

Toward Annotation Efficiency in Biased Learning Settings for Natural Language Processing

Thomas Effland

Submitted in partial fulfillment of the  
requirements for the degree of  
Doctor of Philosophy  
under the Executive Committee  
of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2023

© 2022

Thomas Effland

All Rights Reserved

## **Abstract**

Toward Annotation Efficiency in Biased Learning Settings for Natural Language Processing

Thomas Effland

The goal of this thesis is to improve the feasibility of building applied NLP systems for more diverse and niche real-world use-cases of extracting structured information from text. A core factor in determining this feasibility is the cost of manually annotating enough unbiased labeled data to achieve a desired level of system accuracy, and our goal is to reduce this cost. We focus on reducing this cost by making contributions in two directions:

1. Easing the annotation burden by leveraging high-level expert knowledge in addition to labeled examples, thus making approaches more annotation-efficient.
2. Mitigating known biases in cheaper, imperfectly labeled real-world datasets so that we may use them to our advantage.

A central theme of this thesis is that high-level expert knowledge about the data and task can allow for biased labeling processes that focus experts on only manually labeling aspects of the data that cannot be easily labeled through cheaper means. This combination allows for more accurate models with less human effort. We conduct our research on this general topic through three diverse problems with immediate applications to real-world settings.

First, we study an applied problem in biased text classification. We encounter a rare-event text classification system that has been deployed for several years. We are tasked with improving this system's performance using only the severely biased incidental feedback provided by the

experts over years of system use. We develop a method that combines importance weighting and an unlabeled data imputation scheme that exploits the selection-bias of the feedback to train an unbiased classifier without requiring additional labeled data. We experimentally demonstrate that this method considerably improves the system performance.

Second, we tackle an applied problem in named entity recognition (NER) concerning learning tagging models from data that have very low recall for annotated entities. To solve this issue we propose a novel loss, the Expected Entity Ratio (EER), that uses an uncertain estimate of the proportion of entities in the data to counteract the false-negative bias in the data, encouraging the model to have the correct ratio of entities in expectation. We justify the principles of our approach by providing theory that shows it recovers the true tagging distribution under mild conditions. Additionally we provide extensive empirical results that show it to be practically useful. Empirically, we find that it meets or exceeds performance of state-of-the-art baselines across a variety of languages, annotation scenarios, and amounts of labeled data. We also show that, when combined with our approach, a novel sparse annotation scheme can outperform exhaustive annotation for modest annotation budgets.

Third, we study the challenging problem of syntactic parsing in low-resource languages. We approach the problem from a cross-lingual perspective, building on a state-of-the-art transfer-learning approach that underperforms on “distant” languages that have little to no representation in the training corpus. Motivated by the field of syntactic typology, we introduce a general method called Expected Statistic Regularization (ESR) to regularize the parser on distant languages according to their expected typological syntax statistics. We also contribute general approaches for estimating the loss supervision parameters from the task formalism or small amounts of labeled data. We present seven broad classes of descriptive statistic families and provide extensive experimental evidence showing that using these statistics for regularization is complementary to deep learning approaches in low-resource transfer settings.

In conclusion, this thesis contributes approaches for reducing the annotation cost of building applied NLP systems through the use of high-level expert knowledge to impart

additional learning signal on models and cope with cheaper biased data. We publish implementations of our methods and results, so that they may facilitate future research and applications. It is our hope that the frameworks proposed in this thesis will help to democratize access to NLP for producing structured information from text in wider-reaching applications by making them faster and cheaper to build.

## Table of Contents

Acknowledgments . . . . .	xi
Dedication . . . . .	xii
Chapter 1: Introduction . . . . .	1
1.1 Background . . . . .	3
1.2 Three Real-World Problems and Solutions . . . . .	8
1.3 Contributions . . . . .	11
1.4 Organization . . . . .	13
Chapter 2: Improving a Rare-Event Classification System with Minimal Wasted Labels . .	14
2.1 Introduction . . . . .	14
2.1.1 The Application Domain . . . . .	14
2.1.2 The Technical Problem . . . . .	16
2.2 Materials and Methods . . . . .	18
2.2.1 Yelp System Design . . . . .	18
2.2.2 Classification Methods . . . . .	19
2.2.3 Enhanced Dataset . . . . .	20
2.2.4 Selection Bias Correction for Training and Test Metrics . . . . .	20
2.2.5 Training Regimes . . . . .	24

2.3	Evaluation . . . . .	25
2.3.1	Hyperparameters . . . . .	26
2.4	Results . . . . .	27
2.4.1	Classification Evaluation . . . . .	27
2.4.2	Precision-Recall Tradeoff . . . . .	28
2.4.3	Error analysis of best “Sick” classifier . . . . .	31
2.4.4	Error analysis of best “Multiple” classifier . . . . .	32
2.5	Related Work . . . . .	33
2.5.1	Public Health Informatics using Social Media . . . . .	33
2.5.2	Rare-Event Classification . . . . .	34
2.5.3	Accounting for Sampling Bias . . . . .	35
2.6	Discussion . . . . .	36
2.7	Conclusion . . . . .	37
Chapter 3: Partially Supervised Named Entity Recognition via the Expected Entity Ratio .		38
3.1	Background and Motivation . . . . .	39
3.2	Methods . . . . .	41
3.2.1	Problem Setup and Notation . . . . .	41
3.2.2	Tagging Model . . . . .	43
3.2.3	Supervised Marginal Tag Loss . . . . .	44
3.2.4	Expected Entity Ratio Loss . . . . .	45
3.2.5	Combined Objective and Consistency . . . . .	47
3.3	Benchmark Experiments . . . . .	51

3.3.1	Corpora . . . . .	51
3.3.2	Simulated Annotation Scenarios . . . . .	52
3.3.3	Approaches . . . . .	54
3.3.4	Preprocessing . . . . .	56
3.3.5	Hyperparameters . . . . .	57
3.3.6	Results . . . . .	58
3.3.7	Analysis of EER hyperparameters . . . . .	61
3.4	EE vs. Exhaustive Experiments . . . . .	63
3.4.1	Annotation Speed User Study . . . . .	63
3.4.2	Performance Learning Curves . . . . .	64
3.5	Related Work . . . . .	66
3.6	Conclusions . . . . .	67
Chapter 4: Improving Low-Resource Cross-lingual Parsing with Expected Statistic Regularization . . . . .		69
4.1	Background and Motivation . . . . .	70
4.2	Expected Statistic Regularization . . . . .	72
4.2.1	Semi-Supervised Objective . . . . .	73
4.2.2	The Statistic Function $f$ . . . . .	74
4.2.3	The Distance Function $\ell$ . . . . .	75
4.3	Choosing the Targets and Margins . . . . .	76
4.4	Application to Cross-Lingual Parsing . . . . .	77
4.4.1	Problem Setup and Data . . . . .	78
4.4.2	The Model and Training . . . . .	78

4.4.3	Typological Statistics as Supervision . . . . .	80
4.4.4	The Need for Entropy Regularization . . . . .	82
4.4.5	The SST Relaxation for Optimizing Intractable Expected Statistics . . . . .	83
4.5	Oracle Unsupervised Experiments . . . . .	85
4.5.1	Experimental Setup . . . . .	85
4.5.2	Assessing the Proposed Statistics . . . . .	87
4.5.3	Ablation Studies . . . . .	90
4.6	Realistic Semi-Supervised Experiments . . . . .	91
4.6.1	Learning Curves . . . . .	91
4.6.2	Low-Resource Transfer . . . . .	97
4.7	Related Work . . . . .	99
4.7.1	Weak Supervision . . . . .	99
4.7.2	Cross-Lingual Transfer . . . . .	100
4.8	Conclusion . . . . .	100
Chapter 5: Conclusions . . . . .		102
5.1	Contributions . . . . .	103
5.2	Limitations and Future Work . . . . .	105
Bibliography . . . . .		109
Appendix A: Partially Supervised Named Entity Recognition via the Expected Entity Ratio		125
A.1	Full Benchmark Results . . . . .	125
A.1.1	Non-Native Speaker . . . . .	125

A.1.2	Exploratory Expert . . . . .	128
Appendix B:	Improving Low-Resource Cross-lingual Parsing with Expected Statistic Reg- ularization . . . . .	132
B.1	Detailed Learning Curve Results . . . . .	132

## List of Figures

2.1	<i>Precision-Recall Tradeoffs.</i> Precision-recall curves of “Sick” logistic regression models in the high-recall region. While the “Biased” logistic regression performance lags below, the “Gold” and “Silver” models show relatively mild losses in precision per point of recall gained until the 90-100% recall region. After 92% recall the “Gold” model begins to experience a steep drop in precision while the “Silver” model does not experience a steep drop in precision until a recall of 98%.	30
3.1	An example low-recall sentence with two entities (one is missing) and its NER tags. The Gold row shows the true tags, the Raw row shows a false negative induced by the standard “tokens without entity annotations are non-entities” assumption, and the Latent row reflects our view of unannotated tags as latent variables. . . . .	40
3.2	Test performance as a function of the number of observed training annotations for the Exhaustive vs. EE annotation on CoNLL English. Lines are averages and shaded regions are $\pm 1$ standard error. . . . .	65
4.1	<i>Adaptive Loss Function Visualization.</i> Example statistic sampling distributions (sample size = 8) and their induced loss functions. The histograms show the sampling distribution of statistics for POS tag frequency for three POS tags: PROP, DET, and CONJ. The solid orange lines show the loss functions induced by these distributions. The samples have decreasing variance from left to right, and the respective losses adaptively become narrower in response. The vertical dotted lines show the $t_k \pm \sigma_k$ boundaries where the $\ell_k$ switches from a scaled L2 to an unscaled L1. . . . .	77
4.2	<i>Multi-Source UDPRE Transfer Learning Curves.</i> Baseline approaches are dotted, while ESR variants are solid. All curves show the average of 15 runs across 5 different languages with 3 randomly sampled labeled datasets per language. The plots indicate a significant advantage of ESR over the baselines in low-data regions.	93

- 4.3 “*From Scratch*” mBERT *Transfer Learning Curves*. Baseline approaches are dotted, while ESR variants are solid. All curves show the average of 15 runs across 5 different languages with 3 randomly sampled labeled datasets per language. The plots indicate a significant advantage of ESR over the baselines in low-data regions. 96

## List of Tables

2.1	<i>Model Performance on “Sick” Task.</i> The bold value represents the final selected model from among the variants. This is the model we further analyze in the error analysis. Because the bootstrap distribution of some test statistics exhibited non-normal behavior, their corresponding confidence intervals are wider. . . . .	28
2.2	<i>Model Performance on “Multiple” Task.</i> The bold value represents the final selected model from among the variants. This is the model we further analyze in the error analysis. . . . .	29
2.3	<i>“Sick” Task Confusion Matrix.</i> . . . . .	29
2.4	<i>“Multiple” Task Confusion Matrix.</i> . . . . .	29
3.1	<i>Dataset Statistics.</i> Each row is organized by dataset with CoNLL03 in the top group and Ontonotes5 below. The first column shows the number of entity tags $ \mathcal{Y} $ . The next three columns give the total number of tokens in the train, dev, and test splits for each dataset. The final six columns give the total number of annotated entity tokens in the training split for the gold and simulated low-recall datasets from Section 3.3.2, along with their observed EERs as percentages. . . . .	51
3.2	Benchmark test set F1 scores across different languages and annotation scenarios. Best models in bold. † indicates that for EE the test F1 score is statistically significantly better than SNS-BERT-shortest ( $p < 0.01$ ) (details in footnote 7). Other pairs between SNS-BERT-shortest and EER-BERT-short/shortest were not significant. . . . .	59
3.3	CoNLL English EE EER-short test set F1 across three randomly sampled datasets. *: $\rho = \rho^*$ . †: benchmark experiment setting. . . . .	62
4.1	Treebanks details for train (top) and test (bottom) sets in UD-13. *: the original Russian-SynTagRus dataset has 48.8k sentences, which we down-sample to the same 15k sentences as Ustun et al. (2020) to reduce training time and balance the data. . . . .	86

4.2	<i>Unsupervised Constraint Variant Results.</i> (Top): Baseline methods that do not use ESR. (Bottom): Various statistics used by ESR as unsupervised loss on top of UDPRE. Scores are measured on target treebank development (not test) sets. Bold rows mark statistics used in later experiments. (*): All statistics with * are intractable and utilize the SST relaxation of Paulus et al. (2020). (†): All statistics except those with † also include left/right directional information – those with a † do not have directional information. . . . .	88
4.3	<i>Loss Aggregation Ablation Results.</i> Loss per batch outperforms loss per sentence for both POS and LAS on average. . . . .	91
4.4	<i>Loss Function Ablation Results.</i> The Smooth L1 loss outperforms the other simpler loss variants for both POS and LAS, averaged over 5 languages. . . . .	91
4.5	<i>Low-Resource Semi-Supervised Transfer Results.</i> “FT” refers to the UDPRE-FT fine-tuning baseline, “ESR” refers to our UDPRE-ESR-CLD approach, and $\Delta$ refers to the absolute difference of ESR minus FT. Best performing methods are in bold. . . . .	98
A.1	CoNLL English, Non-native Speaker . . . . .	125
A.2	CoNLL German, Non-native Speaker . . . . .	125
A.3	CoNLL Spanish, Non-native Speaker . . . . .	126
A.4	CoNLL Dutch, Non-native Speaker . . . . .	126
A.5	Ontonotes English, Non-native Speaker . . . . .	126
A.6	Ontonotes Chinese, Non-native Speaker . . . . .	126
A.7	Ontonotes Arabic, Non-native Speaker . . . . .	127
A.8	CoNLL English, Exploratory Expert . . . . .	128
A.9	CoNLL German, Exploratory Expert . . . . .	128
A.10	CoNLL Spanish, Exploratory Expert . . . . .	129
A.11	CoNLL Dutch, Exploratory Expert . . . . .	129
A.12	Ontonotes English, Exploratory Expert . . . . .	130
A.13	Ontonotes Chinese, Exploratory Expert . . . . .	130

A.14 Ontonotes Arabic, Exploratory Expert . . . . .	131
B.1 UDPRE <i>Semi-Supervised Learning Curves</i> . . . . .	133
B.2 MBERT <i>Semi-Supervised Learning Curves</i> . . . . .	134

## **Acknowledgements**

I am incredibly grateful to my advisor Mike Collins. He always allowed me to have total freedom over my choice of project topics, yet was engaged and offered vital and incisive guidance when I needed it most. His influence has made me both a better writer and a more precise mathematician.

I found the community at Columbia to be an immensely stimulating environment. I would like to thank Chris Kedzie, Giannis Karamanolakis, and Lampros Flokas for countless though-provoking conversations. I would also like to thank Luis Gravano, Daniel Hsu, Eugene Wu, Dave Blei, John Cunningham, the many members of the NLP reading group, Dave Blei's machine learning reading group, the database group, and the broader faculty for generously sharing their vastly deep and varied perspectives with me over the years. These experiences challenged me to expand the scope my understanding in so many directions.

Finally, I would like to thank my family and friends for their steadfast support along this journey. I am truly grateful.

## **Dedication**

To my wife, Farah, and my parents, Anne and Bill.

Thank you for everything.

## Chapter 1: Introduction

In recent decades, much of the world’s information has become digitized, amassing unfathomable stores of raw text data in the form of natural language. This massive volume of readily available unstructured text holds within it the promise for vastly expanding human knowledge. The potential applications are virtually infinite: political analysis (Laver et al., 2003; Grimmer and Stewart, 2013; Glavas et al., 2019; Nanni et al., 2021), disaster response (Palen et al., 2010; Imran et al., 2013; Kedzie et al., 2015; Alam et al., 2018), clinical health (Zhou et al., 2014; Pradhan et al., 2015; Halpern et al., 2016; OliverJ.BearDon’tWalk et al., 2021), public health (Gesualdo et al., 2013; Harris et al., 2014; Eichstaedt et al., 2018; Karamanolakis et al., 2019c), consumer opinion/sentiment (Ghani et al., 2006; Putthividhya and Hu, 2011; Petrovski and Bizer, 2017; Çataltaş et al., 2020; Karamanolakis et al., 2020; Fong et al., 2021), and law (Li et al., 2012; Sim et al., 2016; Aletras et al., 2016; Chalkidis et al., 2020), just to name a few. Some of the greatest potential for this information to create value is in areas of specialized knowledge-work where experts would like to comb through significant amounts of written information to help them do their jobs.

Yet, grappling with the scale of this data is the main bottleneck to using large text corpora for producing knowledge (Bush et al., 1945). Individuals and organizations simply do not have enough time and resources to read and organize all of the required information buried in the relevant mountains of text. With their incredible speed and ability to scale, computational solutions that use natural language processing (NLP) to extract structured knowledge from unstructured text are a promising tool for overcoming this scaling challenge.

The goal of this thesis is to improve the feasibility of building NLP systems for more diverse and niche real-world use-cases. A core factor in determining this feasibility is the cost of manually annotating enough unbiased labeled data to achieve a desired level of system accuracy,

and our goal is to reduce that cost. Standard approaches using supervised learning can easily become prohibitively expensive for new applications where practitioners do not have sufficient resources (Dahlmeier, 2017). In these “low-resource” scenarios, practitioners need solutions that:

1. Have high “annotation-efficiency”, requiring fewer manually annotated data to achieve sufficient levels of accuracy.
2. Allow for usage of imperfectly labeled data for training models. That is, data that are biased in the sense that they do not come from the same distribution of inputs and labels as the data that will be seen in practice. This is necessary because it is often possible to find valuable but imperfectly labeled datasets at significantly lower cost than manual annotation. Some examples include datasets that come from biased feedback sources, are only partially annotated, or are from other input domains or languages.

If we are ever to democratize access of NLP for applications that meet the needs of everyone and not just large institutions, we need methods that work in these low-resource situations and alleviate the large manually labeled annotation requirements of standard supervised learning.

We approach this problem by focusing on improving this core cost/accuracy trade-off, with a focus on two directions:

1. Easing the annotation burden by leveraging high-level expert knowledge in addition to labeled examples, thus making approaches more annotation-efficient.
2. Mitigating known biases in cheaper, imperfectly labeled datasets so that we may use them to our advantage.

A central theme of the thesis is that high-level expert knowledge about the data and task can play double-duty — not only can it be used to derive additional supervision signal that improves the annotation-efficiency of models, making them more accurate with fewer labels, but it can also be used to counteract biases in the dataset collection process. Further, their combination may be greater than the sum of the individual parts and used to our advantage: high-level expert knowledge

can allow for biased labeling processes that are even less costly for the experts while still providing key strong supervised learning signal that cannot be not easily obtained by cheaper means. For example, in named entity recognition (Chapter 3) we can allow annotators to quickly skim and inexhaustively annotate documents to produce a wider variety of examples in a short amount of time. This process comes at the expense of completeness, though, and introduces a bias in the data. We can then use expert knowledge about the problem to correct for this bias during training. Ultimately the combination of this quicker, biased process and corrective high-level expert knowledge can allow for more accurate models to be produced at modest annotation budgets compared to exhaustive annotation.

## **1.1 Background**

Supervised learning has been the dominant paradigm in applied NLP, in some way or another underpinning nearly every successful real-world NLP application. Under this paradigm, domain experts break down their information needs into well-defined ontological categories, hierarchies, and structures, called “task specifications,” that define the space of possible outputs they desire the model to predict from text. Given this specification, they must annotate sufficiently large unbiased samples of individual text examples with the desired outputs of the model. A parameterized model is then trained on this labeled corpus to reproduce the outputs given the inputs, with the hope that the learned model will generalize to new inputs from the same domain. In the last decade, significant progress in deep learning model architectures (Mikolov et al., 2013; Bahdanau et al., 2014; Vaswani et al., 2017) has propelled supervised learning to higher and higher accuracies while simultaneously reducing the burden on the applied modeler to design input featurizations of the text (“feature engineering”) or model architectures (“architecture engineering”).

This supervised learning approach does have significant drawbacks, however, that make it difficult to apply in many areas. Supervised learning is in general “annotation hungry” (Liang, 2005; Ratner et al., 2017). Additionally, the more complex the prediction task is, the harder it is to annotate for, and this increases the cost of producing a sufficiently large labeled corpus. Further,

labeling in this paradigm assumes that the task specifications are complete and static which is rarely the case. In practice, task specifications often need iterated revisions (it is difficult to define an ontology with no errors from the outset) or could fundamentally be dynamic in nature, such as situations where new classes of relevant entities emerge over time (Derczynski et al., 2017). In some situations, practitioners attempt to use ad-hoc heuristics and rules for selecting and/or labeling the data in an attempt to alleviate the burden of manually labeling unbiased samples, but this often introduces other problems. When used with standard supervised learning that assumes independent and identically distributed (IID) samples, the biases in the ad-hoc data translate to biased models that underperform in ways that can be hard to measure (Wang et al., 2015; Yin et al., 2022).<sup>1</sup>

On the opposite end of the learning methodology spectrum lie unsupervised approaches, such as clustering (Brown et al., 1992; Toutanova et al., 2004), autoencoders (Vincent et al., 2010), and generative models (Kingma and Welling, 2013). These approaches do not use directly labeled examples, instead attempting to identify inherent patterns in the data. Unfortunately these models are often not directly applicable to most of the “inference” NLP tasks that we are concerned with, as the patterns and representations of the data that they identify are general in nature and often do not directly translate to the specific tasks defined in the task specifications.<sup>2</sup>

However, the representations they extract often do carry some implicit information relevant to the target prediction tasks and have been shown to be useful as features (Liang, 2005) and initializations of supervised models (Devlin et al., 2019). In particular, recent advances in “self-supervised” language model pretraining of large generic models such as Peters et al. (2018a); Devlin et al. (2019); Radford et al. (2019) and Lan et al. (2019) have shown that high-capacity deep learning architectures with a large number of parameters can be “pretrained” on massive raw text corpora with generic objectives and then utilized downstream on more specific applications

---

<sup>1</sup>This bias is often so hard to measure because test data used for measuring performance is typically sampled from the originally collected dataset, implying that it is also biased with respect to real data generated by the in-use application and therefore fundamentally flawed as a measuring instrument.

