Cross-Framework Discourse Relation Classification Though Unifying Dimensions

Anonymous ACL submission

Abstract

Existing discourse corpora annotated under different frameworks adopt distinct but somewhat related taxonomies of relations. The integration of these corpora has been an open research question. Previous studies on the interoperability of different discourse formalisms are mainly theoretical, although such research is performed with the hope of benefiting computational applications. In this paper, we show how the unifying dimensions (UDims) that originate from the Cognitive approach to Coherence Relations (CCR) (Sanders et al., 2018) can facilitate cross-framework discourse relation (DR) classification. To address the challenges of using predicted UDims for DR classification in model learning, we adopt the Bayesian learning framework based on Monte Carlo dropout (Gal and Ghahramani, 2016) to obtain more robust predictions. Data augmentation enabled by our proposed method yields strong performance. We compare different possible models and analyze the experimental results from different perspectives.

1 Introduction

Discourse coherence relates to the way that a monologue or dialogue is organized so that it is a coherent entity, instead of a random collection of clauses or sentences. As such, coherence represents an important aspect of text quality. Various studies have shown the benefits of incorporating discourse-level information or coherence-related training objectives in NLP tasks, such as text generation (Bosselut et al., 2018), language modelling (Iter et al., 2020; Lee et al., 2020; Stevens-Guille et al., 2022), and summarization (Xu et al., 2020).

Discourse-level analysis is typically concerned with discourse relations (Rutherford and Xue, 2015). These relations describe the link with which two textual segments are associated with each other and they form an integral part in discourse modelling frameworks, such as the Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) and the Penn Discourse Treebank (PDTB) (Prasad et al., 2008, 2018). However, RST and PDTB focus on different aspects of discourse coherence and adopt distinctive approaches of discourse modelling (Fu, 2022). As discourse annotation is a demanding task and different discourse modelling frameworks provide distinctive but not incompatible perspectives of discourse phenomena, the interoperability and integration of different discourse modelling frameworks has been a topic of interest for a long time (Bunt and Prasad, 2016; Benamara and Taboada, 2015; Sanders et al., 2018; Chiarcos, 2014).

Most of the studies are theoretical, although it is believed that a good way to test the usefulness of the proposed methods is to merge different corpora based on the methods and apply the data in computational experiments to see whether the increased size of the training data improves the results (Benamara and Taboada, 2015). Demberg et al. (2019) try to validate several existing proposals for integrating discourse corpora against annotated data. One of their research purposes is to enable joint usage of discourse corpora annotated under different frameworks for computational purpose. Nevertheless, results of this strand of research find little computational application.

The UniDim proposal (Sanders et al., 2018), which originate from the Cognitive approach to Coherence Relations (CCR) (Sanders et al., 1992, 1993), is shown to be relatively successful in mapping between PDTB and RST relations (Demberg et al., 2019). Moreover, previous studies (Roze et al., 2019; Fu, 2023) demonstrate the possibility of automatically predicting and incorporating such dimensions in discourse relation (abbreviated as "DR" in the following) classification tasks. Therefore, in this paper, we try to apply the unifying dimensions (abbreviated as "UDims" in the following) in the UniDim proposal for RST and PDTB

DR classification tasks.

As not a small number of UDims are involved and the classification errors of each UDim may propagate to DR classification, the combination of predicted UDims poses additional challenges for model learning. We show that the Bayesian learning approach based on Monte Carlo (MC) dropout (Gal and Ghahramani, 2016) is a viable option for solving this problem. Meanwhile, our approach of utilizing UDims allows data augmentation with cross-framework discourse corpora. Thus, we present results on data augmentation with different types of data. We then analyze different model designs based on the correlation between training losses of the DR classification task and the UDim prediction tasks. Lastly, we show the correlation between specific DR classes and the UDims to provide another perspective of analysis.

2 Related Work

2.1 DR Classification

Early studies on RST and PDTB DR classification focus on feature extraction (Feng and Hirst, 2012; Joty et al., 2012; Lin et al., 2009; Pitler et al., 2009) or representation learning (Ji and Eisenstein, 2014, 2015). With the advancement of contextualized embeddings, an increasing number of studies try to model the relationship between argument representations with attention mechanisms (Guo et al., 2018; Li et al., 2016).

Discourse marker prediction is considered a potentially effective auxiliary task for both RST DR classification (Yu et al., 2022) and PDTB DR classification (Shi and Demberg, 2019; Jiang et al., 2021; Liu and Strube, 2023). However, as RST does not make a clear distinction between implicit DRs and explicit DRs in the annotation, this approach is primarily applied to PDTB implicit DR classification.

To combat the problem of limited amounts of training data for RST parsing, Braud et al. (2016) utilize multi-task learning to benefit from supervision of related tasks such as PDTB DR classification. As RST elementary discourse units (EDUs) and PDTB arguments are determined based on different criteria, they have to make adjustments to PDTB data and use sentences rather than manually annotated arguments in their experiments and ignore intra-sentential PDTB relations. Multi-task learning is also adopted in Liu et al. (2016) for PDTB implicit DR classification, where RST DR classification is treated as an auxiliary task. It shows that RST DR classification improves performance on the classification of some PDTB Level-1 implicit DRs.

