CoMuMDR: Code-mixed Multi-modal Multi-domain corpus for Discourse paRsing in conversations

Anonymous ACL submission

Abstract

Discourse parsing is an important task useful for NLU applications such as summarization, machine comprehension, and emotion recognition. The current discourse parsing datasets based on conversations consists of written English dialogues restricted to a single domain. In this resource paper, we introduce CoMuMDR: Code-mixed Multi-modal Multi-domain corpus for Discourse paRsing in conversations. The corpus (code-mixed in Hindi and English) has both audio and transcribed text and is annotated with nine discourse relations. We experiment with various SoTA baseline models; the poor performance of SoTA models highlights the challenges of multi-domain code-mixed data, pointing towards the need for developing better models for such realistic settings.

1 Introduction

Discourse structures (Mann and Thompson, 1988; Asher and Lascarides, 2005) capture relationships between clauses (e.g., causality, contrast, elaboration, and temporal sequencing) and are crucial to understanding the logical flow of information. These have been utilized in various tasks such as text summarization (Paulus et al., 2018; Li et al., 2016), language understanding, machine reading comprehension (Li et al., 2019), dialog generation (Chernyavskiy and Ilvovsky, 2023; Hassan and Alikhani, 2023; Chen and Yang, 2023) and emotion recognition (Zhang et al., 2023). Researchers have created annotated discourse corpora from human-to-human dialogues for a single language such as English (e.g., STAC (Asher et al., 2016) and Molweni (Li et al., 2020)). However, many modern-day conversations are audio-based and often involve code-mixing of multiple languages, such as Hindi and English (Hinglish). Understanding the discourse structure in such code-mixed audio-based conversations would be interesting. In this paper, we attempt to fill this gap. In a nutshell, we make the following contributions:

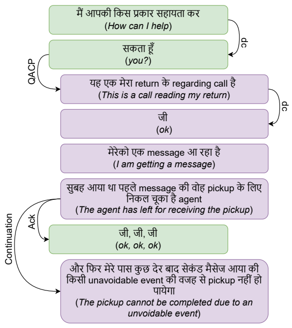

Figure 1: An example of a two-party customer center conversation (in code-mixed language) regarding a complaint on product return. Utterances corresponding to the customer center representative are shown in green boxes and those of the customer in purple boxes. Here, the relation types of "Question answer complaint pair" and "Acknowledgement" are shortened to "QACP" and "Ack" respectively. Here, "dc" is the correction by the annotator on incorrect diarization.

- We present **CoMuMDR**, a large scale code-mixed (Hindi + English = Hinglish), multi-modal (text+audio), multi-domain discourse corpus of multi-party conversations (Table 1). **CoMuMDR** consists of audio recordings and corresponding transcriptions of customer call center interactions from multiple domains, including e-commerce, pharmaceutical, stock broker application support, e-marketplace, and education.
- The corpus is annotated to create a labeled discourse graph for link prediction and discourse relation classification. The annotation is done at the span level with nine discourse relation types that aptly support the flow of information in customer call centers. We merged a few relation types presented in SDRT (Asher and Lascarides, 2005) and added another type "Question answer complaint pair" to support the logical flow. Fig. 1 shows a sample for **CoMuMDR**. The conversations in a practical setting can be complex; for example, there can be an overlap (§3) between utterances (7th utterance) of two speakers. Also, note that since we used ASR and a diarization

	STAC	Molweni	CoMuMDR
# dialogues	1137	10000	799
# utterances	10678	86042	8811
# words	44843	860851	79867
Avg. # utterances/dialogue	11.07	8.83	11.03
Avg. # words/dialogue	39.44	95.65	99.96
Parties	Multi	Multi	Two
Modalities	Uni-modal	Uni-modal	Multi-modal
Languages	English	English	Code-mixed
Source	Catan Game	Ubuntu chats	Call center interactions
Domains	Single domain	Single domain	Multi-domain
Discourse Labels	17 labels	17 labels	9 labels
Annotator Metrics	Kappa	Kappa	Kappa, Jaccard
Data split # dialogues			
Train	909	9000	639
Test	115	500	81
Validation	113	500	79

Table 1: Comparison with previous corpora

model for transcribing and splitting (§3), an utterance (e.g., utterances 1, 2, and 3) could get incorrectly split due to diarization errors. These are resolved during annotations (§3).

- We evaluate existing text-based discourse parsers (and GPT-4o) for link and relation prediction on CoMuMDR using English-only and multilingual text embeddings. We compare this with the performance of existing corpora STAC and Molweni. We observe that SoTA models underperformed on CoMuMDR, pointing towards the development of advanced models.
- We will release the experiment code, audio transcriptions (and text embeddings), and audio features upon acceptance (we do not release the actual audio and unfiltered transcripts due to concerns about the privacy of the company and customers). Currently, we release a sample from CoMuMDR.

The motivation behind CoMuMDR is to create a practical, real-world system that handles audio conversations and is robust to transcription and diarization errors.

2 Related Work

Discourse Parsing has been an active research area in the NLP community (Li et al., 2022). Discourse parsing consists of three main components: discourse segmentation (Wang et al., 2018; Lukasik et al., 2020; Liu et al., 2021), discourse link prediction and discourse relation classification. Discourse segmentation divides a text corpus into Elementary Discourse Units (EDUs) for further processing. Discourse link prediction predicts a directed link between two EDUs, and relation classification assigns a relation type to the link (also check discourse theories in App. A.1).

Datasets: In the context of English, two main text-based corpora have been proposed for Discourse parsing: **STAC** (Asher et al., 2016) and **Molweni** (Li et al., 2020) (check details in App. A.2). Table 1 compares the STAC and Molweni datasets with our proposed dataset. CoMuMDR is code-mixed, audio-based, and covers multiple domains as opposed to mono-lingual single-domain conversations covered by existing text-based datasets. The corpus (having a comparable number of words with STAC) is based on Hindi-English code-mixed audio conversations with imperfect transcription and diarization quality, so CoMuMDR proposes a practical outlook on discourse parsing in conversations. Note that compared to existing datasets (STAC (based on the Catan game) and Molweni (based on Ubuntu chats)), CoMuMDR, besides including audio information, covers more domains and a variety of topics.