<sup>2</sup>Once exception to this are recent results in prompting of large autoregressive language models (Radford et al., 2019). While these results are exciting, the current state of these methods also have disadvantages, which we discuss below.

with impressive benefits.

These self-supervised language model pretraining methods can be grouped into two types, with different advantages. The first are cloze-like objectives where the model is tasked with predicting missing words or phrases given the rest of the utterance (Devlin et al., 2019; Lan et al., 2019; Joshi et al., 2020). These models have been shown to be quite useful as pretrained initializations of target application architectures that are subsequently “fine-tuned” on supervised task data with great benefit to the application’s cost/accuracy trade-off (Devlin et al., 2019) — either models with similar accuracy can be produced with much less labeled data, or can be made significantly more accurate with larger datasets. The second type are traditional language-model objectives that aim to predict the next word given the sentence prefix (Radford et al., 2019). These models tend to underperform as initializers for fine-tuning in inference tasks (Liu et al., 2019a) but, unlike cloze-style models, they are able to generate continuations of arbitrary length. When taken to enormous scales of capacity (i.e., billions of parameters), these models have recently demonstrated the remarkable emergent ability to solve prediction tasks whose specifications and examples are provided through the input text prefix itself as “prompts”, instead of as learning signal that modifies the model parameters through optimization (Radford et al., 2019).

In this thesis, we do not eschew these recent advances and instead aim to make contributions that complement their benefits. In particular we focus on approaches that are compatible with the former “fine-tuning” methodology as opposed to the latter “prompt” methodology for three reasons: (1) fine-tuning methods are more amenable with leveraging large and diverse information sources that can be introduced through learning – prompt methods are not practically tunable; (2) fine-tuning approaches are compatible with other output structures that are not just sequences of tokens, and it is generally more clear how to use them in tandem with structured prediction advances for highly structured tasks such as syntax parsing; and (3) due to their sheer model size, state-of-the-art prompting models require extensive computational resources and thus are less accessible as they typically require usage of externally hosted apis.<sup>3</sup>

---

<sup>3</sup>We do not wish however to downplay the likely benefits of prompting-based methods in future work. In particular, prompt-based fine-tuning methods (e.g., (Shin et al., 2021)) and the use of prompting as a “teacher” for generating

Using unsupervised data as a preliminary feature-induction/transfer step before training a supervised model is only one of many approaches that fall under the umbrella of semi-supervised learning (SSL). Another class of approaches in more traditional conceptions of SSL aim to use supervised data and unsupervised data simultaneously. Bootstrapping methods such as self-training (Yarowsky, 1995; Agichtein and Gravano, 2000), co-training (Blum and Mitchell, 1998; Collins and Singer, 1999; Wang et al., 2011; Clark et al., 2018), tri-training (Zhou and Li, 2005; Saito et al., 2017) all use current iterations of a model or a diverse set of models to label unlabeled examples, growing the training set, and then retrain the models in an iterative cycle that attempts to automatically improve the model with less data. A main disadvantage of bootstrapping approaches is that the model’s predictions create labeling noise and feedback loops that can cause them to drift or fail to converge. Another approach to semi-supervised learning is multi-task learning (Caruana, 2004), where the underlying model parameters are at least partially shared while making predictions for different tasks. This brings the advantage of potentially pooling multiple smaller datasets together to act as a larger dataset (Wang et al., 2018), but has been shown to be highly dependent on the relatedness of the multiple tasks, which is itself hard to quantify or predict (Zhang and Yang, 2021).

One particular flavor of SSL worth noting is “weak” supervision. It differs from traditional SSL in that it attempts to utilize weaker or incomplete sources of signal rather than “strong” signals such as complete labels from humans or other models. Early examples of this methodology include “distant supervision” from databases for relation extraction (Mintz et al., 2009) and rule-based label propagation of Wikipedia hyperlinks for named entity recognition (Nothman et al., 2013). More recently, several approaches to weak supervision, such as Snorkel (Ratner et al., 2016a, 2017; Bach et al., 2019) and others (Pal and Balasubramanian, 2018; Kang et al., 2018; Sun et al., 2018; Karamanolakis et al., 2019b; Awasthi et al., 2020; Karamanolakis et al., 2021), bring back high-level but incomplete expert knowledge in the form of keywords and rules that were traditionally engineered as input features but instead use them as weak models to heuristically label datasets

---

silver-labeled data that can be used to fine-tune a "student" model (He et al., 2022) are exciting directions.

that are then used to supervise other stronger models, such as pretrained language models. These approaches are exciting in that they better combine the best of both worlds, utilizing pretrained language model architectures with strong generalization capabilities in addition to high-level expert knowledge about the target task, which can often be worth many manually labeled examples.

The current state of weak supervision does still have drawbacks though. One primary drawback of current weak supervision approaches is their focus on labeling individual text examples. Many methods are primarily concerned with turning weak signals into concrete estimates of “silver” labels for individual examples that are then used as training data in a standard supervised learning setup. This conception is restrictive in that it does not provide a means of using many abstract types of expert knowledge that bear on the output task distribution by itself, such as general expectations about label proportions in the data irrespective of any particular input. A second concern is that weak labeling rules, though better handled than in ad-hoc setups, may still contain significant biases that ultimately are reflected in the final model. For example, if experts write rules that systematically miss or incorrectly label certain groups data, the final models most likely will as well. A third challenge in weak supervision is using it for structured prediction tasks; most approaches are concerned with classification problems. This is mainly due to the fact that annotating complex structures requires specification of many related variables for every text example, and the use of weak rules typically do not have high enough coverage to successfully specify estimated structures completely, leading to unusable or biased labelings. While there have been some attempts to work with these incomplete labelings in the past (Tsuboi et al., 2008; Sassano and Kurohashi, 2010; Mirroshandel and Nasr, 2011), generally these approaches are concerned with manually generated partial labelings and their combination with weak supervision is under-explored.

In general we advocate for a combination of learning paradigms that brings to bear the complementary advantages of unsupervised learning, weakly supervised learning, and traditional supervised learning. Unsupervised learning can be used to pretrain and initialize models with effective generalization capabilities, then weak supervision and traditional (strong) supervised learning can be used to train the model for the end task, with weak supervision providing broad but shallow in-

struction about the target task while strong supervision provides deep but narrow instruction.

## 1.2 Three Real-World Problems and Solutions

We conduct our research on this general topic through three diverse problems with immediate applications to real-world settings.

**Rare-Event Text Classification for Public Health** In the first part of this thesis, we study an applied problem in biased text classification. We encounter a rare-event text classification system that has been deployed for several years, used by epidemiologists at the NYC Department of Health and Mental Hygiene (DOHMH) to identify foodborne illness in online restaurant reviews. The purpose of the system is to classify all of the Yelp restaurant reviews for NYC and present only those that are likely to discuss food poisoning for review and potential further investigation by epidemiologists.

We are tasked with improving this system’s performance using the labeled feedback provided by the epidemiologists over the years, but these data are heavily biased for two reasons. First, the original system model was biased because it built in an ad-hoc fashion, where keywords related to the positive class, such as “food poisoning” or “vomiting”, were used to search an initial corpus of reviews and then labeled for the target class, breaking IID assumptions of standard supervised training. Second, and perhaps more importantly, the epidemiologists had only reviewed and labeled examples already predicted as positive by the system. This resulted in a highly selection-biased labeled dataset that was unfit for standard supervised training. Further, we cannot ask them to label a large unbiased sample for training since the classes are extremely imbalanced and the epidemiologists have little time for annotating – the feedback they have been giving has only been incidental through their use of the system for their real jobs.

Keeping within these constraints, we develop a method that combines importance weighting and an unlabeled data imputation scheme that exploits the selection-bias of the previous model to train an unbiased classifier using the incidental biased feedback data. We demonstrate that this

method considerably improves the system performance while remaining unbiased. Though our application setting is specific, our contribution can be applied to any deployed rare-event classification system that we want to periodically improve using biased incidental feedback. This work is published in Effland et al. (2018).

**Partially Supervised Named Entity Recognition** In the second part of this thesis, we tackle an applied problem in named entity recognition (NER) concerning learning named entity recognizers in the presence of missing entity annotations. That is, the labeled data have acceptable precision for labeled entities, but very low recall. This presents a serious problem in the context of NER, because of the popular “closed world” annotation scheme that assumes all non-annotated spans of text must be nonentities, which introduces a significant bias and leads to unusable models in many realistic cases. This setting is applicable to situations such as distant supervision with gazetteers or hyperlinks (Nothman et al., 2008), labeling by non-native speakers (Mayhew et al., 2019), and exploratory expert annotators with time constraints that skim documents and inexhaustively annotate to gather more diverse contexts for a given time budget.

We approach this setting as sequence tagging with latent variables and propose a novel loss, the Expected Entity Ratio (EER), to learn models in the presence of systematically missing tags. In addition to marginal likelihood training (Tsuboi et al., 2008), our loss adds a regularization term that uses an estimate of the proportion of entities in the data to counteract the data bias, encouraging the model to have the correct ratio of entities in expectation.

We justify the principles of our approach by providing theory that shows it recovers the true tagging distribution under mild conditions. Additionally we provide extensive empirical results that show it to be practically useful. Experimentally, we find that it meets or exceeds performance of strong and state-of-the-art baselines across a variety of languages, annotation scenarios, and amounts of labeled data. We also show that, when combined with our approach, a novel sparse annotation scheme outperforms exhaustive annotation for modest annotation budgets. This work is published in Effland and Collins (2021).

**Syntactic Parsing in Low-Resource Languages** In the third part of this thesis, we study the challenging problem of syntactic analysis (part of speech (POS) tagging and dependency parsing, henceforth referred to as just “parsing”) in low-resource languages. This problem is of significant importance, as expanding the coverage of linguistic analysis to all of the world’s seven thousand languages is a primary goal toward producing equal representation and access of language technologies around the world, in addition to its inherent scientific value.

We approach the problem from a cross-lingual perspective, building on a state-of-the-art thread of research that combines multilingual pretraining with multilingual, multitask fine-tuning (Konratyuk, 2019) for the Universal Dependencies data (Nivre, 2020). While the parser has impressive accuracy for many languages, it has still greatly underperforms on “distant” languages that have little to no representation in the pretraining and fine-tuning corpus.

Motivated by field of syntactic typology, we introduce a method to regularize the parser on distant languages according to the expected typological statistics of the target language. We call our method Expected Statistic Regularization (ESR), as it uses expectations of high-level descriptive statistics about model behavior on target distributions to guide the model toward more sensible outputs, even in the presence of little to no labeled data. ESR is a significant generalization of the EER loss in our NER work. The class of descriptive statistics usable by ESR are expressive and powerful. For example, they may describe cross-task interactions, encouraging the model to obey structural patterns that are not explicitly tractable in the model factorization. Additionally, the statistics may be derived from constraints dictated by the task formalism itself (such as ruling out invalid substructures) or by numerical parameters that are specific to the target dataset distribution (such as relative substructure frequencies). In the latter case, we also contribute a method for selecting those regularization parameters using small amounts of labeled data, based on the bootstrap (Efron, 1979).

We present seven broad classes of descriptive statistic families, some of which have been used in previous parsers as features but have been sidelined in recent years by deep learning approaches. We provide extensive experimental evidence showing that these statistics are still useful and com-

plementary to deep learning approaches as regularizers in low-resource transfer settings, most notably when the target languages are distant with respect to the pretraining data. Additionally, we provide learning curve experiments that show our method to be quite effective when paired with small amounts of labeled data in the target language, and ablation studies that justify other key design choices of our approach. This work is accepted for publication pending minor revisions at TACL. (The revisions are incorporated in this thesis.)

### 1.3 Contributions

Our key contributions can be summarized as follows:

1. We propose a novel approach to improving a deployed rare-event classification with biased incidental feedback (Effland et al., 2018). Specifically we contribute:
  - (a) A method for improving and debiasing the deployed system using a combination of importance weighting and data imputation that exploits the selection bias of the previous system iteration without requiring additional labels from domain experts.
  - (b) A detailed evaluation and error analysis of the method for two applied tasks in rare-event text classification. Our evaluation shows that our method improves precision and recall of the resulting model and counteracts the training data bias.
  - (c) Considerable improvements in performance of a real-world deployed rare-event text classification system with immediate impact.
2. We propose a novel approach for learning named entity recognition models using biased, partially labeled data (Effland and Collins, 2021). Specifically we contribute:
  - (a) A principled method that utilizes a weak and uncertain expert prior about the relative occurrence rate of entities in the text to train accurate NER models using low-recall data.

- (b) Theory justifying the statistical consistency of the approach, proving that our approach recovers the true tagging distribution in the limit of infinite data under mild conditions.
  - (c) Extensive benchmark comparisons showing that our method equals or outperforms previous state-of-the-art approaches across 7 corpora, 6 languages, and 2 diverse low-recall annotation scenarios.
  - (d) A novel partial annotation scheme that we call “Exploratory Expert” (EE) annotation, which allows experts to inexhaustively skim and annotate documents, generating more varied example contexts for a fixed time budget.
  - (e) A user study, showing that EE is as fast as exhaustive annotation.
  - (f) Learning curve experiments that show EE annotation can outperform exhaustive annotation for modest annotation budgets.
3. We propose a novel approach for improving cross-lingual syntactic parsing in low-resource scenarios by using expected typological statistics in the target language as weak supervision. Specifically we contribute:
- (a) A novel and general regularization framework, “Expected Statistic Regularization” (ESR), that can be used to regularize models on unlabeled target datasets with a broad class of functions that describe expected model behaviors. These statistics allow for the incorporation of various forms of high-level expert knowledge as supervision.
  - (b) A method for estimating target statistic values using small amounts of labeled data.
  - (c) An application of the method that improves state-of-the-art cross-lingual parsing on low-resource languages. We contribute seven families of descriptive statistics that bear on parser behavior and extensively evaluate their impact on transfer, showing most to be useful.
  - (d) An extensive benchmark evaluation on transfer to 44 languages showing that ESR leads to significant improvements over state-of-the-art approaches on many low-resource languages.

- (e) Learning curve experiments that demonstrate the impact of the approach is largest for target datasets with 500 or fewer annotated sentences.
- (f) Ablation studies justifying key design choices for the proposed loss function.

## **1.4 Organization**

The remainder of this thesis is organized as follows. In Chapter 2, we focus on our first application, debiasing and improving a deployed rare-event text classification system. Chapter 3 focuses on our second application, learning named entity recognizers with partially annotated low-recall data. Chapter 4 focuses on the third problem of improving cross-lingual syntactic parsing on low-resource languages using expert knowledge. Finally, in Chapter 5 we discuss conclusions, limitations, and future work. While each chapter is a contribution towards the larger topic of this thesis, they also each engage with different applications, and it is our intention that they may be read separately. To this end, we use self-contained notation and related works within each chapter.

## **Chapter 2: Improving a Rare-Event Classification System with Minimal Wasted Labels**

### **2.1 Introduction**

Identifying rare events in large text corpora is an important application area in NLP. There are many potential applications for mining mountains of text for needle-in-a-haystack instances, enabling domain experts to declutter noisy or broad information sources by filtering out the irrelevant texts. For example, in crisis informatics it may be possible to detect early signs of disaster impact zones and focus responses through the use of social media (Imran et al., 2013, 2016; Alam et al., 2018). In consumer protection, rare-event classification could be used to identify posts about product malfunctions or other complaints (Çataltaş et al., 2020; Fong et al., 2021). In this chapter, we encounter an the application of rare-event classification to epidemiology and public health by discovering foodborne illness in online restaurant reviews. We are tasked with improving the deployed system with the epidemiologist feedback accumulated through years of use, but this is challenging because the data are significantly biased by the filtering process. In the rest of this section, we first detail the application domain, then we describe the general technical problem we face when we try to improve the deployed application over time. The contributions in this chapter are published in Effland et al. (2018).

#### **2.1.1 The Application Domain**

Foodborne illness remains a major public health concern nationwide. The Centers for Disease Control and Prevention (CDC) estimates that there are 48 million illnesses and >3000 deaths caused by the consumption of contaminated food in the United States each year (Scallan et al., 2011). Of the approximately 1200 foodborne outbreaks reported and investigated nationally, 68%

are restaurant-related (Gould et al., 2013). Most restaurant associated outbreaks are identified via health department complaint systems. However, there are potentially valuable data sources emerging that could be incorporated in outbreak detection. Specifically, the increasing use of social media has provided a public platform for users to disclose serious real-life incidents, such as food poisoning, that may not be reported through established complaint systems.

In this application we use data from consumer reviews obtained from the popular website Yelp. A comparison of food vehicles associated with outbreaks from the CDC Foodborne Outbreak Online Database and data extracted from Yelp reviews indicating foodborne illness and implicating a specific food item found that the distribution of food categories was very similar between the 2 sources, supporting the usefulness of these data in public health responses (Nsoesie et al., 2014). Furthermore, Yelp reviews can be directly linked with individual restaurant locations, allowing for targeted and timely response.

Since 2012, the Computer Science Department at Columbia University has been collaborating with the New York City (NYC) Department of Health and Mental Hygiene (DOHMH) to develop a system that applies data mining and uses text classification to identify restaurant reviews on Yelp indicating foodborne illness, which are later manually reviewed and classified by DOHMH epidemiologists. This system was used in a pilot study from July 1, 2012, to March 31, 2013, and found 468 Yelp reviews that described a foodborne illness occurrence (Harrison et al., 2014). Of these 468 reviews, only 3% of the illness incidents had been reported to the DOHMH by calling NYC's citywide complaint system, 311. Investigations as a result of these reviews led to the discovery of 3 previously unknown foodborne illness outbreaks, approximately 10% of the total number of restaurant-associated outbreaks identified during the pilot project's time period. This highlighted the need to mine Yelp reviews to improve the identification and investigation of foodborne illness outbreaks in NYC. Due to the success of the pilot study, DOHMH integrated Yelp reviews into its foodborne illness complaint surveillance system and continues to mine Yelp reviews and investigate those pertaining to foodborne illness; this process has been instrumental in the identification of 10 outbreaks and 8523 reports of foodborne illness associated with NYC

restaurants between July 2012 and January 2018.

### 2.1.2 The Technical Problem

During the course of the deployed system use, epidemiologists have reviewed thousands of Yelp posts that have been marked as positive instances of “descriptions of foodborne illness” by a prototype classification model, providing additional labeled data for the tasks through incidental feedback. Our task is to use these data to improve the system classifiers so that they can better assist epidemiologists at doing their jobs. We want to improve the recall of the system so that it misses fewer foodborne illness reports and the precision of the system so epidemiologists waste less time reading false positive reports.

The technical challenge in this situation is two-fold:

1. We want to use the incidental feedback of the epidemiologists to improve the system, but this labeled data suffers from a severe selection-bias due to its selection from the “positive” prediction set of the previous classifier. Using this data naively to retrain models will lead to significant overrepresentation of the positive class and unacceptable model precision. Additionally, naively using this biased data as a test set would yield biased and overly optimistic of results.
2. The epidemiologists are busy and should not be labeling data outside of their standard workflow. This means that we do not have the budget to ask them to label data outside of this workflow, as these labels would be “wasted” in the sense that they do not the epidemiologists do their jobs. We must work with the data we have and cannot simply annotate a new unbiased dataset so as to minimize wasted labels.

At the technical level, these challenges more generally apply to all deployed rare-event classification systems with limited annotation budgets. By design, these systems serve up systematically biased data to help domain experts efficiently comb through large corpora, and the domain experts can easily give feedback on this data incidentally through use of the system. It is natural then to

want to use this feedback to improve the system over time, but the previous system’s selection-bias will prevent its direct use. It is also natural to not want to waste the experts’ time annotating data that is highly likely to be irrelevant to their jobs.

We address these challenges by combining two separate techniques:

1. We derive importance-weighted training loss and test metric equations that explicitly control for the selection bias of the previous classifier in generating labeled data.
2. The estimates used in the importance-weighting training and evaluation require that we label data from the “complement” set of reviews (those filtered out by the deployed system). We propose a data imputation scheme that exploits the extreme rarity of the positive class in this set to automatically label data with minimal introduced noise. This significantly increases the size of the training dataset without the need for manual annotation.

Combined, these methods allow us to train a model that incorporates this biased feedback as the only manually labeled data while being significantly debiased compared to naive training with this data.<sup>1</sup> The results of our evaluations in Section 2.4 indicate the proposed approach significantly improves upon naive retraining and can significantly improve efficacy of the deployed system.

In summary, our overall contribution is a novel approach to improving a deployed rare-event classification with biased incidental feedback (Effland et al., 2018). Specifically we contribute:

- A method for improving and debiasing the deployed system using a combination of importance weighting and data imputation that exploits the selection bias of the previous system iteration without requiring additional labels from domain experts.
- A detailed evaluation and error analysis of the method for two applied problems in rare-event text classification. Our evaluation shows that our method improves precision and recall of the resulting model and counteracts the training data bias.

---

<sup>1</sup>We note that the proposed approach is not entirely unbiased, since the data imputation scheme can introduce a small, hopefully negligible bias on the training data.

- Considerable improvements in performance of a real-world deployed rare-event text classification system with immediate impact.

## 2.2 Materials and Methods

We first describe the overall DOHMH system design. We then describe the classification models used in our evaluation. Finally, we describe the data used in the evaluation and discuss bias-adjusted training and evaluation objectives

### 2.2.1 Yelp System Design

The system runs a daily process to pull Yelp reviews of NYC restaurants from a privately available application programming interface (API) and applies text classification techniques to classify reviews according to 2 criteria. The first criterion, referred to as the “Sick” task, corresponds to whether the review mentions the occurrence of a person experiencing foodborne illness from the restaurant. The second criterion, the “Multiple” task, corresponds to whether there was a foodborne illness event experienced by more than one person; although they are quite rare, these cases constitute significant evidence of a foodborne illness outbreak and are of special interest to DOHMH epidemiologists. After automatically classifying all new reviews according to these criteria, all reviews classified as “Sick” (ie, having a “Sick” probability  $>0.5$ ) are then presented to DOHMH epidemiologists in a user interface for manual review. Upon reviewing a document, the epidemiologists record the gold standard label for both criteria. Yelp messages are sent to the authors of reviews that appear to report true incidents of foodborne illness, and an interview is attempted with each author to collect information regarding symptoms, other illnesses among the author’s dining group, and a 3-day food history. All sources of restaurant-associated foodborne illness complaints are aggregated in a daily report; outbreak investigations are initiated if multiple complaints indicating foodborne illness are received within a short period of time for one establishment, or if a complaint indicates a large group of individuals experiencing illness after a single event.