2.2 The UniDim Proposal and UDims

Sanders et al. (2018) propose a set of unifying dimensions as an interface for different annotation frameworks to be related with each other. These UDims originate from four cognitive primitives—*basic operations*, *source of coherence*, *order of segments* (called *implication order* in Sanders et al. (2018)) and *polarity*, which are used to define coherence relations in Sanders et al. (1992). To make the taxonomy more expressive, additional dimensions are added, including *temporality*, and *specificity*, *lists* and *alternatives* for additive relations, and *conditionals* and *goal-orientedness* for causal relations. Each of these dimensions has a number of possible values, for instance, the *polarity* dimension has distinctions between *positive*, *negative* or *under-specified*. We refer those interested to Sanders et al. (2018) for a better understanding of the meaning of the UDims. With those UDims, DRs from different annotation frameworks can be decomposed and compared.

Demberg et al. (2019) propose a method for mapping RST and PDTB, and the results of their data-driven investigation exhibit higher consistency with the results obtained with the UniDim proposal, in comparison with the OLiA reference model (Chiarcos, 2014) and the ISO standard proposal (Bunt and Prasad, 2016).

To our best knowledge, the method proposed in Roze et al. (2019) represents the first study on using UDims for DR classification. Fu (2023) reports results of using UDims for cross-framework DR classification. However, their experiments are aimed at testing the effectiveness of the UniDim proposal with computational experiments. The pipeline approach adopted by Roze et al. (2019) achieves no improvement over the baseline for PDTB implicit DR classification, and the high performance shown by Fu (2023) relies on gold UDim values, which are not accessible during inference time in realistic settings.

3 Our Method

Our experimental settings are similar to the those described in Fu (2023), with the exception of using predicted UDims during inference time. Therefore, two tasks are involved: a) UDim prediction and b) DR classification. We follow the rule-based method in Fu (2023) to obtain gold UDim values

for each of the training examples.

For an input sequence X_i in a dataset with size N, i.e., $\{X_i\}_{i=1}^{N}$, X_i is formed by a pair of arguments of lengths m and n, respectively, i.e., $X_i = A_1^{(1)} \ldots A_m^{(1)}, A_1^{(2)} \ldots A_n^{(2)}$. We use a pre-trained language model as the input encoder f_{Enc}. Special tokens are to be inserted based on the requirements of the chosen encoder, and X_i is typically padded to a fixed length. In our experiments, the two arguments are padded separately at the ends. After such preprocessing, the representation of the input sequence, which is denoted as $\widetilde{X_i}$, can be obtained from the encoder:

$$h = f_{Enc}(\widetilde{X_i}) \tag{1}$$

3.1 UDim Prediction

A three-layer feed-forward network g, comprising a fully connected layer, a LeakyReLU activation function, followed by a dropout layer, is applied to transform h to a lower dimensional space:

$$h_{UDim} = g(h) \tag{2}$$

UDims are not independent. For example, the *implication order* dimension is associated with the *basic operation* dimension, because the former only describes properties of causal relations, which form a sub-category under *basic operation*. Inspired by Gerych et al. (2021) and Roze et al. (2019), we exploit knowledge about the relationships between UDims to improve the performance on this task. For instance, in the example, the embedding vector of the predicted *basic operation* $E(\hat{y}_{bop})$ will be passed as features to the classifier f_{impl} for *implication order*:

$$\tilde{y}_{impl} = softmax(f_{impl}(h_{UDim} \oplus E(\hat{y}_{bop}))) \tag{3}$$

An argmax function is typically required to obtain a discrete value from the predicted probability distribution, so that $E(\hat{y}_{UDim})$ can be obtained and passed as features to the classification of another related UDim or DR. However, this operation is non-differentiable and the training signal of one UDim cannot backpropagate to the training of the related UDims or from DRs to UDims when predicted UDims are used as features for DR classification. Therefore, we adopt the Gumbel-Softmax function (Jang et al., 2016), which is a differentiable approximation to the argmax function:

$$y_i = \frac{\exp((\log(p_i) + g_i)/\tau)}{\sum_{j=1}^{k} \exp((\log(p_j) + g_j)/\tau)} \tag{4}$$

where p_i represents a class probability for a categorical variable with k possible outcomes. $g_i \ldots g_k$ are i.i.d samples drawn from a Gumbel(0, 1) distribution, which can be sampled by drawing $\mu \sim$ Uniform(0, 1) and $g = -\log(-\log(\mu))$.

3.2 DR Classification

Similar to UDim prediction, a three-layer feed-forward network ϕ is applied to h first:

$$h_{DR} = \phi(h) \tag{5}$$

We experiment with four ways of leveraging UDims in the DR classification task:

1. *TrainonGoldTestonPred*: During training, gold UDims are used and their embeddings are concatenated with h_{DR} for DR classification, so that the model learns the relationship between the input and the UDims and the target labels. During inference time, the embeddings of the predicted UDims are used.

2. *InputDimCat*: During both training and testing, the embeddings of predicted UDims are used by simple concatenation with h_{DR}.

3. *InputDimAtt*: During both training and testing, the embeddings of predicted UDims are combined with h_{DR} by an attention mechanism based on scaled dot product (Vaswani et al., 2017).

4. *InputForRelCls*: The hypothesis is that as the UDims are closely related to the target DRs, if the model takes UDim predictions as explicit training objectives, the performance on DR classification may be improved, even without using the embeddings of predicted UDims as features. Hence, only h_{DR} is used for DR classification, and UDim prediction and DR classification form a scenario of multi-task learning.

Preliminary experiments show that directly using predicted UDims causes a large performance drop for the DR classification task. The performance deterioration could be attributed to the utilization of *predicted* UDims, where the classification errors of these UDims might introduce noise, and combined usage of these predicted UDims may exacerbate data sparsity, hence amplifying uncertainty in the DR classification task. To address this challenge, we employ the MC dropout method.

