Multitask Semi-Supervised Learning for Class-Imbalanced Discourse Classification

Alexander Spangher, Jonathan May

University of Southern California
{spangher, jonmay}@usc.edu

Sz-rung Shiang, Lingjia Deng

Bloomberg
{sshiang, ldeng43}
@bloomberg.net

Abstract

As labeling schemas evolve over time, small differences can render datasets following older schemas unusable. This prevents researchers from building on top of previous annotation work and results in the existence, in discourse learning in particular, of many small class-imbalanced datasets. In this work, we show that a multitask learning approach can combine discourse datasets from similar and diverse domains to improve discourse classification. We show an improvement of 4.9% Micro F1-score over current state-of-the-art benchmarks on the *NewsDiscourse* dataset, one of the largest discourse datasets recently published, due in part to label correlations across tasks, which improve performance for underrepresented classes. We also offer an extensive review of additional techniques proposed to address resource-poor problems in NLP, and show that none of these approaches can improve classification accuracy in our setting[1].

1 Introduction

Learning the discourse structure of a text has been shown to be helpful for diverse tasks such as event extraction (Choubey et al., 2020), sentiment analysis (Chenlo et al., 2014), natural language generation (Celikyilmaz et al., 2020), summarization (Lu et al., 2019; Isonuma et al., 2019), storyline discovery (Rehm et al., 2019), and even misinformation detection (Abbas, 2020; Zhou et al., 2020).

However, even as recent advances in NLP allow us to achieve impressive results across a variety of tasks, discourse learning, a supervised learning task, faces the following challenges: (1) discourse datasets tend to be very class-imbalanced.[2] (2) Discourse learning is a complex task: human annotators require training and conferencing to achieve moderate agreement (Das et al., 2017). (3) Discourse learning tends to be resource-poor, as annotation complexities make large-scale data collection challenging (Table 1). Compounding the problem, a schema often evolves across different annotation efforts, preventing the compilation of smaller datasets into larger ones.[3]

We observe, however, that certain discourse schemata appear to offer complementary information. For example, Penn Discourse and Rhetorical Structure Theory Treebanks offer intrasentential, low-level discourse information (Prasad et al., 2008; Carlson et al., 2003), while news discourse schemas offer intersentential, high-level, domain-specific discourse information (Choubey et al., 2020; Yarlott et al., 2018). Inspired by Collobert and Weston (2008)'s finding that lower-level NLP tasks (e.g. part of speech tagging) could aid higher-level tasks (e.g. semantic role labeling), we hypothesize that a multitask approach incorporating multiple discourse datasets can address the challenges listed above. Specifically, by introducing *complementary information from auxiliary discourse tasks*, we can increase performance for a primary discourse task's underrepresented classes.

We propose a multitask neural architecture (Section 2) to address this hypothesis. We construct tasks from 6 discourse datasets, an events dataset, and an unlabeled news dataset (Section 3), including a novel discourse dataset we introduce in this work. Although different datasets are developed under divergent schemas and have different goals, our framework learns correlations between schemas, and does not "waste" labeling work done by generations of NLP researchers.

Our experiments show that a multitask approach can help us improve discourse classification on

[1]For code and data, see: https://github.com/alex2awesome/multitask-news-discourse.

[2]For example, of Penn Discourse Tree-Bank's 48 classes, the top 24 are on average 25 times more common than the bottom 24 (Prasad et al., 2008).

[3]See, for instance, datasets based on variations of Van Dijk's news discourse schema (Van Dijk, 2013) released in Choubey et al. (2020), Yarlott et al. (2018) and the present work.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 498–517
November 7–11, 2021.

a primary task, *NewsDiscourse* (Choubey et al., 2020), from a baseline performance of 62.8% Micro F1 to 67.7%, an increase of 4.9 points (Section 4), with the biggest improvements seen in underrepresented classes. On the contrary, two data augmentation approaches, Training Data Augmentation (TDA) and Unsupervised Data Augmentation (UDA), fail to improve performance.

We give insight into why this occurs (Section 5). In the multitask approach, the primary task's underpresented labels are correlated with labels in other datasets. However, if we only provide more data without any correlated labels (TDA and UDA), we overpredict the overrepresented labels. We test many other approaches proposed to address class-imbalance and observe similar negative results (Appendix F). Taken together, this analysis indicates that the signal from labeled datasets is essential for boosting performance in class-imbalanced settings.

In summary, our core contributions are:

- We show a 4.9 F1-score improvement above state-of-the-art on the *NewsDiscourse* dataset and introduce a novel dataset with 67 labeled articles based on an expanded Van Dijk news discourse schema (Van Dijk, 2013).
- *What worked and why*: we show that different discourse datasets in a multitask framework complement each other; correlations between labels in divergent schemas provide support for underrepresented classes in a primary task.
- *What did not work and why:* training data augmentation and semi-supervised data augmentation failed to improve above baseline because they overpredict overrepresented classes, thus hurting overall performance.

2 Methodology

We formulate a multitask approach to discourse learning with the *NewsDiscourse* dataset as our primary task (Section 3). Our multitask architecture uses shared encoder layers and task-specific classification heads[4].

Our objective is to minimize the weighted sum of losses across tasks:

$$\min L(D, \alpha) = \min_{\theta} \sum_{t=1}^{T} \sum_{i=1}^{N_t} \alpha_t L_t((x_i\,[, y_i])_t) \quad (1)$$

where $D = \{D_t\}_{t=1}^{T}$ is our joined dataset, $D_t = \{(x_i\,[, y_i])\}_{i=1}^{N_t}$ are task-specific datasets for tasks $t = 1, ..., T$, each of size N_t (labeled and unlabeled). L_t is the task-specific loss, and hyperparameter $\alpha = \{\alpha_t\}_{t=1}^{T}$, a coefficient vector that weights the loss from each task. In each training step, we randomly sample one task t and one datum $(x_i\,[, y_i])_t$[5] from that task's dataset, D_t.

2.1 Neural Architecture

Our neural architecture (Figure 1) consists of a sentence-embedding layer and, in some experimental variations, embedding augmentations; a classification layer for the primary task; and separate classification layers for auxiliary supervised tasks.

The architecture we use to model our supervised tasks is inspired by previous work in sentence-level tagging and discourse learning (Choubey et al., 2020; Li et al., 2021). We use RoBERTa-base (Liu et al., 2019) to generate sentence embeddings (Figure 1). Sentences in each document are read sequentially by the same model, and the </s> token from each sentence is used as the sentence-level embedding. The sequence of sentence embeddings is passed through a Bi-LSTM layer to provide context. These layers are shared between tasks.[6]

Additionally, we experiment with concatenating different embeddings to the sentence embeddings to provide document-level and sentence-positional information. We concatenate headline embeddings and document embeddings, generated as described in Choubey et al. (2020), and sentence-positional embeddings, described in Vaswani et al. (2017).[7]

Each output embedding is classified using a task-specific feed-forward layer.[8] Some of our tasks (including our primary task) are multiclass and others are multilabel. We discuss our datasets (and tasks) in the next section.

3 Datasets

We use 8 datasets in our multitask setup, shown in Table 1. Four datasets contain sentence-level labels and no relational labels; two contain annotations of clausal relations; one is an events-nugget dataset

[4] Our framework can be seen as a multitask feature learning architecture (Zhang and Yang, 2017).

[5] $[, y_i]$ indicates that for some tasks, labels y_i are not present. See Section 4.3: we decompose UDA into a supervised head and an unsupervised head.

[6] Variations on our method for generating sentence embeddings are reported in Appendix F.1

[7] For more detail, see Appendix E.1.

[8] Variations both of the classification tasks and the loss function, aimed at addressing the class-imbalance inherent in the VD2 dataset, are reported in Appendix F.2.

Dataset Name	Label	#Docs	#Sents	#Labels	Altered	Type	Class Imb.
NewsDiscourse	VD2	802	18,151	9	No	MC	3.01
Van Dijk (Yarlott et al., 2018)	VD1	50	1,341	9	No	MC	3.81
Van Dijk (present work)	VD3	67	2,088	12	No	MC	6.36
Argumentation	ARG	300	11,715	5	No	ML	9.35
Penn Discourse Treebank^{**+}	PDTB-t	194	12,533	5	Yes	ML	2.28
Rhetorical Structure Theory**	RST	223	7,964	12	Yes	ML	2.90
KBP Events 2014/2015**	KBP	677	24,443	4	Yes	ML	4.07
All-The-News**	U	6,000	177,530	N/A	N/A	N/A	N/A

Table 1: List of the datasets used, an acronym, the size, number of labels (k), whether we processed it, whether each sentence is multiclass (MC) or multilabel (ML) and the class-imbalance. ** indicates dataset was filtered. + indicates subset of tags was used. (Class Imb. $:= \frac{\sum_{j=1}^{\lfloor k/2 \rfloor} n_j}{\lfloor k/2 \rfloor} / \frac{\sum_{j=\lfloor k/2 \rfloor+1}^{k} n_j}{\lfloor k/2 \rfloor+1}$. n_j is size of class j; $n_1 > ... > n_k$).

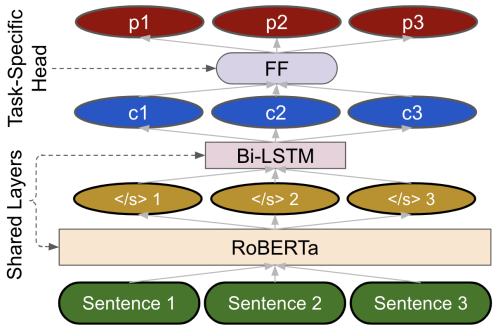


Figure 1: Sentence-Level classification model used for each prediction task. The </s> token in the RoBERTa model is used to generate sentence-level embeddings, $</s>_i$. Bi-LSTM is used to contextualize these embeddings, c_i. Finally, FF is used to make class predictions, p_i. RoBERTa and Bi-LSTM are shared between tasks. FF is the only task-specific layer.

where labels denote the presence of events in sentences; and one is an unlabeled news dataset. See Tables 4 and 5 for all label names.

Van Dijk (VD1, VD2, VD3) and Argumentation (ARG) The Van Dijk Schema, developed by Van Dijk (2013), was applied with no modifications (Yarlott et al., 2018) to 50 news articles sampled from the ACE corpus (VD1). Choubey et al. (2020) expanded Van Dijk's schema to capture anecdotal discourse (Craig, 2006) and released a dataset, *NewsDiscourse* (VD2), consisting of 802 articles from 3 outlets[9]. ***We take VD2 as our primary task due to its size.*** As shown in Table 1, VD2 has 9 classes: *Main Event* (**M1**), *Consequence* (**M2**), *Current Context* (**C1**), *Previous Event* (**C2**), *Historical Event* (**D1**), *Anecdotal Event* (**D2**), *Evaluation* (**D3**), *Expectation* (**D4**) and *Error* (**E**).[10] VD2 is an imbalanced dataset; its highest-support class has 1224 samples while its lowest-support has 77.