### 2.2.2 Classification Methods

Prior to classification, the reviews, or documents, are converted into a representation that is usable by the classification algorithms, known as the featurization of documents. This is done using a bag-of-words (BOW) approach by converting each document into a vector with the counts for each word in the vocabulary. The classifiers built for the operational system at DOHMH, further referred to as “prototype” classifiers, were J4.8 (Quinlan, 2014) decision tree models, chosen for the interpretability of their decision functions. These models were trained using 500 reviews, labeled by DOHMH epidemiologists for both criteria. The 500 reviews were selected using a mix of an unbiased sample of reviews and reviews from keyword searches for terms that are intuitively indicative of foodborne illness, such as “sick,” “vomit,” “diarrhea,” and “food poisoning.” To identify the most effective classifiers for our classification tasks, we experimentally evaluated several standard document classification techniques in addition to the prototype classifiers. First, we considered improvements to the document featurization over basic BOW by including n-grams (n consecutive words) for  $n \in \{1, 2, 3\}$ , and term frequency-inverse document frequency (TF-IDF) weights for the terms (Rajaraman and Ullman, 2011). For both classification tasks, “Sick” and “Multiple,” we evaluated 3 well-known supervised machinelearning classifiers: logistic regression (Cox, 1958), random forest (Breiman, 2001), and support vector machine (SVM) (Cortes and Vapnik, 1995). Logistic regression is a classical statistical regression model where the response variable is categorical. Random forest is an ensemble of weak decision tree classifiers that vote for the final classification of the input document. SVM is a nonprobabilistic classifier that classifies new documents according to their distance from previously seen training documents. By definition, the positive examples for the “Multiple” task are a subset of the positive “Sick” examples, since at least one person must have foodborne illness for multiple people to have foodborne illness. Using this notion, we additionally designed a pipelined set of classifiers, further referred to as “Sick-Pipelined” classifiers, for the “Multiple” task, which first condition their predictions on the best “Sick” classifier. If the “Sick” classifier predicts “Yes,” then the “Multiple” classifier is run. Intuitively, this allows the “Multiple” classifier to focus more on the number of people in-

volved than on whether there was a singular foodborne illness event at all. We evaluated logistic regression for this model class.

### 2.2.3 Enhanced Dataset

Between July 2012 and October 2017, DOHMH epidemiologists labeled 13,526 reviews selected for manual inspection by the prototype “Sick” classifier. These reviews are balanced for the “Sick” task, with 51% “Yes” and 49% “No” documents, but are imbalanced for the “Multiple” task, with only 13% “Yes” and 87% “No” documents. For training and evaluation, we split the data chronologically at January 1, 2017, to mirror future performance when training on historical data. This results in 11,551 training reviews and 1,975 evaluation reviews. The training and evaluation sets have equal class distributions: 51%/49% for “Sick” and 13%/87% for “Multiple.” While these reviews contain useful information, having been selected by the prototype “Sick” classifier before labeling heavily biases them, and so they are not representative of the full (original) Yelp feed. To understand and correct for the impact of such bias, we derive a bias-adjusted training objective and augmented the training and evaluation datasets with a sample of reviews from the complement of the biased datasets in the full Yelp feed.

### 2.2.4 Selection Bias Correction for Training and Test Metrics

To account for the selection bias of the prototype “Sick” classifier in the labeled data, we augment the training data with reviews from the set of Yelp reviews that were labeled “No” by the prototype “Sick” classifier. Reviews from this set, further referred to as “complement-sampled” reviews, likely have nothing to do with foodborne illness, but instead serve as easy “No” examples that the classifiers should predict correctly. Exactly how these 2 datasets are merged, however, requires principled consideration. For classifiers that learn to reduce classification error in training, we can formally model the joint likelihood of the classifier misclassifying some review and that review being selected by the prototype “Sick” classifier. Then, by marginalizing this joint distribution over the indicator that a review is selected by the prototype “Sick” classifier, we arrive at an

unbiased estimate of the classification error. The end result is that we weigh classification mistakes for the biased and complement-sampled reviews by the inverses of their respective probabilities of being selected at random from the full Yelp dataset.

Next we present these derivations:

## Definitions

Let  $T(x)$  be the biased selection process for labelling data at DOHMH. We treat it as a black-box and model it atomically.

Let  $U$  be the set of all Yelp Reviews that have been processed by the system.

Let  $B \subset U$  s.t.  $B = \{(x, y) | T(x) = 1\}$ .  $B$  is the biased Yelp review set, labeled by DOHMH epidemiologists after selection by the prototype classifier.

Let  $B^c \subseteq U \setminus B = \{(x, y) | T(x) = 0\}$ .  $B^c$  is the complement of the biased set: all reviews which were never seen by DOHMH epidemiologists because they were filtered out by the prototype classifier.

## Error Rate

Let  $\bar{B} \subseteq B$  be the labeled points from  $B$  which are seen in the *training* data.

Likewise, let  $\bar{B}^c \subset B^c$  be the sample of labeled points from  $B^c$  which are seen in the *training* data.

We can model the error rate of some classifier  $f$  as:

$$p(f(x) \neq y) = p(f(x) \neq y | T(x) = 1)p(T(x) = 1) + p(f(x) \neq y | T(x) = 0)p(T(x) = 0)$$

and use plugin estimates:

$$\hat{p}(T(x) = 1) = \frac{|B|}{|U|}$$

$$\hat{p}(T(x) = 0) = 1 - \frac{|B|}{|U|}$$

$$\hat{p}(f(x) \neq y | T(x) = 1) = \frac{1}{|\bar{B}|} \sum_{(x,y) \in \bar{B}} I[f(x) \neq y]$$

$$\hat{p}(f(x) \neq y | T(x) = 0) = \frac{1}{|\bar{B}^c|} \sum_{(x,y) \in \bar{B}^c} I[f(x) \neq y]$$

therefore

$$\hat{p}(f(x) \neq y) = \frac{1}{|\bar{B}|} \frac{|B|}{|U|} \sum_{(x,y) \in \bar{B}} I[f(x) \neq y] + \frac{1}{|\bar{B}^c|} \left(1 - \frac{|B|}{|U|}\right) \sum_{(x,y) \in \bar{B}^c} I[f(x) \neq y]$$

Since in practice we will average the errors over the entire training set, we multiply and divide the quantity by  $|\bar{B}| + |\bar{B}^c|$ , yielding:

$$\text{IW-Error Rate} = \frac{1}{|\bar{B}| + |\bar{B}^c|} \left[ w_{\bar{B}} \sum_{(x,y) \in \bar{B}} I[f(x) \neq y] + w_{\bar{B}^c} \sum_{(x,y) \in \bar{B}^c} I[f(x) \neq y] \right]$$

where

$$w_{\bar{B}} = \frac{|\bar{B}| + |\bar{B}^c|}{|\bar{B}|} \frac{|B|}{|U|} \quad \text{and} \quad w_{\bar{B}^c} = \frac{|\bar{B}| + |\bar{B}^c|}{|\bar{B}^c|} \left(1 - \frac{|B|}{|U|}\right)$$

are the importance weights.

## Test Metrics

We will calculate the weights as was done in the above Error Rate calculation, however this time we must recalculate the weights using the observed test data proportions. So, let  $\underline{B} \subseteq B$  be the labeled points from  $B$  which are seen in the *test* data. Likewise, let  $\underline{B}^c \subset B^c$  be the sample of labeled points from  $B^c$  which are seen in the *test* data.

Then we have

$$w_{\underline{B}} = \frac{|\underline{B}| + |\underline{B}^c|}{|\underline{B}|} \frac{|\underline{B}|}{|U|} \quad \text{and} \quad w_{\underline{B}^c} = \frac{|\underline{B}| + |\underline{B}^c|}{|\underline{B}^c|} \left(1 - \frac{|\underline{B}|}{|U|}\right)$$

and precision and recall can be calculated as follows:

### Precision

$$\text{IW-Precision} = \frac{1}{\sum_{\substack{(x_i, y_i) \in \underline{B} \cup \underline{B}^c \\ : f(x_i)=1}} w_i} \left( \sum_{\substack{(x_i, y_i) \in \underline{B} \cup \underline{B}^c \\ : f(x_i)=1}} w_i I[y_i = 1] \right)$$

### Recall

$$\text{IW-Recall} = \frac{1}{\sum_{\substack{(x_i, y_i) \in \underline{B} \cup \underline{B}^c \\ : y_i=1}} w_i} \left( \sum_{\substack{(x_i, y_i) \in \underline{B} \cup \underline{B}^c \\ : y_i=1}} w_i I[f(x_i) = 1] \right)$$

### F1

Using the above plugin estimates to calculate the importance weighted precision and recall, we can calculate the bias-adjusted F1-score using:

$$\text{IW-F1} = 2 * \frac{\text{IW-Precision} * \text{IW-Recall}}{\text{IW-Precision} + \text{IW-Recall}}$$

### AUPR

Finally, we can obtain a series of ordered bias-adjusted IW-Precision-Recall points  $E = \{(p, r)_i | r_i \geq r_{i'}, i' < i\}$  by varying the classification threshold  $t \in (1, 0]$  and then using trapezoidal integration to approximate the Area Under the Recall vs. Precision curve (AUPR).

$$\text{IW-AUPR} = \sum_{(p_i, r_i) \in E, i < |E|} .5 * (r_{i+1} - r_i) * (p_i + p_{i+1})$$

## Bootstrap

For the final evaluation, we would like confidence intervals about the IW-F1 and IW-AUPR. We find these by using the percentile bootstrap (Efron, 1979). We calculate bootstrap confidence intervals for the IW-Precision, IW-Recall, and IW-AUPR statistics as follows:

First we calculate the statistic  $\bar{x}$  (for each of IW-Precision, IW-Recall, and IW-AUPR). Then we resample the test dataset with replacement  $B$  times and obtain the bootstrap statistic estimates for each set. Call these  $x_1, \dots, x_B$ . Then we can compute confidence intervals around  $\bar{x}$  the usual way by finding the  $\alpha = .025$  boundary quantiles  $\delta_\alpha, \delta_{1-\alpha}$ , such that

$$P(\bar{x}^* - \delta_{1-\alpha} \leq \bar{x} \leq \bar{x}^* - \delta_\alpha) = .95$$

### 2.2.5 Training Regimes

Using the above sample weights, we incorporate both the biased label data and the complement-sampled data to train our classifiers under 3 different regimes. The first, “Biased,” used only the data from the 11,551 reviews selected by the prototype “Sick” classifier. The second, “Gold,” used the “Biased” data plus 1,000 reviews sampled from the complement-sampled Yelp feed and labeled by DOHMH epidemiologists. In this sample of 1,000 reviews, only 4 were labeled “Yes” for the “Sick” task and 1 was labeled “Yes” for the “Multiple” task. In the third regime, “Silver,” we randomly sampled 10,000 reviews from the complement-sampled Yelp feed before January 1, 2017, and assumed all were negative examples of both tasks. Intuitively, this regime can be helpful if it regularizes out statistical quirks of the “Biased” data more than the noise it may introduce through false negatives. Importantly, this regime also does not require any additional wasted manual annotation effort.

## 2.3 Evaluation

The performance of each classifier was evaluated on the 1,975 biased reviews from after January 1, 2017, along with another sample of 1,000 reviews from the complement-sampled Yelp feed after January 1, 2017. These 1,000 reviews were again labeled by DOHMH epidemiologists for both tasks. However, there were no positive examples of either task among the 1,000 reviews. We evaluated the models for both tasks using 4 performance metrics common to class-imbalanced binary classification problems: precision, recall, F1-score, and area under the precision-recall curve (AUPR). Precision (often called “positive predictive value”) is the proportion of true positives out of the total number of positive predictions. Recall (often called “sensitivity”) is the true positive rate. F1-score is the harmonic mean of precision and recall. Precision, recall, and F1-score were calculated at a classification threshold of 0.5, meaning that we classified reviews with “Yes” probabilities 0.5 as “Yes.” The AUPR was measured by first graphing precision versus recall by varying the classification threshold from 0 to 1, then calculating the area under the curve. For all 4 metrics, 0 is the worst possible score and 1 is a perfect score. Since our evaluation data are also biased, we use the bias-adjusted variants described in Section 2.2.4.

After selecting the best hyperparameter settings for each model variation using best average bias-adjusted F1-score across the development folds, we retrained the models on their full training datasets. We compared the resulting model variations to each other and the prototype classifiers on the 4 evaluation metrics. We calculated 95% confidence intervals for F1-score and AUPR using the percentile bootstrap method (Efron and Tibshirani, 1994) with 1,000 sampled test datasets. We then selected the best variation for both tasks based on test bias-adjusted F1-score as our final classifiers. We report the confusion matrices, perform a detailed error analysis, and identify insightful top features for the final classifiers on both tasks.

### 2.3.1 Hyperparameters

To select the best performing hyperparameters for each model variation, we ran 500 trials of random search with important hyperparameters sampled from reasonable distributions. We selected the best settings of each model variant using best average bias-adjusted F1-score over 5-fold cross validation on the training data, stratified by class label and biased/complement-sampled label.

For each model class, task, and training regime (21 variations total), we performed hyperparameter tuning experiments using 500 trials of random search from reasonable sampling distributions using 5-fold cross-validation on the training data, stratified by class label and biased/complement-sampled label.

### Featurization of Documents

Document featurization and text normalization operations were evaluated to determine their impact on system performance. For all trials, we converted tokens to lowercase and filtered stop words (i.e., articles or function words such as “the,” “an,” “at,” etc.). For each trial, we sampled a hyperparameter value at random for the following settings:

- Max document frequency (removing words that occur in more than a threshold percent of documents), sampled uniformly in  $[.75, 1.0]$ .
- N-gram range (using contiguous word phrases as features with phrases up to length  $n$ ), sampled uniform categorically from  $n \in \{1, 2, 3\}$ .
- TF-IDF normalization (how to normalize the TF-IDF vectors), sampled uniform categorically from  $\{L_1, L_2, None\}$ .
- Whether to use IDF reweighting or not, sampled uniform categorically from  $\{Yes, No\}$ .

## Classifiers

- **Logistic Regression**

- Regularization strength, sampled log-uniformly from  $\lambda \in [10^{-3}, \dots, 10^4]$ .
- Regularization norm type, sampled uniform categorically from  $\{L_1, L_2\}$ .

- **Random Forest**

- Number of trees in forest, sampled uniform integer in  $[10, \dots, 200]$ .
- Max number of features per tree as a function of the total number of features,  $D$ , sampled uniform categorically from  $\{\sqrt{D}, \log_2 D\}$ .

- **SVC**

- Regularization strength, sampled log-uniformly from  $L_1, L_2, None$ .
- Kernel function always set to linear.

## 2.4 Results

We found that the best classifiers achieved bias-adjusted F1-scores of 87% and 66% on the “Sick” and “Multiple” classification tasks, respectively.

### 2.4.1 Classification Evaluation

The performance of the classifier variations for the “Sick” and “Multiple” tasks is presented in Tables 2.1 and 2.2, respectively. All models were evaluated on the test data from after January 1, 2017. For the “Sick” task, we found that the logistic regression model trained using the “Silver” regime achieved the highest F1-score, 87%. With the addition of 10 000 silver-labeled complement-sampled reviews, this model gained 77% in bias-adjusted F1-score over its “Biased” counterpart, a significant increase. The low bias-adjusted F1-score of 10% for the “Biased” “Sick”

Model	Regime	Precision	Recall	F1-Score (95% CI)	AUPR (95% CI)
J4.8	Prototype	0.48	0.99	0.65 (0.63-0.67)	0.83 (0.81-0.85)
Logistic Regression	Biased	0.05	0.94	0.10 (0.09-0.11)	0.63 (0.55-0.76)
Logistic Regression	Gold	0.83	0.88	0.85 (0.83-0.87)	0.90 (0.88-0.92)
Logistic Regression	Silver	0.85	0.88	<b>0.87</b> (0.85-0.88)	0.91 (0.90-0.93)
Random Forest	Biased	0.04	0.91	0.07 (0.06-0.09)	0.59 (0.54-0.70)
Random Forest	Gold	0.36	0.89	0.51 (0.38-0.68)	0.81 (0.78-0.84)
Random Forest	Silver	0.70	0.88	0.78 (0.66-0.85)	0.87 (0.85-0.89)
SVM	Biased	0.09	0.95	0.16 (0.13-0.20)	0.82 (0.79-0.87)
SVM	Gold	0.33	0.93	0.49 (0.37-0.67)	0.88 (0.85-0.91)
SVM	Silver	0.96	0.74	0.83 (0.81-0.85)	0.93 (0.92-0.95)

Table 2.1: *Model Performance on “Sick” Task*. The bold value represents the final selected model from among the variants. This is the model we further analyze in the error analysis. Because the bootstrap distribution of some test statistics exhibited non-normal behavior, their corresponding confidence intervals are wider.

logistic regression is due to the misrepresentation of the full Yelp dataset by the “Biased” training, which causes the model to highly over-predict “Yes” on the complement-sampled test data. This behavior is heavily penalized by the bias-adjustment because each false positive in the small complement-sampled test data is representative of many more false positives in the full Yelp dataset. For the “Multiple” task, we found that the “Sick-Pipelined” logistic regression model trained using the “Silver” regime achieved the highest F1-score, 66%. The use of pipelined training and prediction caused a gain of 5% for the “Silver” “Sick-Pipelined” logistic regression over its single-step counterpart.

#### 2.4.2 Precision-Recall Tradeoff

Given the rarity of reviews discussing foodborne illness, it is desirable to explore settings of the “Sick” classifiers that favor recall over precision, since DOHMH epidemiologists are willing to accept some extra false positives to reduce the risk of missing an important positive “Sick” review. We analyzed this trade-off by examining the precision-recall curves of the “Sick” logistic regression classifiers, presented in Figure 2.1. From the plot, we can see that “Gold” and “Silver” models begin to experience an approximately equal trade-off of precision for recall in the region of

Model	Regime	Precision	Recall	F1-Score (95% CI)	AUPR (95% CI)
J4.8	Prototype	< 0.01	0.69	0.01 (0.01, 0.01)	0.01 (0.01, 0.01)
Logistic Regression	Biased	0.08	0.56	0.15 (0.09-0.26)	0.25 (0.19-0.40)
Logistic Regression	Gold	0.42	0.58	0.48 (0.30-0.67)	0.56 (0.49-0.67)
Logistic Regression	Silver	0.64	0.58	0.61 (0.56-0.66)	0.58 (0.52-0.65)
Sick-Pipelined LR	Biased	0.07	0.61	0.13 (0.09-0.23)	0.18 (0.13, 0.43)
Sick-Pipelined LR	Gold	0.77	0.56	0.65 (0.60-0.70)	0.65 (0.59-0.70)
Sick-Pipelined LR	Silver	0.75	0.59	<b>0.66</b> (0.61-0.70)	0.71 (0.65-0.76)
Random Forest	Biased	0.04	0.37	0.07 (0.05-0.12)	0.03 (0.02-0.18)
Random Forest	Gold	0.75	0.24	0.36 (0.29-0.42)	0.31 (0.23-0.45)
Random Forest	Silver	0.74	0.25	0.37 (0.31-0.43)	0.40 (0.31-0.43)
SVM	Biased	0.07	0.65	0.12 (0.08-0.20)	0.18 (0.12-0.48)
SVM	Gold	0.35	0.34	0.35 (0.21-0.54)	0.29 (0.21-0.57)
SVM	Silver	0.20	0.30	0.24 (0.13-0.47)	0.39 (0.30-0.64)

Table 2.2: *Model Performance on “Multiple” Task.* The bold value represents the final selected model from among the variants. This is the model we further analyze in the error analysis.

Actual Class	Predicted Class	
	No	Yes
No	1882 (true negatives) 93%	144 (false positives) 7%
Yes	112 (false negatives) 12%	837 (true positives) 88%

Table 2.3: *“Sick” Task Confusion Matrix.*

Actual Class	Predicted Class	
	No	Yes
No	2643 (true negatives) 98%	55 (false positives) 2%
Yes	114 (false negatives) 42%	163 (true positives) 58%

Table 2.4: *“Multiple” Task Confusion Matrix.*

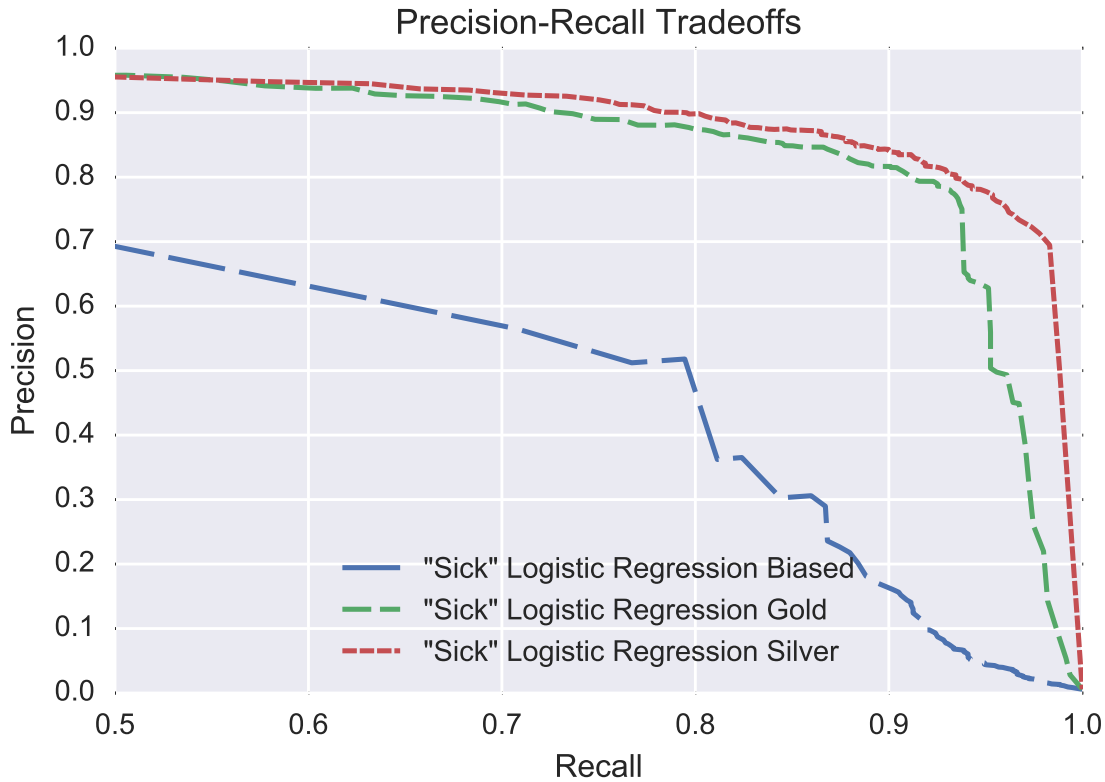


Figure 2.1: *Precision-Recall Tradeoffs*. Precision-recall curves of “Sick” logistic regression models in the high-recall region. While the “Biased” logistic regression performance lags below, the “Gold” and “Silver” models show relatively mild losses in precision per point of recall gained until the 90-100% recall region. After 92% recall the “Gold” model begins to experience a steep drop in precision while the “Silver” model does not experience a steep drop in precision until a recall of 98%.