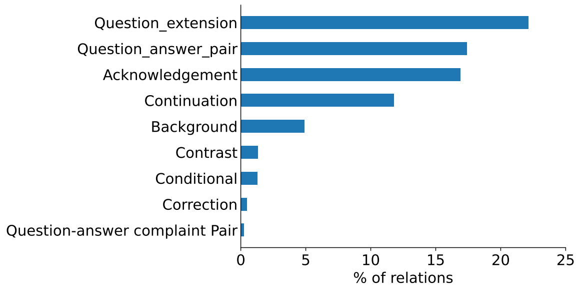

Figure 2: Distribution of discourse labels in CoMuMDR.

Discourse Parsing Models: Various approaches have been proposed for Discourse parsing such as deep sequential model (Shi and Huang, 2019), hierarchical model (Liu and Chen, 2021), Structure-aware model (Wang et al., 2021), SSP-BERT+SCIJE model by Yu et al. (2022) and SDDP model by (Chi and Rudnicky, 2022). Due to space constraints, details are given in App. A.3. We benchmark using each of the above models.

3 CoMuMDR Creation

CoMuMDR consists of two-party customer call center interactions. We obtained the data via a joint research collaboration with a call center company (they own the data) that wants to automate customer call understanding. The calls mainly cater to Indian customers and companies. We ensure that the privacy of customers and companies mentioned in a call is maintained during annotation. The audio data is transcribed using the existing ASR (Automatic Speech Recognition) system Verma et al. (2023) (details in App. B.1) and diarized into utterances using Koluguri et al. (2021) (details in App. B.2). Subsequently, the data is anonymized for customer names and other private information. A team of 3 professional annotators further annotated the transcribed and diarized data to develop a DAG for discourse parsing.

Annotation Details: The annotators were tasked to identify the EDUs, predict links between them to form a DAG, and label a relation for the links. A team of two annotators independently annotated the data, and another annotator verified the annotations and marked batches for re-annotation if deemed necessary. Previous studies have shown that identifying complex discourse units (CDUs) is a challenging task (Muller et al., 2012; Afantenos et al., 2015). Prior discourse parsing models have used various strategies to convert CDUs to EDUs for efficient parsing (Shi and Huang, 2019; Liu and Chen, 2021). We identified similar issues and instructed the annotators to connect an EDU with only the head (i.e., the first EDU) of a CDU. Appendix C presents more details regarding the annotation. We used the inception software (Klie et al., 2018) for annotation (§C.1). We provide details of annotators, instructions, and processes in the App. C.2, App. C.3, and App. C.4, respectively. We use nine discourse labels to annotate our data (see Fig. 2, also see Fig. 3 for comparison with other corpora). App. Table 5 lists the discourse labels (and definitions). The labels are based on Semantic Discourse Representation Theory (SDRT) (Asher et al., 2016). App. G provides details of the distribution of relative distance between linked EDU pairs for each relationship type. In addition, we propose a new relation type, "Question-Answer Complaint Pair," to classify complaints separately. Additionally, we removed the "Narration" discourse label as it was not required in two-party customer conversations. During a pilot annotation experiment, we found that several discourse labels conveyed overlapping meanings. Hence, we merged them.

Question Extension is made by merging "Clarification question" and "Question elaboration", as all the instances of customer call center conversations posed clarification questions as elaborations, and the answers to them were more akin to answers to elaborative questions. **Conditional** is made by merging "Alternation" and "Conditional". Because, in code-mixed conversations, it is hard to differentiate between a conditional and an alternation. It is due to the nature of Hindi-English code-mixed conversations. **Continuation** is made by merging "Comment", "Elaboration", "Parallel" and "Result". Because call center conversations rarely contain examples of comments compared to STAC and Molweni. Customer calls do not revolve around multiple ideas or topics; hence, there is no notion of parallel. Customer call center representatives continuously assure the customers about quick resolution of complaints, and the result of the conversation is typically implicit.

Dialogues are divided into utterances using a diarization model. However, the audio contains overlapping utterances where both speakers are speaking simultaneously. Hence, we added another relation termed "diarization continuation" to fix the diarization issues. However, this relation label is not part of the discourse and is not used in calculating the results (§4). We plan to use these annotations to improve the diarization model.

Metric	Score
Jaccard	0.9569
Span exact match (see App. C.5)	0.6294
Span partial match (see App. C.5)	0.8224
Relation exact match (see App. C.5)	0.5500
Relation partial match (see App. C.5)	0.5321
Structured Kappa (Asher et al., 2016; Li et al., 2020)	0.4044
Relationship Kappa (Asher et al., 2016; Li et al., 2020)	0.3190

Table 2: Inter-annotator agreement metrics

Inter-Annotator Agreement: Table 2 shows inter-annotator agreement using various metrics. Given our complex setting, existing metrics (e.g., Kappa) show a relatively low performance compared to previous datasets. We computed Kappa using the span and relation exact match metrics as in STAC and Molweni.

4 Experiments, Results and Analysis

Discourse Modeling: A dialogue consists of a list of utterances between two speakers. The utterances are further divided into elementary discourse units (i.e., clauses (Asher and Lascarides, 2005)) $\{u_0, u_1, ..., u_n\}$, where u_0 is a dummy root EDU. Discourse parsing models predict a directed link between two EDUs u_j and u_i and classify a relation label r_{ji}.

