAC-squished (*S-Sq)*

6 Related work

Older approaches to multiparty conversation (Afantenos et al., 2015; Perret et al., 2016) used manual features about EDU pairs and simple ML models to build a local attachment model for predicting attachment and relation labels. They also added a decoding mechanism. BERTLine's local model uses transformer-style embeddings.

Shi and Huang (2019) were the first to obtain significant discourse parsing results using a neural approach on the corpus *S*. They attempted to capture incremental and contextual effects in their model by training a supplemental *Structured Encoder* that incrementally updates attachment paths (sequences of parent-child EDUs). However, Wang et al. (2021) showed that the model obtains similar scores with or without the *Structured encoder*; the encoder didn't capture what it intended. Moreover, they implemented this method with Python 2 and Tensorflow 1.3, which are not in use anymore.

Liu and Chen (2021)'s model encodes EDUs using a pre-trained RoBERTa model (Liu et al., 2019) and a bi-GRU cell to capture contextual information but limits the size of the input dialogues. It uses two linear layers for link and relation prediction. We could not reproduce their results with their model or our reimplementation [2] perhaps because of an evaluation metric that does not consider multiple parents or all gold EDU pairs.

The Structure Self-Aware Graph Neural Network (SSA-GNN) by Wang et al. (2021) proposes a complex GNN-based architecture and model that uses both EDU and edge embeddings. The model is comprised of a Hierachichal GRU gate to obtain contextual EDU representations. They then apply the SSA-GNN to capture implicit structural information between EDUs, using a Structure-Aware Scaled Dot-Product Attention (Zhu et al., 2019; Wang et al., 2020) to update edge and EDU representations. A teacher network is also trained and supplements the standard classification loss with an auxiliary loss to enhance learning performances. Our model is simpler with better results. Moreover, none of the models by (Shi and Huang, 2019; Liu and Chen, 2021; Guz et al., 2020; Wang et al., 2021) predict multiple parents for attachments.

7 Conclusions

We have described a simple yet effective discourse parser that provides multiple attachments and codependent learning of the labeling and attachment tasks. Our model is the only neural parser that does this. We also wanted to show the power of local information when cleverly used. Indeed, discourse parsing requires contextual information, but our results show that current research does not yet leverage that information to achieve gains that convincingly outstrip those of a local model.

[2] https://github.com/zineb198/F1_recompute

8 Limitations

Some of the limitations of this work are the lack of diversity in expert-annotated discourse data sets on multiparty and situated dialogue. Since current data sets come from forums and chat messages, we still have to see how our model behaves in a spoken conversation context. More investigation efforts must be made in order to better analyze and evaluate *S-Sq*. While it has a more predictable structure due to nonlinguistic elements, it also seems at first glance to contain better suited semantic relation labeling for the linguistic elements. We need to do an in-depth error analysis on BERTLine's performance on the three versions of the STAC corpus. We would also like to investigate training on one corpus and then running the model on the other corpus, and we would like to do the same with the Molweni corpus.

9 Ethics Statement

Bertline's performance does not seem to pose any ethical difficulties or questions.

Acknowledgements

We gratefully acknowledge support from the 3AI institute ANITI (ANR-19-PI3A-0004) and the ANR project SUMM-RE (ANR-20-CE23-0017).

References

S. Afantenos, E. Kow, N. Asher, and J. Perret. 2015. Discourse parsing for multi-party chat dialogues. In *Empirical Methods in Natural Language Processing*, pages 928–937. Association for Computational Linguistics.

Nicholas Asher. 1993. *Reference to abstract objects in English*, volume 50. D. Reidel, Dordrecht.

Nicholas Asher, Julie Hunter, Mathieu Morey, Farah Benamara, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: The STAC corpus. In *Language Resources and Evaluation Conference*, pages 2721–2727.

Nicholas Asher and Alex Lascarides. 2003. *Logics of Conversation*. Cambridge University Press.

Sonia Badene, Kate Thompson, Jean-Pierre Lorré, and Nicholas Asher. 2019. Weak supervision for learning discourse structure. In *Empirical Methods in Natural Language Processing*.

Grigorii Guz, Patrick Huber, and Giuseppe Carenini. 2020. Unleashing the power of neural discourse parsers–a context and structure aware approach using large scale pretraining. *arXiv preprint arXiv:2011.03203*.

Jiaqi Li, Ming Liu, Min-Yen Kan, Zihao Zheng, Zekun Wang, Wenqiang Lei, Ting Liu, and Bing Qin. 2020. Molweni: A challenge multiparty dialogues-based machine reading comprehension dataset with discourse structure. pages 2642–2652.

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

Zhengyuan Liu and Nancy Chen. 2021. Improving multi-party dialogue discourse parsing via domain integration. In *Proceedings of the 2nd Workshop on Computational Approaches to Discourse*, pages 122–127, Punta Cana, Dominican Republic and Online. Association for Computational Linguistics.

William Mann and Sandra Thompson. 1987. Rhetorical structure theory: A framework for the analysis of texts. *International Pragmatics Association Papers in Pragmatics*, 1:79–105.

Daniel Marcu. 1999. A decision-based approach to rhetorical parsing. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 365–372, College Park, Maryland, USA. Association for Computational Linguistics.

Jérémy Perret, Stergos Afantenos, Nicholas Asher, and Mathieu Morey. 2016. Integer linear programming for discourse parsing. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 99–109, San Diego, California. ACL.

Zhouxing Shi and Minlie Huang. 2019. A deep sequential model for discourse parsing on multi-party dialogues. In *AAAI*.

Ante Wang, Linfeng Song, Hui Jiang, Shaopeng Lai, Junfeng Yao, Min Zhang, and Jinsong Su. 2021. A structure self-aware model for discourse parsing on multi-party dialogues. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 3943–3949. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Tianming Wang, Xiaojun Wan, and Hanqi Jin. 2020. Amr-to-text generation with graph transformer. *Transactions of the Association for Computational Linguistics*, 8:19–33.

Jie Zhu, Junhui Li, Muhua Zhu, Longhua Qian, Min Zhang, and Guodong Zhou. 2019. Modeling graph structure in transformer for better amr-to-text generation. *arXiv preprint arXiv:1909.00136*.