80%–90% recall, illustrated by the slope of the curves being close to 1 point of precision lost per point of recall gained. In the 90%–100% recall region, the “Gold” model begins to experience a steep drop in precision at a recall of 92% while the “Silver” model does not experience a steep drop in precision until a recall of 98%. At this point, the precision of the “Silver” logistic regression is still 69%, 21% higher than the prototype classifier which has 48% precision at 99% recall. This indicates that even in a high-recall setting the “Silver” “Sick” classifier should provide better performance over the “Sick” prototype.

### 2.4.3 Error analysis of best “Sick” classifier

Of the 2,975 reviews in the test dataset, there are 949 positive examples and 2,026 negative examples for the “Sick” task. The best “Sick” classifier, “Silver” trained logistic regression, achieved an F1-score of 87%, a statistically significant 22% absolute increase over the prototype classifier, with an F1-score score of 65%. On this test dataset, the best “Sick” classifier correctly classified many reviews containing major sources of false positives for the prototype classifier. These gains are not surprising, given that this model uses 40 times more data and better document representations (TF-IDF and trigrams rather than vanilla BOW). This large performance increase will qualitatively change the efficacy of the system for DOHMH epidemiologists. Examination of the 144 false positives identified various causes. Many of these false positives cannot be identified by a classifier only using n-grams with  $n \in \{1, 2, 3\}$ . For example, one reviewer wrote, “I didn’t get food poisoning,” which would require 4-grams for the classifier to capture the negation. This example illustrates a major shortcoming of n-gram models: important dependencies or relationships between words often span large distances across a sentence. Another major source of false positives are reviews that do talk about food poisoning but are not current enough to meet the DOHMH criteria for follow-up, and thus are labeled “No.” A third type of false positive occurs when a review talks about food poisoning in a hypothetical or future sense. For example, one reviewer reported that the food “had a weird chunky consistency...hopefully we won’t get sick tonight.” Multiple causes of the 112 false negatives were also identified. One notable cause is misspellings of key words related to food poisoning in the review, such as “diherrea.” Another major cause is grave references to food poisoning but the classifier predicts “No” because of a prevalence of negatively weighted n-grams, such as “almost threw up.” A final source of false negatives is human error in the labeling of reviews for the test data. For example, one review’s only reference to illness was “she began to feel sick” while at the restaurant, yet the review was labeled positive. Many of the reviews contained negation, which the best “Sick” classifier can detect due to the use of n-grams. N-grams also allow the classifier to identify that the pattern “sick of,” as in “sick of the pizza,” does not typically refer to actual food poisoning, compared to “got sick,” which typically

does. Finally, we examined the highest-weighted n-grams of the best “Sick” classifier. The most highly positive-weighted features were phrases indicative of foodborne illness, such as “diarrhea,” “food poisoning,” and “got sick,” while the most highly negative features were either very positive phrases or indicative of false positives, such as “amazing” and “sick of.” These top features are encouraging, as they show the model has identified features that epidemiologists would also deem important.

#### 2.4.4 Error analysis of best “Multiple” classifier

Of the 2,975 reviews in the test dataset, there are 277 positive examples and 2,698 negative examples for the “Multiple” task. The best “Multiple” classifier, “Silver” trained “Sick-Pipelined” logistic regression, achieved an F1-score of 66%. We examined the reason behind the 114 false negative reviews. Many false negatives were due to incorrect predictions made by the pipelined “Sick” classifier. Most other false negatives were caused by the inability of trigram models to capture longer phrases. Phrases indicating multiple illnesses, such as “we both got really sick,” typically span more than 3 contiguous words, leaving no way for a classifier using trigrams to detect them directly. Of the 277 true positives, 163 were correctly classified. Reviews containing phrases clearly indicating multiple illnesses in a bigram or trigram, such as “both got sick,” scored highest; however, such concise n-grams are rare. The classifier’s highly weighted features are n-grams that simply refer to multiple people without referring to food poisoning. The classifier can capture references to multiple people in a trigram, but these references are often devoid of context, making it hard to determine if multiple people simply did something together or multiple people became ill. Analysis of the true positive test reviews with respect to these feature weights suggests that the classifier tends to select reviews that contain an abundance of ngrams about multiple people. Examination of these features shows that the n-gram model class is not sufficient for the “Multiple” task, indicated by its low performance relative to the “Sick” task and the need for detection of long phrases, which it cannot do. While it is tempting to simply extend the n-gram range to longer sequences, this approach fails due to a well-known statistical issue called “sparsity”:

specific longer phrases become extremely rare in the data and are not seen in enough quantity for models to learn from them.

## **2.5 Related Work**

We describe related work in three areas: usage of social media as data for public health informatics, and machine learning approaches in the context of rare-event classification and biased sampling settings.

### **2.5.1 Public Health Informatics using Social Media**

As a result of the increasing interest and potential value of social media data, research institutions are partnering with public health agencies to develop methods and applications to use data from social media to monitor outbreaks of infectious diseases. Textual data from Internet search engines and social media have been used to monitor outbreaks of various infectious diseases, such as influenza (Santillana et al., 2015). An evaluation comparing the use of informal and unconventional outbreak detection methods against traditional methods found that the informal source was the first to report in 70% of outbreaks, supporting the usefulness of such systems (Bahk et al., 2015). The incorporation of social media data into public health surveillance systems is becoming more common. Multiple projects focus on identifying incidents of foodborne illness using data from Twitter. Harvard Medical School developed and maintains a machine learning platform, HealthMap Foodborne Dashboard, to identify complaints and occurrences of foodborne illness and send a survey link where Twitter users can provide more information; this platform is freely available for research (Freifeld et al., 2008). The Chicago Department of Public Health partnered with the Smart Chicago Collaborative to develop Foodborne Chicago, which also uses machine learning to identify tweets indicating foodborne illness and also sends a survey link where Twitter users can provide more information (Harris et al., 2014). The Southern Nevada Health District developed nEmesis, an application that associates a user’s previous locations with subsequent tweets indicating foodborne illness (Sadilek et al., 2016).

### 2.5.2 Rare-Event Classification

Rare-event classification in our context is an instance of binary classification under extreme class-imbalance, where the minority class (the rare event) is the positive category. The problem of binary classification with high negative-class skew has received extensive attention in the literature in the context of a wide variety of application domains (Kubat et al., 1998; Ezawa et al., 1996; Fawcett and Provost, 1996; Domingos, 1999, *inter alia*).<sup>2</sup>

There are two main types of methodological challenges to machine learning under class imbalance: training classifiers with high performance, and properly measuring that performance.

The challenge with training models under extreme class imbalance is that for complex (non-linearly separable) tasks, model optimization tends to overly favor detection of the majority class (Anand et al., 1993), which in turn leads to unacceptable performance since the minority class is the one of interest (Japkowicz and Stephen, 2002). In the literature, several types of approaches have been proposed to counter this issue by “rebalancing” the learning setting (Japkowicz and Stephen, 2002; Johnson and Khoshgoftaar, 2019):

1. Data-level approaches that attempt to rebalance data, such as under-sampling of the majority class (Kubat et al., 1998) or over-sampling of the minority class (Japkowicz and Stephen, 2002). These approaches can run into issues in that under-sampling can reduce the overall information about the target variable that the dataset carries, while over-sampling of small minority-class samples can cause overfitting to specific idiosyncrasies of sample that do not generalize to the class more broadly (Japkowicz and Stephen, 2002), however, the Synthetic Minority Over-sampling Technique (SMOTE) method (Chawla et al., 2002) and its variants (e.g., Han et al., 2005; He et al., 2008) interpolate examples in the feature space and can help reduce this issue. Over- and under-sampling can of course be combined to additional benefit (Ling and Li, 1998).

2. Algorithm-level approaches that modify the learning algorithm with unequal costs for major-

---

<sup>2</sup>Work in this area began in the mid 90’s. The rest of this review focuses primarily on seminal work from this period (with notable exceptions), since the litany of subsequent work is generally more narrowly focused and applied.

ity and minority classes during parameter optimization (Ming Ting and Zheng, 1998; Provost and Fawcett, 1998; Domingos, 1999, *inter alia*). A main concern with these approaches is that they can still suffer from overfitting if there are too few minority class examples. Another issue is that the choice of misclassification costs are not always known apriori and must then be treated as hyperparameters.

3. Hybrid approaches which combine the data- and algorithm-level rebalancing (Akbari et al., 2004; Tang et al., 2009; Ahumada et al., 2008, *inter alia*). Our approach can be regarded as an instance of this hybrid class of techniques.

The main challenge with measuring performance in class-imbalanced settings, is that accuracy can obscure performance of the classifier on the minority class of interest, purely by nature of the majority class error rate dominating the total error rate. Provost and Fawcett (1998) are the first to identify this failing and suggest the use of Area under the Receiver Operating Characteristic Curve (AUC) to mitigate this effect by varying all possible classification thresholds. However, as Davis and Goadrich (2006) observe, this too can yield overly optimistic results in the face of extreme class-imbalance, and suggest the Area under the Precision-Recall Curve (AUPR) to better focus the metric on the minority class of interest.<sup>3</sup>

### 2.5.3 Accounting for Sampling Bias

Although the general problem of detecting foodborne illness in online restaurant review is an instance of classification of with rare positive classes, this is not the only challenge at play in our situation. In addition to this class skew in the true data distribution, we also must contend with significant selection bias in our training and test samples due to the feedback mechanism of the in-use system.

This problem has also been studied significantly in the classical statistics literature. It is an

---

<sup>3</sup>This overly optimistic aspect of AUC is due to the fact that it compares true positive to false positive rates. When the number of negative instances significantly outweighs the number of positives, even significant differences in the number of predicted positives can have little effect on the false positive rate. Precision does not have this issue, since it does not include negatives in the denominator.

example of parameter estimation with systematically missing observations, and our solution is to use stratified samples (two strata based on the prediction of the deployed classifier) with estimated prior strata occurrence rates. To adjust for this selection bias, the bias-adjusted error rate that we’ve derived ultimately boils down to the use of inverse propensity weighting (IPW), a form of importance weights (IW), which has been used in statistical inference approaches, such as the seminal Horovitz-Thompson estimator (Horvitz and Thompson, 1952) and estimation of the condition mean with missing observations due to Robins et al. (1994). IPW has also been used in countless experimental designs that employ unequal sample sizes in stratified sampling for increased sampling efficiency and reduced cost (Kish, 1965).

## **2.6 Discussion**

In this study, we have presented an automated text-classification system for the surveillance and detection of foodborne illness in online NYC restaurant reviews from Yelp. Using this system, NYC DOHMH epidemiologists are able to monitor millions of reviews, a previously impossible task, to aid in the identification and investigation of foodborne illness outbreaks in NYC. As of May 21, 2017, this system has been instrumental in the identification of 10 outbreaks and 8523 reports of foodborne illness associated with NYC restaurants since July 2012. Aided by simple prototype classifiers, DOHMH epidemiologists have evaluated and labeled 13 526 Yelp reviews for 2 key indicators of foodborne illness since July 2012. Although these data are biased by the prototype classifier’s selection criterion, we showed how these biased data and additional complement-sampled data could be combined in a bias-adjusted training regime to build significantly higher-performing classifiers, an issue that commonly plagues deployed needle-in-a-haystack systems. We evaluated the performance of our prototype classifiers and several other well-known classification models on 2 tasks, namely “Sick” and “Multiple.” We found that logistic regression trained with the “Silver” regime performed best for the “Sick” task and that the “Silver” “Sick-Pipelined” logistic regression performed best on the “Multiple” task, with bias-adjusted F1-scores of 87% and 66%, respectively. Although the raw Yelp data are not publicly

available, all code used to reproduce the final experiments in this manuscript can be found at [https://github.com/teffland/FoodborneNYC/tree/master/jamia\\_2017/](https://github.com/teffland/FoodborneNYC/tree/master/jamia_2017/).

## **2.7 Conclusion**

The importance of effective information extraction regarding foodborne illness from social media sites is increasing with the rising popularity of online restaurant review sites and the decreasing likelihood that younger people will report food poisoning via official government channels. In this chapter, we described details of the DOHMH system for foodborne illness surveillance in on-line restaurant reviews from Yelp. Our system has been instrumental in the identification of 10 outbreaks and 8,523 reports of foodborne illness associated with NYC restaurants between July 2012 and January 2018. Our evaluation has identified strong classifiers for both tasks, whose deployment will allow DOHMH epidemiologists to more effectively monitor Yelp for improved foodborne illness investigations.

In solving this applied problem, we have contributed a more general approach for improving deployed rare-event text classification systems without additional wasted labels. The approach, which combines principled importance weighting in addition to a data imputation scheme, has high annotation-efficiency because it does not require additional labeling procedures outside of the incidental feedback gathered from day-to-day system use. It also successfully harnesses the known systematic bias of the system to our advantage by admitting a data-imputation scheme that generates significantly more data with little noise.

## **Chapter 3: Partially Supervised Named Entity Recognition via the Expected Entity Ratio**

In Chapter 2, we developed an approach for improving a deployed rare-event text classification system using only the incidental annotations provided through its use by experts. In this chapter, we again encounter a setting where the training data suffers from a systematic and severe selection bias due to its annotation scheme, and we again do not wish to ask annotators to go back and manually correct the bias. However, we move from the simpler text classification problem to the more complex multi-class structured prediction task of named entity recognition (NER) with a tagging model. This transition to predicting structures complicates the problem, as we may no longer apply a simple weighting scheme to the loss on individual examples as before, and we no longer can make hard assumptions about labels for the missing data. Instead we must use a soft and uncertain prior assumption about the overall label proportions in the data to drive the model toward correct label proportions in expectation.

The rest of this chapter is organized as follows. In Section 3.1 we discuss the problem background and motivation. In Section 3.2 we describe the proposed approach and provide theoretical justification for its principle. In Sections 3.3 and 3.4 we extensively evaluate the proposed approach against state-of-the-art baselines across a variety of languages, annotation scenarios, and annotation-availability settings. Finally, we discuss related work and conclusions in Sections 3.5 and 3.6, respectively. The contributions described in this chapter are published in Effland and Collins (2021).

### 3.1 Background and Motivation

Named entity recognition (NER) is a critical subtask of many domain-specific natural language understanding tasks in NLP, such as information extraction, entity linking, semantic parsing, and question answering. For large, exhaustively annotated benchmark datasets, this problem has been largely solved by fine-tuning of high-capacity pretrained sentence encoders from massive-scale language modeling tasks (Peters et al., 2018b; Devlin et al., 2019; Liu et al., 2019c). However, fully annotated datasets themselves are expensive to obtain at scale, creating a barrier to rapid development of models in low-resource situations.

Partial annotations, instead, may be much cheaper to obtain. For example, when building a dataset for a new entity extraction task, a domain expert may be able to annotate entity spans with high precision at a lower recall by scanning through documents inexhaustively, creating a higher diversity of contexts and surface forms by limiting the amount of time spent on repetitive individual documents. One way inexhaustive settings like this happen is when annotations are being created "when it is convenient," while the domain expert is performing another information-seeking task. In another scenario studied by Mayhew et al. (2019), non-speaker annotators for low-resource languages may only be able to recognize some of the more common entities in the target language, but will miss many less common ones. In both of these situations, we wish to leverage partially annotated training data with high precision but low recall for entity spans. Because of the low recall, unannotated tokens are ambiguous and it is not reasonable to assume they are non-entities (the  $\circ$  tag). We give an example of this in Figure 3.1.

We address the problem of training NER taggers with partially labeled, low-recall data by treating unannotated tags as latent variables for a discriminative tagging model. We propose to combine marginal tag likelihood training (Tsuboi et al., 2008) with a novel discriminative criterion, the Expected Entity Ratio (EER), to control the relative proportion of entity tags in the sentence. The proposed loss is (1) flexibly able to incorporate prior knowledge about expected entity rates under uncertainty; (2) theoretically recovers the true tagging distribution under mild conditions;

	<b>Tim</b>	<b>Cook</b>	<b>is</b>	<b>the</b>	<b>CEO</b>	<b>of</b>	<b>Apple</b>
<b>Gold</b>	B-PER	L-PER	O	O	O	O	U-ORG
<b>Raw</b>	O	O	O	O	O	O	U-ORG
	<i>False Negative</i>						
<b>Latent</b>	–	–	–	–	–	–	U-ORG

Figure 3.1: An example low-recall sentence with two entities (one is missing) and its NER tags. The Gold row shows the true tags, the Raw row shows a false negative induced by the standard “tokens without entity annotations are non-entities” assumption, and the Latent row reflects our view of unannotated tags as latent variables.

and (3) easy to implement, fast to compute, and amenable to standard gradient-based optimization. We evaluate our method across 7 corpora in 6 languages along two diverse low-recall annotation scenarios, one of which we introduce. We show that our method performs as well or better than the previous state-of-the-art methods from Mayhew et al. (2019) and the recent work of Li et al. (2021) across the studied languages, scenarios, and amounts of labeled entities. Further, we show that our novel partial annotation scheme, when combined with our method, outperforms exhaustive annotation for modest annotation budgets.

In summary, we make the following contributions:

- A principled method that utilizes a weak expert prior about the relative occurrence rate of entities in the text to train accurate NER models using low-recall data.
- Theory justifying the statistical consistency of the approach, proving that our approach recovers the true tagging distribution in the limit of infinite data under mild conditions.
- Extensive benchmark comparisons showing that our method equals or outperforms previous state-of-the-art approaches across 7 corpora, 6 languages, and 2 diverse low-recall annotation scenarios.
- A novel partial annotation scheme that we call “Exploratory Expert” (EE) annotation, which

allows experts to inexhaustively skim and annotate documents, generating more varied example contexts for a fixed time budget.

- A user study, showing that EE is as fast as exhaustive annotation.
- Learning curve experiments that show EE annotation can outperform exhaustive annotation for modest annotation budgets.

## 3.2 Methods

In this section, we describe the proposed approach. We begin with a description of the problem and notation in Section 3.2.1, followed by the NER tagging model in Section 3.2.2. We then describe the supervised marginal tag loss and our proposed auxiliary loss, used for learning on positive-only annotations, in Section 3.2.3 and Section 3.2.4, respectively. Finally, in Section 3.2.5 we describe the full objective and give theory showing that our approach recovers the true tagging distribution in the large-sample limit.

### 3.2.1 Problem Setup and Notation

We formulate NER as a tagging problem, as is extremely common (McCallum and Li, 2003; Lample et al., 2016; Devlin et al., 2019; Mayhew et al., 2019, *inter alia*). In fully supervised tagging for NER, we are given an input sentence  $x_{1:n} = x_1 \dots x_n, x_i \in \mathcal{X}$  of length  $n$  tokens paired with a sequence  $y_{1:n}, y_i \in \mathcal{Y}$  of tags that encode the typed entity spans in the sentence. Following previous work, we use the BILUO scheme (Ratinov and Roth, 2009). Under this formulation, a NER dataset of fully annotated sentences is a set of pairs of token and tag sequences:

$$\mathcal{D}_s^m = \{(x_{1:n_k}^k, y_{1:n_k}^k)\}_{k=1}^m$$

## Partial Annotations

Normally, fully annotated tag sequences are derived from exhaustive annotation schemes, where annotators mark all positive entity spans in the text and then the filler  $\circ$  tag can be perfectly inferred at all unannotated tokens. Training a model on such fully annotated data is easy enough with traditional maximum likelihood estimation (McCallum and Li, 2003; Lample et al., 2016).

In many cases, however, it is desirable to be able to learn on incomplete, partially annotated training data that has high precision for entity spans, but low recall (Section 3.3.2 discusses two such scenarios). Because of the low recall, unannotated tokens are ambiguous and it is not reasonable to assume they are non-entities (the  $\circ$  tag). Even in this low-recall situation, prior works (Jie et al., 2019; Mayhew et al., 2019) assume that unannotated tokens are given this non-entity tag. Their approaches then try to estimate which of these tags are “incorrect” through self-training-like schemes, iteratively down-weighting the contribution of these noisy tags to the loss with a meta training loop.

In contrast to prior work, we make no direct assumptions about unannotated tokens and treat all such positions as latent tags. In this view, a partially annotated sentence is a token sequence  $x_{1:n}$  paired with a set of observed (tag, position) pairs. Given a sentence  $x_{1:n}$ , we define

$$y_O \subset \{(y, i) \mid y \in \mathcal{Y}, 1 \leq i \leq n\}$$

as the set of observed tags  $y$  at positions  $i$ . For example, in Figure 3.1 we would have  $y_O = \{(\text{U-ORG}, 7)\}$ . Under this formulation, we will be given a partially observed dataset:

$$\mathcal{D}^m = \{(x_{1:n_k}^k, y_{O_k}^k)\}_{k=1}^m$$

We use data of this form for the rest of the work.