We introduce a novel news discourse dataset (VD3) following the Van Dijk Schema. We expand the schema to capture discourse elements related to "Explanatory Journalism" (Forde, 2007). VD3 contains 67 news articles with sentence-level labels, sampled from the ACE corpus without redundancy to VD1. We additionally label 10 articles from VD1 and find an interannotator agreement of $\kappa = .69$[11].

A substantial volume of news discourse is not factual assertion, but analysis, explanation, and prediction (Steele and Barnhurst, 1996). We thus include the Argumentation dataset (ARG) (Al Khatib et al., 2016), a dataset consisting of 5 labels applied to 300 news editorials.[12]

Each of these four datasets assigns a single label to each sentence. We treat them as multiclass datasets, as shown in Table 1.

Penn Discourse Treebank (PDTB) and Rhetorical Structure Theory Treebank (RST) These discourse datasets each consist of spans of text in articles; labels indicate how different spans relate to each other. We process each so that sentences are annotated with the set of all relations occurring at least once in the sentence[13]. Then, we downsample documents in each of these dataset so that the distribution of document length matches VD2.[14] We match document lengths to control for biases introduced by shorter documents, as the full PDTB and RST consist of a large amount of short documents that are not representative of documents in VD2.

Some of Van Dijk's discourse elements differ

[9] nytimes.com, reuters.com and xinhuanet.com

[10] For a detailed class description, see Choubey et al. (2020).

[11] For more information on the dataset we introduce in this paper, see Appendix B.1.

[12] This dataset contains articles from 3 news outlets: aljazeera.com, foxnews.com and theguardian.com

[13] For more details, see Appendix B.2.

[14] Specifically, if $p_m(n)$ and $p_a(n)$ are the likelihood of a document d with n sentences in the main and auxiliary datasets respectively, we sample with weight $w_d = p_m(n)/p_a(n)$ (Austin and Stuart, 2015). $p_m(n)$ and $p_a(n)$ were determined empirically by N_n/N_{total} (N_n: # of docs with sentence-length n in a or m, N_{total}: # of docs in a or m).

based on temporal relation: for example, some elements describe events occurring before a main event (e.g. *Previous Event* (**C2**)) while others describe events occurring after (e.g. *Consequence* (**M2**)). To introduce more information about temporality, we use PDTB's tags pertaining to *Temporal* relations (we call this filtered dataset PDTB-t).

When processed as described above, each of these datasets assign multiple labels to each sentence. We treat them as multilabel datasets.

Knowledge Base Population (KBP) 2014/2015 Some of Van Dijk's discourse elements differ based on the presence or absence of an event. For example, the elements *Previous Event* (**C2**) and *Current Context* (**C1**) both describe the context before a main event, but the former describes events while the latter describes general circumstances. We hypothesize that a dataset identifying event occurrence can help our model differentiate these elements. We collect an additional non-discourse dataset, the KBP 2014/2015 Event Nugget dataset, which annotates the trigger words for events by event-type. We treat this as a multilabel dataset.

All-The-News (U) For semi-supervised data-ablation experiments, described in Section 4.3, we sample 6,000 documents from an unlabeled news dataset.[15] We downsample in the manner described above for PDTB and RST.

4 Experiments and Results

In this section, we briefly discuss experiments using VD2 as a single classification task. Then, we discuss the experiments using VD2 in a multitask setting. Finally, we discuss our experiments with data augmentation as ablations. We leave a more detailed analysis of single-task experiments for Appendix F, focusing here on multi-task experiments.

4.1 Single Task Experiments

We observe, perhaps unsurprisingly, a 2-point F1-score improvement by using RoBERTa as a contextualized embedding layer rather than Choubey et al. (2020)'s baseline, ELMo (Peters et al., 2018) (**Roberta** in Table 2). We observe an additional 1.5 F1 score improvement by freezing layers in RoBERTa (**+Frozen** in Table 2). We find that freezing layers closer to the input results in greater improvement, replicating Lee et al. (2019). Finally, we observe a .5 F1 improvement by incorporating document, headline, and sinusoidal information (**+EmbAug** in Table 2).[16]

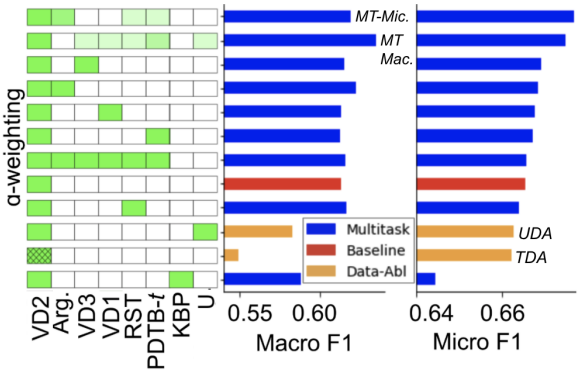


Figure 2: Loss coefficient weightings (α vector) across tasks and Macro vs. Micro F1 Score shown for: (a) a mix of trials, (First two blue bars; **MT-Micro** and **MT-Macro** trials) (b) pairwise multitask tasks (other blue bars), (c) baseline (red bar) (d) data ablation (yellow bar; UDA and TDA). Tasks are green in strength proportional to their α value. When **U** is used, it is used with UDA head. Hashed VD2, for TDA, is data-augmented as described in Section 4.3. Pairwise tasks shown in some rows to emphasize that a soft-weighting α achieves maximal F1 scores.

4.2 Multi-Task Experiments

As shown in Table 2, multitask achieves better results than any single-task experiment. We conduct our multitask experiment by performing a grid-search over loss-weighting, α (defined in Equation 1). We select top-performing α for Micro F1-score as well as Macro F1-score based on a validation split, and report results on a test split[17]. As can be seen, in Figure 2, the weighting achieving the top Micro F1-score includes datasets VD2, ARG, RST and PDTB-t, while the weighting achieving the top Macro F1-score includes datasets VD2, ARG, VD3, and RST.

To understand the effect of each dataset individually, we run linear regression on the α and F1-scores found in our grid search[18]. The regression coefficients, β, displayed in Table 3, approximate the effect each dataset has. We conduct over 600 trials in our grid search and thus have confidence in these results.

[15] kaggle.com/snapcrack/all-the-news. Dataset originally collected from archive.org. We filter to articles from nytimes.com and reuters.com.

[16] The .5 F1 improvement is observed across different sentence embeddings variations. See Appendix sections F and E, specifically Figure 8 and Table 9.

[17] Train, test and validation splits are specified by (Choubey et al., 2020).

[18] I.e. $y = \beta X$, where $X = \alpha$, the loss-weighting scheme for each trial, and $y =$ F1-score.

	M1	M2	C2	C1	D1	D2	D3	D4	E	F1-Macro	F1-Micro
Support	460	77	1149	284	406	174	1224	540	396	4710	4710
ELMo	50.6	27.0	58.9	35.2	63.4	50.3	70.5	64.3	94.6	57.21	62.85
RoBERTa	52.1	9.4	65.1	27.7	68.1	51.6	72.4	65.4	96.0	56.43	64.97
+Frozen	51.2	29.3	64.3	29.8	72.2	65.8	**73.7**	**67.1**	96.5	61.08	66.54
+EmbAug	54.1	28.0	64.7	**35.9**	71.8	66.3	72.9	65.9	96.3	61.76	66.92
TDA	8.5	5.2	57.1	29.8	61.1	44.3	66.1	58.2	16.4	56.53	59.22
UDA	49.4	0.0	65.0	28.4	56.0	0.0	70.8	69.8	96.2	48.39	62.72
+TSA	51.9	34.2	63.6	33.1	70.7	66.9	72.5	66.7	96.3	61.77	66.29
MT-Mac	54.9	**35.5**	63.8	**35.9**	**73.7**	**70.7**	**73.7**	66.3	**96.7**	**63.46**	67.51
MT-Mic	**55.4**	25.0	**67.1**	32.8	72.5	68.9	73.6	65.8	96.0	61.89	**67.70**
Human (Blind)	**58.8**	**36.1**	28.3	10.5	**75.0**	40.0	48.6	22.2	**100.0**	46.18	46.76
Human (Post-Rec.)	**68.7**	**75.0**	70.3	33.3	**81.2**	**79.2**	**83.0**	**79.7**	**100.0**	**73.69**	**77.63**

Table 2: **Overview:** F1-scores of individual class labels in VD2 and Macro-averaged F1-score (Mac.) and Micro F1-score (Mic.). **ELMo** is the baseline used in (Choubey et al., 2020). **RoBERTa+Frozen+EmbAug** is our subsequent baseline. **TDA** refers to Training Data Augmentation. **UDA** is Unsupervised Data Augmentation (+TSA is for "Fine-Tuned UDA with TSA", described in Section 4.3). **MT** stands for multitask: **MT-Mac** is a trial with α chosen to maximize Macro F1-score while **MT-Mic** is a trial with α chosen to maximize Micro F1-score. **Human** is our agreement with Choubey et al. (2020): **Human (Blind)** shows agreement after reading VD2's annotation guidelines, conferencing and not observing labels. **Human (Post-Rec.)** is after observing VD2 labels.

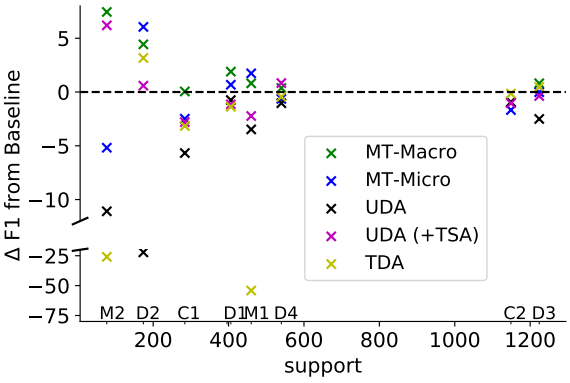


Figure 3: Comparison of class-level accuracy vs. label for three models: MT-Micro, TDA (which underperforms baseline for lower-represented labels like M2, C1), and MT-Macro (which overperforms baseline for lower represented labels M1, M2, D1, D2). Split y-axis shown for clarity, due to TDA outliers.