3.3 MC Dropout

Dropout was originally proposed to reduce overfitting in model training while achieving the effect of training an ensemble model (Srivastava et al., 2014). During training, random neural units along with its incoming and outgoing connections are temporarily removed from the neural network. Thus, a model with n units can be seen as a collection of 2^n possible smaller neural networks, which are sampled and trained. During inference time, dropout is deactivated.

Specifically, for a model with l layers, the model weights ω can be expressed as a set of weight matrices for each layer: $\omega = \{\mathbf{W}_i\}_{i=1}^{l}$. If the model is trained with a dropout probability of p, the final weights will be $\omega' = \omega \times p$, yielding a single neural net. For a new input $\mathbf{x}$ at test time, the predicted $\hat{y}$ is obtained with:

$$\hat{y} = \underset{y'}{\operatorname{argmax}}\, p(y'|\mathbf{x}, \omega') \tag{6}$$

For uncertainty estimation, Bayesian networks represent a natural choice. For a model trained with Bayesian approach on input set $\mathbf{X}$ and corresponding target set $\mathbf{Y}$, the predictive distribution for a new input $\mathbf{x}$ is obtained with:

$$p(y'|\mathbf{x}, \mathbf{X}, \mathbf{Y}) = \int p(y'|\mathbf{x}, \omega) p(\omega|\mathbf{X}, \mathbf{Y}) d_\omega \tag{7}$$

where $p(\omega|\mathbf{X}, \mathbf{Y})$ is generally approximated with a computationally simpler distribution $q(\omega)$. The distribution over ω naturally encodes model uncertainty. However, these methods typically come with large computational costs, and for transformer-based models, the computation costs can be prohibitive.

Gal and Ghahramani (2016) introduces the MC dropout method to tackle the challenge of uncertainty estimation in deep neural networks. Different from the standard dropout method, dropout is activated during inference time. The MC dropout method represents a lightweight Bayesian approximation. For an input representation from the previous layer h_{i-1}, the output h_i of the i_{th} layer is determined with:

$$h_i = \sigma(h_{i-1}, \mathbf{W}_i, \mathbf{M}_i) \tag{8}$$

where $\mathbf{M}_i$ is a masking matrix, with its entries being sampled from a Bernoulli distribution, and the probability of being zero is the dropout probability p. σ denotes the activation function of this layer.

During inference, one can sample T sets of $\{\mathbf{M}_i\}_{i=1}^{l}$ for T stochastic forward passes and the mean predicted distribution is obtained by averaging over the T passes:

$$p(y'|\mathbf{x}, \mathbf{X}, \mathbf{Y}) = \frac{1}{N} \sum_{t=1}^{T} p(y'|\mathbf{x}, [\mathbf{W}_i^t, \mathbf{M}_i^t], \ldots, [\mathbf{W}_l^t, \mathbf{M}_l^t]) \tag{9}$$

The variance can be used as an indicator of model uncertainty. This method is similar to an ensemble of approximated functions with shared parameters (Choubineh et al., 2023) but without increasing computational complexity or sacrificing model performance.

As indicated in Shelmanov et al. (2021), applying the MC dropout to all the dropout layers of a transformer model yields better performance on uncertainty estimation. Even though our focus is not uncertainty estimation but to obtain more robust predictions, this approach of applying the MC dropout method may better approximate an ensemble model and we use mean predictive distribution over multiple runs for UDim and DR classification.

3.4 Data Augmentation

Although RST and PDTB follow different criteria for discourse unit segmentation, data from both frameworks can be used together for the UDim prediction task. Figure 1 shows the pipeline for the data augmentation method.

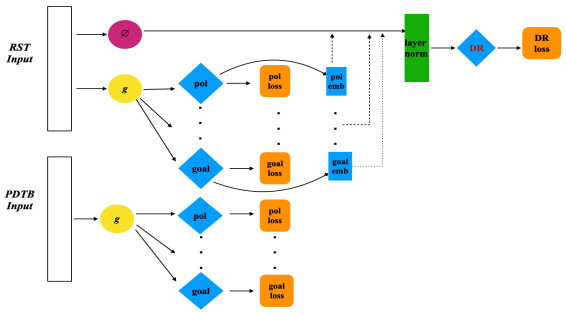

Figure 1: Pipeline for data augmentation with PDTB data and the final task is RST DR classification. *pol...goal* represent the UDims *polarity...goal-orientedness*. As we explore different ways of leveraging predicted UDims, the embeddings of the UDims are not necessarily fed as features to the DR classification task, hence represented with dashed arrow lines. The losses shown in orange boxes are to be minimized through model training.

In the example shown in Figure 1, for RST DR classification, PDTB data (explicit, implicit, or both) can be used for training the model on UDim prediction. Increased data amount and more diversified training data may increase model robustness in UDim prediction, which could improve model performance on DR classification.

3.5 Training

Cross-entropy loss is used for model training. To jointly train the model for UDim prediction and DR classification, we adopt a multi-task loss:

$$\mathcal{L}_{total} = \mathcal{L}_{UDims} + 2.0 * \mathcal{L}_{DR} \quad (10)$$

Note that there are multiple UDims involved in the experiments, even though the loss term shows them collectively as $\mathcal{L}_{UDims}$. In order to guide the model training towards DR classification, we increase the weight for DR classification loss.