### 3.2.2 Tagging Model

We use a simple, relatively off-the-shelf tagging model for  $p(y_{1:n}|x_{1:n}; \theta)$ . Our model, BERT-CRF, first encodes the token sequence using a contextual Transformer-based (Vaswani et al., 2017) encoder, initialized from a pretrained language-model objective (Devlin et al., 2019; Liu et al., 2019c). Given the output representations from the last layer of the encoder, we then score each tag individually with a linear layer, as in Devlin et al. (2019). Finally, we model the distribution  $p(y_{1:n}|x_{1:n})$  with a linear-chain CRF (Lafferty et al., 2001), using the individual tag scores and learned transition parameters  $T$  as potentials. Mathematically, our tagging model is given by:

$$\begin{aligned}
h_{1:n} &= \text{BERT}(x_{1:n}; \theta_{\text{BERT}}) \\
\phi(i, y) &= v_y^\top h_i \\
\phi(i, y, y') &= \phi(i, y) + T_{y, y'} \\
p(y|x) &= \frac{\exp\{\sum_{i=1}^{n-1} \phi(i, y_i, y_{i+1}) + \phi(n, y_n)\}}{Z(\phi)} \\
Z(\phi) &= \sum_{\substack{y'_{1:n} \\ \in \mathcal{Y}^n}} \exp\{\sum_{i=1}^{n-1} \phi(i, y'_i, y'_{i+1}) + \phi(n, y'_n)\}
\end{aligned}$$

where  $\phi \in \mathbb{R}^{n \times |\mathcal{Y}| \times |\mathcal{Y}|}$  is the tensor of individual potentials and  $\theta = \{\theta_{\text{BERT}}, T\} \cup \{v_y\}_{y \in \mathcal{Y}}$  are the full set of model parameters.

A few important things to note:

1. While we call the encoder ‘‘BERT’’, in practice we utilize various BERT-like pretrained transformer language models from the HuggingFace Transformers (Inc., 2019) library.
2. We apply grammaticality constraints to the transition parameters  $T$  that cause the model to put zero mass on invalid transitions.
3. We do not use special start and end states, as pretrained transformers already bookend the

sentence with `SOS` and `EOS` tokens that can be assumed to always be `O` tags. This combined with the transition constraints guarantees that the tagger outputs valid sequences.

We choose this model architecture because it closely reflects recent standard practice in applied NER (Devlin et al., 2019; Inc., 2019), where a pretrained transformer is fine-tuned to the tagging dataset. However, we improve this practice by using a CRF layer on top instead of predicting all tags independently. We stress that the additional CRF layer has multiple benefits – the transition parameters and global normalization improve model capacity and, importantly, prevent invalid predictions. In preliminary experiments, we found that invalid predictions were common in some of the few-annotation scenarios we study here.

### 3.2.3 Supervised Marginal Tag Loss

We train our tagger on partially annotated data by maximizing the marginal likelihood (Tsuboi et al., 2008) of the observed tags under the model:

$$L_p(\theta; \mathcal{D}^m) = \frac{1}{m} \sum_{(x_{1:n_k}^k, y_{O_k}^k) \in \mathcal{D}^m} -\log p(y_{O_k}^k | x_{1:n_k}^k; \theta) \quad (3.1)$$

with

$$\log p(y_O | x_{1:n}) = \log \sum_{y_{1:n} \models y_O} p(y_{1:n} | x_{1:n}) \quad (3.2)$$

where  $y_{1:n} \models y_O$  means all taggings satisfying the observations  $y_O$ .

While it is possible to optimize only this loss for the given partially annotated data, doing so alone has deleterious effects in our scenario – the resulting model will not learn to meaningfully predict the `O` tag, by far the most common tag (Jie et al., 2019) and thus fail to have acceptable performance, with high recall at nearly zero precision. We need another term in the loss to encourage the model to predict `O` tags, which we introduce next.

**A Note on Implementation:** For linear CRFs (and tree-shaped CRFs more generally), this loss is tractable and has an interesting decomposition that makes it simple to implement without ad-

ditional custom dynamic programming. If we substitute the full expression for  $p(y_{1:n}|x_{1:n})$  into Equation 3.2, the loss can be re-expressed as the difference between log-partition functions for two CRFs, the first with potentials constrained by the observations and the second with the original unconstrained potentials. The derivation is as follows:

$$\begin{aligned}
\log p(y_O|x_{1:n}) &= \log \sum_{y_{1:n} \models y_O} p(y_{1:n}|x_{1:n}) \\
&= \log \sum_{y_{1:n} \models y_O} \exp\left\{\sum_{i=1}^{n-1} \phi(i, y_i, y_{i+1}) + \phi(n, y_n)\right\} \\
&\quad - \log \sum_{y_{1:n} \in \mathcal{Y}^n} \exp\left\{\sum_{i=1}^{n-1} \phi(i, y_i, y_{i+1}) + \phi(n, y_n)\right\} \\
&= \log \sum_{y_{1:n} \in \mathcal{Y}^n} \mathbb{1}\{y_{1:n} \models y_O\} \exp\left\{\sum_{i=1}^{n-1} \phi(i, y_i, y_{i+1}) + \phi(n, y_n)\right\} - \log Z(\phi) \\
&= \log \sum_{y_{1:n} \in \mathcal{Y}^n} \exp\left\{\sum_{i=1}^{n-1} \log \mathbb{1}\{y_i \models y_O\} + \phi(i, y_i, y_{i+1}) + \log \mathbb{1}\{y_n \models y_O\} + \phi(n, y_n)\right\} - \log Z(\phi) \\
&\Rightarrow \log p(y_O|x_{1:n}) = \log Z(\phi^{y_O}) - \log Z(\phi)
\end{aligned}$$

where  $\phi^{y_O}(i, y, y') = \begin{cases} -\infty & \text{if } i \in O \wedge (y, i) \notin y_O \\ \phi(i, y, y') & \text{else} \end{cases}$

This property allows for convenient implementation; we only need to call an implementation for computing partition function (which is well known) with two different sets of potentials.

### 3.2.4 Expected Entity Ratio Loss

As has been observed in prior work (Augenstein et al., 2017; Peng et al., 2019; Mayhew et al., 2019), the number of named entity tags (versus  $\circ$  tags) over the entire distribution of sentences occur at relatively stable rates for different named entity datasets with the same task specification. For any specific dataset, we call this proportion the “expected entity ratio” (EER), which is simply

the marginal distribution of some tag  $y$  being part of an entity span,  $p(y \neq \circ)$ . Given an estimate of this EER,  $\rho = p(y \neq \circ)$ , for the dataset in question, we propose to impose a second loss that directly encourages the tag marginals under the model to match the given EER, up to a margin of uncertainty  $\gamma$ . This loss is given by:

$$L_u(\theta; \mathcal{D}^m, \rho, \gamma) = \max\{0, |\rho - \hat{\rho}_\theta| - \gamma\} \quad (3.3)$$

where

$$\hat{\rho}_\theta = \frac{\sum_{\substack{(x_{1:n_k}^k, y_{\circ_k}^k) \\ \in \mathcal{D}^m}} \mathbb{E}_{p(y_{1:n_k}^k | x_{1:n_k}^k; \theta)} [\sum_{i=1}^{n_k} \mathbb{1}\{y_i^k \neq \circ\}]}{\sum_{(x_{1:n_k}^k, y_{\circ_k}^k) \in \mathcal{D}^m} n_k} \quad (3.4)$$

is the model's expected rate of entity tags.

For linear-chain CRFs, the inner expected count

$$\mathbb{E}_{p(y_{1:n}|x)} [\sum_{i=1}^n \mathbb{1}\{y_i \neq \circ\}] = \sum_{i=1}^n \sum_{y \in \mathcal{Y} \setminus \{\circ\}} p(y_i | x) \quad (3.5)$$

can be computed exactly, because it factors over the model potentials and reduces to a simple sum over the tag marginals under the model. This follows from linearity of expectations:

$$\mathbb{E}_{y_{1:n}} [\sum_i f(y_i)] = \sum_i \mathbb{E}_{y_{1:n}} [f(y_i)] = \sum_i \mathbb{E}_{y_i} [f(y_i)]$$

The outer expectation in Equation 3.4 is not feasible for large datasets on modern hardware, so we approximate it with Monte-Carlo estimates from mini-batches and optimize using stochastic gradient descent (Robbins and Monro, 1951).

We also note that the loss in Equation 3.3 takes the same form as the  $\epsilon$ -insensitive hinge loss for support vector regression machines (Vapnik, 1995; Drucker et al., 1996), though our use-case is quite different. Additionally, this loss function is differentiable everywhere except at the  $\rho \pm \gamma$  points.

### 3.2.5 Combined Objective and Consistency

The final loss, presented in Equation 3.6, combines Equations 3.1 and 3.3 with a balancing coefficient  $\lambda_u$ .

$$L(\theta; \mathcal{D}, \lambda_u, \rho, \gamma) = L_p(\theta; \mathcal{D}) + \lambda_u L_u(\theta; \mathcal{D}, \rho, \gamma) \quad (3.6)$$

This loss has an intuitive explanation. The supervised loss  $L_p$  optimizes the entity recall of the model. The addition of the EER loss  $L_u$  further controls the precision of the model. Together, they form a principled objective whose optimum recovers the true distribution under mild conditions.

We now present a theorem that gives insight into why the loss in Equation 3.6 is justified. First, we introduce the following set of assumptions:<sup>1</sup>

**Assumption 1.** *Assume there are finite vocabularies of words  $\mathcal{X}$  and tags  $\mathcal{Y}$ , and that  $\mathcal{Y}$  contains a special tag  $\circ$ . We have some model  $p(y_{1:n}|x_{1:n}; \theta)$  with parameter space  $\Theta$ . Assume some distribution  $p_{X,Y}(x_{1:n}, y_{1:n})$  over sequence pairs  $x_{1:n} \in \mathcal{X}^+$ ,  $y_{1:n} \in \mathcal{Y}^+$ , and define  $\mathcal{S} = \{x_{1:n} \in \mathcal{X}^+ : p_X(x_{1:n}) > 0\}$ . Assume in addition the following:*

(a)  *$p_{Y|X}$  is deterministic: that is, for any  $x_{1:n} \in \mathcal{S}$ , there exists some  $y_{1:n} \in \mathcal{Y}^+$  such that*

$$p_{Y|X}(y_{1:n}|x_{1:n}) = 1.$$

(b) *There is some parameter setting  $\theta \in \Theta$  such that  $p(y_{1:n}|x_{1:n}; \theta) = p_{Y|X}(y_{1:n}|x_{1:n})$  for all*

$$(x_{1:n}, y_{1:n}) \in \mathcal{S} \times \mathcal{Y}^+.$$

(c) *We have a set of training examples  $\mathcal{D}^m = \{(x_{1:n_k}^k, y_{1:n_k}^k)\}_{k=1}^m$  drawn from the distribution*

$$p_X(x_{1:n}) \times \tilde{p}_{Y|X}(y_{1:n}|x_{1:n}) \text{ where } \tilde{p}_{Y|X} \text{ has the following properties:}$$

(c1) *No false positives: for all  $x_{1:n} \in \mathcal{S}$ , for all  $i \in \{1 \dots n\}$ , if  $p_{Y|X}(y_i = \circ|x_{1:n}) = 1$ , then*

$$\tilde{p}(y_i = \circ|x_{1:n}) = 1.$$

---

<sup>1</sup>We make use of the following definition: For any finite set  $\mathcal{A}$ , define  $\mathcal{A}^+$  to be the set of finite length sequences of symbols drawn from  $\mathcal{A}$ . That is,  $\mathcal{A}^+ = \{a_{1:n} : n > 0, \forall i, a_i \in \mathcal{A}\}$ .

(c2) *Positive entity support: for all  $x_{1:n} \in \mathcal{S}$ , for all  $i \in \{1 \dots n\}$ , if there is some  $y \in \mathcal{Y}$  such that  $y \neq \circ$  and  $p_{Y|X}(y_i = y|x_{1:n}) = 1$ , then  $\tilde{p}(y_i = y|x_{1:n}) > 0$ , and  $\tilde{p}(y_i = \circ|x_{1:n}) = 1 - \tilde{p}(y_i = y|x_{1:n})$ . That is, only  $y$  and  $\circ$  are possible under  $\tilde{p}$ , and the tag  $y$  has probability strictly greater than zero.*

Given these assumptions, define  $L^\infty$  to be the expected loss under the distribution  $\tilde{p}$ :

$$L^\infty(\theta; \lambda_u, \rho, \gamma) = \mathbb{E}_{\mathcal{D}^m \sim \tilde{p}} [L(\theta; \mathcal{D}^m, \lambda_u, \rho, \gamma)]$$

We can then state the following theorem.

**Theorem 2.** *Assume that all conditions in assumption 1 hold. Define  $\rho = \rho^*$  where  $\rho^*$  is the known marginal entity tag distribution,  $\gamma = 0$ , and  $\lambda_u > 0$ .*

*Then for any  $\theta \in \arg \min L^\infty(\theta; \lambda_u, \rho, \gamma)$ , the following holds:*

$$\begin{aligned} \forall (x_{1:n}, y_{1:n}) &\in \mathcal{S} \times \mathcal{Y}^+, \\ p(y_{1:n}|x_{1:n}; \theta) &= p_{Y|X}(y_{1:n}|x_{1:n}) \end{aligned}$$

The proof of the theorem is given below.

Intuitively, this result is important because it shows that in the limit of infinite data, parameter estimates optimizing the loss function will recover the correct underlying distribution  $p_{Y|X}$ . More formally, this theorem is the first critical step in proving consistency of an estimation method based on optimization of the loss function. In particular (see for example Section 4 of Ma and Collins (2018)) it should be relatively straightforward to derive a result of the form

$$P\left(\lim_{m \rightarrow \infty} d(\hat{p}_{Y|X}^m, p_{Y|X}) = 0\right) = 1$$

under some appropriate definition of distance between distributions  $d$ , where  $\hat{p}_{Y|X}^m$  is the distribution under parameters  $\theta^m$  derived from a random sample  $\mathcal{D}^m$  of size  $m$ . However, we leave this to

future work.<sup>2</sup>

## Proof of Theorem 2

We have

$$L^\infty(\theta; \lambda_u, \rho, \gamma) = g(\theta) + h(\theta)$$

where  $g(\theta) = \mathbb{E}[L_p(\theta; \mathcal{D}^m)]$  and  $h(\theta) = \mathbb{E}[\lambda_u L_u(\theta; \mathcal{D}^m, \rho, \gamma)]$ .

Note that

$$g(\theta) = \sum_{x_{1:n}, y_{1:n}} \tilde{p}(x_{1:n}, y_{1:n}) g'(x_{1:n}, y_{1:n}, \theta) \quad (3.7)$$

where  $\tilde{p}(x_{1:n}, y_{1:n}) = p_X(x_{1:n}) \times \tilde{p}(y_{1:n}|x_{1:n})$  and

$$g'(x_{1:n}, y_{1:n}, \theta) = -\log \sum_{y'_{1:n} \models y_{1:n}} p(y'_{1:n}|x_{1:n}; \theta) \quad (3.8)$$

Define  $\theta^*$  to be such that  $\forall x_{1:n} \in \mathcal{X}, \forall y_{1:n}, p(y_{1:n}|x_{1:n}; \theta^*) = p_{Y|X}(y_{1:n}|x_{1:n})$  (by assumption 1(b) such a parameter setting must exist).

The following properties are easily verified to hold:

1.  $\forall \theta, g(\theta) \geq 0, h(\theta) \geq 0$
2.  $g(\theta^*) = h(\theta^*) = 0$ . Hence  $\theta^*$  is a minimizer of  $g(\theta) + h(\theta)$ .

We now show that any minimizer  $\theta'$  of  $g(\theta) + h(\theta)$  must satisfy the property that  $\forall x_{1:n} \in \mathcal{X}, \forall y_{1:n}, p(y_{1:n}|x_{1:n}; \theta') = p_{Y|X}(y_{1:n}|x_{1:n})$ .

For  $\theta'$  to be a minimizer of  $g(\theta) + h(\theta)$  it must be the case that  $g(\theta') = h(\theta') = 0$ . We then note the following steps:

1. By Lemma 3, if  $g(\theta') = 0$  it must hold that  $\forall x_{1:n} \in \mathcal{X}, \forall i \in \{1 \dots n\}$  such that  $p_{Y|X}(y_i = y|x_{1:n}) = 1$  and  $y \neq \circ, p(y_i = y|x_{1:n}; \theta') = 1$ .

---

<sup>2</sup>One additional remark: Assumption 1 conditions (a) and (b) do not strictly speaking include log-linear models, as probabilities in these models cannot be strictly equal to 1 or 0. However, probabilities under these models can approach arbitrarily close to 1 or 0; for simplicity we present this version of the theorem here, but a more complete analysis could use techniques similar to those in Della Pietra et al. (1997) that make use of the closure of the set of distributions of the model, which include points on the boundary.

2. It remains to be shown that  $\forall x_{1:n} \in \mathcal{X}, \forall i \in \{1 \dots n\}$  such that  $p_{Y|X}(y_i = y|x_{1:n}) = 1$  and  $y = \circ, p(y_i = y|x_{1:n}; \theta') = 1$ .
3. Property (ii) follows from (i) through proof by contradiction. If  $\exists x_{1:n} \in \mathcal{X}$  together with  $i \in \{1 \dots n\}$  such that  $p_{Y|X}(y_i = y|x_{1:n}) = 1$  and  $y = \circ$ , and  $p(y_i = y|x_{1:n}; \theta) < 1$  it must be the case that  $h(\theta') > 0$ , because the expected number of  $\circ$  tags under  $\theta'$  is strictly less than the expected number of  $\circ$  tags under  $p_{Y|X}$ .

□

**Lemma 3.** Define  $g(\theta)$  and  $g'(x_{1:n}, y_{1:n}, \theta)$  as in Eqs. 3.7 and 3.8 above.

$$g(\theta) = \sum_{x_{1:n}, y_{1:n}} \tilde{p}(x_{1:n}, y_{1:n}) g'(x_{1:n}, y_{1:n}, \theta)$$

where  $\tilde{p}(x_{1:n}, y_{1:n}) = p_X(x_{1:n}) \times \tilde{p}(y_{1:n}|x_{1:n})$  and

$$g'(x_{1:n}, y_{1:n}, \theta) = -\log \sum_{y'_{1:n} \models y_{1:n}} p(y'_{1:n}|x_{1:n}; \theta)$$

Then, for any value of  $\theta$  such that  $g(\theta) = 0, \forall x_{1:n} \in \mathcal{X}, \forall i \in \{1 \dots n\}$  such that  $p_{Y|X}(y_i = y|x_{1:n}) = 1$  and  $y \neq \circ, p(y_i = y|x_{1:n}; \theta) = 1$ .

Proof: If  $g(\theta) = 0$ , then for all  $x_{1:n}, y_{1:n}$  such that  $\tilde{p}(x_{1:n}, y_{1:n}) > 0$ , it must be the case that  $-\log \sum_{y'_{1:n} \models y_{1:n}} p(y'_{1:n}|x_{1:n}; \theta) = 0$  and hence  $\sum_{y'_{1:n} \models y_{1:n}} p(y'_{1:n}|x_{1:n}; \theta) = 1$ . The proof is then by contradiction: if there exists some  $x_{1:n} \in \mathcal{X}, i \in \{1 \dots n\}$  such that  $p_{Y|X}(y_i = y|x_{1:n}) = 1$  and  $y \neq \circ$  and  $p(y_i = y|x_{1:n}; \theta) < 1$ , it must be the case that there exists some  $y_{1:n}$  such that  $\tilde{p}(x_{1:n}, y_{1:n}) > 0$ ,  $y_i = y$ , and  $\sum_{y'_{1:n} \models y_{1:n}} p(y'_{1:n}|x_{1:n}; \theta) < 1$ .

□

Dataset	$ \mathcal{Y} $	Train	Dev	Test	Gold	$\rho^*$	NNS	$\rho_{\text{NNS}}$	EE	$\rho_{\text{EE}}$
eng-c	17	203.6K	51.4K	46.4K	34.0K	16.7%	19.3K	3.6%	1.4K	0.7%
deu	17	206.1K	51.3K	51.4K	16.7K	8.1%	9.3K	4.6%	1.4K	0.7%
esp	17	264.4K	52.7K	51.3K	32.8K	12.4%	12.7K	6.7%	1.8K	0.7%
ned	17	201.2K	37.2K	68.6K	19.1K	9.5%	10.8K	5.4%	1.4K	0.7%
eng-o	73	1,644.2K	251.0K	172.1K	239.8K	14.6%	131.4K	8.0%	1.9K	0.1%
chi	73	782.7K	113K	93.0K	91.7K	11.7%	52.8K	6.8%	1.4K	0.2%
ara	73	242.0K	28.3K	28.3K	40.6K	16.8%	22.8K	9.4%	1.9K	0.8%

Table 3.1: *Dataset Statistics*. Each row is organized by dataset with CoNLL03 in the top group and Ontonotes5 below. The first column shows the number of entity tags  $|\mathcal{Y}|$ . The next three columns give the total number of tokens in the train, dev, and test splits for each dataset. The final six columns give the total number of annotated entity tokens in the training split for the gold and simulated low-recall datasets from Section 3.3.2, along with their observed EERs as percentages.

### 3.3 Benchmark Experiments

We evaluate our approach on 7 datasets in 6 languages for two diverse annotation scenarios (14 datasets in total) and compare to strong and state-of-the-art baselines.

#### 3.3.1 Corpora

Our original datasets come from two benchmark NER corpora in 6 languages. We use the English (eng-c), Spanish (esp), German (deu) and Dutch (ned) languages from the CoNLL 2003 shared tasks (Tjong Kim Sang and De Meulder, 2003). We also use the NER annotations for English (eng-o), Mandarin Chinese (chi), and Arabic (ara) from the Ontonotes5 corpus (Hovy et al., 2006).