Experimental Setup: We experimented with state-of-the-art discourse parsing models: deep sequential model (Shi and Huang, 2019), hierarchical model (Liu and Chen, 2021), Structure-aware model (Wang et al., 2021), SSP-BERT+SCIJE model by Yu et al. (2022) and SDDP model by Chi and Rudnicky (2022). We implemented all the discourse parsing models on STAC, Molweni, and **CoMuMDR** and trained them from scratch (details in App. D). We followed the data-split (train/validation/test) as given in Table 1. Validation set was used to tune the models. We implemented the models in two settings for encoding the text in the elementary discourse units: English-only and multilingual embeddings. English-only embeddings include GLoVe (Pennington et al., 2014) or `Roberta-base` embeddings (Liu et al., 2019), same as those used in the original implementations. On the other hand, multilingual sentence-level embeddings include `paraphrase-xlm-r-multilingual-v1` (Reimers

	Link only			Link + relation		
	STAC	Molweni	CoMuMDR	STAC	Molweni	CoMuMDR
Multi-lingual embeddings						
Hierarchical	0.6841	0.7000	0.9036	0.5221	0.5733	0.4263
Structure-aware	0.7125	0.8050	<u>0.9434</u>	0.5314	0.5614	0.5005
SSP-BERT + SCIJE	0.7250	0.8205	**0.9495**	<u>0.6151</u>	**0.6634**	0.5547
SDDP	0.7304	0.7898	0.9416	0.5670	0.5770	0.3781
English-only embeddings						
Deep Sequential	0.7496	0.7577	0.7330	**0.6318**	0.5162	0.4796
Hierarchical	0.7505	0.8097	<u>0.9443</u>	0.5704	0.5690	0.5786
Structure-aware	0.7267	0.8232	0.7782	0.5582	<u>0.5934</u>	0.4072
SSP-BERT + SCIJE	0.7201	0.8293	**0.9452**	0.5623	0.5925	0.5675
SDDP	0.7488	0.8233	0.7918	0.5887	0.5770	0.2941

Table 3: F1-score of various discourse parsing models. Values in **bold** highlight the top-performing model in each method, while values in <u>underline</u> highlight the next top-performing model.

and Gurevych, 2019), which convert a complete EDU's text to a 768-dimension vector.

Results: Table 3 shows the F1-score (App. E) on test sets for link and link+relation prediction. For both the settings, CoMuMDR scores are lowest across all the models, highlighting the challenge of discourse parsing on multi-domain, multilingual conversations. As can be observed, our data has an equivalent score across models for link prediction. SDDP and Hierarchical model outperform on CoMuMDR compared to STAC and Molweni. However, in relation classification (link+relation), CoMuMDR has the lowest performance, possibly due to the presence of multiple domains and the challenge of domain adaptation (Liu and Chen, 2021).

Error Analysis: On further analysis, we observed that the hierarchical model (Liu and Chen, 2021) could not predict the same relation links for "Correction" and "Contrast" on CoMuMDR, leading to a loss of performance in link prediction and relation classification involving correction and contrast. The hierarchical model easily identifies "Acknowledgement" relations among the correctly predicted links. It could be due to the strong presence ($\sim$ 18%) of "Acknowledgement" in the dataset. Similarly, in SSP-BERT, the model misclassified some "Acknowledgement" relations as "Question answer pairs". App. Fig. 4 is an example of a conversation snippet with the gold and predicted relations marked on the left and right sides, respectively. The model incorrectly classified an "Acknowledgement" relation as a "Question answer pair", possibly due to the presence of "ma'am" in the acknowledgment clause (also see App. Fig. 4). In Table 3, we have computed the results of link and link+relation prediction of SDDP using Roberta and `paraphrase-xlm-r-multilingual-v1`. Roberta handles English-only text and cannot handle code-mixed or Hindi text. On the other hand, `paraphrase-xlm-r-multilingual-v1` can handle multilingual text but often fails at effectively processing code-mixed text. Thus, Table 3 shows lower performance on relation prediction for SDDP. Baseline methods, including Deep sequential, hierarchical, Structure-aware, SSP-Bert + SCIJE, perform relation prediction after link prediction, i.e., they classify the relation type for each predicted link. On the other hand, SDDP performs link+relation prediction simultaneously as a single task, which is much more complicated. Hence, SDDP shows significantly lower performance on link+relation prediction than other baseline methods. Additionally, SDDP assumes the discourse relations to form a tree and performs tree parsing during inference, while most of the discourse relations in CoMuMDR cannot adhere to tree structures. Hence, SDDP on CoMuMDR shows low scores on link+relation prediction.

	Link only	Link+relation
STAC	0.6012	0.2729
Molweni	0.5176	0.1474
CoMuMDR	**0.7217**	**0.2808**

Table 4: Performance of GPT-4o as a discourse parser

Results of GPT-4 Model: We evaluated GPT-4o on the test set (81 dialogues, 890 utterances). We prompted GPT-4o in a 3-shot setting (template in App. F) to behave as a discourse parser (results in Table 4). GPT-4o performs worse on both tasks compared to the SoTA models. On examining the confusion matrix (App. Table 8) for GPT-4o on CoMuMDR, we observed misclassification of "Question extension" as "Continuation", possibly due to the overlapping semantics of these relations in a two-party conversation.

5 Future Directions and Conclusion

This paper presents CoMuMDR, a new discourse corpus for multi-modal, multi-domain, and code-mixed conversations from various customer call centers. We transcribed the audio and diarized the text into utterances. We annotated the EDUs using nine discourse labels by combining a few closely related labels from the SDRT format as they formed a more appropriate flow of discourse in a two-party conversation on customer support calls. In this work, we experimented with SoTA models; however, these do not perform well on CoMuMDR. In the future, we plan to develop more advanced models incorporating audio modality information.

Limitations

We developed **CoMuMDR** by capturing audio conversations between a customer and a customer care representative. The audio is then transcribed for annotation.

Our corpus is not as big as the existing Discourse corpora but our corpus is code-mixed, multi-domain, and multi-modal. The corpus is sizable enough to develop meaningful models. Nevertheless, we plan to keep growing our corpus. Discourse annotations is a very time consuming process and hence it takes time to expand the corpus.

CoMuMDR consists of nine discourse relation labels, far fewer than STAC and Molweni, which contain 17 labels. We found during our pilot annotation process that the "Narration" discourse label had no role in customer-centered conversations. Also, we found that in two-party conversations, some of the discourse labels had quite confusing meanings, which led to poor inter-annotator agreements. Hence, we combined the labels to create our presented nine labels presented in Table 5.