4 Experiments

4.1 Data Preprocessing

The experiments on RST are carried out on the RST Discourse Treebank (RST-DT) (Carlson et al., 2001), which is the standard benchmark for RST parsing, consisting of 385 documents. We follow the gold division of the corpus for training and test sets and take 20% from the training set for validation. We utilize the preprocesing method in Ji and Eisenstein (2014) and binarize the RST trees in order to obtain pairs of discourse segments linked by DRs. The 78 relations are mapped to 18 broad classes based on the template in Braud et al. (2016), but as *Same-Unit* and *Attribution* are not covered in Sanders et al. (2018), the two relations are excluded in our experiments, leaving a set of 16 RST relations.

The experiments on PDTB are performed on PDTB 3.0 (Prasad et al., 2018), which is the latest version characterized by a new sense hierarchy and newly annotated intra-sentential implicit relations. We follow the data split used in Ji and Eisenstein (2015), i.e., sections 2-20 for training, sections 0-1 for validation, and sections 21-22 for testing, and discard DRs with fewer than 100 instances to alleviate data imbalance, as proposed in Kim et al. (2020), which leaves 14 senses from Level-2 (L2) of the sense hierarchy. Since PDTB explicit DR classification can be achieved with high performance, we only focus on implicit DR classification. Moreover, PDTB L2 senses are more fine-grained than Level-1 (L1), which makes them potentially more useful. Therefore, we focus on 14-way classification of PDTB L2 implicit DRs.

As PDTB annotation involves a much larger number of files from the Penn Treebank (Marcus et al., 1993)[1], to mitigate the confounding effect of data amount in our experiments on data augmentation with different types of data, we try to increase the data amount for RST by back-translating data from the training set (English->French->English, translated by Google Translate), thus doubling the training data amount for RST.

Following Fu (2023), we exclude the UDim *list*, and merge sub-categories under *specificity*, making *specificity* a binary property, similar to *alternative*, *conditional* and *goal-orientedness*, which is also the approach adopted in Roze et al. (2019).

UDims	Sub-Categories	Parents
polarity(*pol*)	NS, positive, negative	-
basic operation(*bop*)	NS, additive, causal	-
source of coherence(*soc*)	NS, objective, subjective	-
implication order(*imp*)	NS, NA, basic, non-basic	bop
temporality(*temp*)	NS, anti-chronological, chronological, synchronous	-
specificity(*spec*)	specificity, non-specificity	bop
alternative(*alt*)	alternative, non-alternative	bop
conditional(*con*)	conditional, non-conditional	bop
goal-orientedness(*goal*)	goal-oriented, non-goal-oriented	bop

Table 1: UDims used in the experiments. Their abbreviations used in the paper are shown in the brackets in italics. "-" in the last column suggests that no parent passing is performed for this UDim.

Table 1 shows the UDims, their abbreviations in the paper and the sets of sub-categories to predict from. The parent UDims described in section 3.1, which are passed as features for predicting the UDims, are shown in the last column.

4.2 Implementation Details

We use the pre-trained $\text{BERT}_{\text{BASE}}$ model (Devlin et al., 2019) and $\text{RoBERTa}_{\text{BASE}}$ model (Liu et al., 2019) from the Transformers library (Wolf et al., 2020) as the input encoder.

The embeddings of the UDims are derived from separate embedding layers, which are configured with learnable parameters, and the embedding vectors are initialized from uniform distributions.

Baseline The baseline is thus DR classification based on the input, without involving training and prediction of UDims. To ensure fair comparison, we also apply MC dropout to the baseline models, i.e., the pre-trained $\text{BERT}_{\text{BASE}}$ model and $\text{RoBERTa}_{\text{BASE}}$ model, and run the same number of passes to obtain the mean predictive distribution.

[1] 2159 files in total

Model	$F1_{BERT}$	Acc_{BERT}	$F1_{RoBERTa}$	$Acc_{RoBERTa}$
Baseline	50.24	62.30	53.72	65.56
TrainonGoldTestonPred	50.30	64.15	**55.21**	66.27
InputDimCat	51.02	64.36	54.49	66.16
InputDimAtt	49.78	61.64	54.65	66.27
InputForRelCls	51.56	62.73	54.89	**66.32**

Table 2: Results for RST DR classification.

Model	$F1_{BERT}$	Acc_{BERT}	$F1_{RoBERTa}$	$Acc_{RoBERTa}$
Baseline	45.53	56.42	52.36	60.47
TrainonGoldTestonPred	46.40	55.87	51.80	59.09
InputDimCat	44.71	54.77	52.82	**61.43**
InputDimAtt	44.74	54.56	52.93	60.67
InputForRelCls	45.02	54.02	**53.44**	60.26

Table 3: Results for PDTB implicit DR classification.

Hyper-parameters The arguments of the input sequences are padded to a fixed length of 250 tokens, and all the model parameters are initialized with the Xavier uniform initialization (Glorot and Bengio, 2010). The output size of the feed-forward networks g and ϕ described in section 3.1 and section 3.2 is set to 128 through manual tuning. The dropout probability is kept at 0.2 for all the experiments, which means that greater regularization is adopted than the pre-trained BERT and RoBERTa models. We keep all the dropout layers active during inference time, and run the model for UDim prediction three times and obtain the average predictive distributions. The UDim embeddings are set with a dimension size of 100 in all the experiments, except for *InputDimCat* in section 3.2, where the dimension sizes of the UDim embeddings are set to be 2 * number of subcategories, which we find sufficient through experimentation. Similarly, we also run the DR classifier three times and obtain the average predictive distribution. The batch size is set to the largest value that the GPU machine can accommodate.