	Mic.	Mac.		Mic.	Mac.
Main	.83	1.15	ARG	.05	.83
RST	.50	.73	PDTB	-.69	-1.41
VD3	.49	.53	U	-1.14	.68
VD1	.21	.61	KBP	-2.17	-2.94
β_0	66.26	61.13			

Table 3: We run LinReg (LR) on the α weights from multitask trials to predict Micro and Macro F1-scores (i.e. LR(α) = Mic. F1-score, Mac. F1-score). LR coefficients (β) for each dataset show the effects of each dataset on the scores. E.g. increasing RST's weight by +1 yields .5 Mic. F1-score improvement.

4.3 Data Ablation Experiments

To test our hypothesis that labeled information in the multitask setup helps us achieve higher accuracy, we perform the following ablation: we test using additional data that does not contain new label information. We test two methods of data augmentation: Training Data Augmentation (TDA) and Unsupervised Data Augmentation (UDA).

TDA enhances supervised learning (DeVries and Taylor, 2017) by increasing the size of the training dataset through data augmentations on the training data; it exploits the smoothness assumption in semi-supervised learning to help our model be more robust to local data perturbations (Van Engelen and Hoos, 2020). For each datapoint (x_i, y_i) in our primary dataset, we generate $k = 10$ noisy samples $(x_{i1}, y_i), ..., (x_{ik}, y_i)$. We use a sampling-based backtranslation function to generate augmentations for TDA and UDA. (Edunov et al., 2018).[19]

UDA is a form of semisupervised learning that propagates signal from labeled to unlabeled datapoints, making use of the manifold assumption in semi-supervised learning (Xie et al., 2020; Van Engelen and Hoos, 2020). UDA seeks to promote consistency between model predictions on unlabeled datapoints $p_\theta(x_i)$ and their augmentations $\{p_\theta(\hat{x}_i)\}_{j=1}^{k}$ by minimizing their KL-divergence.[20]

[19]To perform backtranslation, we use Fairseq's English to German and English to Russian models (Ott et al., 2019). Inspired by Chen et al. (2020), we generate backtranslations using random sampling with a tunable temperature parameter instead of beam search, to ensure diversity in augmented sentences.

[20]KL-divergence is minimized via consistency loss:

Both techniques were chosen as they have been shown to boost performance of low-resource NLP classifiers above other semi-supervised methods (DeVries and Taylor, 2017; Berthelot et al., 2019; Chen et al., 2020; Xie et al., 2020; Hyun et al., 2020). Because both techniques introduce more data without introducing more labels, they address the question: did multitask learning improve accuracy only by introducing more data?

As shown in Table 2 and Figure 2, **TDA** and **UDA** fail to improve performance above single-task experiments (**RoBERTa+EmbAug**). To interrogate further, we explored approaches introduced by Xie et al. (2020) and Hyun et al. (2020) to improve convergence of UDA. Specifically, we use a confidence threshold, r, to mask out uncertain unlabeled data; Training Signal Annealing (TSA), to mask out uncertain labeled data; suppression coefficient β, to decrease unsupervised loss contributions for low-support classes; and other methods.[21]

We test a range of values for each of these hyperparameters. In particular, we find that TSA with a *Linear* schedule has a dramatic effect on accuracy, nearly rescuing the performance of UDA. We show UDA with and without TSA (Figure 3, Table 2) to demonstrate, yet we are unable to achieve a setting whereby UDA or TDA beats multitask. Additionally, we add UDA as an unsupervised head in our multitask setup, similar to Collobert and Weston (2008) introducing language modeling as an unsupervised head. We find only one setting where it contributes to our multitask accuracy (MT-Macro in Figure 2 and Table 3).

5 Discussion

As shown in Figure 3, a multitask approach significantly increases performance for underrepresented classes while not hurting performance for others. This is in contrast to pure data augmentation approaches, like UDA or TDA. Improving performance in low-support classes improves overall Macro F1, as expected, and Micro F1 (Table 2).

Multitask learning can help learn part of the data manifold where an underrepresented class exists by learning signal from a class which is correlated. Tables 4 and 5 show the correlation between class labels predicted by our multitask model on the same dataset using different heads.

One insight from Table 4 is a simple sanity check: the Van Dijk datasets largely agree on the labels that share similar definitions. For example, there is a strong correlation between sentences labeled *Main Event* (**M1**) by the VD2 head and those labeled *Main Event* by the VD3 head.

However, a more interesting insight is the strong correlation existing between underrepresented classes in the VD2 dataset and classes in other datasets. Classes *Consequence* (**M2**) and *Anecdotal Event* (**D2**) are two of the lowest-support classes, yet they each have strong correlations with labels in every other dataset.

We pause to comment on the differences in task weightings observed in Figure 2 for **MT-Micro** and **MT-Macro**. For example, ARG is one of the most important datasets for **MT-Micro**, but ignored in

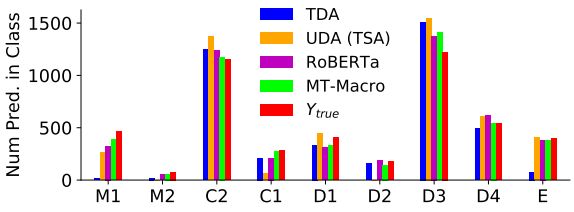


Figure 4: UDA and TDA over-predict better-represented classes (C2, D3) relative to Y_{true}, and underpredicts lesser represented classes (M1, M2, C1, D1). MT-Macro prediction rates are closer to Y_{true}. ($D_{\text{KL}}(C_{\text{UDA}}||C_{Y_{true}}) = .45$, $D_{\text{KL}}(C_{\text{TDA}}||C_{Y_{true}}) = .27$, $D_{\text{KL}}(C_{\text{MT-Macro}}||C_{Y_{true}}) = .01$), where C is the empirical distribution over class-predictions).

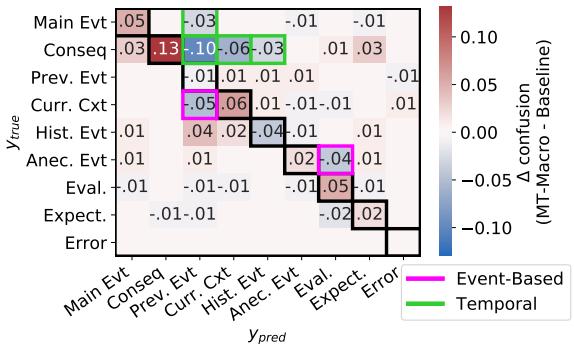


Figure 5: Change in Confusion between **MT-Macro** and **Baseline** (RoBERTa+EmbAug). Except for Historical Event, all classes show an improvement. Classes with Event-Based and Temporal error improvement highlighted (see Section 5 for discussion on confusion categories.)

$L_{con} = \mathbb{E}_k[D(p_\theta(x_i)||p_\theta(\hat{x}_{i,k}))]$

[21] See Appendix G for a detailed discussion on these approaches and our reported explorations. The top-performing hyperparameters we found were: $r = .8$, $TSA = Linear$, $\beta = 0$, $k = 5$, $p = 8$, $\alpha_{UDA} = .8$, $\tau = .8$,; Xie et al. (2020) do not share their explorations; we find that the choice of p (the number of unlabeled data) and k (the number of augmentations per datum) have significant impact on performance.

	Anecdote	Assumption	Common Gr.	Statistics	Testimony	Lede	Main Event	Consequence	Previous E.	Secondary E.	Current Cxt	Historical E.	Evaluation	Expectation	Explanation	Verbal React.	Lede	Main Event	Consequence	Previous E.	Current Cxt	Historical E.	Evaluation	Expectation	Verbal React.	Support
Main Event (M1)						.3	.4		-.4								.4						-.4			1782
Consequence (M2)			.5			.4		.4		.5							.3	.5	.5							387
Previous Event (C2)	.5		.3	.5					.3		.7			-.4		-.6				.4	.7				-.4	4486
Current Context (C1)	.4			.4			.4		.4		.6			-.5		-.5		.3		.5			-.4	-.4		1094
Historical Event (D1)								-.4	.7			.6								.5		.8				1499
Anecdotal Event (D2)					.3						-.4					.6					-.4				.5	609
Evaluation (D3)	-.4		-.5		.5	-.4	-.4			-.5						.5	-.4	-.6	-.4	-.4			.3		.6	4697
Expectation (D4)	-.4				.3			.3	-.5		-.4	-.6		.7		.5				-.5				.7	.4	1981
	Argumentation					VD3 Dataset											VD1 Dataset									

Table 4: Spearman correlations between labels predicted with VD2 head and Argumentation, VD3 and VD1 heads. Note that the two Van Dijk datasets have high correlations between most labels that they have in common. Correlations above $|r| > .2$ shown.

	Elaboration	Joint	Topic Chg.	Attribution	Contrast	Explanation	Background	Evaluation	Summary	Cause	Comment	Temporal	Temporal	Asynch.	Precedence	Synchrony	Succession	Support
Main Event (M1)						-.4												1782
Consequence (M2)			.4		-.4								.2	.2		.2	.3	387
Previous Event (C2)				-.4		.4	-.6				-.5							4486
Current Context (C1)																		1094
Historic Event (D1)						.3						.3	.2	.3	.3		.2	1499
Anecdotal Event (D2)				.3	.3	.4	.5			.3	.4							609
Evaluation (D3)				.4	.5	.4	.6						-.3	-.4	-.4	-.3	-.4	4697
Expectation (D4)				.3			.4					-.4						1981
	RST Dataset											PDTB-t Dataset						

Table 5: Spearman correlation between labels predicted with VD2 head and RST head and PDTB-t head, on the Evaluation split of VD2. Note that PDTB-t relations, which tend to be temporally-based, have a positive correlation with *Consequence* and *Historical Event* labels, which are both defined in temporal relation to the *Main Event* label. Correlations above $|r| > .2$ shown.

MT-Macro. In class imbalanced settings, Micro F1-score is weighted more towards high-support classes while Macro F1-score favors each class equally. Because different auxiliary tasks boost performance for different classes, it is reasonable to assume that the same α will lead to different Macro F1 and Micro F1 scores[22]

One future direction is to identify criteria for including promising discourse tasks in a multitask framework. Bingel and Søgaard (2017) performed such an analysis for multitask setups including POS-tagging and Keyphrase detection and the present work demonstrates the impact such criteria could have in aiding discourse tagging. One criteria for inclusion might be based on the label correlations between the main discourse task and a candidate task. However, obtaining correlations would require training a multitask model; at that point, directly calculating the accuracy boost would be trivial. Identifying discourse-relevant features in the input data, x, as Bingel and Søgaard (2017) did in their work, might be more fruitful.

A competing explanation to our hypothesis that multitask improves performance through label correlations is that additional datasets simply expose the model to more of the data-input space, x. Both UDA and TDA serve as ablation studies for this.