By studying across this wide array of corpora, we test the approaches in a variety of language settings, as well as dataset and task sizes. The CoNLL corpus specifies 4 entity classes while the Ontonotes corpus has 18 different classes and they span 7.4K to 82K training sentences. Full details are in Table 3.1.

We use standard train/dev/test document splits. For each corpus, we generate two partially annotated datasets according to the scenarios from Section 3.3.2.

### 3.3.2 Simulated Annotation Scenarios

We simulate two partial annotation scenarios that model diverse real-world situations. The first is the “Non-Native Speaker” (NNS) scenario from Mayhew et al. (2019) and the second, “Exploratory Expert” (EE), is a novel scenario inspired by industry. We choose these two samplers to make our results more applicable to practitioners. The simpler alternative – dropping entity annotations uniformly at random (as in Jie et al. (2019); Li et al. (2021)) – is not realistic, leaving an overly diverse set of surface mentions with none of the biases incurred by real-world partial labeling. While there are other partial annotation scenarios compatible with our method that we could have considered here as well, such as using Wikipedia or gazetteers for silver-labeled supervision, we chose to work with simulated scenarios that allow us to study a large array of datasets without introducing the confounding effects of choices for outside resources.

#### **Scenario 1: Non-Native Speaker (NNS)**

Our first low-recall scenario is the one proposed by Mayhew et al. (2019), wherein they study NER datasets that simulate non-native speaker annotators. To simulate data for this scenario, Mayhew et al. (2019) downsample annotations grouped by mention until a recall of 50%. For example, if “New York” is sampled, then all annotations with “New York” as their mention in the text are dropped. After the recall is dropped to 50%, the precision is lowered to 90% by adding short randomly typed false-positive spans. The reasoning for this slightly more complicated scheme is that it better reflects the biases incurred via non-native speaker annotation. When non-native speakers exhaustively annotate for NER, they often systematically miss unrecognized entities and occasionally incorrectly annotate false-positive spans. This happens because some entities are foreign words that are easy to recognize compared to their neighboring words (Mayhew et al., 2019). It is worth noting that the NNS scenario is also quite close to a silver-labeled scenario using a seed dictionary with 50% recall, only it has some additional false positive noise.

The original sampling code used in Mayhew et al. (2019) is not available and we have introduced datasets that were not in their study, so we reimplemented their sampler and used our version

across all of our corpora for consistency. We do, however, run their model code on our datasets, so our results with respect to their approach still hold.<sup>3</sup>

## Scenario 2: Exploratory Expert (EE)

In addition to Mayhew et al. (2019)’s non-native speaker scenario, we introduce a significantly different scenario that reflects another common real-world low-recall NER situation. Though it has not been studied before in the literature, it is inspired by accounts of partially annotated datasets encountered in industry.

In the “Exploratory Expert” (EE) scenario, we suppose a new NER task to be annotated by a domain expert with limited time. Here, in the initial “exploratory” phase of annotation, the expert may wish to cover more ground by inexhaustively scanning through documents in the corpus, annotating the first few entities they see in a document before moving on, stopping once they have added  $M$  total entity spans. The advantage of this approach is that, by being inexhaustive, the resulting set of mentions will occur in a larger diversity of contexts than by using exhaustive annotation when the number of annotations is small. This is because there tends to be a larger diversity of language across documents than within them.<sup>x</sup> Compared to exhaustive annotation, the disadvantage is annotators may miss entities and the annotations are biased toward the top of documents.

We simulate this scenario by first removing all annotations from the dataset, then adding back entity spans with the following process. First, we select a document at random without replacement, then scan this document left to right, adding back entity spans with probability 0.8, until 10 entities have been added, then moving on to the next random document. The process halts when  $M = 1,000$  total entity spans have been added back to the dataset. We note that this assumes that the expert annotators are skimming, sometimes missing entities (20% of the time), but also assumes that the expert does not make flagrant mistakes and so do not insert random false-positive spans.

---

<sup>3</sup>The authors of Mayhew et al. (2019) graciously provided us with their experiment code.

An important aspect of this scenario in our experiments is the scale of the number of kept annotations. In previous works (Jie et al., 2019; Mayhew et al., 2019; Li et al., 2021), the number of kept annotations is not dropped below 50% of the complete dataset. By keeping only 1K entities, this scenario is significantly more impoverished than those previously studied (1K entities leaves less than 10% of annotations for all datasets, ranging from 0.8% to 8.5%, depending on the corpus).

### 3.3.3 Approaches

We compare several modeling approaches on the benchmark corpora, detailed below.

#### **Gold**

For comparison, we report our tagging model trained with supervised sequence likelihood on the original gold datasets. This provides an upperbound on tagging performance and puts any performance degradation from partially-supervised datasets into perspective. We do not expect any of the other methods to outperform this.

#### **Raw**

In the **Raw-BERT** baseline, we make the naive assumption that all unobserved tags in the low-recall datasets are the  $\circ$  tag, reflecting the second row of Figure 3.1, and train with supervised likelihood. This is a weak baseline that we expect to have low recall.

#### **Cost-aware Decoding (Raw+CD)**

This stronger baseline explores a simple modification to the Raw baseline at test time: we increase the cost of predicting an  $\circ$  tag during inference in an attempt to artificially increase the recall. That is, we introduce an additional hyperparameter  $b_{\circ} \geq 0$  that is subtracted from the  $\circ$  tag

potentials, biasing the model away from predicting  $\circ$  tags:

$$\phi(i, y) = \begin{cases} v_y^\top h_i - b_\circ & y = \circ \\ v_y^\top h_i & \text{else} \end{cases}$$

Intuitively, this approach will work well if the tag potentials consistently rank false negative entity tokens higher than true  $\circ$  tokens. To select  $b_\circ$ , we perform a model-based hyperparameter search (Head et al., 2020) using a Gaussian process with 30 evaluations on the validation set F1 score for each dataset’s trained Raw-BERT model.

### Constrained Binary Learning (CBL)

The **CBL** baseline is a state-of-the-art approach to partially supervised NER from Mayhew et al. (2019). The main idea of the approach is to estimate which  $\circ$  tags are false negatives, and remove them from training.

Constrained Binary Learning (CBL) approaches this through a constrained, self-training-like meta-algorithm, based on Constraint-Driven Learning (Chang et al., 2007a). The algorithm starts off with a binarized version of the problem ( $\circ$  tag vs not) and initializes instance weights of 1 for all  $\circ$  tags. It then estimates their final weights by iteratively training a model, predicting tags for the training data, then down-weighting some tags based on the confidence of these predictions according to a linear-programming constraint on the total number of allowed  $\circ$  tags. At each iteration, the number of allowed  $\circ$  tags is decreased slightly, and this loop is repeated until the final target entity ratio (our  $\rho$ ) is satisfied by the weights. A final tagger is then trained on the original tag set using a weighted modification of the supervised tagging likelihood.

For this method, we used the code exactly as was provided, with the following exception. For all non-english languages, we were not able to obtain the original embeddings used in their experiments, and so we have used language-specific pretrained embeddings from the FastText library (Grave et al., 2018). The base tagging model from Mayhew et al. (2019) utilizes the BiLSTM-CRF approach from Ma and Hovy (2016). The CBL meta-algorithm, however, is ag-

nostic to the underlying scoring architecture of the CRF, and so we test the CBL algorithm both with their BiLSTM scoring architecture and with our BERT-based scoring architecture, which we call **CBL-LSTM** and **CBL-BERT** respectively. By testing the CBL meta-algorithm with our tagging model, we control for the different modeling choices and get a clear view of how their CBL approach compares to ours.

### **Span-based Negative Sampling (SNS)**

The **SNS-BERT** baseline is a recent state-of-the-art approach to partially supervised NER from Li et al. (2021). It uses the same BERT-based encoding architecture, but has a different modeling layer on top. Instead of tagging each token, they instead use a span-based scheme, treating each possible pair of tokens as potential entity and classifying all of the spans independently, using an ad-hoc decoding step based on confidence to eliminate overlapping spans. To deal with the resulting class imbalance ( $\bigcirc$  spans are overwhelmingly common) and low-recall entity annotations, they propose to sample spans from the set of unlabeled spans as negatives. While it is possible that they incorrectly sample false negative entities, they argue that this has very low probability. For this method, we used the code as provided but controlled for the same encoding pretrained weights as our other models.

### **Expected Entity Ratio (EER)**

The **EER-BERT** model implements our proposed approach, using the proposed tagger (Section 3.2.2) and loss function described in Equation 3.6.

#### 3.3.4 Preprocessing

All datasets came in documents, pre-tokenized into words, with gold sentence boundaries. Recent work (Akbik et al., 2019; Luoma and Pyysalo, 2020) has demonstrated that larger inter-sentential document context is useful for maximizing performance, so we work with full docu-

ments instead of individual sentences.<sup>4</sup> For approaches that used a pretrained transformer, some documents did not fit into the 512 token maximum length. In these cases, we split documents into maximal contiguous chunks at sentence boundaries. Also, for pretrained transformer approaches we expand the tag sequences to match the subword tokenizations.

Because the low-recall data in the EE scenario concentrates annotations at the top of only a few documents, it is possible to identify and omit large unannotated portions of text from the training data. We hypothesize that this will significantly improve model outcomes for the baselines because it significantly cuts down on the number of false negative annotations. Therefore, we explore three preprocessing variants for all EE models: (1) **all** uses the full dataset as given; (2) **short** drops all documents with no annotations; and (3) **shortest** drops all sentences after the last annotation in a document (subsuming **short**). Model names are suffixed with their preprocessing variants. We note that these approaches do not apply to the NNS scenario, as it has many more annotations spread more evenly throughout the data.

### 3.3.5 Hyperparameters

All hyperparameters were given reasonable defaults, using recommendations from previous work. For pretrained transformer models, we used the Huggingface (Inc., 2019) implementations of `roberta-base` (Liu et al., 2019c) on English datasets and `bert-base-multilingual-cased` (Devlin et al., 2019) for the other languages. The vector representations used by these models are 768-dimensional and we used matching dimensions for other vector sizes throughout the model. We used a learning rate of  $2 \times 10^{-5}$  with slanted triangular schedule peaking at 10% of the iterations (Devlin et al., 2019). For batch size, we use the maximum batch size that will allow us to train in memory on a Tesla V100 GPU (14 for CoNLL data, 2 for Ontonote5 data). We found that training for more epochs than originally recommended (Devlin et al., 2019) was necessary for convergence and used 20 epochs for the **all** variants and 50 epochs for the significantly smaller

---

<sup>4</sup>With the exception of the SNS (Li et al., 2021) baseline where we had to restrict to sentences because it is  $O(n^2)$  span-based model and could not handle long text sequences, running into memory issues.

**short** and **shortest** variants.<sup>5</sup>

The only hyperparameter we adjusted (from a preliminary experiment measuring dev set performance) was setting  $\lambda_u = 10$ . We originally tried a weight of  $\lambda_u = 1$ , but then found that the scale of the  $L_p$  loss massively overpowered  $L_u$ , so we increased it to  $\lambda_u = 10$ , which yielded good performance. We did not try other values after that.

In important contrast to benchmark experiments from prior work (Jie et al., 2019; Mayhew et al., 2019), we do not assume we know the gold entity tag ratio for each dataset when setting  $\rho$ . Instead, to make the evaluation more realistic, we use a reasonable guess of  $\rho = 0.15$  with a margin of uncertainty  $\gamma = 0.05$  for all approaches and datasets. We choose this range because it covers most of the gold ratios observed in the datasets.<sup>6</sup>

### 3.3.6 Results

The results of our evaluation are presented in Table 3.2. The first row shows the result of training our tagger with the original gold data. These results are competitive with previously published results from similar pretrained transformers (Devlin et al., 2019) that do not use ensembles or NER-specific pretraining (Luoma and Pyysalo, 2020; Baevski et al., 2019; Yamada et al., 2020). Interestingly, we also found that our tagging CRF outperformed the span-based independent distribution of Li et al. (2021) on all gold datasets.

**NNS Performance.** The second set of rows shows test F1 scores of models from Section 3.3.3 for the NNS sampled datasets. We first note that the CBL-LSTM approach from Mayhew et al. (2019) significantly underperformed for all non-english languages (and are much lower than the results from their paper with similar data). We used their code as is, only changing the pretrained word vectors, and so suspect that this is due to lower quality word vectors obtained from FastText instead of their custom-fit vectors. This is confirmed by the results of using their CBL meta-algorithm with our proposed tagging architecture, which is competitive with EER-BERT in this

---

<sup>5</sup>For the CBL-LSTM approach, we use the hyperparameters from Mayhew et al. (2019): these are more epochs (45), and a higher learning rate of  $10^{-3}$ .

<sup>6</sup>In early experiments we found that the CBL code from Mayhew et al. (2019) used the gold ratio *plus* 0.05. This additional 0.05 turned out to be critical to getting competitive performance, so in practice we use a  $\rho = 0.2$  for CBL.

Approach / Language	eng-c	deu	esp	ned	eng-o	chi	ara	avg
Gold-BERT-all	92.7	83.9	88.3	91.1	90.7	79.4	72.9	85.6
Gold-SNS-BERT-all	91.1	82.3	87.9	89.5	89.7	77.1	62.1	82.8
Non-Native Speaker Scenario (NNS): Recall=50%, Precision=90%								
Raw-BERT-all	81.9	69.1	71.2	70.1	68.0	61.9	52.8	67.9
Raw+CD-BERT-all	86.3	78.4	79.9	77.2	80.9	64.9	60.1	75.4
CBL-LSTM-all	79.2	38.4	54.6	48.2	67.9	53.5	39.4	54.5
CBL-BERT-all	84.8	<b>77.5</b>	78.7	75.3	76.3	<b>68.9</b>	<b>61.9</b>	74.8
SNS-BERT-all	86.0	77.0	80.8	<b>77.9</b>	81.5	66.4	56.0	75.1
EER-BERT-all	<b>88.0</b>	77.3	<b>80.9</b>	76.9	<b>84.5</b>	66.6	56.6	<b>75.8</b>
Exploratory Expert Scenario (EE): 1, 000 Annotations								
Raw-BERT-all	0.4	02.6	00.7	0.0	0.4	2.4	5.3	1.7
Raw-BERT-short	44.1	37.2	44.4	0.0	28.4	32.4	15.4	28.8
Raw-BERT-shortest	80.7	65.4	73.0	69.1	67.5	57.1	42.0	65.0
Raw+CD-BERT-shortest	82.4	67.9	76.6	70.0	68.9	58.3	43.9	66.9
CBL-LSTM-all	60.2	27.5	41.2	33.3	23.1	29.9	15.3	32.9
CBL-LSTM-shortest	67.8	20.1	36.2	26.7	42.0	24.6	9.7	32.4
CBL-BERT-all	36.4	52.8	40.9	52.5	22.4	29.3	20.8	36.4
CBL-BERT-short	43.7	64.7	56.4	60.8	16.0	31.2	30.2	43.3
CBL-BERT-shortest	80.6	65.1	74.7	71.2	28.4	53.6	39.2	59.0
SNS-BERT-all	59.5	63.8	70.8	70.3	14.0	28.8	0.0	43.9
SNS-BERT-short	64.4	62.6	70.8	64.1	40.7	46.4	0.0	49.9
SNS-BERT-shortest	83.9	70.1	76.8	77.1	75.6	63.3	40.7	69.6
EER-BERT-all	86.3	73.2	<b>80.2</b>	80.2	61.2	56.2	42.9	68.6
EER-BERT-short	<b>89.0</b> <sup>†</sup>	72.2	76.5	<b>80.3</b> <sup>†</sup>	<b>75.9</b>	61.4	<b>46.8</b> <sup>†</sup>	<b>71.7</b> <sup>†</sup>
EER-BERT-shortest	87.3 <sup>†</sup>	<b>73.6</b> <sup>†</sup>	76.5	74.2	74.0	<b>64.3</b>	42.1	70.3

Table 3.2: Benchmark test set F1 scores across different languages and annotation scenarios. Best models in bold.

<sup>†</sup> indicates that for EE the test F1 score is statistically significantly better than SNS-BERT-shortest ( $p < 0.01$ ) (details in footnote 7). Other pairs between SNS-BERT-shortest and EER-BERT-short/shortest were not significant.

setting. Otherwise, we found that all strong baselines and our method performed quite similarly. This suggests that performance in the NNS regime with relatively high recall (50%) and little label noise per positively labeled mention is not bottlenecked by approaches to resolving missing mentions. Further improvements in this regime will likely come from other sources, such as better pretraining or supplemental corpora. Because of this we recommend that future evaluations for

partially supervised NER focus on more impoverished annotation counts, such as the EE scenario we study next.

**EE Performance.** In the third group of rows, we show test F1 scores for each model using the more challenging EE scenario with only 1,000 kept annotations. In this setting, using the dataset as is for supervised training (Raw-BERT-all), fails to converge, but smarter preprocessing largely alleviates this problem, with Raw-BERT-shortest obtaining an average F1 of 65.0. Adding cost-aware decoding (Raw+CD-BERT-shortest) further improves upon the standard baseline (F1 66.9).

Even with only 1,000 biased and incomplete annotations – less than 10% of the original annotations for all datasets – we find that our approach (EER-BERT-short) still achieves an F1 score of 71.7 on average. This outperforms the best strong baselines: Raw+CD, CBL, and SNS, by 4.8, 12.7, and 2.3 F1 score, respectively. The closest baseline, SNS-BERT-shortest from Li et al. (2021), is competitive with EER-BERT-short on four of the datasets, but performs significantly worse on the other three as well as overall,<sup>7</sup> leading us to conclude that our method has a performance edge in this regime. Further, EER-BERT-short performs only 4.1 average F1 worse on EE data than EER-BERT-all on NNS data. We also note that EER-BERT-shortest significantly outperformed SNS-BERT-shortest on two datasets, but failed to reject the null hypothesis overall.

Another important finding is that EER-BERT is much more robust to preprocessing choices than the baselines. The baselines all view missing entities as  $\emptyset$  tags/spans (at least to start) and these relatively common false negatives severely throw off convergence. By removing most of the unannotated text with preprocessing, we effectively create a much smaller corpus that has nearly 80% recall (for shortest). In contrast, EER-BERT’s view of the data makes no assertions about the class of individual unobserved tokens and so is less sensitive to the relative proportion of false negative annotations. This is useful in practice, as our approach should better handle partial

---

<sup>7</sup>We assessed significance between model pairs using a percentile bootstrap of F1 score differences, resampling test set documents with replacement 100K times (Efron and Tibshirani, 1994) and measuring the paired F1 scores differences of EER-BERT-short/shortest and SNS-BERT-shortest. Significance was assessed by whether the two-sided 99% confidence interval contained 0.0. To assess overall significance, we concatenated the test datasets before bootstrapping.

annotation scenarios with wider varieties of false negative proportions that may not be so easily addressed with simple preprocessing.

**Speed.** A pragmatic appeal of our approach compared to CBL (Mayhew et al., 2019) is training time. On NNS data, EER-BERT-all is on average 7.6 times faster than CBL-BERT-all and on EE data EER-BERT-short is 2.2 times faster than CBL-BERT-shortest, even though it uses more data. This is because EER does not require a costly outer self-training loop.<sup>8</sup>

**Conclusion.** These results illustrate that our approach outperforms the previous strong and state-of-the-art baselines in the challenging low-recall EE setting with only 1K annotations while also being more robust to the relative proportions of false negatives in the training corpus.<sup>9</sup>

### 3.3.7 Analysis of EER hyperparameters

Recall that the definition of our EER loss in Equation 3.3 defines an acceptable region  $\hat{\rho}_\theta \in [\rho - \gamma, \rho + \gamma]$  of learned models and that in our this experiment, we used  $\rho = 0.15$  and  $\gamma = 0.05$  for all datasets, regardless of the true entity ratios  $\rho^*$ . Two interesting questions then are:

1. How sensitive is the procedure to choices of  $\rho$  and  $\gamma$ ?
2. How closely do the final learned models reflect the true entity ratios for the data?

We address these next.

#### Robustness to choices of $\rho$ and $\gamma$

To study robustness we varied choices of  $\rho$  and  $\gamma$  for EER-BERT-short on the CoNLL English EE dataset with three randomly sampled datasets. Table 3.3.7 shows test F1 scores across seeds for various settings of  $\rho \pm \gamma$ . We first show three point estimates with  $\gamma = 0.0$ , the first at  $\rho = \rho^* = 0.23$ , then shifted around  $\rho^*$  left and right to  $\rho = 0.15$  and  $\rho = 0.30$ , respectively. We then widen the

---

<sup>8</sup>We unfortunately cannot comment on relative speed of SNS because runtimes cannot be inferred from the SNS code output, though we do not expect a fundamental speed advantage of one over the other, as neither use self-training.

<sup>9</sup>We also note that the EE scenario averages for all models are significantly affected by the poor performance on the Arabic Ontonotes5 (ara) dataset. After further inspection of the training curves, we found that all models exhibited very slow convergence on this dataset and/or failed to converge in the allotted number of epochs.

range with  $\gamma = 0.05$  and show the benchmark result  $\rho = 0.15$ , followed by shifts of  $\rho \pm 0.1$ . Finally we show a very wide range of  $\rho = 0.15, \gamma = 0.15$ .

Varying EER HPs Test F1 Scores				
$[\rho - \gamma, \rho + \gamma]$	RS0	RS1	RS2	Avg.
$[0.23, 0.23]^*$	86.8	87.4	87.0	87.1
$[0.15, 0.15]$	89.3	87.1	87.8	88.1
$[0.30, 0.30]$	79.1	79.4	79.7	79.4
$[0.10, 0.20]^\dagger$	87.6	88.2	87.8	87.9
$[0.20, 0.30]$	83.9	83.8	84.1	83.9
$[0.00, 0.10]$	89.2	87.1	87.8	88.0
$[0.00, 0.30]$	83.9	83.8	84.0	83.9

Table 3.3: CoNLL English EE EER-short test set F1 across three randomly sampled datasets. \*:  $\rho = \rho^*$ .  $\dagger$ : benchmark experiment setting.

