Exploiting Knowledge about Discourse Relations for Implicit Discourse Relation Classification

Nobel Jacob Varghese and **Frances Yung** and **Kaveri Anuranjana** and **Vera Demberg**
Language Science and Technology, Saarland University
nobeljacobv@gmail.com, {frances, kaveri, vera}@coli.uni-saarland.de

Abstract

In discourse relation recognition, the classification labels are typically represented as one-hot vectors. However, the categories are in fact not all independent of one another – on the contrary, there are several frameworks that describe the labels' similarities (by e.g. sorting them into a hierarchy or describing them in terms of features (Sanders et al., 2021)). Recently, several methods for representing the similarities between labels have been proposed (Zhang et al., 2018; Wang et al., 2018; Xiong et al., 2021). We here explore and extend the *Label Confusion Model* (Guo et al., 2021) for learning a representation for discourse relation labels. We explore alternative ways of informing the model about the similarities between relations, by representing relations in terms of their names (and parent category), their typical markers, or in terms of CCR features that describe the relations. Experimental results show that exploiting label similarity improves classification results.

1 Introduction

Discourse relations (DRs) are logical relations between units of text ("arguments 1 and 2") that make the whole text coherent, see e.g. the concession relation in (1).

(1) [John prepared for his final exam hoping to get at least a pass.]$_{Arg1}$ [He got an E.]$_{Arg2}$ (DR: COMPARISON.CONCESSION.ARG2-AS-DENIER)

The task of implicit DR recognition (IDRR) is particularly challenging because informative discourse connectives (DCs), such as *"however"* are missing. Implicit discourse relation classification tasks using the Penn Discourse Treebank (PDTB) framework (Prasad et al., 2008) typically distinguish between 11 different labels. However, these labels are not completely independent of one another – some relations tend to co-occur or be confused more than others. The similarities between relations are represented in the PDTB relation hierarchy, which groups the labels into four top-level classes, or by the CCR feature representation proposed in Sanders et al. (2021). However, these well-known similarities are typically not exploited for discourse relation classification tasks – instead, all labels are treated as if they were independent of one another.

Guo et al. (2021) recently proposed the Label Confusion Model (LCM), which seems well-suited for the characteristics of the IDRR task: Guo et al. (2021) showed that the method is particularly suitable for problems with many labels, classification problems in which labels are ambiguous and tend to be confused with each other, and/or when there is semantic overlap between the labels. They demonstrated the benefit of the method on several text classification tasks.

The goal of the present paper is to test whether the LCM approach is indeed helpful for IDRR and experiment with three different ways of capturing the label similarities. (1) We use label embeddings: DR labels are not random words but terms that lexically describe the meaning of the DRs, such as REASON, PRECEDENCE, CONDITION, and so on. Using label embeddings assumes that similar relations also tend to have names with similar lexical embeddings. However, some relation labels may additionally be associated with a quite different meaning in normal language use (e.g., "concession"), and their embedding may hence not capture the technical meaning well. (2) We characterize a DR by a set of prototypical connectives (e.g., *however* and *nevertheless* for a CONCESSION relation). (3) We encode DRs via their cognitive features (e.g., a concession relation would be described as a negative causal relation).

Proceedings of the 4th Workshop on Computational Approaches to Discourse (CODI 2023), pages 99–105

2 Related work

2.1 Discourse Relation Classification

Our work is not the first to use information from typical connectives for enriching classification: For example, the implicit DCs that are annotated together with the sense labels in PDTB have been incorporated into the training objective (Kishimoto et al., 2020; Jiang et al., 2021a; Kurfalı and Östling, 2021; Jiang et al., 2021b). Several works also utilize the label hierarchy of the PDTB to train the model to learn the difference between the labels by contrastive learning (Long and Webber, 2022) or operate on the label hierarchy for learning sounder embeddings to direct the prediction (Wu et al., 2021). In this work, we operate on the label names to incorporate the information of the DCs and the PDTB hierarchy.

In addition, DRs can be described in terms of features. The Cognitive approach to Coherence Relations (CCR) (Sanders et al., 1992, 2021) characterizes the nature of DRs by "dimensions" such as *basic operation*, *source of coherence*, *order* and *polarity*. For example, a CONTRAST relation can be described as a *negative* relation of *addition* operation with *objective* source of coherence. We also explore the potential of encoding these unifying dimensions of DRs into the label names for IDRR.