Hyun et al. (2020) show that, for class-imbalanced problems, regions of the data manifold that contain the underrepresented classes generalize poorly when data augmentation is used. Indeed, we show in Figure 4 that TDA and UDA overpredict overrepresented classes, perhaps showing that the algorithms misjudge the extent of underrepresented classes on the data manifold.

One approach to improving semi-supervision would be to consider a more sophisticated annealing algorithm. As discussed in Section 4.3, TSA nearly rescued UDA's performance for all labels. Another would be to generate more augmentations for underrepresented classes (Shorten and Khoshgoftaar, 2019); on the training data for TDA (Chawla et al., 2002) or using a model to identify promising unlabeled points for UDA. Upsampling underrepresented labels in sequences, which our data are, presents a challenge because we can only

[22]For more information, see Appendix D.

sample the entire sequence (i.e. the document). Thus, if we try to upsample individual underrepresented classes (i.e. sentences), we will also be upsampling overrepresented classes in the sequence.

As a final piece of analysis on our multitask setup, we show the reduction of confusion between **MT-Macro** and **Baseline** in Figure 5.[23] We identify reductions in two main classes of confusion: **Temporal** confusion, or confusion between temporal ordering of discourse elements (i.e. *Previous Event* and *Consequence*); and **Event-based** confusion, or confusion between tags semantically similar except for the presence of an event (i.e. *Current Context* and *Previous Event*). While we hypothesize the reduction is due to the addition of temporal information in PDTB-t and event information in RST, more experimentation is needed to confirm.

We close our discussion with an analysis of VD2's task difficulty. We ask expert annotators to relabel VD2 data. Our annotators read Choubey et al. (2020)'s annotation guidelines and labeled a few trial examples. Then they sampled and annotated 30 documents from VD2 without observing VD2's labels. Annotations in this **Blind** pass were significantly worse than predictions made by our best model (Table 2). Then, our annotators observed VD2's labels on the 30 articles, discussed, and changed where necessary. Surprisingly, even in this **Post-Reconciliation** pass, our annotators rarely scored more than 80% F1-score.

Thus, Van Dijk labeling task might face an inherent level of legitimate disagreement, which **MT-Macro** seems to be approaching. However, there are two classes, M1 and M2, where **MT-Macro** underperformed even the **Blind** annotation. For these classes, at least, we expect that there is further room for modeling improvement through: (1) annotating more data, (2) incorporating more auxiliary tasks in the multitask setup, or (3) learning from unlabeled data, by fine-tuning RoBERTa (Mosbach et al., 2021), using an adapter-based method (Wang et al., 2020) or another semi-supervised algorithm (one candidate besides UDA is Berthelot et al. (2019)).

6 Related Work

Ruder (2017) gives a good overview of multitask learning in NLP more broadly. A major early work by Collobert and Weston (2008) uses a single CNN architecture to jointly learn 5 different supervised NLP tasks (e.g. *Part-of-Speech Tagging*) and one unsupervised task (*Language Modeling*), improving performance in their main task. Our work differs in several key aspects: (1) we are concerned with sentence-level tasks; (2) we consider a softer approach to task inclusion, α; (3) we perform a deeper analysis of why multitask helps, including examining inter-task prediction correlations and class-imbalance.

Aside from using different datasets that share the same language, researchers have also used datasets from one language to perform tasks in another. From Information Extraction (Wiedemann et al., 2018; Névéol et al., 2017; Poibeau et al., 2012), Event Detection (Liu et al., 2018; Lejeune et al., 2015; M'hamdi et al., 2019), Part-of-Speech tagging (Cardenas et al., 2019; Plank et al., 2016; Naseem et al., 2009), to even Discourse Analysis (Liu et al., 2020), English datasets have been translated into a target language, a target language has been translated into English, or a joint multilingual space has been learned. Our task may also have benefited from multilingual discourse datasets.

Most state-of-the-art research in discourse analysis specifically has focused on classifying the discourse relations between pairs of clauses, as is practice in the Penn Discourse Treebank (PDTB) (Prasad et al., 2008) and Rhetorical Structure Theory (RST) dataset (Carlson et al., 2003). Corpora and methods have been developed to predict explicit discourse connectives (Miltsakaki et al., 2004; Lin et al., 2009; Das et al., 2018; Malmi et al., 2018; Wang et al., 2018) as well as implicit discourse relations (Rutherford and Xue, 2016; Liu et al., 2016; Lan et al., 2017; Lei et al., 2017). Choubey et al. (2020) built a news article corpus where each sentence was labeled with a discourse label defined in Van Dijk schema (Van Dijk, 2013).

Since discourse analysis has limited resources, some work has explored a multitask framework to learn from more than one discourse corpus. Liu et al. (2016) propose a CNN based multitask model and Lan et al. (2017) propose an attention-based multitask model to learn implicit relations in PDTB and RST. The main difference in our work is the coverage and flexibility of our framework. This work is able to learn both explicit and implicit discourse relations; multilabel and multiclass tasks; and labeled data and unlabeled data in one framework, which makes it possible to fully take advantage of corpora like PDTB and RST as well as corpora developed using the Van Dijk schema.

[23] For a more extended analysis, see Appendix C

7 Conclusion

We have shown a state-of-the-art improvement of 4.9 Micro F1-score above baseline, from 62.8% F1-score to 67.7% F1-score, for discourse tagging on the *NewsDiscourse* dataset, the largest dataset currently available for Van Dijk discourse tagging. This dataset has a number of challenges: distinctions between discourse labels are complex and multifaceted and this dataset is class-imbalanced, with the overrepresented classes being 3 times more likely than the underrepresented classes.

We showed that a multitask approach is especially helpful in this circumstance, improving performance for underrepresented labels. One reason might be the high correlations observed between label predictions between tasks, indicating that auxiliary tasks are giving signal to our primary task's underrepresented labels. This includes a novel dataset that we introduce based on the same schema with some minor alterations. We show an additional benefit that our approach can reconcile datasets with slightly different schema, allowing NLP researchers not to "waste" valuable annotations.

Finally, we perform a comparative analysis of other strategies proposed in the literature for dealing with small datasets or class-imbalanced problems. We show in exhaustive experiments, in Appendix F, that these approaches do not help us improve above baseline. These negative experiments include extensive analyses and provide a justification for the necessity of our multitask approach.

8 Acknowledgements

This work was performed while Alexander Spangher interned at Bloomberg. We would like to thank Nanyun Peng and Temma Choji for generous and helpful discussions throughout the ideation and execution of this research. We would like to thank all of our anonymous reviewers for very thoughtful and helpful advice throughout the review process (including previous review cycles during which earlier versions of this paper was rejected). We would like to thank Ruihong Huang for generous discussions before this project started, and for sharing datasets with us. We would like to thank all the Bloomberg interns of the class of 2020, as well as students in Nanyun Peng and Jon May's labs for comments and feedback during public discussions of this work. Finally, the first author would like to thank Bloomberg for a generous 3 year fellowship, which made this research possible.

9 Ethics Statement

The source material for pre-existing annotated corpora and VD3, the annotation we provide, is either (1) archived by LDC, which, through an arrangement with the original publisher has licensed the data for use by members, (2) is granted a CC-by-attribution license and is released by Google Datasets, or (3) is collected by `archive.org` and is licensed freely for academic purposes. We only release annotations on the data, not the data itself. Annotation was done by the authors of this paper, who have been compensated for their work as part of their research roles. All the datasets are in the domain of news and news discourse and in formal news English; we would expect degradation of performance from that presented in this work were the models to be evaluated on other domains (i.e. non-news English or any domain of non-English) though the degree of the degradation has not been measured, as this work is chiefly concerned with English language news discourse.

References

Ali Haif Abbas. 2020. Politicizing the Pandemic: A Schemata Analysis of COVID-19 News in Two Selected Newspapers. *International Journal for the Semiotics of Law-Revue internationale de Sémiotique juridique*, pages 1–20.

Khalid Al Khatib, Henning Wachsmuth, Johannes Kiesel, Matthias Hagen, and Benno Stein. 2016. A News Editorial Corpus for Mining Argumentation Strategies. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3433–3443.

Peter C Austin and Elizabeth A Stuart. 2015. Moving Towards Best Practice When Using Inverse Probability of Treatment Weighting (IPTW) Using the Propensity Score to Estimate Causal Treatment Effects in Observational Studies. *Statistics in medicine*, 34(28):3661–3679.

David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. 2019. MixMatch: A Holistic Approach to Semi-Supervised Learning. In *NeurIPS*.

Joachim Bingel and Anders Søgaard. 2017. Identifying Beneficial Task Relations for Multi-Task Learning in Deep Deural Networks. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 164–169, Valencia, Spain. Association for Computational Linguistics.

Avrim Blum and Tom Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-Training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pages 92–100.

Ronald Cardenas, Ying Lin, Heng Ji, and Jonathan May. 2019. A Grounded Unsupervised Universal Part-of-Speech Tagger for Low-Resource Languages. 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 2428–2439, Minneapolis, Minnesota. Association for Computational Linguistics.

Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. 2003. Building a Discourse-Tagged Corpus in the Framework of Rhetorical Structure Theory. In *Current and new directions in discourse and dialogue*, pages 85–112. Springer.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of Text Generation: A Survey. *arXiv preprint arXiv:2006.14799*.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. 2002. SMOTE: Synthetic Minority Over-Sampling Technique. *Journal of artificial intelligence research*, 16:321–357.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. MixText: Linguistically-Informed Interpolation of Hidden Space for Semi-Supervised Text Classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157, Online. Association for Computational Linguistics.

Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long Short-Term Memory-Networks for Machine Reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561, Austin, Texas. Association for Computational Linguistics.

José M Chenlo, Alexander Hogenboom, and David E Losada. 2014. Rhetorical Structure Theory for Polarity Estimation: An Experimental Study. *Data & Knowledge Engineering*, 94:135–147.

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

Ronan Collobert and Jason Weston. 2008. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

David Craig. 2006. *The Ethics of the Story: Using Narrative Techniques Responsibly in Journalism*. Rowman & Littlefield.

Debopam Das, Tatjana Scheffler, Peter Bourgonje, and Manfred Stede. 2018. Constructing a Lexicon of English Discourse Connectives. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 360–365.

Debopam Das, Manfred Stede, and Maite Taboada. 2017. The Good, the Bad, and the Disagreement: Complex Ground Truth in Rhetorical Structure Analysis. In *Proceedings of the 6th Workshop on Recent Advances in RST and Related Formalisms*, pages 11–19.

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.

Terrance DeVries and Graham W Taylor. 2017. Dataset Augmentation in Feature Space. *arXiv preprint arXiv:1702.05538*.