From the table we can glean two interesting points. The first is that in settings where the high end of range of acceptable EER's is greater then  $\rho^*$  (when  $\rho + \gamma = 0.30$ ) there is a substantial drop in performance (mean = 82.3). The second is that the complement group of settings, where  $\rho + \gamma \leq \rho^*$  are all high-performing with little variance (mean = 87.8, std = 0.4). Together they suggest that the true sensitivity of the proposed EER approach to the high end of the interval and that it is best to conservatively estimate that value, whereas the low end of the range is unimportant. This result agrees well with the intuitions provided in Section 3.2.5: since  $L_p$  is encouraging models with high recall without regard for precision ( $\hat{\rho}_\theta \rightarrow 1$ ), it is best to set  $\rho + \gamma$  such that  $L_u$  introduces a tension in the combined loss by encouraging  $\hat{\rho}_\theta \leq \rho^*$ . This is not the whole story, however, as we discuss next.

## Convergence towards $\rho^*$

The results from the previous experiment suggests that  $L_u$  simply serves to drive  $\hat{\rho}_\theta \rightarrow \rho + \gamma$ . Since we used  $\rho + \gamma = 0.2$  for all datasets in the benchmark, we would then expect to see a result that  $\hat{\rho}_\theta \approx 0.2$  for all models.

We tested this hypothesis by calculating the entity ratio  $\hat{\rho}_\theta$  of final trained EER-BERT-short models for the EE datasets (leaving out ara, since it failed to converge) and calculated the average difference of each  $\hat{\rho}_\theta$  with respect to the corresponding true  $\rho^*$ , resulting in mean absolute error of only 0.018. This is much closer on average than if the models just converged to 0.2 (the mean absolute error then would be 0.048), indicating that our approach tends to converge more closely to the true entity ratio  $\rho^*$  than the estimate given by  $\rho + \gamma$ . In particular, we found that all final models had  $\hat{\rho}_\theta < 0.2$  except CoNLL English, where  $\hat{\rho}_\theta = 0.23$ , quite close to the gold  $\rho^*$  even though it was outside of the target range. This result is encouraging in that it suggests the EER loss, in balance with the supervised marginal tag loss, does more to recover  $\rho^*$  than just drive  $\hat{\rho}_\theta \rightarrow \rho + \gamma$ .

## 3.4 EE vs. Exhaustive Experiments

In situations where we only have partially annotated data without the option for exhaustive annotations, the utility of being able to train with the data as provided is self-evident. However, given the potential upsides of partial annotation relative to exhaustive annotation – mentally less taxing and increased contextual diversity for a fixed annotation budget – it is natural to ask whether it is actually *better* to go with a sparse annotation scheme.

### 3.4.1 Annotation Speed User Study

We begin with a user study of annotation speed, comparing EE to the standard exhaustive annotation scheme. Following methodology from Li et al. (2020), we recorded 8 annotation sessions from 4 NLP researchers familiar with NER. Using the Ontonotes5 English corpus, we asked each annotator to annotate for two 20 minute sessions using the BRAT (Stenetorp et al., 2012) annota-

tion tool, one exhaustively and the other following the EE scheme. We split documents into two randomized groups and systematically varied which group was annotated with each scheme and in what order to control for document and ordering variation effects. Then, for each annotator, we measured the number of annotated entities per minute for both schemes and report the ratio of EE annotations per minute to exhaustive annotations per minute (i.e., the relative speed of EE to exhaustive). We found that, although speed varied greatly between annotators (ranging from roughly 4 annotations/min to 9 annotations/min across sessions), EE annotation and exhaustive annotation were essentially the same speed, with EE being 3% faster on average. Thus we may fairly compare exhaustive and EE schemes using model performance at the same number of annotations, which we do next.<sup>10</sup>

### 3.4.2 Performance Learning Curves

In this experiment, we compare the best traditional supervised training from the benchmark (Raw-BERT-shortest) with our proposed approach (EER-BERT-short) on EE-annotated and exhaustively annotated documents from CoNLL’03 English (eng-c) at several annotation budgets,  $M \in \{100 (0.4\%), 500 (2.1\%), 1K (4.3\%), 5K (21.3\%), 10K (42.6\%)\}$ . For each annotation budget, we sampled three datasets with different random seeds for both annotation schemes and trained both modeling approaches. This allows us to study how all four combinations of annotation style and training methods perform at varying magnitudes of annotation counts. In addition to low-recall annotations, we compared our EER approach to supervised training on the gold data.

In Figure 3.2, we show learning curves for the average test performance of all four annotation/training variants. From the plot, we can infer several points. First, on EE-annotated data, using our EER loss substantially outperforms traditional likelihood training at all amounts of partial annotation, but the opposite is true on exhaustively annotated data. This indicates that the training method should be tailored to the annotation scheme.

The comparison between EE data with EER training versus exhaustive data with likelihood

---

<sup>10</sup>The exact number of annotated entities among the four annotation sessions for EE were 91, 90, 109, and 179. For Exhaustive the matching annotation counts were 83, 85, 117, and 170.

training is more nuanced. At only 100 annotations, exhaustive annotation worked best on average in our sample, but all methods exhibit high variance due to the large variation in *which* entities were annotated. Interestingly, at modest sizes of only 500 and 1K annotations, EE annotated data with our proposed EER-short approach outperformed exhaustive annotation with traditional supervised training, with gains of +1.8 and +1.5 average F1 for 500 and 1K annotations, respectively. These results, however, reverse as the annotation counts grow: at 5K annotations, the two approaches perform the same (90.8) and, at even larger annotation counts, exhaustive annotation with traditional training outperforms our approach by +0.5 at 10K annotations and +0.8 on the gold dataset. This indicates that EE annotation, paired with our EER loss, is competitive and potentially advantageous to exhaustive annotation and traditional training at modest annotation counts, but that

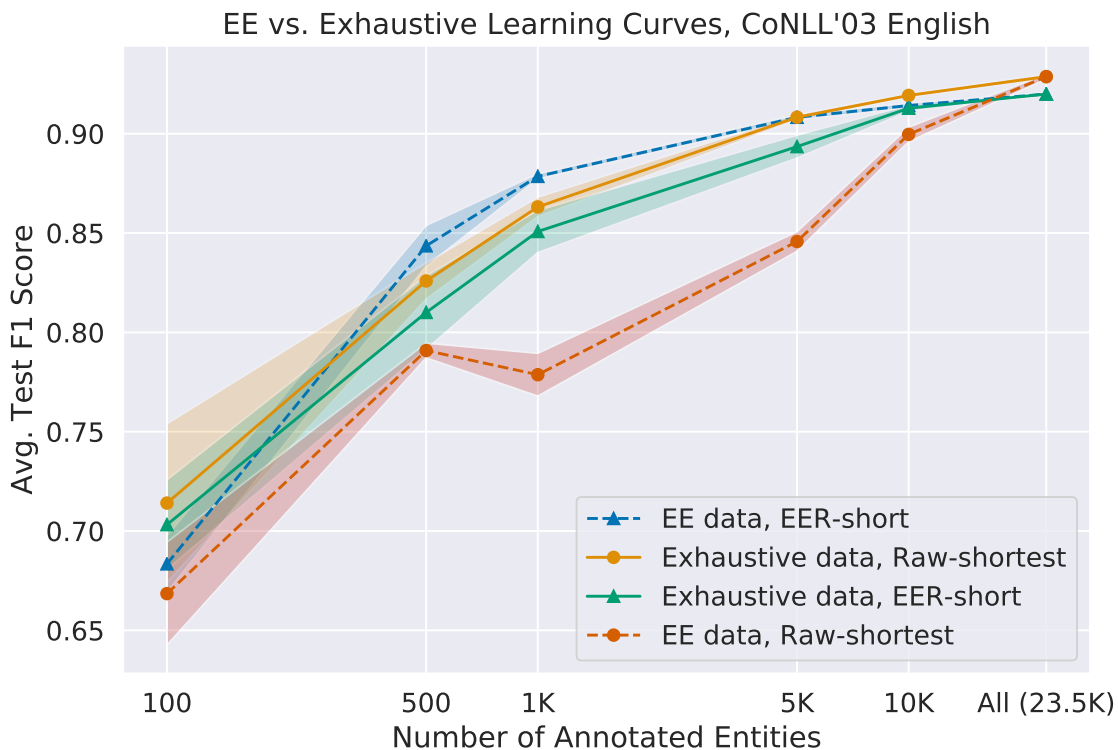


Figure 3.2: Test performance as a function of the number of observed training annotations for the Exhaustive vs. EE annotation on CoNLL English. Lines are averages and shaded regions are  $\pm 1$  standard error.

exhaustive annotation with traditional training is better at large annotation counts. This suggests that a hybrid annotation approach where we sparsely annotate data at first, but eventually switch to exhaustive annotations as the process progresses, is a promising direction of future work. We note that our EER loss can easily incorporate observed  $\circ$  tags from exhaustively annotated documents in  $y_O$  and so would work in this setup without modification.

### 3.5 Related Work

A common paradigm for low-recall NER is automatically creating silver-labeled data using outside resources. Bellare and McCallum (2007) approach the problem by distantly supervising spans using a database with marginal tag training. Carlson et al. (2009) similarly use a gazetteer and adapt the structured perceptron (Collins, 2002) to handle partially labeled sequences, while Nothman et al. (2008) use Wikipedia and label-propagation heuristics. Peng et al. (2019) also use distant supervision to silver-labeled entities, but use PU-learning with specified class priors to estimate individual classifiers with ad-hoc decoding. Yang et al. (2018); Nooralahzadeh et al. (2019) optimize the marginal likelihood (Tsuboi et al., 2008) of the distantly annotated tags but require gazatteers and some fully labeled data to handle proper prediction of the  $\circ$  tag. Greenberg et al. (2018a) use a marginal likelihood objective to pool overlapping NER tasks and datasets, but must exploit cross-dataset constraints. Snorkel (Ratner et al., 2017) uses many sources of weak supervision, but relies on high-recall and overlap to work. In contrast to these works, we do not use outside resources.

Our problem setting has connections to PU-learning, which is classically an approach to classification (Liu et al., 2002, 2003; Elkan and Noto, 2008; Grave, 2014), but here we work with tagging structures. Our approach is also related to constraint-satisfaction methods for shaping the model distribution such as CoDL (Chang et al., 2007a), used by Mayhew et al. (2019), and is also related to Posterior Regularization (Ganchev et al., 2010a), with main differences being that we do not use the KL-divergence and use gradient-based updates to a nonlinear model instead of closed-form updates to a log-linear model.

The problem setup from Jie et al. (2019); Mayhew et al. (2019) is the same as ours, but Jie et al. (2019) use a cross-validated self-training approach and Mayhew et al. (2019) use an iterative constraint-driven self-training approach to down-weight possible false-negative  $O$  tags, which they show to outperform Jie et al. (2019). Mayhew et al. (2019) is the current state of the art on the CoNLL 2003 NER datasets (Tjong Kim Sang and De Meulder, 2003) and we compare to their work in the experiments. Recently, Li et al. (2021) have published a span-based method that uses negative sampling of non-entity spans, but they do not provide any supporting theoretical guarantees. We also compare to them in the experiments.

### 3.6 Conclusions

We study learning NER taggers in the presence of partially labeled data and propose a simple, fast, and theoretically principled approach, the Expected Entity Ratio loss, to deal with low-recall annotations. We show empirically that it outperforms the previous state of the art across a variety of languages, annotation scenarios, and amounts of labeled data. Additionally, we give evidence that sparse annotations, when paired with our approach, are a viable alternative to exhaustive annotation for modest annotation budgets.

Though we study two simulated annotation scenarios to provide controlled experiments, our proposed EER approach is compatible with a variety of other incomplete annotation scenarios, such as incidental annotations (e.g., from web links on Wikipedia), initialized by seed annotations from incomplete distant supervision/gazetteers, or embedded as a learning procedure in an active/iterative learning framework, which we intend to explore in future work.

The method developed in this chapter pushes the boundaries of annotation-efficiency in NER. As we have seen, the proposed approach allows us to trade off completeness of the annotated data for biased low-recall annotation processes that can generate a higher variety of labeled input texts on fixed annotation budgets and draw on other sparse and incomplete data sources without incurring the same drawbacks as previous works in the literature. Further our approach allows us to build usable models on datasets that are significantly more sparsely annotated than the previous

state-of-the-art methods, going as low as only observing only 0.6% of the true entities, while maintaining 84.7% of the fully supervised F1 score. It is our hope that this method can expand the accessibility of NER to more diverse and niche applications by reducing the annotation burden on would-be NER practitioners.

## Chapter 4: Improving Low-Resource Cross-lingual Parsing with Expected Statistic Regularization

In Chapter 3, we developed an approach for learning NER taggers on partially annotated data with very low recall and systematically missing tags. The proposed approach for addressing the partially supervised NER problem was to penalize the marginal entity tag distribution of the model if it deviated outside a range of acceptable values. This computation of the marginal entity tag distribution can be thought of as a “descriptive *statistic*” for the model – a function that quantitatively describes a property of the model distribution given a sample of the data. Further the EER loss can be thought of as *regularizing* the model by encouraging the marginal entity tag statistic to fall within a reasonable range of prior expected values.

In this chapter, we propose to take this abstract idea behind EER and generalize it far beyond a single statistic function. We propose a novel and general regularization technique “Expected Statistic Regularization” (ESR), that uses a wide class of statistic functions and their expected values to keep various aspects of model behavior from diverging outside of our expectations on unsupervised data. Interestingly, these statistic functions and their expectations can be formulated using expert knowledge about the task, so that we can impart high level information about the target task on the model without directly labeling examples, allowing us to increase the annotation-efficiency of learning. Further, as we saw in Chapter 3, these statistics can be used to counter biases in the model that are induced by biases in the training data, allowing us to train in biased settings.

Though we propose a general method here, we study it empirically in the context of an important structured prediction problem in NLP: syntactic part-of-speech tagging and dependency parsing in low-resource languages. This problem is a perfect application area for studying ESR because its rich structure allows for the analysis of many interesting types of statistics, based on

the theory of syntactic typology, that bear on different aspects of model behavior. Further, because this problem has already seen great progress in recent years from applications of multi-lingual deep learning and cross-lingual transfer approaches, we have an opportunity to see if the use of high-level expert knowledge about the problem, in the form of ESR, can be complimentary to this strong baseline. As we will see in the experiments, ESR helps to improve cross-lingual transfer in precisely the areas where multi-lingual pretraining and fine-tuning fail by discouraging the model from making erratic and unsensible predictions in regions of the input space that are “distant” or poorly represented in the training data.

The rest of this chapter is organized as follows. In Section 4.1 we discuss the problem background and motivation. In Sections 4.2 and 4.3 we describe the general proposed framework of ESR and a general approach for estimating supervision targets from small labeled samples, respectively. Then, in Section 4.4 we describe its proposed application to state-of-the-art cross-lingual syntactic analysis. In Section 4.5 we present oracle unsupervised experiments that analyze the effectiveness of the proposed statistics along with other controlled ablations. In Section 4.6 we conduct extensive experiments in realistic low-resource transfer scenarios across a wide range of languages and amounts of labeled data. Finally, in Section 4.7 we discuss related work and in Section 4.8 we discuss concluding thoughts.

## **4.1 Background and Motivation**

In recent years, great strides have been made on linguistic analysis for low-resource languages. These gains are largely attributable to transfer approaches from:

1. Massive pretrained multilingual language model (PLM) encoders (Devlin et al., 2019; Liu et al., 2019c)
2. Multi-task training across related syntactic analysis tasks (Kondratyuk, 2019)
3. Multi-lingual training on diverse high-resource languages (Wu and Dredze, 2019; Ahmad et al., 2019; Kondratyuk, 2019)

Combined, these approaches yield a methodology that has been shown to be particularly effective for cross-lingual syntactic analysis, as exemplified by UDify (Kondratyuk, 2019).

However, even with the improvements brought about by these techniques, transferred models still make syntactically implausible predictions on low-resource languages, and these error rates increase dramatically as the target languages become more distant from the source languages (He et al., 2019; Meng et al., 2019). In particular, the transferred models fail to match many low-order statistics concerning the typology of the task structures. We hypothesize that enforcing regularity with respect to estimates of these structural statistics — effectively using them as weak supervision — is complementary to current transfer approaches for low-resource cross-lingual parsing.

To this end, we introduce Expected Statistic Regularization (ESR), a novel differentiable loss that regularizes models on unlabeled target datasets by minimizing deviation of descriptive statistics of model behavior from target values. The class of descriptive statistics usable by ESR are expressive and powerful. For example, they may describe cross-task interactions, encouraging the model to obey structural patterns that are not explicitly tractable in the model factorization. Additionally, the statistics may be derived from constraints dictated by the task formalism itself (such as ruling out invalid substructures) or by numerical parameters that are specific to the target dataset distribution (such as relative substructure frequencies). In the latter case, we also contribute a method for selecting those parameters using small amounts of labeled data, based on the bootstrap (Efron, 1979).

Although ESR is applicable to a variety of problems, we study it using modern cross-lingual syntactic analysis on the Universal Dependencies data, building off of the strong model-transfer framework of UDify (Kondratyuk, 2019). We show that ESR is complementary to transfer-based approaches for building parsers on low-resource languages and that they can be used together. We present several interesting classes of statistics for the tasks and perform extensive experiments in both oracle unsupervised and realistic semi-supervised cross-lingual multi-task parsing scenarios, with particularly encouraging results that significantly outperform state-of-the-art approaches for semi-supervised scenarios. We also present ablations that justify key design choices.

Specifically, we make the following contributions:

- A novel and general regularization framework, “Expected Statistic Regularization” (ESR), that can be used to regularize models on unlabeled target datasets with a broad class of functions that describe expected model behavior. These statistics allow for the incorporation of various forms of high-level expert knowledge as supervision.
- A method for estimating target statistic values using small amounts of labeled data.
- An application of the method to that improves state-of-the-art cross-lingual parsing on low-resource languages. We contribute seven families of descriptive statistics that bear on parser behavior and extensively evaluate their impact on transfer, showing most to be useful.
- An extensive benchmark evaluation on transfer to 44 languages showing that ESR leads to significant improvements over state-of-the-art approaches on many low-resource languages.
- Learning curve experiments that demonstrate the impact of the approach is largest for target datasets with 500 or fewer annotated sentences.
- Ablation studies justifying key design choices for the proposed loss function.

## 4.2 Expected Statistic Regularization

We consider structured prediction in an abstract setting where we have inputs  $x \in \mathcal{X}$ , output structures  $y \in \mathcal{Y}$ , and a conditional model  $p_\theta(y|x) \in \mathbb{P}$  with parameters  $\theta \in \Theta$ , where  $\mathbb{P}$  is the distribution space and  $\Theta$  is the parameter space. In this section we assume that the setting is semi-supervised, with a small labeled dataset  $\mathcal{D}_L$  and a large unlabeled dataset  $\mathcal{D}_U$ ; let  $\mathcal{D}_L = \{(x_i, y_i)\}_{i=1}^m$  be the labeled dataset of size  $m$  and similarly define  $\mathcal{D}_U = \{x_i\}_{i=m+1}^{m+n}$  as the unlabeled dataset.

Our approach centers around a vectorized statistic function  $f$  that maps unlabeled samples and

models to real vectors of dimension  $d_f$ :

$$f : \mathbb{D} \times \mathbb{P} \rightarrow \mathbb{R}^{d_f} \quad (4.1)$$

where  $\mathbb{D}$  is the set of unlabeled datasets of any size, (i.e.,  $\mathcal{D}_U \in \mathbb{D}$ ). The purpose of  $f$  is to summarize various properties of the model using the sample. For example, if the task is part-of-speech tagging, one possible component of  $f$  could be the expected proportion of NOUN tags in the unlabeled data  $\mathcal{D}_U$ . In addition to  $f$ , we assume that we are given vectors of target statistics  $t \in \mathbb{R}^d$  and margins of uncertainty  $\sigma \in \mathbb{R}^d$  as its supervision signal. We will discuss the details of  $f$ ,  $t$ , and  $\sigma$  shortly but first describe the overall objective.

#### 4.2.1 Semi-Supervised Objective

Given labeled and unlabeled data  $\mathcal{D}_L$  and  $\mathcal{D}_U$ , we propose the following semi-supervised objective  $O$ , which breaks down into a sum of supervised and unsupervised terms  $L$  and  $C$ :

$$\hat{\theta} = \arg \min_{\theta \in \Theta} O(\theta; \mathcal{D}_L, \mathcal{D}_U) \quad (4.2)$$

$$O(\theta; \mathcal{D}_L, \mathcal{D}_U) = L(\theta; \mathcal{D}_L) + \alpha C(\theta; \mathcal{D}_U) \quad (4.3)$$

where  $\alpha > 0$  is a balancing coefficient. The supervised objective  $L$  can be any suitable supervised loss; here we will use the negative log-likelihood of the data under the model. Our contribution is the unsupervised objective  $C$ .

For  $C$ , we propose to minimize some distance function  $\ell$  between the target statistics  $t$  and the value of the statistics  $f$  calculated using unlabeled data and the model  $p_\theta$ . ( $\ell$  will also take into account the uncertainty margins  $\sigma$ .) A simple objective would be:

$$C(\theta; \mathcal{D}_U) = \ell(t, \sigma, f(\mathcal{D}_U, p_\theta))$$

This is a dataset-level loss penalizing divergences from the target level statistics. The problem with

this approach is that this is not amenable to modern hardware constraints requiring SGD. Instead, we propose to optimize this loss in expectation over unlabeled mini-batch samples  $\mathcal{D}_U^k$ , where  $k$  is the mini-batch size and  $\mathcal{D}_U^k$  is sampled uniformly with replacement from  $\mathcal{D}_U$ . Then,  $C$  is given by:

$$C(\theta; \mathcal{D}_U) = \mathbb{E}_{\mathcal{D}_U^k} [\ell(t, \sigma, f(\mathcal{D}_U^k, p_\theta))] \quad (4.4)$$

This objective penalizes the model if the statistic  $f$ , when applied to samples of unlabeled data  $\mathcal{D}_U^k$ , deviates from the targets  $t$  and thus pushes the model toward satisfying these target statistics.