To build the dataset, we collected audio recordings from customer care centers. The audio was then transcribed and diarized. We found that the state-of-the-art diarization model gave imperfect diarizations during our pilot annotation process. It is because the audio data we collected consists of overlapping audio, i.e., both speakers are speaking simultaneously, and the transcription model returns text for both speakers. Hence, we added another annotation termed "diarization continuation," and the annotators were tasked to fix the diarization issues along with discourse relation annotation.

The RST and SDRT theories (Mann and Thompson, 1988; Asher and Lascarides, 2005) define clauses as the textual span to be used as elementary discourse units (EDU). However, due to the nature of **CoMuMDR** and the imperfect diarizations resulting from the same, we could not use off-the-shelf clause identification algorithms. Hence, our annotation effort also includes the manual identification of EDUs and discourse relation annotation. It led to annotator-level differences in selecting clause spans. Hence, we report different annotation metrics in Appendix C.

Ethical Considerations

CoMuMDR is constructed by obtaining audio conversation data from customer call center offices. The data is obtained under the agreement between us and the research collaborator (call center company). All the data that was used for experimentation complies with the terms of use and licensing agreements.

The audio transcriptions in **CoMuMDR** are anonymized for of all personally identifiable information. We also removed instances of toxic language, offensive or harmful content, and sensitive or wrong information from **CoMuMDR**.

CoMuMDR consists of Hindi-English code-mixed conversations taken from a specific geographical section. The data contains conversations from companies in pharmaceutical, e-commerce, stock broker applications, e-marketplaces, and education counseling services.

We made sure to remove any bias in the data. Any bias, toxic language, offensive or harmful content, sensitive information, and misinformation in **CoMuMDR** is entirely unintentional.

Due to licensing agreements and ethical constraints, we will not be releasing the original audio data in **CoMuMDR**. We will only release the anonymized text transcriptions, corresponding text embeddings and audio features along with appropriate annotations in **CoMuMDR**.

References

Stergos Afantenos, Eric Kow, Nicholas Asher, and Jérémy Perret. 2015. Discourse parsing for multi-party chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 928–937, Lisbon, Portugal. Association for Computational Linguistics.

Nicholas Asher, Julie Hunter, Mathieu Morey, Benamara Farah, and Stergos Afantenos. 2016. Discourse structure and dialogue acts in multiparty dialogue: the STAC corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2721–2727, Portorož, Slovenia. European Language Resources Association (ELRA).

Nicholas Asher and Alex Lascarides. 2005. *Logics of Conversation*. Cambridge University Press, Cambridge, England, UK.

Jiaao Chen and Diyi Yang. 2023. Controllable conversation generation with conversation structures via diffusion models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7238–7251, Toronto, Canada. Association for Computational Linguistics.

Sanyuan Chen, Chengyi Wang, Zhengyang Chen, Yu Wu, Shujie Liu, Zhuo Chen, Jinyu Li, Naoyuki Kanda, Takuya Yoshioka, Xiong Xiao, Jian Wu, Long

Zhou, Shuo Ren, Yanmin Qian, Yao Qian, Jian Wu, Michael Zeng, Xiangzhan Yu, and Furu Wei. 2022. Wavlm: Large-scale self-supervised pre-training for full stack speech processing. *IEEE Journal of Selected Topics in Signal Processing*, 16(6):1505–1518.

Alexander Chernyavskiy and Dmitry Ilvovsky. 2023. Transformer-based multi-party conversation generation using dialogue discourse acts planning. In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 519–529, Prague, Czechia. Association for Computational Linguistics.

Ta-Chung Chi and Alexander Rudnicky. 2022. Structured dialogue discourse parsing. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 325–335, Edinburgh, UK. Association for Computational Linguistics.

Sabit Hassan and Malihe Alikhani. 2023. DisCGen: A framework for discourse-informed counterspeech generation. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 420–429, Nusa Dua, Bali. Association for Computational Linguistics.

Kenneth Heafield. 2011. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 187–197, Edinburgh, Scotland. Association for Computational Linguistics.

Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The INCEpTION platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th International Conference on Computational Linguistics: System Demonstrations*, pages 5–9, Santa Fe, New Mexico. Association for Computational Linguistics.

Nithin Rao Koluguri, Taejin Park, and Boris Ginsburg. 2021. Titanet: Neural model for speaker representation with 1d depth-wise separable convolutions and global context. *Preprint*, arXiv:2110.04410.

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. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2642–2652, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jiaqi Li, Ming Liu, Bing Qin, and Ting Liu. 2022. A survey of discourse parsing. *Frontiers of Computer Science*, 16(5):165329.

Jiaqi Li, Ming Liu, Bing Qin, Zihao Zheng, and Ting Liu. 2019. An annotation scheme of a large-scale multi-party dialogues dataset for discourse parsing and machine comprehension. *Preprint*, arXiv:1911.03514.

Junyi Jessy Li, Kapil Thadani, and Amanda Stent. 2016. The role of discourse units in near-extractive summarization. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 137–147, Los Angeles. Association for Computational Linguistics.

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. *CoRR*, abs/1907.11692.

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.

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2021. DMRST: A joint framework for document-level multilingual RST discourse segmentation and parsing. In *Proceedings of the 2nd Workshop on Computational Approaches to Discourse*, pages 154–164, Punta Cana, Dominican Republic and Online. Association for Computational Linguistics.

Michal Lukasik, Boris Dadachev, Kishore Papineni, and Gonçalo Simões. 2020. Text segmentation by cross segment attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4707–4716, Online. Association for Computational Linguistics.

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

Philippe Muller, Stergos Afantenos, Pascal Denis, and Nicholas Asher. 2012. Constrained decoding for text-level discourse parsing. In *Proceedings of COLING 2012*, pages 1883–1900, Mumbai, India. The COLING 2012 Organizing Committee.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Alan Lee Rashmi Prasad, Bonnie Webber and Aravind Joshi. 2019. Penn Discourse Treebank Version 3.0 - Linguistic Data Consortium.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Zhouxing Shi and Minlie Huang. 2019. A deep sequential model for discourse parsing on multi-party dialogues. In *AAAI 2019*, AAAI'19/IAAI'19/EAAI'19. AAAI Press.