Phillipa J Easterbrook, Ramana Gopalan, JA Berlin, and David R Matthews. 1991. Publication Bias in Clinical Research. *The Lancet*, 337(8746):867–872.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding Back-Translation at Scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Kathy Roberts Forde. 2007. Discovering the Explanatory Report in American Newspapers. *Journalism Practice*, 1(2):227–244.

Minsung Hyun, Jisoo Jeong, and Nojun Kwak. 2020. Class-Imbalanced Semi-Supervised Learning. *arXiv preprint arXiv:2002.06815*.

Masaru Isonuma, Junichiro Mori, and Ichiro Sakata. 2019. Unsupervised Neural Single-Document Summarization of Reviews via Learning Latent Discourse Structure and its Ranking. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2142–2152, Florence, Italy. Association for Computational Linguistics.

Samuli Laine and Timo Aila. 2017. Temporal Ensembling for Semi-Supervised Learning. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Man Lan, Jianxiang Wang, Yuanbin Wu, Zheng-Yu Niu, and Haifeng Wang. 2017. Multi-Task Attention-Based Neural Networks for Implicit Discourse Relationship Representation and Identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1299–1308.

Jaejun Lee, Raphael Tang, and Jimmy Lin. 2019. What Would Elsa Do? Freezing Layers During Transformer Fine-Tuning. *arXiv preprint arXiv:1911.03090*.

Wenqiang Lei, Xuancong Wang, Meichun Liu, Ilija Ilievski, Xiangnan He, and Min-Yen Kan. 2017. SWIM: A Simple Word Interaction Model for Implicit Discourse Relation Recognition. In *IJCAI*, pages 4026–4032.

Gaël Lejeune, Romain Brixtel, Antoine Doucet, and Nadine Lucas. 2015. Multilingual Event Extraction for Epidemic Detection. *Artificial intelligence in medicine*, 65(2):131–143.

Xiangci Li, Gully Burns, and Nanyun Peng. 2021. Scientific Discourse Tagging for Evidence Extraction. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2550–2562, Online. Association for Computational Linguistics.

Xiaoya Li, Xiaofei Sun, Yuxian Meng, Junjun Liang, Fei Wu, and Jiwei Li. 2020. Dice Loss for Data-imbalanced NLP Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 465–476, Online. 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.

Jian Liu, Yubo Chen, Kang Liu, and Jun Zhao. 2018. Event Detection via Gated Multilingual Attention Mechanism. In *Thirty-Second AAAI conference on artificial intelligence*.

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.

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2020. Multilingual Neural RST Discourse Parsing. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6730–6738.

Ruqian Lu, Shengluan Hou, Chuanqing Wang, Yu Huang, Chaoqun Fei, and Songmao Zhang. 2019. Attributed Rhetorical Structure Grammar for Domain Text Summarization. *arXiv preprint arXiv:1909.00923*.

Eric Malmi, Daniele Pighin, Sebastian Krause, and Mikhail Kozhevnikov. 2018. Automatic Prediction of Discourse Connectives. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Meryem M'hamdi, Marjorie Freedman, and Jonathan May. 2019. Contextualized Cross-Lingual Event Trigger Extraction with Minimal Resources. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 656–665, Hong Kong, China. Association for Computational Linguistics.

Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. 2016. V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. In *2016 fourth international conference on 3D vision (3DV)*, pages 565–571. IEEE.

Eleni Miltsakaki, Aravind Joshi, Rashmi Prasad, and Bonnie Webber. 2004. Annotating Discourse Connectives and Their Arguments. In *Proceedings of the Workshop Frontiers in Corpus Annotation at HLT-NAACL 2004*, pages 9–16.

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the Stability of Fine-tuning BERT: Misconceptions, Explanations, and Strong Baselines. In *9th International Conference on Learning Representations*, CONF.

Tahira Naseem, Benjamin Snyder, Jacob Eisenstein, and Regina Barzilay. 2009. Multilingual Part-of-Speech Tagging: Two Unsupervised Approaches. *Journal of Artificial Intelligence Research*, 36:341–385.

Aurélie Névéol, Aude Robert, Robert Anderson, Kevin Bretonnel Cohen, Cyril Grouin, Thomas Lavergne, Grégoire Rey, Claire Rondet, and Pierre Zweigenbaum. 2017. CLEF eHealth 2017 Multilingual Information Extraction task Overview: ICD10 Coding of Death Certificates in English and French. In *CLEF (Working Notes)*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Barbara Plank, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss. In *Proceedings of ACL 2016*. Association for Computational Linguistics (ACL).

Thierry Poibeau, Horacio Saggion, Jakub Piskorski, and Roman Yangarber. 2012. *Multi-Source, Multilingual Information Extraction and Summarization*. Springer Science & Business Media.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. 2008. The Penn Discourse TreeBank 2.0. In *LREC*. Citeseer.

Georg Rehm, Karolina Zaczynska, and Julián Moreno-Schneider. 2019. Semantic Storytelling: Towards Identifying Storylines in Large Amounts of Text Content.

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.

Sebastian Ruder. 2017. An Overview of Multi-Task Learning in Deep Neural Networks.

Sebastian Ruder and Barbara Plank. 2018. Strong Baselines for Neural Semi-Supervised Learning under Domain Shift. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1044–1054.

Attapol Rutherford and Nianwen Xue. 2016. Robust Non-Explicit Neural Discourse Parser in English and Chinese. In *Proceedings of the CoNLL-16 shared task*, pages 55–59.

Connor Shorten and Taghi M Khoshgoftaar. 2019. A Survey on Image Data Augmentation for Deep Learning. *Journal of Big Data*, 6(1):1–48.

Daniel Silva-Palacios, Cesar Ferri, and María José Ramírez-Quintana. 2017. Improving Performance of Multiclass Classification by Inducing Class Hierarchies. *Procedia Computer Science*, 108:1692–1701.

Catherine A Steele and Kevin G Barnhurst. 1996. The Journalism of Opinion: Network News Coverage of US Presidential Campaigns, 1968–1988. *Critical Studies in Media Communication*, 13(3):187–209.

Carole H Sudre, Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M Jorge Cardoso. 2017. Generalised Dice Overlap as a Deep Learning Loss Function for Highly Unbalanced Segmentations. In *Deep learning in medical image analysis and multimodal learning for clinical decision support*, pages 240–248. Springer.

Antti Tarvainen and Harri Valpola. 2017. Mean Teachers are Better Role Models: Weight-Averaged Consistency Targets Improve Semi-Supervised Deep Learning Results. In *Advances in neural information processing systems*, pages 1195–1204.

Teun A Van Dijk. 2013. *News as Discourse*. Routledge.

Jesper E Van Engelen and Holger H Hoos. 2020. A Survey on Semi-Supervised Learning. *Machine Learning*, 109(2):373–440.

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*, 30:5998–6008.

Bin Wang and C-C Jay Kuo. 2020. SBERT-WK: A Sentence Embedding Method by Dissecting BERT-Based Word Models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2146–2157.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Guihong Cao, Daxin Jiang, Ming Zhou, et al. 2020. K-adapter: Infusing knowledge into pre-trained models with adapters. *arXiv preprint arXiv:2002.01808*.

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.

Gregor Wiedemann, Seid Muhie Yimam, and Chris Biemann. 2018. A Multilingual Information Extraction Pipeline for Investigative Journalism. *arXiv preprint arXiv:1809.00221*.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised Data Augmentation for Consistency Training. *Advances in Neural Information Processing Systems*, 33.

W Victor Yarlott, Cristina Cornelio, Tian Gao, and Mark Finlayson. 2018. Identifying the Discourse Function of News Article Paragraphs. In *Proceedings of the Workshop Events and Stories in the News 2018*, pages 25–33.

Yu Zhang and Qiang Yang. 2017. A Survey on Multi-Task Learning. *arXiv preprint arXiv:1707.08114*.

Xinyi Zhou, Atishay Jain, Vir V Phoha, and Reza Zafarani. 2020. Fake News Early Detection: A Theory-Driven Model. *Digital Threats: Research and Practice*, 1(2):1–25.

A Appendices Overview

The appendices convey two broad areas of analysis: (1) Additional explanatory information for our multitask setup and (2) Negative Experiments and Results.

Appendix B contains more information on the datasets used, including labelsets for previously published work, the schema and annotation guidelines for the novel dataset we introduce, and processing information for RST and PDTB. Appendix C and D contain explanatory analysis. Appendix C shows that our multitask setup is reducing confusion between several important pairs of tags, giving further information and discussion beyond Figure 5 in the main body. Appendix D shows, for each tag, which α-weighting across tasks yields the highest score.

Appendix F provides more information about the negative results we obtained throughout our research and the explorations we performed, including details and mathematical definitions characterizing the additional experiments we ran. We believe that it is important to publish about negative results, to help fight against publication bias (Easterbrook et al., 1991) and to help other researchers considering similar techniques. Where possible, we conducted explorations to understand why such results were negative, and what hyperparameters might be tuned to produce a positive results.

B Additional Information on Multi-task Datasets

We summarize the tag-set in each of the datasets we used in Table 6. For all previously published datasets, the tag schema can be found in reference datasets.

B.1 Schema Definition Introduced in VD3

We provide additional information into VD3, the novel dataset we provide in this work. Tagging was done by the first author, who has worked at *The New York Times*, a major newspaper, for 4 years. We consider him an expert annotator, and as mentioned in Section 3, he checked his process by relabeling 10 articles from VD1, finding an inteannotator agreement of $\kappa = .69$.

The schema used for VD3 was based off the schema introduced by Van Dijk (2013). As such, the classification guidelines were:

Lede: A hook to engage the reader in the main event: can be an anecdote, question or observation.

Main Event: The major subject of the news report. It can be the most recent event that gave rise to the news report, or, in the case of an analytical news report, it can be a general phenomenon, a projected event, or a subject.

Consequence: An event or phenomenon that is caused by the main event or that directly succeeds the main event.

Previous Event: A specific event that occurred shortly before the main event. It either directly caused the main event, or provides context and understanding for the main event.

Circumstances: The general context or world-state immediately preceding the main event. Similar to **Previous Event**, but not necessarily tied to a specific event.

Secondary Event: An event occurring in parallel to the main event, also succeeding and/or being caused by previous events or circumstances, usually used discursively to illustrate a trend. For example, "lax oversight" (circumstance) might be the cause of "major oil spill #1" (main event), and also "minor oil spills #2, #3 and #4" (secondary events).

Historical Event: An event occurring more than 2 weeks prior to the main event. Might still impact or cause the main event, but is more distal.

Expectation: An analytical insight into future consequences or projections made by the journalist.