The model learning rate is set to $1e-5$ and it is trained for a maximum of 30 epochs, with an early-stopping scheme monitoring performance improvement for DR classification on the validation set with a threshold of 7 epochs. The AdamW optimizer is used and a warmup ratio of 0.06 is set for the scheduler. A weight decay of 0.1 is applied, and gradients are clipped to a maximum of 1.0. The implementation is based on the PyTorch machine learning framework (Paszke et al., 2019). A single A5000 GPU with a capacity of 24GB is used for all the experiments.

5 Results

We select models based on their performance measured by F1 in DR classification, and thus, they do not necessarily perform the best in terms of accuracy.

5.1 DR Classification

Table 2 shows the results for RST DR classification.

As indicated in Table 2, much higher performance is achieved using RoBERTa than BERT. The best performance is achieved with *TrainonGoldTestonPred*, followed by *InputForRelCls*. In both cases, the predicted UDims are not used as features for DR classification during training. Compared with the baseline method, the models are trained for UDim prediction. The results support our hypothesis that the association between UDims and DRs can aid in the DR classification task.

Table 3 shows the results for PDTB implicit DR classification. Similar to the results on RST, results obtained with RoBERTa are much higher than BERT. However, a performance drop compared with the baseline is visible with the approach *TrainonGoldTestonPred*. As shown in Sanders et al. (2018, p.52, section 5.3), implicit relations pose a challenge for the UniDim proposal, and it is likely that model performance on UDim prediction trained on PDTB implicit DR data alone is not high, causing a large discrepancy between training and inference time, which may result in a performance drop with *TrainonGoldTestonPred* here.

5.2 Data Augmentation

Based on the results for DR classification, we choose the RoBERTa encoder and focus on the *InputForRelCls* approach in this set of experiments. The hypothesis is that because of the association between UDims and DRs, if the model is trained on UDim prediction tasks, its performance on DR classification can be improved, and the data augmentation method is primarily used for improving model performance on UDim prediction. The results with *InputDimAtt* are shown for comparison.

Table 4 shows the RST DR classification results under augmentation with different types of PDTB data.

As can be seen from Table 4, data augmentation improves F1 score, but an increase in F1 does not necessarily lead to higher accuracy, which is not rare for imbalanced classification, suggesting that the model is trained to distinguish smaller classes. For *InputForRelCls*, data augmentation with total

Model	$F1_{RoBERTa}$	$Acc_{RoBERTa}$
InputForRelCls	54.89	66.32
InputForRelCls+PDTBExpl	55.28	**65.72**
InputForRelCls+PDTBTotal	**55.75**	65.61
InputForRelCls+PDTBImpl	54.57	65.02
InputDimAtt	54.65	66.27
InputDimAtt+PDTBExpl	**54.84**	**66.16**
InputDimAtt+PDTBTotal	53.81	65.78
InputDimAtt+PDTBImpl	54.65	65.56
$Baseline_{RoBERTa}$	53.72	65.56

Table 4: Results for RST DR classification with dat augmentation. *Baseline* refers to the approach withou using UDims in training and testing in Table 2. *PDTI Expl*, *PDTBImpl* and *PDTBTotal* denote PDTB explic data, implicit data and the combination of both part respectively.

Model	$F1_{RoBERTa}$	$Acc_{RoBERTa}$
InputForRelCls	53.44	60.26
InputForRelCls+RST	52.12	61.02
InputForRelCls+PDTBExpl	**55.01**	61.22
InputForRelCls+PDTBExpl&RST	53.05	**61.70**
InputDimAtt	52.93	60.67
InputDimAtt+RST	51.18	59.30
InputDimAtt+PDTBExpl	**54.21**	**61.43**
InputDimAtt+PDTBExpl&RST	51.22	60.54
$Baseline_{RoBERTa}$	52.36	60.47

Table 5: Results for PDTB implicit DR classification with data augmentation. Baseline refers to the approach without using UDims in training and testing in Table 3.

PDTB data yields the highest performance, which is expected. However, it is noticeable that adding PDTB implicit DR data causes a performance drop. This might be attributed to the high ambiguity in mapping UDims with implicit relations discussed in Sanders et al. (2018). For *InputDimAtt*, it is clear that adding PDTB explicit DR data is helpful, but adding total PDTB data causes a performance drop. Using predicted UDims as features may introduce noise and aggravate data sparsity for model learning for DR classification. If total PDTB data is used, significantly more data will be used for UDim prediction[2], and RST data and the corresponding UDims will be sampled less in each batch compared with data augmentation with only PDTB explicit or implicit DR data, where the amount of data augmentation is similar to RST data.

Table 5 shows the results for PDTB implicit DR classification under augmentation with different types of data.

As is shown in Table 5, adding PDTB explicit DR data is the most helpful form of data augmentation for both *InputForRelCls* and *InputDimAtt*, but adding RST data causes performance drops, possibly due to the high dissimilarity between RST data and PDTB implicit DR data.

[2]In our experiments, RST has 24,062 training instances, while PDTB total data has 35,080 instances.

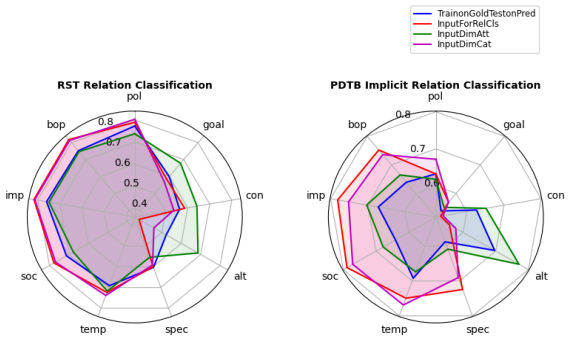

Figure 2: Correlation between DR classification loss and UDim classification losses for RST and PDTB. The abbreviations of the UDims have been explained in Table 1, and the scales represent the Pearson correlation coefficient scores. Note that the areas of different models cannot be compared between RST and PDTB, since the scales on the two plots are arranged in different ways to suit the range of the real data.