Tushar Verma, Atul Shree, and Ashutosh Modi. 2023. Asr for low resource and multilingual noisy code-mixed speech. In *INTERSPEECH 2023*, pages 3242–3246.

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.

Yizhong Wang, Sujian Li, and Jingfeng Yang. 2018. Toward fast and accurate neural discourse segmentation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 962–967, Brussels, Belgium. Association for Computational Linguistics.

Chang-Sung Yu. 1986. Graph theory, by w. t. tutte, encyclopedia of mathematics and its applications, volume 21, addison-wesley publishing company, menlo park, ca., 1984, 333 pp. price: 45.00. *Networks*, 16(1):107–108.

Nan Yu, Guohong Fu, and Min Zhang. 2022. Speaker-aware discourse parsing on multi-party dialogues. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 5372–5382, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Duzhen Zhang, Feilong Chen, and Xiuyi Chen. 2023. DualGATs: Dual graph attention networks for emotion recognition in conversations. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7395–7408, Toronto, Canada. Association for Computational Linguistics.

Appendix

Table of Contents

List of Tables

List of Figures

A Related Work

A.1 Discourse Parsing Theories

There are two prominent theories around discourse parsing and structures. The RST theory (Mann and Thompson, 1988) defines EDUs as clauses (made of subject, object, and predicate). EDUs are then linked to form a discourse tree. The Penn Discourse Treebank developed a parser to divide a text corpus into EDUs and establish relationships between them using the grammar from RST (Rashmi Prasad and Joshi, 2019). The Semantic Discourse Representation Theory (SDRT) realizes the need for discourse in AI-based tools dealing with discourse (Asher and Lascarides, 2005). SDRT defines the theoretical background of discourse relations. The relationship is driven by dynamic logical semantics and a discourse structure.

A.2 Other Corpora

STAC (Asher et al., 2016): The STAC corpus is built on the online game of "Settlers of Catan". The game revolves around multiple players with dynamic resources to play and survive on a newly occupied land. Participants interact with each other on a chat system. The interaction includes gameplay interactions and general conversations. Hence, one can replay the entire game by noting the chat interactions. The STAC corpus is built on the recordings of the chat interface and hence includes gameplay-related interactions and general conversations. Asher et al. (2016) used the SDRT discourse theory to annotate 17 relation types between EDUs.

Molweni (Li et al., 2020): The Molweni dataset is based on Ubuntu support chat. This is a multiparty chat environment and is domain-specific. The annotation is based on the SDRT discourse theory and contains 17 relation types between EDUs.

Table 1 compares the STAC and Molweni datasets with our proposed dataset. `CoMuMDR` is built by transcribing audio call interactions between a customer and a call center representative. We sourced data from multiple customer call centers catering to domains, including e-commerce, pharmaceutical, stock broker application support, e-marketplace, and education counseling. On the other hand, STAC and Molweni datasets consist of single domains, namely Catan conversations and Ubuntu support. `CoMuMDR` is built from Hindi-English code-mixed audio conversations with imperfect transcription and diarization quality, imposing a practical outlook on discourse parsing in conversations.

A.3 Previous Methods

Deep Sequential (Shi and Huang, 2019): develops non-structured and structured EDU representations for jointly optimizing link prediction and relation classification. The model sequentially predicts the link and classifies relations for each EDU in a dialog. Glove embeddings are taken for tokens in the EDU and used for downstream models.

Hierarchical (Liu and Chen, 2021): The authors employ a hierarchical text embeddings approach by first encoding the text using a transformer followed by a BiGRU layer to compute EDU representations. Links are predicted by concatenating the representations of an EDU with all the previous EDUs and passing them through a linear layer. A discourse relation is classified by concatenating the representations of two connected EDUs. The authors experiment on STAC and Molweni datasets and highlight a need for domain adaptive models. Since STAC and Molweni are single-domain datasets, they are ineffective in training a model for cross-domain discourse parsing.

Structure-aware (Wang et al., 2021) jointly optimizes link and relation prediction. The EDUs are passed through a Heirarchial GRU to obtain context-aware dialog-level embeddings. This is then passed through a GNN containing a structure-aware dot product attention module to compute relation embeddings. As a discourse graph is a DAG, the relation embeddings here are computed for the forward and backward directions. These relation embeddings are then used for link prediction and relation classification.

SSP-BERT+SCIJE (Yu et al., 2022): The authors fintune a BERT model to predict if 2 EDUs have the same speaker, which is termed as SSP-BERT. The model then concatenates the embeddings of different speakers and the same speaker using a standard BERT and SSP-BERT model to predict links and classify discourse relation labels jointly.

SDDP (Chi and Rudnicky, 2022): This model jointly optimizes link and relation prediction on tree-level distributions. They discard a fraction of the edges to convert the discourse graph from a directed-acyclic graph (DAG) to a minimum spanning tree (MST) to efficiently learn and decode the discourse structure. The discourse tree is learned by minimizing the KL divergence between the predicted and reference tree distributions. The probability distribution of the tree is calculated by computing a tree's score and dividing it by the score of all possible tree structures, i.e., the partition function. The partition function is approximated using the Matrix-Tree theorem (Yu, 1986).

B Corpus Creation

B.1 Automatic Speech Recognition (ASR)

Our ASR system leverages the WavLM model (Chen et al., 2022) to generate frame-level embeddings from 8 kHz audio data (Verma et al., 2023). For each 50ms frame, WavLM predicts character probabilities, which are decoded using a beam search algorithm to produce the transcript. To enhance transcription accuracy, we integrate KenLM (Heafield, 2011), a statistical language model that effectively handles the linguistic diversity of Indian code-mixed speech. The transcription process begins with a reduced character set based on Devanagari, which facilitates phonetic alignment and reduces transcription errors. Subsequently, this text is converted to the native language, where spoken words are mapped to their respective languages. Finally, the text undergoes a romanization process to ensure consistency and maintain the pronunciation of English words, enabling seamless handling of multilingual utterances (Verma et al., 2023).