Evaluation: A summary, opinion or comment made by the journalist on any of the other discourse components.

Explanation: A comment or opinion made by the journalist or source seeking to either establish a causal relation or justify in some other manner why events are occurring.

Verbal Reaction: A comment made by a source in a news article that does not necessarily serve another discursive purpose. Note: VD2 discards this category and includes another dimension ($y_{i,2}$ = "Speech" or "Not Speech") on each sentence to capture this tag.

B.2 Additional Information on PDTB and RST Processing

Both PDTB and RST are relational discourse datasets, which provide span-level annotations and relational links between spans. We process each as shown in Figure 6 to better fit these datasets into our multitask framework. We process each so that relational labels are mapped onto sentences if a span

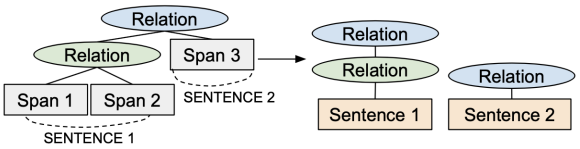


Figure 6: We processed the Penn Discourse Treebank and Rhetorical Structure Theory datasets, which are both hierarchical and relation-focused, to be sentence-level annotation tags.

exists within that sentence that is originally part of that relation. As shown in the figure, this holds for intersentential and intrasentential relations, and it results in a multilabel schema.

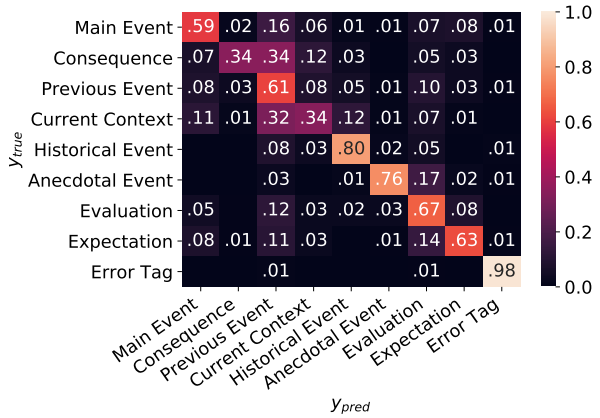
In Table 7, we show the heuristic mapping scheme that we developed to reduce the dimensionality of the RST dataset.

C Confusion Matrices

We identify two main classes of error, **Temporal** and **Event-based** error, from the confusion matrix shown in Figure 7a.

In the first case, **temporal** error, we observe confusion based on the temporal relation of events in discourse elements. For example, *Previous Events*, *Historical Events* and *Current Contexts* happen before the *Main Event*, while *Consequences* and *Expectations* happen after. The confusion between *Previous Event* and *Consequence* is one example of a temporal confusion, as is the confusion between *Expectation* and *Previous Event*. To address this confusion, we introduced a filtered down PDTB to include temporal relations. As can be seen in Table 4, PDTB-t is positively correlated with *Consequence*, and as shown in Table 5, PDTB-t contributes to temporal tags, like *Previous Event* and *Expectation*.

In the second case, **Event-based** error, we observe confusion between discourse elements with similar meaning except for the present or absence of an event. For example, *Current Context* and *Previous Event* contextualize a *Main Event*, but *Previous Event* contains the literal description of an event, while *Current Context* does not. A similar confusion can be seen between *Anecdotal Event* and *Evaluation*. We hypothesized that adding an KBP, a dataset specifically focused on identifying events, would reduce this error type, however, that was not observed. Further work tuning the size of the event dataset, or further tuning α, might yield more favorable results.



(a) **Baseline** confusion matrix (for RoBERTa +EmbAug.) Major classes of confusion are: (a)**Temporal**, ex. between *Consequence* and *Previous Event* (b) **Semantic**, ex. between *Current Context* and *Previous Event*.

(b) **MT-Macro** confusion matrix. We see a significant reduction in uncertainty for both event-based and temporal confusions.

Figure 7: Confusion Matrices for Baseline RoBERTa compared with MT-Macro.

Overall, the addition of the multitask datasets decreased confusion in these two main error classes, as shown in Figure 7b.

D Interrogating Multitask Dataset Contributions

In the main body of the paper, we interpreted the effects of the multitask setup by examining the overall increase in performance (Figure 2), the regressive effects of each dataset (Table 3) and the correlations between tag-predictions (Tables 4, 5). Another way to examine the contributions of each task is to analyze which combination of datasets results in the highest F1-score for each tag.

In Table 8, we show the α-weighting that results in the optimal F1-score for each tag. This gives us a sense of which datasets are important for that tag and how much of an improvement they give over

Schema Name	Tagset
Van Dijk Schema	{ *Lede*, *Main Event* (**M1**), *Consequence* (**M2**), *Circumstances* (**C1**), *Previous Event* (**C2**), *Historical Event* (**D1**), *Expectation* (**D4**), *Evaluation* (**D3**), *Verbal Reaction* }
VD2	Van Dijk $\oplus$ { *Anecdotal Event* (**D2**) }
VD3	Van Dijk $\oplus$ { *Explanation*, *Secondary Event* }
Argumentation	{ *Anecdote*, *Assumption*, *Common-Ground*, *Statistics*, *Testimony* }
Penn Discourse Treebank	{ *Temporal*, *Asynchronous*, *Precedence*, *Synchrony*, *Succession* }
Rhetorical Structure Theory	{ *Elaboration*, *Joint*, *Topic Change*, *Attribution*, *Contrast*, *Explanation*, *Background*, *Evaluation*, *Summary*, *Cause*, *Topic-Comment*, *Temporal* }
KBP Event Nugget	{ *Actual Event*, *Generic Event*, *Event Mention*, *Other* }

Table 6: Overview of the tagsets for each of the datasets used.

RST Tag-Class	RST Tags in Class
Attribution	Attribution, Attribution-negative
Evaluation	Evaluation, Interpretation, Conclusion, Comment
Background	Background, Circumstance
Explanation	Evidence, Reason, Explanation-argumentative
Cause	Cause, Result, Consequence, Cause-result
Joint	List, Disjunction
Comparison	Comparison, Preference, Analogy, Proportion
Manner-Means	Manner, Mean, Means
Condition	Condition, Hypothetical, Contingency, Otherwise
Topic-Comment	Topic-comment, Problem-solution, Comment-topic, Rhetorical-question, Question-answer
Contrast	Contrast, Concession, Antithesis
Summary	Summary, Restatement, Statement-response
Elaboration	Elaboration-additional, Elaboration-general-specific, Elaboration-set-member, Example, Definition, Elaboration-object-attribute, Elaboration-part-whole, Elaboration-process-step
Temporal	Temporal-before, Temporal-after, Temporal-same-time, Sequence, Inverted-sequence
Enablement	Purpose, Enablement
Topic Change	Topic-shift, Topic-drift

Table 7: The mapping we developed to reduce dimensionality of the RST Treebank. The left column shows the tag-class which we ended up using for classification and the right column shows the RST tags that we mapped to that category. We determined this tag-mapping heuristically.

the baseline MT-Micro.

For instance, a strong .3 weight for PDTB-t increases the performance for the *Expectation* tag and a strong .27 weight for RST increases the performance of the *Historical Event* tag. This is possibly because both the *Expectation* tag and the *Historical Event* tag describes events either far in the future or far in the past relative to the *Main Event*, and both PDTB-t and RST contain information about temporal relations.

Interestingly, and perhaps conversely, a strong α-weighting for the ARG dataset ($> .25$) increases performance for *Main Event*, *Previous Event*, and *Current Context*. This set of tags might seem counterintuitive, since they are all dealing with factual statements and events, and by definition contain less commentary and opinion than tags like *Expectation* and *Evaluation*. However, if we cross-reference Table 8 with Table 4, we see strong positive correlations between these tags and ARG tags like *Common Ground*, *Statistics* and *Anecdote*[24]

E Additional Explanatory Results for Single Task Experiments

E.1 Embedding Augmentations

We experiment with concatenating different embeddings to our sentence-level embeddings. These help us incorporate information on document-topic and sentence-position: headline embeddings (H_i) generated via the same method as sentence-embeddings; sentence-level positional embeddings (vanilla ($P_{i,j}$) and sinusoidal ($P^{(s)}_{i,j}$) (Vaswani et al., 2017)); document embeddings (D_i), and document arithmetic ($A_{i,j}$).[25]

[24] We were surprised that ARG's *Anecdote* tag does ***not*** correlate with VD2's *Anecdotal Event* tag, but perhaps the definitions are different enough that, despite the semantic similarity between the labels, they are in fact capturing different phenomena.

[25] D_i and $A_{i,j}$ are generated for sentence j of document i by using self-attention on input sentence-embeddings to

	ARG	VD3	VD1	RST	PDTB-t	KBP	Unsup.	**Tag F1-Score**	(MT-Micro)
Main Event	.28			.19			.05	58.37	(54.91)
Consequence			.18	.27			.09	40.00	(35.48)
Previous Event	.30			.010	.010			67.06	(63.76)
Current Context	.27	.09	.09				.09	38.75	(35.94)
Historical Event			.18	.27			.09	77.02	(73.71)
Anecdotal Event	.09	.09	.09	.09	.09		.09	75.84	(70.73)
Evaluation		.18	.09		.18		.09	74.78	(73.71)
Expectation	.010			.010	.30			68.94	(66.26)

Table 8: Maximum multitask weighting, α, by tag, for secondary datasets. **Tag F1-score** shows the maximum F1-score for the tag, and the left columns show the α that achieves this weighting. Right-most column is shown simply for comparison. Note that PDTB-t contributes most to *Expectation*, while Argumentation contributions most to *Main Event, Previous Event* and *Current Context*.

Embedding Augmentations	δ Micro F1
$\oplus P_i^{(s)} \oplus D_i \oplus A_i \oplus H_i$	.38
$\oplus P_i^{(s)} \oplus D_i \oplus A_i$	.37
$\oplus P_i^{(s)} \oplus D_i \oplus H_i$	.35
$\oplus P_i \oplus D_i \oplus A_i \oplus H_i$	.33
$\oplus H_i$	.11
$\oplus D_i$	.00
$\oplus D_i \oplus A_i$	.00
$\oplus P_i$	-.01
$\oplus P_i^{(s)}$	-.08

Table 9: **Sample** of embedding augmentation combinations. Micro F1-score increase gained by adding the embedding augmentation above **+Frozen**. $P_i^{(s)}$ is sinusoidal and P_i is vanilla positional embeddings. D_i is document embeddings and A_i is document embeddings arithmetic. H_i is headline embeddings.