2.2 Exploiting label Similarity

Text classification tasks typically distinguish between a large number of categories or labels. Various approaches have been proposed to model the relation between the semantics of the labels and the text to be classified. Zhang et al. (2018) compares the vectors of the inputs and labels in a multitask learning setting. Wang et al. (2018) use label-based attention scores to embed the label information. Xiong et al. (2021) append the labels to the inputs, such that the embeddings of the labels are learned using the self-attention mechanism of BERT.

Our work builds on the Label Confusion Model (LCM; Guo et al., 2021), which was proposed for learning about the similarity of instances and labels simultaneously during training and which can be expected to be particularly useful in classification tasks with many similar labels. The LCM generates an alternate semantically informed vector in place of one-hot vectors.

For every input to the base model, the LCM inputs all the labels of the corresponding classification tasks, i.e., the LCM is run in parallel with a base model, as seen on the right side of Figure 1. The LCM model consists of a label encoder and a Simulated Label Distribution (SLD) block. The encoder, which comprises an input layer, an embedding layer and a linear layer produces a representation for all the labels.

The representation produced by the base model before the soft-max layer and the representation generated by the LCM encoder is made compatible such that they have dimensions that enable a similarity calculation. A similarity calculation is performed in the SLD block between the representation produced by the base model and the label encoder to generate the SLD distribution in place of one-hot vectors. A controlling parameter is α modulates the balance between the original label one-hot vector and the generated SLD.

Then, KL-divergence loss is computed between the predicted label distribution (PLD) of the base model and the generated SLD. The final labels are predicted using the soft-max classifier of the base model. The LCM trains in parallel with the base model until the LCM-stop epoch, which is determined by a hyper-parameter.

Experiments and analyses on data sets like DBPedia[1], THUCNews[2], etc., show that the LCM can generate representations that capture the dependencies between the labels and assist the base model to better understand the obscure meaning of the target labels compared with one-hot representation. In this work, we train an IDRR model with the LCM to exploit the semantics of the DR labels.

3 Methodology

3.1 LCM for IDRR

We train an 11-way classification model which predicts one of the second-level DR labels defined in PDTB 2.0, as shown in the first column of Table 1, given the two spans of text (called *Arg 1* and *Arg 2*) the DR links. To do so, we trained the LCM with a state-of-the-art IDRR model, which is the Bilateral Matching and Gated Fusion (BMGF) RoBERTa model (Liu et al., 2020).

The BMGF-RoBERTa is a complex model that comprises six layers: a hybrid representation layer, a context representation layer, a matching layer, a fusion layer, an aggregation layer, and a prediction layer. As shown in Fig 1, the LCM runs concurrently with the BMGF-RoBERTa for each input

[1]https://www.dbpedia.org/
[2]http://thuctc.thunlp.org/

instance. We initialize the embedding layer of the LCM with pre-trained GloVe (Pennington et al., 2014) word embeddings of the labels and their variations as described in table 1. The learned representation generated by the prediction layer of the BMGF-RoBERTa is fed as the input to the SLD block of the LCM. KL-divergence loss is calculated between the predicted label distribution (PLD) of the BMGF-RoBERTa and the generated SLD. The final labels are predicted using the soft-max classifier of the BMGF-RoBERTa and the SLD produced by the LCM is utilized for optimizing the loss until the LCM-stop epoch, which is determined by a hyper-parameter. After the LCM-stop epoch, only the BMGF-RoBERTa is trained further and the LCM is inactive.

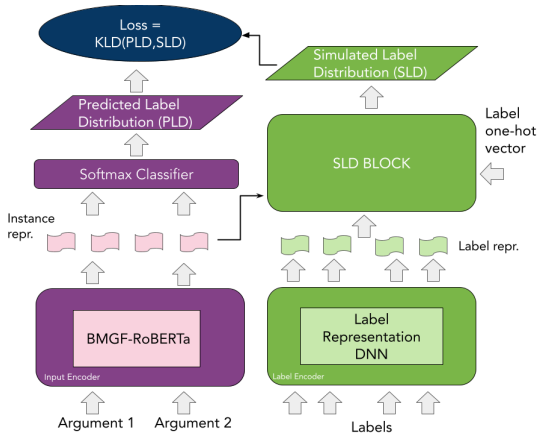

Figure 1: Combined architecture of the BMGF-RoBERTa and the LCM

3.2 Encoding other DR knowledge

Training with the LCM allows the IDRR model to learn the association between the input arguments and the semantics of the label tokens, such as CONJUNCTION and CAUSE. We hypothesize that more detailed relationships could be learned with more expressive label tokens. We explore three alternative ways of encoding label similarity: via label encodings, via encodings of prototypical connectives and via CCR features.