6 Analysis

The test performance of the selected models on the UDim prediction task is shown in Table 8 and Table 9 in Appendix A for reference.

6.1 Model Performance

We examine the correlation between the DR classification task and each of the UDim classification tasks. The losses at each training step are collected and the Pearson correlation coefficient is computed between the DR classification loss and the UDim classification losses. The results for the different models used in section 5.1 are shown in Figure 2. Note that the data is collected for models using RoBERTa as the input encoder.

As is clear from Figure 2, for RST DR classification, the models show high correlation between DR classification and the classification of five major UDims, including *polarity*, *basic operation*, *implication order*, *source of coherence* and *temporality*, while correlation with the other UDims is not prominent. The pattern with *InputDimAtt* is different, where correlation with the UDims is basically evenly distributed, except for *specificity*, which might be attributed to importance weighting with the attention mechanism.

For PDTB implicit DR classification, different models show divergence in their correlation

strengths with different UDims. In the case of the best performing model *InputForRelCls*, the correlation with *polarity* is low but the correlation with *specificity* is high. We find that the model performance on *polarity* is relatively low, and this could be a reason why the model learns to rely less on this UDim.

Similar to the patterns for RST, apart from the five major UDims, the other UDims do not show high correlation with DR classification, but in *TrainonGoldTestonPred* and *InputDimAtt*, relatively high correlation with *conditional* and *alternative* in particular, is observable. The performance with *TrainonGoldTestonPred* is lower than the baseline and we can see that the total area of correlation for this model is the smallest, which is expected. With *InputDimAtt*, the association area is also small, which may suggest that the attention mechanism gives more weight to h_{DR} than the embeddings of the predicted UDims.

6.2 Correlation Between DRs and UDims

In this section, we investigate the correlation between the prediction of specific DRs and that of UDims. Following Ulmer et al. (2022), we use the softmax gap as a metric for uncertainty estimation (UE), and compute Kendall's τ between uncertainty values of DR prediction and the prediction of their respective UDims. As a higher softmax gap indicates lower uncertainty, we apply a negative sign to the computed softmax gap and add one to the result, thus maintaining the numeric range but reversing the sign, as in Ulmer et al. (2022). We choose the best-performing models, i.e., *InputForRelCls+PDTBTotal* for RST and *InputForRelCls+PDTBExpl* for PDTB implicit DR classification. The uncertainty values are computed on the test sets.

The relations with >100 data points in the test sets are chosen and the UDims whose Kendall's τ association has $p-$value below 0.05 are shown here. Table 6 shows the results for RST. It can be seen that most of the RST DRs are correlated with these five major UDims discussed in section 6.1.

The results for PDTB are shown in Table 7. Among these DRs, *Instantiation* has higher correlation with *alt* and *spec*. The high association with *spec* is expected, but *alt* is to be associated with *Disjunction*. We suspect that the correlations shown here are influenced by model classification of the other relations, not just based on the direct association between UDims and DRs. The higher correlation between *Asynchronous* and *temp* is expected.

DRs (Counts)	UDims
Contrast (146)	pol(0.56), temp(0.48), soc(0.41), spec(0.24), goal(0.21), con(0.19)
Joint(212)	soc(0.46), temp(0.41), pol(0.40), spec(0.40), imp(0.36), bop(0.29), con(0.12), goal(0.12)
Elaboration (796)	bop(0.57), imp(0.56), pol(0.51), alt(0.50), temp(0.42), goal(0.38), con(0.36), soc(0.25), spec(0.17)
Explanation (110)	imp(0.47), temp(0.45), bop(0.37), pol(0.31), con(0.29), goal(0.22), soc(0.21), alt(0.15), spec(0.14)
Background (111)	pol(0.38), bop(0.35), soc(0.35), imp(0.32), temp(0.26), goal(0.23)

Table 6: Correlation between DR classification uncertainty and UDim prediction uncertainty for RST.

DRs (Counts)	UDims
Conjunction (236)	soc(0.62), temp(0.57), pol(0.51), imp(0.39), bop(0.35), goal(0.34), alt(0.30), spec(0.25), con(0.25)
Cause(406)	soc(0.52), bop(0.48), imp(0.45), pol(0.42), temp(0.41), spec(0.36), alt(0.22), goal(0.16)
Instantiation (124)	alt(0.32), spec(0.31), bop(0.29), soc(0.29), pol(0.27), imp(0.26), temp(0.17), goal(0.18)
Level-of-detail (208)	soc(0.51), bop(0.50), imp(0.47), spec(0.43), pol(0.42), temp(0.39), alt(0.34), goal(0.30)
Asynchronous (105)	temp(0.45), soc(0.39), pol(0.31), imp(0.26), bop(0.24)

Table 7: Correlation between DR classification uncertainty and UDim prediction uncertainty for PDTB implicit DR data.