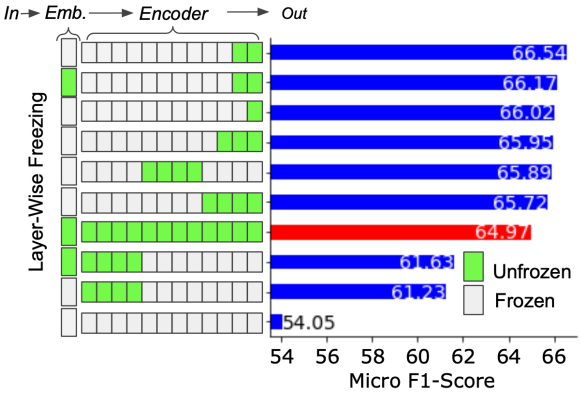


Figure 8: Here we show a sample of the different layer-wise freezing that we performed. "Emb." block is the embedding lookup table for word-pieces. "Encoder" blocks closer to the input are visualized on the left, and blocks to the right are closer to the output. The red bar indicates unfrozen RoBERTa.

Headline embeddings are generated for documents with a headline via the same method as sentence-embeddings, and treated as sentence 0. Vanilla positional embeddings and sinusoidal embeddings are as described in (Vaswani et al., 2017), but on the sentence-level rather than the word level.

Table 9 shows the results of these embedding augmentation experiments. As can be seen, these embeddings interact to increase accuracy: while no embedding along increases the accuracy, combinations of different additional embeddings have a higher increase in F1 improvement. Such augmentations, as we and others have demonstrated, are very important for document-level tasks such as discourse analysis, likely because they increase the amount of document-level information that is available (Choubey et al., 2020; Li et al., 2021).

generate a document-level embedding, and performing the following arithmetic: $D_i = \text{Self-Att}(\{S_{i,j}\}_{j=1}^{N_i})$, and $A_{i,j} = D_i * S_{i,j} \oplus D_i - S_{i,j}$, as described in Choubey et al. (2020). $S_{i,j}$ is the sentence-embedding for sentence j of document i, and self-attention is defined by Cheng et al. (2016).

E.2 Layer-Wise Freezing

As explored by Lee et al. (2019), layerwise-freezing for BERT-based architectures can have a dramatic effect on the training accuracy. This is especially true when the datasets are small. We experimented with freezing different layers of our RoBERTa architecture. As shown in Figure 8, we observed a 1.5 F1-score boost from freezing all but the top-two layers. We found that freezing combinations of higher-level layers yielded similar boosts, while freezing combinations of lower-level layers was detrimental. As suggested by Lee et al. (2019), this is likely due to the higher-level semantic information contained in the higher-level layers. This finding is especially relevant in a discourse task, where the labels convey abstract semantic information.

F Additional Negative Results

In this section, we describe additional negative experiments. We hope that by sharing our exploration in this Appendix, we might inspire researchers working with similar tasks to consider these methods, or advancements of them. Table 11 shows the results of the experiments described in this section.

F.1 Sentence Embedding Variations

There are, as of this writing, three different techniques that use BERT-based word embeddings to perform sentence-embeddings in the literature (i.e. they go beyond simply using BERT's [CLS] token): Sentence-BERT, Sentence Weighted-BERT (Reimers and Gurevych, 2019), and SBERT-WK (Wang and Kuo, 2020). Sentence-BERT trains a Siamese network to directly update the [CLS] token. Sentence Weighted BERT learns a weight function for the word embeddings in a sentence. SBERT-WK proposes heuristics for combining the word embeddings to generate a sentence-embedding.

None of the sentence-embedding variations yielded any improvement above the RoBERTa <s> token. It's possible that these models, which were designed and trained for NLI tasks, do not generalize well to discourse tasks. Additionally, we test two baselines: using the CLS token from BERT-base embeddings and generating sentence-embeddings using self-attention on Elmo word-embeddings, as described in (Choubey et al., 2020). These baselines show no improvement above RoBERTa. We see a need for a general pre-trained sentence embedding model that can transfer well across tasks. We envision a sort of masked-sentence model, instead of a masked-word model. Such a model would extend next sentence prediction (Devlin et al., 2019); instead of simply predicting the next sentence based on the previous embedding, we would predict arbitrarily masked sentences from a sequence of sentences, thus giving greater contextualization. We leave this direction to future research.

F.2 Supervised Head Variations

F.2.1 Classification Task Variations

For variations on the classification task, we consider using a Conditional Random Field layer instead of a simple FF layer, which has been shown to improve results (Li et al., 2021). However, we do not see an improvement in this case, possibly because the Bi-LSTM layer prior to classification already induces sequential information to be shared.

We also experiment with a hierarchical classification approach. Inspired by Silva-Palacios et al. (2017), we construct K clusters, $c_0, ..., c_k$, of semantically-related labels in labelset Y such that each class falls into one cluster of size $N_{c_0}, ...N_{c_k}$.[26] We construct variables from each class-label y_i: $\hat{y_i}^{(c)}$, $\hat{y_i}^{(c_0)}...\hat{y_i}^{(c_k)}$:

$$\hat{y_i}^{(c)} = \{\mathbb{1}(y_i \in \text{cluster } j)\}_{j=1}^{K}$$
$$\hat{y_i}^{(c_0)} = \{\mathbb{1}(y_i = l)\}_{l=1}^{N_{c_0}}$$
$$...$$
$$\hat{y_i}^{(c_k)} = \{\mathbb{1}(y_i = l)\}_{l=N_{c_0}+...+N_{c_{k-1}}}^{N_{c_0}+...+N_{c_k}}$$

where $L = N_{c_0} + ... + N_{c_{k-1}}$ is the original number of labels. We try modeling these variables two ways. (1) As a 2-level hierarchy, where the top-level, $\hat{y_i}^{(c)}$, is one task and each sublayer, $\hat{y_i}^{(c_0)}...\hat{y_i}^{(c_k)}$, is a separate task or (2) as a multilabel classification task of $\hat{y_i}$, where $\hat{y_i} = \hat{y_i}^{(c)} \oplus \hat{y_i}^{(c_0)} \oplus ... \oplus \hat{y_i}^{(c_k)}$.

Our hierarchical classification shows no improvement above vanilla multiclass classification. It's possible that the transformer architecture is already learning the label hierarchy implicitly, and the information we try to pass in by structuring the output space does not improve the prediction.

F.2.2 Loss Variations

Method	Mac.	Mic.
GDL	55.45	64.41
GDL$^{(2)}$	49.90	62.82
GADL	29.39	41.97
(MT-Micro)	*61.89*	*67.70*

Table 10: Macro-F1 (Mac.) and Micro F1 (Mic.) scores for variations of Multiclass Dice Loss. DL: Vanilla Dice Loss, $DL^{(2)}$: the Square Form of Dice Loss, ADL: self-adjusting dice loss (Li et al., 2020). Multiclass generalized as in (Sudre et al., 2017).

We consider losses other than a vanilla Cross-Entropy loss for the multiclass tasks and Binary Cross-Entropy loss for the multilabel tasks. Specifically, we experiment with variations of Dice Loss for the multiclass tasks, which has been proposed for class-imbalanced classification problems in computer vision (Milletari et al., 2016) and NLP

[26] Semantic-relatedness is given *a priori* by the tag definitions (Yarlott et al., 2018; Choubey et al., 2020).

	M1	M2	C2	C1	D1	D2	D3	D4	E	F1-Macro	F1-Micro
SBERT	52.0	11.2	61.7	31.1	67.9	43.1	69.9	64.9	96.6	55.39	63.38
+Frozen	54.8	19.3	62.6	29.9	70.2	53.5	70.0	61.8	96.2	57.59	64.14
+EmbAug	54.6	25.0	62.8	33.0	69.8	45.7	71.9	65.2	95.7	58.20	64.95
SWBERT	51.3	14.5	61.3	30.2	70.1	55.1	71.2	64.3	97.0	57.23	64.14
+Frozen	52.4	20.6	62.6	31.5	68.7	61.1	73.9	66.0	95.9	59.17	65.62
+EmbAug	52.2	12.0	64.6	31.7	72.2	50.0	73.0	66.8	96.7	57.68	65.79
Hier.	47.5	0.0	59.4	24.3	68.3	66.0	71.6	63.8	91.3	54.68	62.51
Dice	55.4	18.5	63.7	29.5	70.8	25.2	72.9	64.2	95.6	55.09	64.41
CRF	54.6	16.4	62.8	30.0	70.1	65.5	72.3	64.2	96.2	59.13	65.43
class scale	53.8	33.8	62.1	32.1	71.4	68.5	72.8	65.5	95.9	61.76	66.06
MT-Mic	*55.35*	*25.0*	*67.06*	*32.78*	*72.5*	*68.88*	*73.63*	*65.8*	*96.0*	*61.89*	*67.70*

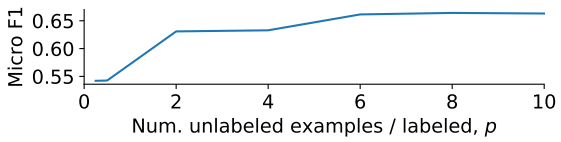
Table 11: **Negative Results:** We show the results of experiments and manipulations that did not increase the accuracy of our model. For all variations that we report, we report the maximum score observed under an array of hyperparameter settings. All of these tasks include +Freezing and +EmbAug.

(Li et al., 2020). Dice Loss seeks to directly optimize F1-score. It differentiates F1, with binary $y_i \in \{0,1\}$, by defining precision of a single prediction as $\text{Prec}(x_i) = p(y_i = 1|x_i) = p_i$, and recall as $\text{Recall}(x_i) = y_i$. Then, $F1(x_i) = \frac{2\text{Prec}(x_i)\times\text{Recall}(x_i)}{\text{Prec}(x_i)+\text{Recall}(x_i)} = \frac{2p_{i,1}y_{i,1}}{p_{i,1}+y_{i,1}} = \text{Dice Score}(x_i)$.

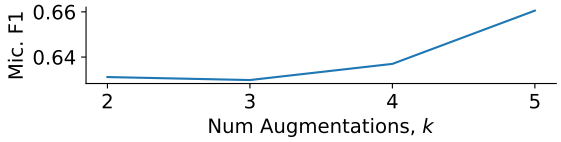
Across a dataset, Binary Dice Loss can be expressed as $DL(X) = 1 - \frac{2\sum_i p_{i,1}y_{i,1}+N\gamma}{\sum_i p_{i,1}+\sum_i y_{i,1}+N\gamma}$, where γ is a hyperparameter (typically $\gamma = 1$) to ensure that negative examples ($y_i = 0$) also contribute to the loss. Binary Dice loss can also be expressed in the square form (Milletari et al., 2016), $DL^{(2)}(X) = 1 - \frac{2\sum_i p_{i,1}y_{i,1}+N\gamma}{\sum_i p_{i,1}^2+\sum_i y_{i,1}^2+N\gamma}$. Additionally, (Li et al., 2020) proposed a self-adjusting Binary Dice Loss (ADL) by multiplying p_i by $(1-p_i)$ to downweight "easy" examples, or examples where p_i is close to 0 or 1.