DR labels The PDTB 2.0 labels are arranged in a three-level hierarchical structure, where the 11-way labels, which are usually used in classification tasks, belong to the second level and are children of one of the four parent categories, namely TEMPORAL, COMPARISON, CONTINGENCY, and EXPANSION. Labels under the same parent category are more closely related than labels of different parent categories. In our experiments, we compare the use of only the level-2 labels with the combination of level-2 and level-1 "parent" labels.

Prototypical DCs DCs are used in both traditional and crowd-sourced annotations to facilitate the identification of the implicit DRs (Prasad et al., 2007; Yung et al., 2019). Most relations can be characterized by some prototypical DCs. For example, a CAUSAL relation is best represented by *because* and *therefore*. We define a subset of prototypical DCs for each label and replace the label tokens with the DC tokens. We do not include preposition tokens present in multi-word DCs in order not to dilute the overall semantic representation of the labels (e.g. *example* instead of *for example*).

Cognitive approach to Coherence Relations (CCR) Sanders et al. (2021) decompose each third-level DR in the PDTB 2.0 with five unifying dimensions. We specify each second-level relation by the dimension values shared by its children. Two or three dimensions are enough to specify the second-level relations. We use these CRR tokens in addition to the original DR tokens because certain second-level relations, such as CONJUNCTION and RESTATEMENT, have the same set of dimension values. We do not include value tokens that semantically overlap with the relation label. For example, we do not include the value of the *temporal order* dimension of the SYNCHRONOUS relation, because it is also *synchronous*.

Table 1 shows the lexical terms we use for each setting. We combine the representation of the multiple tokens per label by summing up the GloVe embeddings of the individual tokens[3].

3.3 Data and setting

We train and evaluate the proposed model on the PDTB 2.0 data set (Prasad et al., 2008). We use sections 2-20 for the training, 21-22 for testing, and 0-1 for validation, following e.g. Ji and Eisenstein (2015). The models are trained for the 11-way classification of the second-level sense labels.

We use the codes of the BMGF-RoBERTa released on GitHub[5], which was implemented in PyTorch, and re-implemented the original LCM from

[3]We also experimented with vector averaging. Similar results were obtained.

[4]For integrity, we use single tokens in the original labels. For the PRAGMATIC CAUSE relation, we used the token *pragmatic* instead of *cause* since there is already a CAUSE relation.

[5]https://github.com/HKUST-KnowComp/BMGF-RoBERTa

Original labels	Parent labels	Prototype DCs	CCR features
concession	comparison	despite, even, though, however	negative, causal
contrast	comparison	contrast, comparison, but	negative, addition, objective
cause	contingency	because, result, therefore	positive, causal, objective
pragmatic (cause)	contingency	considering, accordingly	positive, causal, subjective
alternative	expansion	alternatively, instead, rather	positive, addition
conjunction	expansion	addition, also, furthermore	positive, addition
instantiation	expansion	example, instance	positive, addition
list	expansion	firstly, secondly, thirdly	positive, addition
restatement	expansion	other, words, means	positive, addition
asynchronous	temporal	subsequently, afterwards, previously	positive, addition
synchrony	temporal	same, time, simultaneously, meanwhile	positive, addition

Table 1: Tokens used in each label representation strategy. The *prototype DC* tokens replace the original labels while the *CCR* and *parent* tokens are used in addition to the original labels [4].

TensorFlow to PyTorch in order to integrate the two models.

For training, we have utilized 3 × NVIDIA Tesla V100, with a batch size of 16. The pre-trained embedding utilized where GloVe (Pennington et al., 2014) common crawl with 42B tokens. Whenever we utilized the pre-trained word embeddings for the labels, the weights of the embedding layer were frozen and not updated during the training. The values of the hyper-parameters α is optimized to 4 using initialization in the range of 1–6. The LCM-stop parameter is set to 100, which is chosen based on the implementation of Guo et al. (2021). The results reported below are averaged over five runs.

4 Results

Table 2 compares the results of the models evaluated by accuracy and macro F1. It can be observed that all versions of the LCM improved the baseline model. In particular, the LCM model using prototype DCs outperforms the other models.

Model	Accuracy	macro F1
BL (Liu et al., 2020)	55.20 (.013)	36.07 (.010)
+ LCM (orig.)	57.20 (0.006)	38.92 (0.014)
+ LCM (orig.+parent)	57.55 (.010)	40.48 (.006)
+ LCM (orig.+CCR)	57.69 (.004)	39.45 (.015)
+ LCM (protyp. DC)	**57.80** (0.013)	**40.63** (0.025)

Table 2: 11-way classification results on PDTB 2.0[6]. The standard deviation of the five runs is shown in brackets respectively.