7 Conclusion

We propose a method for incorporating the UDims in the UniDim proposal into DR classification for RST and PDTB, which allows convenient cross-framework data augmentation. With data augmentation, we obtain strong performance in F1 for DR classification (55.75 for RST and 55.01 for PDTB implicit DR classification). Our experiments suggest that because of the strong association between UDims and DRs, training the model with objectives of UDim prediction helps the model in DR classification, and adding PDTB explicit DR data is helpful for both RST and PDTB implicit DR classification. We are aware that there are additional techniques we can explore, such as adding contrastive loss, or leveraging uncertainty estimation during model training to improve the performance on DR classification. We try to reduce the use of tricks to show the influence of UDims, and leave this part to future work.

8 Limitations

With our approach, multiple runs have to be performed, which requires longer training time, even though the model parameters are not increased.

9 Ethics Statement

We do not foresee any ethical concerns with this study.

References

Farah Benamara and Maite Taboada. 2015. Mapping different rhetorical relation annotations: A proposal. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 147–152, Denver, Colorado. Association for Computational Linguistics.

Antoine Bosselut, Asli Celikyilmaz, Xiaodong He, Jianfeng Gao, Po-Sen Huang, and Yejin Choi. 2018. Discourse-aware neural rewards for coherent text generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 173–184, New Orleans, Louisiana. Association for Computational Linguistics.

Chloé Braud, Barbara Plank, and Anders Søgaard. 2016. Multi-view and multi-task training of RST discourse parsers. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1903–1913, Osaka, Japan. The COLING 2016 Organizing Committee.

Harry Bunt and Rashmi Prasad. 2016. ISO DR-Core (ISO 24617-8): Core concepts for the annotation of discourse relations. In *Proceedings 12th Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-12)*, pages 45–54.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurovsky. 2001. Building a discourse-tagged corpus in the framework of Rhetorical Structure Theory. In *Proceedings of the Second SIGdial Workshop on Discourse and Dialogue*.

Christian Chiarcos. 2014. Towards interoperable discourse annotation. discourse features in the ontologies of linguistic annotation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4569–4577, Reykjavik, Iceland. European Language Resources Association (ELRA).

Abouzar Choubineh, Jie Chen, Frans Coenen, and Fei Ma. 2023. Applying Monte Carlo dropout to quantify the uncertainty of skip connection-based convolutional neural networks optimized by big data. *Electronics*, 12(6):1453.

Vera Demberg, Merel CJ Scholman, and Fatemeh Torabi Asr. 2019. How compatible are our discourse annotation frameworks? Insights from mapping RST-DT and PDTB annotations. *Dialogue & Discourse*, 10(1):87–135.

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.

Vanessa Wei Feng and Graeme Hirst. 2012. Text-level discourse parsing with rich linguistic features. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 60–68, Jeju Island, Korea. Association for Computational Linguistics.

Yingxue Fu. 2022. Towards unification of discourse annotation frameworks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 132–142, Dublin, Ireland. Association for Computational Linguistics.

Yingxue Fu. 2023. Discourse relations classification and cross-framework discourse relation classification through the lens of cognitive dimensions: An empirical investigation. *arXiv preprint arXiv:2311.00451*.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059. PMLR.

Walter Gerych, Tom Hartvigsen, Luke Buquicchio, Emmanuel Agu, and Elke A Rundensteiner. 2021. Recurrent bayesian classifier chains for exact multi-label classification. *Advances in Neural Information Processing Systems*, 34:15981–15992.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings.

Fengyu Guo, Ruifang He, Di Jin, Jianwu Dang, Longbiao Wang, and Xiangang Li. 2018. Implicit discourse relation recognition using neural tensor network with interactive attention and sparse learning. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 547–558, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Dan Iter, Kelvin Guu, Larry Lansing, and Dan Jurafsky. 2020. Pretraining with contrastive sentence objectives improves discourse performance of language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4859–4870, Online. Association for Computational Linguistics.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representations*.

Yangfeng Ji and Jacob Eisenstein. 2014. Representation learning for text-level discourse parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13–24, Baltimore, Maryland. Association for Computational Linguistics.

Yangfeng Ji and Jacob Eisenstein. 2015. One vector is not enough: Entity-augmented distributed semantics for discourse relations. *Transactions of the Association for Computational Linguistics*, 3:329–344.

Congcong Jiang, Tieyun Qian, Zhuang Chen, Kejian Tang, Shaohui Zhan, and Tao Zhan. 2021. Generating pseudo connectives with MLMs for implicit discourse relation recognition. In *PRICAI 2021: Trends in Artificial Intelligence: 18th Pacific Rim International Conference on Artificial Intelligence, PRICAI 2021, Hanoi, Vietnam, November 8–12, 2021, Proceedings, Part II 18*, pages 113–126. Springer.

Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2012. A novel discriminative framework for sentence-level discourse analysis. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 904–915, Jeju Island, Korea. Association for Computational Linguistics.

Najoung Kim, Song Feng, Chulaka Gunasekara, and Luis Lastras. 2020. Implicit discourse relation classification: We need to talk about evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5404–5414, Online. Association for Computational Linguistics.

Haejun Lee, Drew A. Hudson, Kangwook Lee, and Christopher D. Manning. 2020. SLM: Learning a discourse language representation with sentence unshuffling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1551–1562, Online. Association for Computational Linguistics.

Qi Li, Tianshi Li, and Baobao Chang. 2016. Discourse parsing with attention-based hierarchical neural networks. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 362–371, Austin, Texas. Association for Computational Linguistics.

Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. 2009. Recognizing implicit discourse relations in the Penn Discourse Treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 343–351, Singapore. Association for Computational Linguistics.

Wei Liu and Michael Strube. 2023. Annotation-inspired implicit discourse relation classification with auxiliary discourse connective generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15696–15712, Toronto, Canada. Association for Computational Linguistics.