B.2 Speaker Diarization

We adopt a tailored approach for speaker diarization, addressing both dual-channel and mono-channel audio scenarios. In dual-channel diarization, each speaker's voice is recorded on a separate channel, and timestamps are assigned to speakers, prioritizing the high-energy speaker in overlapping segments. For mono-channel audio, we employ a clustering-based method using Titanet (Koluguri et al., 2021) to generate embeddings for fixed-length audio windows. By comparing these embeddings with the agent's pre-existing voiceprint, we accurately attribute speech segments to either the agent or the customer.

C Annotation details

C.1 Annotation Software

We used the inception software (Klie et al., 2018) to annotate **CoMuMDR**. The software provided the annotators a platform to select the text spans corresponding to an EDU, establish a link between two EDUs, and annotate a relation label for the link. The platform also displayed the description of each annotation label during annotation to keep reminding them of its definition.

C.2 Annotator Profiles and Payment

The annotators were hired as freelance employees to annotate 20 batches of data for a fixed payment of \$ 1,179.13. Each batch consists of 50 dialogues and consumes 5 hours per annotator. Hence, the annotators were paid \$ 11.79 per hour or \$ 0.60 per dialogue.

The annotators had previous experience annotating conversation data for various domains, including the domains covered in **CoMuMDR**. They are proficient in reading, speaking, and listening to English and Hindi and use both languages in a code-mixed style in everyday communication.

C.3 Annotation Instructions

The annotators were given the below instructions to annotate their batch:

- Dialogue Overview
  - Each dialogue consists of approximately 10 utterances.
  - An utterance is a sequence of phrases, with each phrase separated by punctuation marks.
- Span Identification
  - A span may consist of an entire utterance or one or more phrases within an utterance.
  - Carefully identify spans where a relation might be possible with another span in the dialogue.
- Relation Creation
  - Once relevant spans are identified, create a relational edge between these spans.
  - Select the appropriate label from the defined relation types to describe the connection.
- Edge Constraints
  - No back edges should be created, meaning edges should only flow forward in the dialogue.
- Special Instructions on Acknowledgement vs. Question-Answer Pair
  - Acknowledgement is used for statements that function as conversation continuators, indicating understanding.
  - If an utterance is framed as a question, even if the reply is a simple continuator (e.g., "hmm," "okay," "I see"), the relation should be labeled as Question-Answer Pair rather than Acknowledgement.
- By following these steps, you will ensure consistent and accurate annotations across the dialogues. Read the entire dialogue first, identify

Algorithm 1 Span exact match

```
Require: List of spans A, B
procedure COUNTEXACTMATCHES(A, B)
    ExactCount ← 0
    for a ∈ A do
        if a ∈ B then
            ExactCount ← ExactCount + 1
        end if
    end for
    return ExactCount
end procedure
```

Algorithm 2 Span partial match

```
Require: List of spans A, B
procedure COUNTPARTIALMATCHES(A, B, threshold)
    PartialCount ← 0
    for a ∈ A do
        BestMatchScore ← max_{b∈B} Jaccard(a, b)
        if BestMatchScore ≥ threshold then
            PartialCount ← PartialCount + 1
        end if
    end for
    return PartialCount
end procedure
```

potential relations, mark the spans, and then apply the relevant relation edge labels.

The annotators were also given the list of relation labels, their definitions, and appropriate examples as listed in Table 5.

C.4 Annotation Process

A two-party dialogue consists of a list of utterances spoken by two speakers. An utterance is a continuous set of words spoken by a speaker, which may include multiple sentences. The annotators identified elementary discourse units (EDUs) from the utterances for discourse linking and relation labeling. We used clauses as the EDUs based on the definition in Segmented Discourse Relation Theory (SDRT) (Asher and Lascarides, 2005).

C.5 Inter Annotator Agreement Metrics

Table 2 highlights the inter-annotator metrics that we define in Algorithms 1 and 2. We did not rely on off-the-shelf models and algorithms to segment the text into EDUs because of the nature of **CoMuMDR**. It consists of overlapping utterances and imperfect diarizations, which caused segmentation models to split a potentially single EDU into two parts. The annotators were tasked to select the EDU span, build links between EDUs, and classify relation labels. Thus, we calculated the Kappa inter-annotator agreement based on the overlap between the selected spans of each annotator and the links and relation types between EDUs.

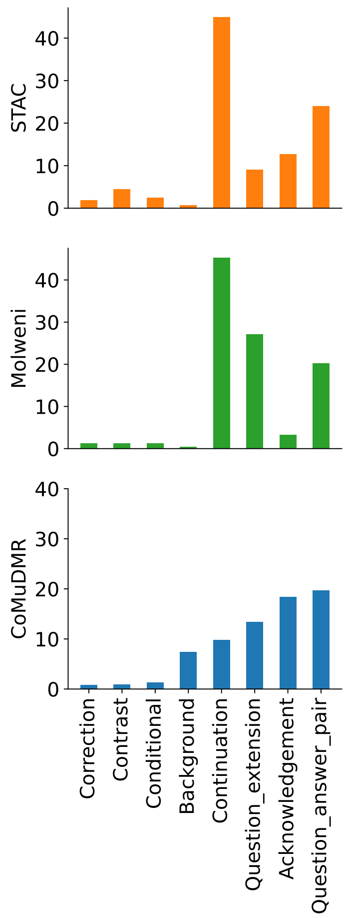

Figure 3: Distribution of the discourse relation labels for STAC and Molweni datasets. In this plot, we have combined the labels based on our labeling strategy mentioned in 5.

D Model Training Details

We used the same hyperparameter settings as mentioned in the model papers. All the experiments were carried out on a Nvidia 3090 GPU. We mentioned the relevant hyperparameters in Table 6.

E Evaluation Metric

We compute link prediction as a binary classification task between two EDUs. If a link is present in the gold annotations and prediction, it is a `True positive` link. Similarly, if a link is predicted between two EDUs but is not in the gold annotation, it is a `False positive` link. Using these definitions, we construct the confusion matrix and calculate the F1-score for link prediction.