Figure 2 compares the distribution of the labels predicted by the baseline and the *LCM (protyp. DC)* models as well as the gold labels of the five most frequent DRs in the test set[7]. It shows that the baseline model over-predicts CONTRAST, CAUSAL, and CONJUNCTION. Inspection of the samples reveals that many of the over-predicted CONTRAST are actually CAUSAL, while the over-predicted CAUSAL and CONJUNCTION are mostly RESTATEMENT and these are correctly classified by the model with LCM. However, the LCM also leads to over-prediction of RESTATEMENT. We will look at some concrete examples in the next section.

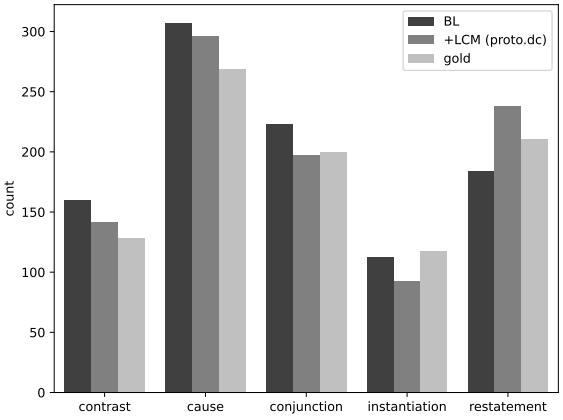

Figure 2: Distribution of the predictions produced by the *BL* and *LCM (protyp. DC)* model compared with gold on the five most frequent DR labels. The counts are the average values of the five runs of each model.

The predicted label distributions suggest that the LCM allows the IDRR model to learn the difference and similarity between COMPARISON and EXPANSION, but not among different types of EXPANSION. The parent relation and the CCR features of all EXPANSION relations are in fact the same. That could explain why the performances of these two versions on the EXPANSION items are similar, while LCM$_{DC}$ performs slightly better.

[6]The published accuracy of the BMGF-RoBERTa is 58.13. We found that the discrepancy is because, according to the released codes, the result of the best **test** epoch has been reported. For fair evaluation, we report the results of all the models based on the best **validation** epoch (based on macro F1).

[7]These are the gold labels of 90% of the test set instances.

5 Qualitative Analysis

In this section, we analyze some examples that demonstrate that the LCM has better captured the implicit DRs between two arguments.

First, as mentioned in the previous section, the false positive CONTRAST relations predicted by the baseline model are mostly CAUSAL relations. In most of these cases, the *Arg2* contains the tokens *now* or *still*, which are often used to mark *contrast*, as in the following example.

(2) [Last week that company and union negotiations had overcome the major hurdle, ...]$_{Arg1}$ [Now only minor points remain to be cleaned up]$_{Arg2}$
(gold: : CONTINGENCY.CAUSE
LCM: CONTINGENCY.CAUSE
baseline: COMPARISON.CONTRAST)

In Example 2, the baseline model's prediction might have been based on the local markers *now* and the lexical pair *major* and *minor*, while the LCM model infers the positive relation between *overcome major hurdle* and *only minor points remain*.

Secondly, the LCM models overpredict RESTATEMENT relations, which are annotated as other relations in the PDTB. We found that for some of these cases, a restatement label could actually be justifiable as a secondary label.

(3) [Treating employees with respect is crucial for managers.]$_{Arg1}$ [It's in their top five work values.]$_{Arg2}$
(gold: : CONTINGENCY.CAUSE
LCM: EXPANSION.RESTATEMENT
baseline: CONTINGENCY.CAUSE)

(4) [Sotheby's defends itself and Mr. Paul in the matter.]$_{Arg1}$ [Mr. Wachter says Mr. Paul was a quick study who worked intensely and bought the best pictures available at the moment.]$_{Arg2}$
(gold: : EXPANSION.INSTANTIATION
LCM: EXPANSION.RESTATEMENT
baseline: EXPANSION.INSTANTIATION)

In Example (3), *respect being crucial* is the reason that it is counted as a *top value*, but these two arguments can also be viewed as different ways to state that it is important for managers to repect their employees. In Example (4), *Mr. Wachter's comment* could be an example of how *Sotheby's defends Mr. Paul*. However, depending on the context, Arg2 can also be interpreted as a RESTATEMENT. These cases suggest that the LCM tends to confuse relations most easily when they are similar or have semantic overlap.

However, we do note that there are cases where the LCM model indeed overpredicts restatement relations, see example (5).