Yang Liu, Sujian Li, Xiaodong Zhang, and Zhifang Sui. 2016. Implicit discourse relation classification via multi-task neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30.

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. *arXiv preprint arXiv:1907.11692*.

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

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. *Advances in Neural Information Processing Systems*, 32.

Emily Pitler, Annie Louis, and Ani Nenkova. 2009. Automatic sense prediction for implicit discourse relations in text. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 683–691, Suntec, Singapore. Association for Computational Linguistics.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. 2008. The Penn Discourse TreeBank 2.0. In *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC'08)*, Marrakech, Morocco. European Language Resources Association (ELRA).

Rashmi Prasad, Bonnie Webber, and Alan Lee. 2018. Discourse annotation in the PDTB: The next generation. In *Proceedings 14th Joint ACL - ISO Workshop on Interoperable Semantic Annotation*, pages 87–97, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Charlotte Roze, Chloé Braud, and Philippe Muller. 2019. Which aspects of discourse relations are hard to learn? primitive decomposition for discourse relation classification. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 432–441, Stockholm, Sweden. Association for Computational Linguistics.

Attapol Rutherford and Nianwen Xue. 2015. Improving the inference of implicit discourse relations via classifying explicit discourse connectives. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages

799–808, Denver, Colorado. Association for Computational Linguistics.

Ted JM Sanders, Vera Demberg, Jet Hoek, Merel CJ Scholman, Fatemeh Torabi Asr, Sandrine Zufferey, and Jacqueline Evers-Vermeul. 2018. Unifying dimensions in coherence relations: How various annotation frameworks are related. *Corpus Linguistics and Linguistic Theory*.

Ted JM Sanders, Wilbert PM Spooren, and Leo GM Noordman. 1992. Toward a taxonomy of coherence relations. *Discourse processes*, 15(1):1–35.

Ted JM Sanders, Wilbert PM Spooren, and Leo GM Noordman. 1993. Coherence relations in a cognitive theory of discourse representation.

Artem Shelmanov, Evgenii Tsymbalov, Dmitri Puzyrev, Kirill Fedyanin, Alexander Panchenko, and Maxim Panov. 2021. How certain is your Transformer? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1833–1840, Online. Association for Computational Linguistics.

Wei Shi and Vera Demberg. 2019. Learning to explicitate connectives with Seq2Seq network for implicit discourse relation classification. In *Proceedings of the 13th International Conference on Computational Semantics - Long Papers*, pages 188–199, Gothenburg, Sweden. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Symon Stevens-Guille, Aleksandre Maskharashvili, Xintong Li, and Michael White. 2022. Generating discourse connectives with pre-trained language models: Conditioning on discourse relations helps reconstruct the PDTB. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 500–515, Edinburgh, UK. Association for Computational Linguistics.

Dennis Ulmer, Jes Frellsen, and Christian Hardmeier. 2022. Exploring predictive uncertainty and calibration in NLP: A study on the impact of method & data scarcity. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2707–2735, Abu Dhabi, United Arab Emirates. 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. *Advances in Neural Information Processing Systems*, 30.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Discourse-aware neural extractive text summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5021–5031, 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.

A Results for UDim Prediction on RST and PDTB

Table 8 shows the test performance of the selected model on UDim prediction for RST.

Model	pol F1	bop F1	impl F1	soc F1	temp F1	spec F1	alt F1	con F1	goal F1	pol acc	bop acc	impl acc	soc acc	temp acc	spec acc	alt acc	con acc	goal acc
TrainonGold TestonPred	73.65	57.78	57.06	60.69	47.18	82.50	64.22	87.82	85.16	87.43	78.45	77.53	73.78	88.79	85.91	99.73	98.91	98.37
InputDimCat	75.33	57.72	56.88	62.57	48.43	82.64	79.95	89.30	83.52	88.41	78.13	77.64	74.48	87.49	86.02	99.78	98.97	98.20
InputDimAtt	74.09	59.02	56.32	60.85	45.42	82.72	74.95	88.42	86.15	87.60	77.86	76.71	75.46	87.21	86.40	99.78	98.86	98.48
InputFor RelCls	73.19	60.34	58.33	61.39	46.41	82.53	83.29	88.36	85.16	87.54	78.84	78.13	75.14	87.00	86.45	99.84	98.91	98.37

Table 8: Results for UDim prediction on RST.

Table 9 shows the test performance of the selected model on UDim prediction for PDTB implicit DR data.

Model	pol F1	bop F1	impl F1	soc F1	temp F1	spec F1	alt F1	con F1	goal F1	pol acc	bop acc	impl acc	soc acc	temp acc	spec acc	alt acc	con acc	goal acc
TrainonGold TestonPred	66.84	66.41	63.50	69.61	58.10	81.42	100.00	84.66	77.10	86.55	75.77	72.82	74.95	76.53	87.17	100.00	99.52	87.17
InputDimCat	68.59	69.71	66.43	72.12	60.72	82.46	100.00	81.68	79.84	87.71	77.97	74.74	76.53	79.75	87.37	100.00	99.45	88.81
InputDimAtt	69.03	68.40	65.66	74.24	59.93	80.16	100.00	77.10	81.08	88.95	77.49	74.19	77.08	77.21	85.86	100.00	99.31	89.09
InputFor RelCls	66.27	65.25	62.48	70.66	58.76	82.10	100.00	84.66	78.92	86.41	75.77	73.03	75.22	77.97	87.71	100.00	99.52	88.81

Table 9: Results for UDim prediction on PDTB implicit DR data.