A multiclass Dice Loss for k classes can be derived either through macro-averaging, micro-averaging, or a squared sum, $\text{GDL}(X) = \sum_{j=1}^{k} \frac{1}{N_j^2} * DL(p_j, y_j)$, introduced by (Sudre et al., 2017). As shown in Table 10, Dice Loss (DL) and Self-Adjusting Dice Loss (SDL) fail to improve above Cross-Entropy Loss. The top-scoring loss was the Vanilla DL formulation, with Sudre et al. (2017)'s generalization scheme. In all trials, DL and DL(2) are comparable but SDL underperforms.

The addition in ADL of the term $(1-p_{i,1})$ downweights tags that the model is more confident about. This idea has a similar aim as TSA (Xie et al., 2020), which excludes high-confidence predictions. The model becomes more confident as it is trained further; however, under ADL, it thus gets downweighted more. It's possible that with a TSA-like decay schedule, ADL would not underperform.



(a) The effect of different dataset-size ratios on F1-score, p.



(b) The effect of different numbers of augmentations per each unlabeled datapoint on F1-score, k.

Figure 9: We investigate two different proportions of unlabeled dataset size. While we have found p's plateau, we have not yet found k's.

F.3 Multitask Head Freezing

Additionally, we experiment with freezing auxiliary heads (heads for tasks that are not VD2) in order to propagate more of the gradient into the shared layers. Note, according to Figure 1, that this is only the FF layer, which is not a major architectural change. We find that this yields no improvement.

G Unsupervised Data Augmentation: Analysis

Semi-supervised learning approaches can often achieve high accuracy with a only a small labeled dataset (Van Engelen and Hoos, 2020). For instance, Blum and Mitchell (1998) achieve a 95% accuracy with a labeled set 63 times smaller than their unlabeled set. However, there are cases, such as in domain-shifted settings, where more unlabeled data might hurt semisupervised training (Ruder and Plank, 2018).

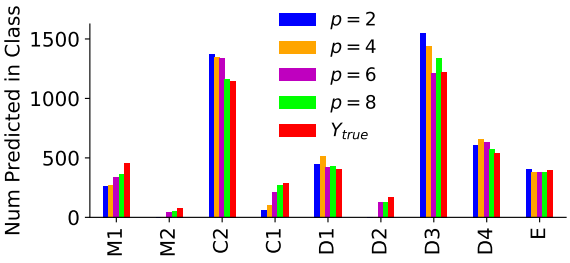


Figure 10: The effect of increasing the unsupervised/supervised dataset ratio p, on class predictions. As the p increases, the number of underrepresented classes predicted approaches the true number, Y_{true}.

G.1 Dataset size exploration

In their original paper, (Xie et al., 2020) do not give insight into how many unlabeled datapoints researchers should use in their semisupervised setups. Here we explore that by varying the size of our semi-supervised dataset in two dimensions: (a) the size of the unlabeled set relative to the labeled set, p and (b) the number of augmentations, k per unlabeled datum. We show our results in Figure 9. As shown in Figure 9a, we reach a plateau between $p = 6$–10. We do not observe a plateau for the number of augmentations per datum (Figure 9b).

We hypothesize that the effect of increasing p is to help the model better predict underrepresented classes. As shown in Figure 10, as p increases, UDA is much better able to generalize the data manifold, and not to overpredict overrepresented classes. We did not explore, however, the effects of varying p for different classes; there are still many underepresented classes where the optimal p is higher than 10.

We hypothesize that adding more augmentations per unlabeled datapoint is helpful in training because more augmentations might help the model more robustly explore the region of space around each unlabeled datapoint, thus mapping that region better. It's also possible that with more augmentations, say, $k = 10, 20, 30$, we would have enough signal to propagate to even more unlabeled data. We leave this question to future work.

G.2 Learning-techniques

Minimizing consistency loss, as a specific approach to semi-supervised learning, has been explored prior to the proposal of UDA, most notably with the Mean Teacher method (Tarvainen and Valpola, 2017) and the Π method (Laine and Aila, 2017), and UDA mirrors such methods in the core optimization setup. However, simply minimizing consistency loss along with supervised loss fails to converge to a global optimum.

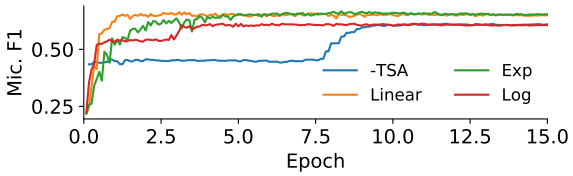


Figure 11: F1-Score on evaluation data across training for UDA with different TSA training schedules: Linear, Exponential and Log. UDA without TSA is shown for comparison.

To address this problem, Xie et al. (2020) introduce curriculum learning techniques, including: (1) Training Signal Annealing (TSA), (2) softmax temperature sharpening, and (3) confidence-based thresholding. Authors do not show how parameters to these effect the training output, so we produce an analysis here. Based on our analysis, we find that their most important tool for our task is TSA.

G.2.1 TSA

TSA is defined as:

$$\min_{\theta} \frac{1}{Z} \sum_{x,y^* \in B} \left[-\mathbb{1}\left[p_\theta(y^*|x) < \eta_t\right] \log p_\theta(y^*|x)\right]$$

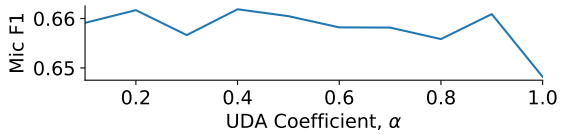
$$Z = \sum_{x,y^* \in B} \mathbb{1}\left[p_\theta(y^*|x) < \eta_t\right]$$

In other words, training examples are only considered if the model is confidet in them, $p_\theta(y^*|x)$, is above a certain value, η_t. η_t is increased throughout training; it is set to $\eta_t = \alpha_t * (1 - \frac{1}{K}) + \frac{1}{K}$, where K is the number of classes ($y \in \{1, ..., K\}$) and α_t increases either with a linear ($\frac{t}{T}$), log ($1 - \exp(-\frac{t}{T} * 5)$) or exponential ($\exp((\frac{t}{T} - 1) * 5)$) schedule, where T is the number of training iterations.

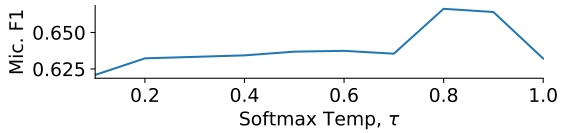
We show the results of using a linear-decay schedule, an exponential-decay schedule and a logarithmic-decay schedule in Figure 11. As can be seen, both linear-decay and exponential decay achieve the same optimum, but the linear schedule arrives faster; the log schedule achieves the same optimum as UDA without TSA.

G.2.2 UDA Coefficient, ζ

We show the effects of other UDA hyperparameter tuning in Figure 12. The UDA coefficient, ζ, shown

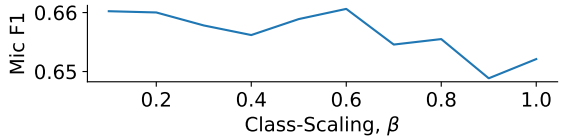


(a) Effect of increasing the weight of the unsupervised head in UDA relative to the supervised head.



(b) Effect of UDA's softmax temperature parameter on Micro F1-score.

(c) Effect of UDA's confidence threshold, r on Micro F1-score.



(d) Effect of SCL's class-scaling parameter, β, on Micro F1-Score (Hyun et al., 2020).

Figure 12: Different hyperparameter configurations and their effect on the Micro F1 for VD2. Overall, parameters either do not affect the performance (like r) or hurt performance. For example, when ζ is 0, then UDA is a supervised task; when β is 1, SCL is just CL.

in Figure 12a, simply weights the consistency loss contribution to the overall loss:

$$L_{UDA} = L_{CE} + \alpha L_{con}$$
$$L_{con} = \mathbb{E}_k[D(p_\theta(x_i)|p_\theta(\hat{x}_{i,k}))]$$

where L_{CE} is cross-entropy loss and k is the number of data augmentations. A lower ζ, which results in a higher-performing model, corresponds to less contribution by consistency loss.

G.2.3 Softmax Temperature, τ

The next two parameters, softmax temperature, τ and confidence threshold, r are designed to increase the weight of the unlabeled dataset. The softmax temperature sharpens the predictions on original unlabeled datapoint (or in one implementation, the augmented datapoint[27]) through the following operation:

$$p_\theta^{(sharp)}(y|x) = \frac{\exp(z_y/\tau)}{\sum_{y'}\exp(z_{y'}/\tau)}$$

where z_y is the logit output of the neural network. So, a lower temperature increases the values in each exponent, and sharpens the probability distribution over the classes, resulting in a higher consistency loss. According to Figure 12b, the performance increases as τ increases, peaking at .8.

G.2.4 Confidence Threshold, r

The confidence threshold, r, masks out predictions on unlabeled data that the model is not confident about.

$$L_{UDA} = L_{CE} + \mathbb{I}(\max_{y'} p_\theta(y'|x) > r)L_{con}$$

(It is important to note that TSA does exactly the same thing but in reverse, but TSA is on the supervised data while the confidence threshold is on the unlabeled data.) There is essentially no pattern observed between changing r and the model performance, according to Figure 12c.

G.2.5 Suppressed Consistency Loss (SCL)

We try a simple alteration to semi-supervised learning with consistency loss (UDA) called *suppressed consistency loss*, (SCL). SCL was suggested by Hyun et al. (2020) to reduce the impact of consistency training on lower-represented classes, where, authors claim, the manifold of the latent space is underlearned and semi-supervised learning can be harmful..

$$L_{SCL}(X_i) = g(N_c) * L_{con}(X_i)$$

where $c = \text{argmax}(f_\theta(X_i))$ and $g(z)$ is a function inversely proportional to z: $g(z) = \beta^{1-\frac{z}{N_{max}}}$ (with $\beta \in (0, 1]$). N_c is the number of training samples in the class predicted by the model and N_max is the number of samples of the class with the most frequency.

The higher β is the more class imbalance is used to downweight consistency loss. As can be seen, performance roughly increases as β approaches 0, indicating that suppressed consistency loss is not helpful.

[27] https://github.com/SanghunYun/UDA_pytorch/blob/master/main.py#L113