A relation r_{ji} between two EDUs (u_j, u_i) is classified only if the model predicts a link between (u_j, u_i). Hence, we first find all the intersecting links between the gold annotated data and predicted links, i.e., $\forall j, i$ if there is a link u_j and u_i in gold

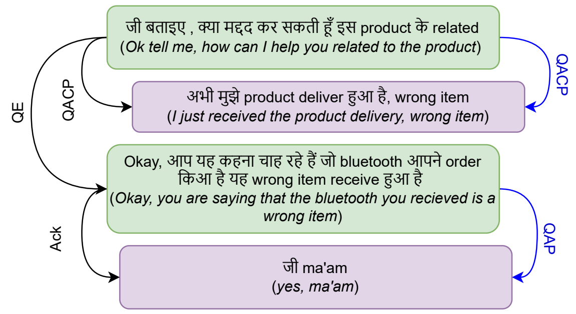

Figure 4: A sample conversation taken from **CoMuMDR**. Utterances from the customer are marked in purple, and those of the customer center representative are green. The gold and predicted relations are marked on the left and right sides.

and predicted data then capture the gold and predicted relations (r_{ji}, r'_{ji}). We calculate the link + relation F1-score by using the pairs of gold and predicted relations.

F GPT-4 Template

We experimented with using GPT-4 for discourse parsing on STAC, Molweni and **CoMuMDR**. We used the prompt template mentioned in Figure 5.

G Distance between linked EDUs

Fig. 6 (and Table 7) shows the distribution of relative distance between linked EDU pairs for each relationship type. We merged the statistics of the merged relations (mentioned in Table 5 for STAC and Molweni. We observe a significant distribution overlap between STAC and Molweni datasets for “Correction”, “Question Extension”, “Acknowledgement” and “Question_answer_pair” suggesting their relative similarity. However, for “Conditional”, “Continuation” and “Contrast” there is a difference in the distributions. We also plot the same for **CoMuMDR**. We notice a significant difference in the distributions; notably that most of the relations have a distance of 1. We also look at the mean and standard-deviation of the relation distances in Table 7. The median distance between linked EDUs for all relations is 1 in all the datasets.

```
You are given a dialogue conversation
between an agent and a customer. You
have to do the link and relation
prediction using SDRT format.  You
will be given the relations and you
have to stricty use those relations
only to do the prediction. You will
be given the nodes as well in the
form of extracted text spans. During
link prediction, you have to identify
between which nodes there exists a
link and what would be the relation.

you have to return the answer
in the SDRT format like json. Do not
return any extra text or explanation.

Dialogue:
{dia}

Spans:
{spans}

relations:
{rels}

Following is just an example of
annotation:
{examples}

Note: For all the instances where a
sentence spoken by the same person
is broken down into multiple lines,
then use dia-continuation relation.
```

Figure 5: Prompt template used for evaluating GPT-4 as a discourse parser.