(5) [It's no longer enough to beat the guy down the street.]$_{Arg1}$ [You have to beat everyone around the world.]$_{Arg2}$
(gold: : EXPANSION.ALTERNATIVE
LCM: EXPANSION.RESTATEMENT
baseline: EXPANSION.ALTERNATIVE)

Finally, comparing the different versions of the LCM models, the LCM$_{DC}$ model outperforms the other two models in predicting CAUSAL and CONJUNCTION relations. A possible explanation is that the DC tokens used to represent these relations are indeed strongly prototypical compared with other relations. This suggests that the choice of prototype DCs has a strong effect on the model performance. On the other hand, the LCM$_{parent}$ model has the highest recall of INSTANTIATION relations, but these are often co-occurring with RESTATEMENT, which is predicted by the other two variants.

6 Conclusion

We proposed to inform an IDRR model with knowledge about the DRs encoded in the classification labels using the LCM, instead of treating each class independently. In addition, we explored various strategies to encode different types of knowledge into the model and found that they are all beneficial. This approach is flexible and can also be applied to other base models. Furthermore, learning the lexical semantics of the label tokens allows a model to train on multiple datasets even if they do not share the same label set, and this is the direction of our future work.

7 Limitations

The encoder of the LCM which we have utilized for our experiments is a basic deep neural network. Replacing it with more robust and effective architectures could help achieve better performance. Furthermore, instead of using pre-trained GloVe embeddings for the encoder, using IDDR-specific

embeddings could have been a more efficient approach. Lastly, our models have been trained and evaluated on PDTB 2.0, instead of the latest PDTB 3.0, which includes also intra-sentential implicit relations and has a more systematic sense hierarchy.

Acknowledgments

This project is supported by the German Research Foundation (DFG) under Grant SFB 1102 ("Information Density and Linguistic Encoding", Project-ID 232722074).

References

Biyang Guo, Songqiao Han, Xiao Han, Hailiang Huang, and Ting Lu. 2021. Label confusion learning to enhance text classification models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12929–12936.

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. 2021a. 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.

Feng Jiang, Yaxin Fan, Xiaomin Chu, Peifeng Li, and Qiaoming Zhu. 2021b. Not just classification: Recognizing implicit discourse relation on joint modeling of classification and generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2418–2431.

Yudai Kishimoto, Yugo Murawaki, and Sadao Kurohashi. 2020. Adapting bert to implicit discourse relation classification with a focus on discourse connectives. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1152–1158.

Murathan Kurfalı and Robert Östling. 2021. Let's be explicit about that: Distant supervision for implicit discourse relation classification via connective prediction. In *Proceedings of the 1st Workshop on Understanding Implicit and Underspecified Language*, pages 1–10.

Xin Liu, Jiefu Ou, Yangqiu Song, and Xin Jiang. 2020. On the importance of word and sentence representation learning in implicit discourse relation classification.

Wanqiu Long and Bonnie Webber. 2022. Facilitating contrastive learning of discourse relational senses by exploiting the hierarchy of sense relations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10704–10716, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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.

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, Eleni Miltsakaki, Nikhil Dinesh, Alan Lee, and Aravind Joshi. 2007. The penn discourse treebank 2.0 annotation manual.

Ted JM Sanders, Vera Demberg, Jet Hoek, Merel CJ Scholman, Fatemeh Torabi Asr, Sandrine Zufferey, and Jacqueline Evers-Vermeul. 2021. Unifying dimensions in coherence relations: How various annotation frameworks are related. *Corpus Linguistics and Linguistic Theory*, 17(1):1–71.

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

Guoyin Wang, Chunyuan Li, Wenlin Wang, Yizhe Zhang, Dinghan Shen, Xinyuan Zhang, Ricardo Henao, and Lawrence Carin. 2018. Joint embedding of words and labels for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2321–2331, Melbourne, Australia. Association for Computational Linguistics.

Changxing Wu, Liuwen Cao, Yubin Ge, Yang Liu, Min Zhang, and Jinsong Su. 2021. A label dependence-aware sequence generation model for multi-level implicit discourse relation recognition.

Yijin Xiong, Yukun Feng, Hao Wu, Hidetaka Kamigaito, and Manabu Okumura. 2021. Fusing label embedding into BERT: An efficient improvement for text classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1743–1750, Online. Association for Computational Linguistics.

Frances Yung, Vera Demberg, and Merel Scholman. 2019. Crowdsourcing discourse relation annotations by a two-step connective insertion task. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 16–25.

Honglun Zhang, Liqiang Xiao, Wenqing Chen, Yongkun Wang, and Yaohui Jin. 2018. Multi-task label embedding for text classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4545–4553, Brussels, Belgium. Association for Computational Linguistics.