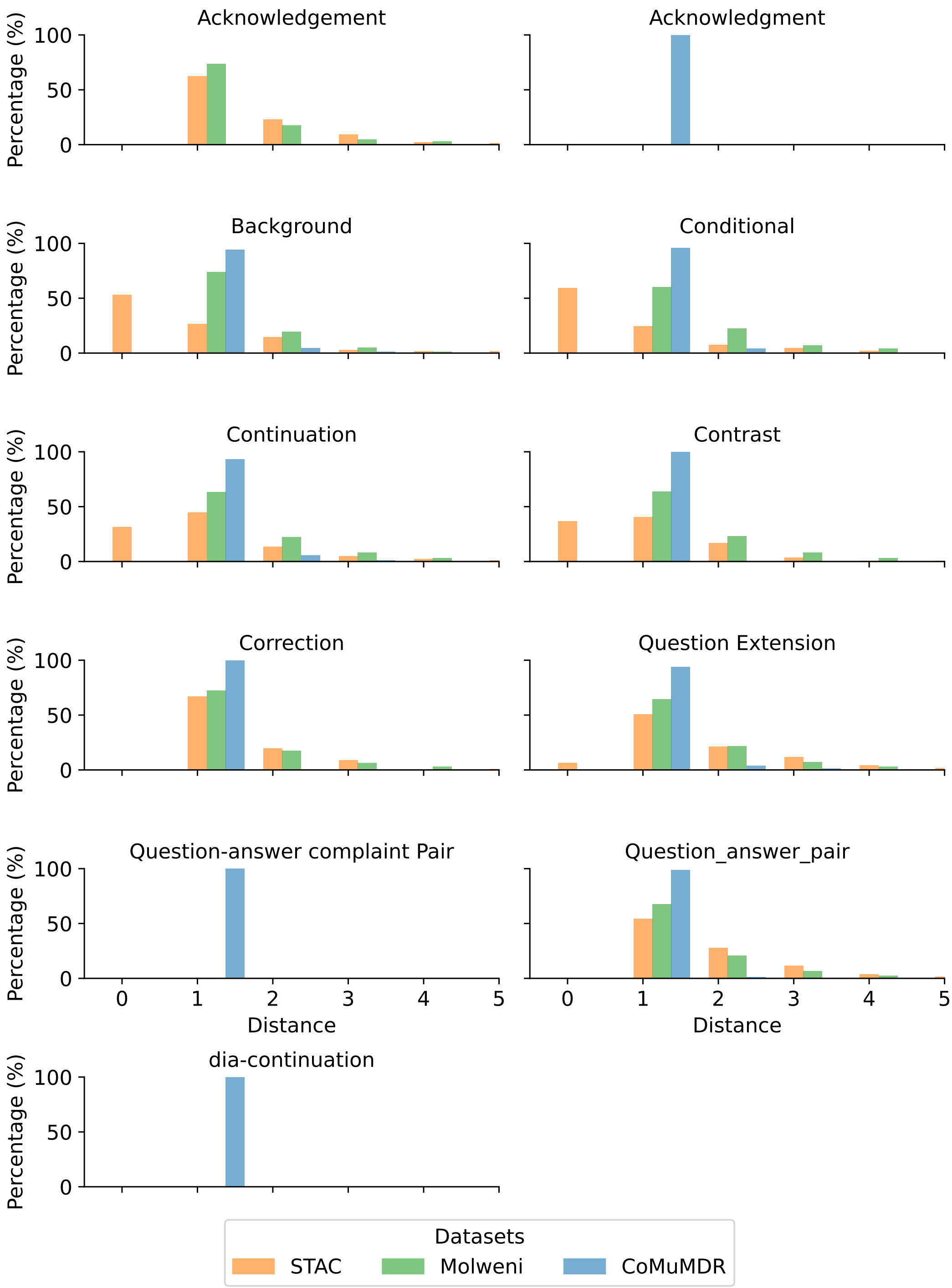

Figure 6: Distance between linked EDUs for different corpora

Discourse Label	Description	Example
Acknowledgment	The tail clause is an agreement or disagreement to the head clause	जी नाम confirm करने के लिए धन्यवाद (*Ok, thank you for confirming your name*)
Question-Answer Pair	The tail is an answer clause to the question in the head clause	मैं आपकी किस प्रकार सहायता कर सकता हूँ → यह मेरा return के regarding call है *How can I help you? → This is a call regarding my return*
Question-Answer Complaint Pair	Similar to the Question-Answer Pair, however, the head clause is a customer complaint question	Sixth मुझे last time दिखा रहा था लेकिन अब ninth दिखा रहा → हाँ सर, मेने high priority issue raise कर दिया है (*It was showing me on the sixth, now it is showing ninth → Yes sir, I have raised a high priority issue*)
Background	The tail provides supplementary context or information to the subject or object in the head clause. The subject or object in the head clause is the main topic of discussion in the dialogue	इस विषय में आपने already issue highlight किया है → 29th October की date में ही issue highlight हुआ है, तोह system में show हो रहा है (*You have highlighted an issue regarding this → The systems shows a issue highlighted on 29th October*)
Contrast	The tail highlights a difference between the subject, predicate, and object interaction in the head clause	यह complaint आप कर सकते हो या मुझे online करनी होगी → आपको करनी पड़ेगी (*Can you raise the complaint or do I have to do it online? → You'll have to do it*)
Correction	The tail clause is a correction or refinement of the head clause	आपके headphone ख़राब है → नहीं, deliver नहीं हुए (*Your headphones are broken → No, headphones are not delivered*)
Question extension (Clarification Question, Question elaboration)	The tail and head are question clauses from the same speaker. The tail enquires more details, seeks clarity, or elaborates on the head clause with option choices.	You are receiving complete wrong item right? → Pickup address will be same?
Conditional (Alternation, Conditional)	The tail provides choices for the actions dictated in the head or sets up a situation that affects the head clause.	"Either we go now, or we wait for tomorrow" "If it rains, we'll stay inside"
Continuation (Comment, continuation, elaboration, parallel, result)	The tail adds a remark, extends or elaborates, clarifies, adds related information, or shows the outcome of a previous action	सुबह आया था पहले message की वोह पिच्कुप् के लिए निकल चूका है agent → और फिर मेरे पास कुछ देर बाद second message आया की किसी unavoidable event की वजह से pickup नहीं हो पायेगा (*I got a message in the morning that the agent has left for receiving the pickup → Then I got a message saying that the pickup cannot be completed due to an unavoidable event*)

Table 5: Discourse relation labels and their descriptions. We use a subset of the labels presented in the STAC corpus and add another label, "Question answer Complaint Pair" to capture a specific case in customer center data. The annotators were given these descriptions and examples during the annotation process. In the first column, we highlight the combined discourse labels for annotating the dataset within parenthesis.

Model	**Optimizer**	**learning-rate**	**lr-decay**	**epochs**	**batch size**
Deep Sequential	AdamW	1e-1	0.98	50	4
Hierarchical	AdamW	2e-4	1.00	20	1
Structure-aware	SGD	1e-1	0.98	10	1
SSP-BERT SCIJE	Adam	1e-3	0.75	100	4
SDDP	AdamW	2e-5	1e-8	3	4

Table 6: Hyperparameter settings used to experiment all the discourse parsing models on STAC, Molweni, and **CoMuMDR** datasets.

Relation	**STAC**	**Molweni**	**CoMuMDR**
Continuation	1.17 ∓ 1.53	1.65 ∓ 1.14	1.06 ∓ 0.42
Question_answer_pair	1.78 ∓ 1.20	1.56 ∓ 1.09	0.99 ∓ 0.27
Acknowledgement	1.67 ∓ 1.31	1.41 ∓ 0.81	0.95 ∓ 0.32
Background	0.72 ∓ 1.14	1.35 ∓ 0.66	1.07 ∓ 0.39
Correction	1.67 ∓ 1.84	1.41 ∓ 0.77	1.00 ∓ 0.00
Question Extension	1.86 ∓ 1.86	1.62 ∓ 1.12	1.05 ∓ 0.48
Conditional	0.67 ∓ 1.35	1.78 ∓ 1.33	1.03 ∓ 0.33
Contrast	0.97 ∓ 1.15	1.60 ∓ 1.05	0.98 ∓ 0.19
Question-answer complaint Pair	-	-	1.21 ∓ 0.54
dia-continuation	-	-	0.97 ∓ 0.26

Table 7: Mean and standard deviation of distribution of distance between linked EDUs for all corpus

	Acknowledgement	**dc**	**QAP**	**QACP**	**QE**	**Correction**	**Continuation**	**Conditional**	**Background**	**Contrast**
Acknowledgement	59	28	3	0	0	0	3	0	3	1
dc	20	70	10	1	2	4	27	0	7	1
QAP	28	17	42	0	1	6	15	1	0	0
QACP	1	0	1	0	0	0	0	0	0	0
QE	11	20	15	1	14	3	28	3	3	2
Correction	0	0	0	0	0	1	0	0	0	0
Continuation	9	20	1	0	4	2	26	1	4	3
Conditional	0	2	0	0	0	0	1	1	1	0
Background	2	9	0	0	0	0	7	0	3	0
Contrast	0	1	1	0	0	3	2	0	0	0

Table 8: Confusion matrix of discourse link+relation classification done by GPT-4o. We have turned some relations into their relevant acronyms for viewing: QACP-question answer complaint pair, QAP-question answer pair, QE-question extension, and dc-diarization continuation.