Importantly, the objective in Equation 4.4 is more general than typical objectives in that the outer loss function  $\ell$  does not necessarily break down into a sum over individual input examples — the aggregation over examples is done inside  $f$ :

$$\ell(t, \sigma, f(\mathcal{D}_U, p_\theta)) \neq \sum_{x \in \mathcal{D}_U} \ell(t, \sigma, f(x, p_\theta)) \quad (4.5)$$

This generality is useful because components of  $f$  may describe statistics that aggregate over inputs, estimating expected quantities concerning sample-level regularities of the structures. In contrast, the right-hand side of Equation 4.5 is more stringent, imposing that the statistic be the same for all instances of  $x$ . In practice, this loss reduces noise compared to a per-sentence loss, as is shown in Section 4.5.3.

#### 4.2.2 The Statistic Function $f$

In principle the vectorized statistic function  $f$  could be almost any function of the unlabeled data and model, provided it is possible to obtain its gradients w.r.t. the model parameters  $\theta$ , however, in this work we will assume  $f$  has the following three-layer structure.

First, let  $g$  be another vectorized function of "sub-statistics" that may have a different dimensionality than  $f$  and takes individual  $x, y$  pairs as input:

$$g : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^{d_g} \quad (4.6)$$

Then let  $\bar{g}$  be the expected value of  $g$  under the model  $p_\theta$  summed over the sample  $\mathcal{D}_U$ :

$$\bar{g} = \sum_{x \in \mathcal{D}_U} \mathbb{E}_{p_\theta(y|x)}[g(x, y)] \quad (4.7)$$

Given  $\bar{g}$ , let the  $f$ 's  $j$ 'th component be the result of an aggregating function  $h_j : \mathbb{R}^{d_g} \rightarrow \mathbb{R}$  on  $\bar{g}$ :

$$f_j(\mathcal{D}_U, p_\theta) = h_j(\bar{g}) \quad (4.8)$$

The individual components  $g_i$  will mostly be counting functions that tally various substructures in the data. The  $\bar{g}_i$ 's then are expected substructure counts in the sample, and the  $h_j$ 's aggregate small subsets of these intermediate counts in different ways to compute various marginal probabilities. Again, in general  $f$  does not need to follow this structure and any suitable statistic function can be incorporated into the regularization term proposed in Equation 4.4.

In some cases—when the structure of  $g$  does not follow the model factorization either additively or multiplicatively—computation of the model expectation  $\mathbb{E}_{p_\theta(y|x)}[g(x, y)]$  in Equation 4.7 is intractable. In these situations, standard Monte Carlo approximation breaks differentiability of the objective w.r.t. the model parameters  $\theta$  and cannot be used. To remedy this, we propose to use the “Stochastic Softmax” differentiable sampling approximation from Paulus et al. (2020) to allow optimization of these functions. We propose several such statistics in the application (see Section 4.4.3 for the statistics and Section 4.4.5 for a description of this approximation).

#### 4.2.3 The Distance Function $\ell$

For the distance function  $\ell$ , we propose to use a smoothed hinge loss (Girshick, 2015) that adapts with the margins  $\sigma$ . Letting  $\bar{f} = f(\mathcal{D}_U^k, p_\theta)$ , the  $i$ 'th component of  $\ell$  is given by:

$$\ell_i = \begin{cases} \frac{(\bar{f}_i - t_i)^2}{2\sigma_i} & \text{if } |\bar{f}_i - t_i| < \sigma_i \\ |\bar{f}_i - t_i| - \sigma_i & \text{else} \end{cases} \quad (4.9)$$

The total loss  $\ell$  is then the sum of its components:

$$\ell(t, \sigma, f(\mathcal{D}_U^k, p_\theta)) = \sum_i \ell_i(t_i, \sigma_i, \bar{f}_i) \quad (4.10)$$

We choose this function because it is robust to outliers, adapts its width to the margin parameter  $\sigma_i$ , and expresses a preference for  $f_i = t_i$  (as opposed to max-margin losses). We give an ablation study in Section 4.5.3 justifying its use.

### 4.3 Choosing the Targets and Margins

There are several possible approaches to choosing the targets  $t$  and margins  $\sigma$ , and in general they can differ based on the individual statistics. For some statistics it may be possible to specify the targets and margins using prior knowledge or formal constraints from the task. In other cases, estimating the targets and margins may be more difficult. Depending on the problem context, one may be able to estimate them from related tasks or domains (such as neighboring languages for cross-lingual parsing). Here, we propose a general method that estimates the statistics using labeled data, and is applicable to semi-supervised scenarios where at least a small amount of labeled data is available.

The ideal targets are the expected statistics under the “true” model  $p^*$  are:  $t^* = \mathbb{E}_{\mathcal{D}_U^k} [f(\mathcal{D}_U^k, p^*)]$ , where  $k$  is the batch size. We can estimate this expectation using labeled data  $\mathcal{D}_L$  and bootstrap sampling (Efron, 1979). Utilizing  $\mathcal{D}_L$  as a set of point estimates for  $p^*$ , we sample  $B$  total minibatches of  $k$  labeled examples uniformly with replacement from  $\mathcal{D}_L$  and calculate the statistic  $f$  for each of these minibatch datasets. We then compute the target statistic as the sample mean:

$$t = \frac{1}{B} \sum_{i=1}^B f(\mathcal{D}_L^{(i)}) , \quad |\mathcal{D}_L^{(i)}| = k, \quad \forall i \quad (4.11)$$

where we have slightly abused notation by writing  $f(\mathcal{D}_L)$  to mean  $f$  computed using the inputs  $\{x : (x, y) \in \mathcal{D}_L\}$  and the point estimates  $p^*(y|x) = 1, \forall (x, y) \in \mathcal{D}_L$ .

In addition to estimating the target statistics for small batch sizes, the bootstrap gives us a way

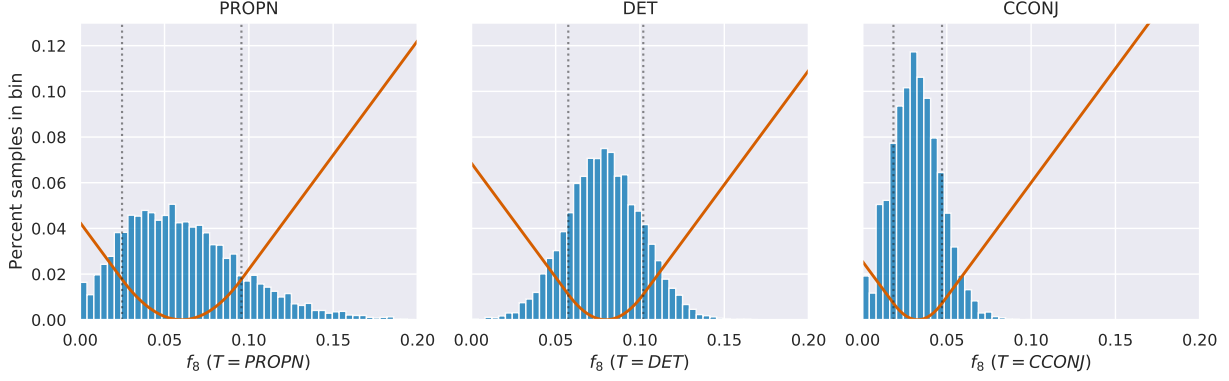


Figure 4.1: *Adaptive Loss Function Visualization*. Example statistic sampling distributions (sample size = 8) and their induced loss functions. The histograms show the sampling distribution of statistics for POS tag frequency for three POS tags: PROP, DET, and CCONJ. The solid orange lines show the loss functions induced by these distributions. The samples have decreasing variance from left to right, and the respective losses adaptively become narrower in response. The vertical dotted lines show the  $t_k \pm \sigma_k$  boundaries where the  $\ell_k$  switches from a scaled L2 to an unscaled L1.

to estimate the natural variation of the statistics for small sample sizes. To this end, we propose to utilize the standard deviations from the bootstrap samples as our margins of uncertainty  $\sigma$ :

$$\sigma = \sqrt{\frac{1}{B-1} \sum_{i=1}^B (f(\mathcal{D}_L^{(i)}) - t)^2} \quad (4.12)$$

This allows our loss function  $\ell$  to adapt to more or less certain statistics. If some statistics are naturally too variable to serve as effective supervision, they will automatically have weak contribution to  $\ell$  and little impact on the model training. The adaptivity of the loss function with respect to the sampling distribution is visualized in Figure 4.1.

#### 4.4 Application to Cross-Lingual Parsing

Now that we have described our general approach, in this section we lay out a proposal for applying it to cross-lingual joint POS tagging and dependency parsing. We choose to apply our method to this problem because it is an ideal testbed for controlled experiments in semi-supervised structured prediction. By their nature, the parsing tasks admit many types of interesting statistics

that capture cross-task, universal, and language-specific facts about the target test distributions. Also, because the data is highly multilingual, we can study our method under domain adaptation conditions in addition to the more typical “from scratch” scenario.

We will evaluate in two different transfer settings: oracle unsupervised and realistic semi-supervised. In the oracle unsupervised settings, there is no supervised training data available for the target languages (and the  $L$  term is dropped from Equation 4.3), but we use target values and margins calculated from the held-out supervised data. This setting allows us to understand the impact of our regularizer in isolation without the confounding effects of direct supervision or inaccurate targets. In the semi-supervised experiments, we vary the amounts of supervised data, and calculate the targets from the small supervised data samples. This is a realistic application of our approach that may be applied to low-resource learning scenarios.

#### 4.4.1 Problem Setup and Data

We use the Universal Dependencies (Nivre, 2020) v2.8 (UD) corpus as data. In UD, syntactic annotation is formulated as a labeled bilexical dependency tree, connecting words in a sentence, with additional part-of-speech (POS) *tags* annotated for each word. The labeled tree can be broken down into two parts: the *arcs* that connect the *head* words to *child* words, forming a tree, and the dependency *labels* assigned to each of those arcs. Due to the definition of UD syntax, each word is the child of exactly one arc, and so both the attachments and labels can be written as sequences that align with the words in the sentence.

More formally then, for each labeled sentence  $x_{1:n}$  of length  $n$ , the full structure  $y$  is given by the three sequences  $y = (t_{1:n}, e_{1:n}, r_{1:n})$ , where  $t_{1:n}$ ,  $t_i \in \mathcal{T}$  are the POS tags,  $e_{1:n}$ ,  $e_i \in \{1, \dots, n\}$  are the head attachments, and  $r_{1:n}$ ,  $r_i \in \mathcal{R}$  are the dependency labels.

#### 4.4.2 The Model and Training

We now turn to the parsing model that is used as the basis for our approach. Though the general ideas of our approach are adaptable to other models, we choose to use the UDify architecture

because it is one of the state-of-the-art multilingual parsers for UD.

## The UDify Model

The UDify model is based on trends in state-of-the-art parsing, combining a multilingual pre-trained transformer language model encoder (mBERT) with a deep biaffine arc-factored parsing decoder, following Dozat and Manning (2017). These encodings are additionally used to predict POS tags with a separate decoder. The full details are given in Kondratyuk (2019), but here it suffices to characterize the parser by its top-level probabilistic factorization:

$$\begin{aligned} p(t_{1:n}, e_{1:n}, r_{1:n} | x_{1:n}) \\ = p(e_{1:n} | x_{1:n}) p(t_{1:n} | x_{1:n}) p(r_{1:n} | e_{1:n}, x_{1:n}) \end{aligned} \quad (4.13)$$

$$= p(e_{1:n} | x_{1:n}) \prod_{i=1}^n p(t_i | x_{1:n}) p(r_i | e_i, x_{1:n}) \quad (4.14)$$

This model is scant on explicit joint factors, following recent trends in structured prediction that forgo higher-arity factors, instead opting for shared underlying contextual representations produced by a mBERT that implicitly contain information about the sentence and structure as a whole. This factorization will prove useful in Section 4.4.3 where it will allow us to compute many of the supervision statistics under the model exactly.

## Training

The UDify approach to training is simple: it begins with a multilingual PLM, mBERT, then fine-tunes the parsing architecture on the concatenation of the source languages. With vanilla UDify, transfer to target languages is zero-shot.

Our approach begins with these two training steps from UDify, then adds a third: adapting to the target language using the target statistics and possibly small amounts of supervised data (Equation 4.3).

### 4.4.3 Typological Statistics as Supervision

We now discuss a series of statistics that we will use as weak supervision. Most of the proposed statistics describe various marginal probabilities for different (but related) grammatical substructures and can ultimately be broken down into ratios of “count” functions (sums of indicators), which tally various types of events in the data. We propose statistics that cover surface level (POS-only), single-arc, two-arc, and single-head substructures, as well as conditional variants. In preliminary experiments, we try 32 different specific instances of these more general families. The exact list of statistics is given in Table 4.2.

#### **Surface Level**

One simple set of descriptive statistics are the unigram and bigram distributions over POS tags. POS unigrams can capture some basic relative frequencies, such as our expectation that nouns and verbs are common to all languages. POS bigrams will allow us to capture simple word-order preferences.

#### **Single-Arc**

This next set of statistical families all capture information about various choices in single-arc substructures. A single arc substructure carries up to 5 pieces of information: the arc’s direction, label, and distance, as well as the tags for the head and child words. Various subsets of these capture differing forms of regularity, such as “the probability of seeing tag  $t_h$  head an arc with label  $r$  in direction  $d$ .”

#### **Universally Impossible Arcs**

In addition to many single-arc variants, we also consider the specific subset of (head tag, label, child tag) single-arc triples that are never seen in any UD data. These combinations, correspond to the impossible arrangements that do not “type-check” within the UD formalism and are interesting in that they could in principle be specified by a linguist without any labeled data whatsoever. As

such, they represent a particularly attractive use-case of our approach, where a domain expert could rule out all invalid substructures dictated from the task formalism without the model having to learn it implicitly from the training data. With complex structures, this can be a large proportion of the possibilities: in UD we can rule out 93.2% (9,966/10,693) of the combinations.

## Two-Arc

We also consider substructures spanning two connected arcs in the tree. They may be useful because they cover many important typological phenomena, such as subject-object-verb ordering. They also have been known to be strong features in higher-order parsing models, such as the parser of Carreras (2007), but are also known to be intractable in non-projective parsers (McDonald and Pereira, 2006).

Following McDonald and Pereira (2006), we distinguish between two different patterns of neighboring arcs: *siblings* and *grandchildren*. Sibling arc pairs consist of two arcs which share a single head word, while grandchild arc pairs share an intermediate word that is the child of one arc and the head of another.

## Head-Valency

One interesting statistic that does not fall into the other categories is the valency of a particular head tag. This corresponds to the count of outgoing arcs headed by some tag. We convert this into a probability by using a binning function that allows us to quantify the “probability that some tag heads between  $a$  and  $b$  children”. Like the two-arc statistics, expected valency statistics are intractable under the model and we must approximate their computation.

## Conditional Variants

Further, each of these statistics can be described in conditional terms, as opposed to their full joint realizations. To do this, we simply divide the joint counts by the counts of the conditioned-upon sub-events. Conditional variants may be useful because they do not express preferences for

probabilities of the sub-events on the right side of the conditioning bar, which may be hard to estimate.

### **A Note on Implementing Marginal Statistics Efficiently**

Efficiently computing the above statistics for all of their possible arguments for full-sized sentences in a modern deep learning setup requires careful implementation. To do so we express computation of the count-based statistics as multi-tensor contractions and use optimized einsum orderings to improve performance. Additionally, we found it necessary to cache common intermediate counts to minimize memory usage. By optimizing the implementation, we are able to simultaneously compute hundreds of thousands of statistics without memory issues and in reasonable time (under 1 second for all statistics in Table 4.2 at once on a batch size of 8 sentences on a Tesla T4 GPU with 16GB memory).

### **Average Entropy**

In addition to the above proposed relative frequency statistics, we also include average per-token, per-edge, and MST tree entropies as additional regularization statistics that are always used. Though we do not show it here, each of these functions may be formulated as a statistic within our approach. The inclusion of these statistics amounts to a form of Entropy Regularization (Grandvalet and Bengio, 2004a) that keep the models from optimizing the other ESR constraints with degenerate constant predictions (Mann and McCallum, 2010). In the next section, we go into detail as to why we include these statistics as regularizers.

#### **4.4.4 The Need for Entropy Regularization**

Optimizing the marginal statistic functions alone can be problematic because they admit a degenerate optimum wherein the model learns to satisfy the expected quantity by predicting it constantly, regardless of input. For example, if we set the expected proportion of NOUN POS tags to be 0.25, the model could satisfy this by predicting  $p(NOUN|x) = 0.25$  for every token in the

batch.

Mann and McCallum (2010) encounter the same issue and propose to remedy it by forcing the model to have lower entropy using a temperature parameter  $\tau > 0$ :  $p_\theta(y|x) \propto \exp\{\frac{1}{\tau}\phi(y, x)\}$ . This approach does not necessarily address the issue, however, since it is possible for the model to simply rescale the potentials  $\phi$  by a factor of  $\tau$  to achieve the same constant value.

Instead, as a more robust alternative, we propose to regularize the per-position entropy of the model to be below the entropy induced by the constant target values. This additional constraint is a data-driven instance of Entropy Regularization (ER) (Williams and Peng, 1991; Grandvalet and Bengio, 2004b). For the entropy constraints, we use a hard max-margin loss instead of the smoothed variant in Equation 4.9, because we only wish to keep it below the margin without pushing it all the way to zero.<sup>1</sup>

#### 4.4.5 The SST Relaxation for Optimizing Intractable Expected Statistics

To use SGD for optimizing Eq. 4.4, we require an approach to calculating the gradients of the loss  $C$  w.r.t. the model parameters  $\theta$ ,  $\nabla_\theta C$ . Generally we will seek “pathwise” gradients (Mohamed et al., 2020) that utilize backpropagation to compute  $\nabla_\theta C$ .<sup>2</sup>

This requires two things:

1. That  $\ell$  and the  $h_k$ ’s are all differentiable w.r.t. their inputs, which will be the case by design.
2. That  $\nabla_\theta \mathbb{E}_{p_\theta(y|x)}[g(x, y)]$  can be computed exactly in practice.<sup>3</sup>

Assuming  $p_\theta$  is an exponential family model, condition (2) will hold exactly in the cases where the structure of  $g$  obeys the structure of the model  $p_\theta$ , either additively or multiplicatively (Zmigrod

---

<sup>1</sup>In preliminary experiments, we found that the more forgiving max-margin loss reduced the numerical instabilities caused by the smooth-l1 driving the entropy to zero while still preventing constant predictions.

<sup>2</sup>This is in opposition to “score-function” gradient estimators that treat  $f$  as a blackbox and often exhibit high variance with single samples. Further, since our statistic functions are differentiable, we would like to use this extra information as opposed to treating them as blackboxes.

<sup>3</sup>Computing  $\nabla_\theta \mathbb{E}_{p_\theta(y|x)}[g(x, y)]$  relies on computing  $\mathbb{E}_{p_\theta(y|x)}[g(x, y)]$  exactly. The most obvious approximation, standard Monte Carlo sampling, breaks differentiability w.r.t  $\theta$  when  $y$  is discrete and cannot be applied directly.

et al., 2021).<sup>4</sup> However, many interesting functions do not factor in this way. For example, the two-arc and head-valency statistics considered in the experiments are intractable to compute exactly in the edge-factored graph model we use. In these cases, condition (2) is not satisfied.

For cases where  $\mathbb{E}_{p_\theta(y|x)}[g(x, y)]$  is intractable, we propose to apply the differentiable, relaxed Monte-Carlo (MC) approximation from Paulus et al. (2020). Called a ‘‘Stochastic Softmax Trick’’ (SST), we approximate  $\mathbb{E}_{p_\theta(y|x)}[g(x, y)]$  with a single relaxed sample  $\tilde{y}_{\tau, \gamma}$  and relaxed function  $\tilde{g}_\tau$ :

$$\mathbb{E}_{p_\theta(y|x)}[g(x, y)] \approx \tilde{g}_\tau(x, \tilde{y}_{\tau, \gamma}) \quad (4.15)$$

where  $\tilde{y}_{\tau, \gamma}$  is computed the same way as the distribution  $p_\theta$  but with potentials perturbed by additive Gumbel noise  $\gamma$  and modulated by a relaxation temperature  $\tau > 0$ . That is, if  $p_\theta(y|x) \propto \exp\{\sum_i \phi_i(\theta, x)\}$ , where  $\phi_i$  are its potential functions, then a relaxed sample  $\tilde{y}_{\tau, \gamma}$  is given by:

$$\tilde{y}_{\tau, \gamma} \propto \exp\left\{\frac{1}{\tau} \sum_i \phi_i(\theta, x) + \gamma_i\right\}, \quad \gamma_i \sim \mathcal{G} \quad (4.16)$$

where  $\mathcal{G}$  is the standard Gumbel distribution (Gumbel, 1954).<sup>5</sup> In this way  $\tilde{y}_{\tau, \gamma}$  is a relaxation of discrete samples  $y \in \mathcal{Y}$ , which lie on the vertices of the marginal polytope of  $\mathcal{Y}$ ,  $\mathcal{M}(\mathcal{Y})$ , to *soft* samples from the interior of  $\mathcal{M}(\mathcal{Y})$ . The temperature  $\tau$  controls the ‘‘softness’’ of the relaxation, converging to discrete binary samples from the vertices of  $\mathcal{M}(\mathcal{Y})$  as  $\tau \rightarrow 0$ .

Likewise,  $\tilde{g}_\tau$  is a relaxation of  $g$  that is modified to accept inputs from the marginal polytope  $\mathcal{M}(\mathcal{Y})$  instead of  $\mathcal{Y}$ . In many cases this modification is simple.<sup>6</sup> In some cases, however, if  $\tilde{g}_\tau$  is still not be differentiable w.r.t. its input  $\tilde{y}_{\tau, \gamma}$ , it must be further relaxed using the temperature  $\tau$ .

When  $\tau > 0$ , both  $\partial \tilde{g}_\tau / \partial \tilde{y}_{\tau, \gamma}$  and  $\partial \tilde{y}_{\tau, \gamma} / \partial \theta$  are well-defined and we can compute pathwise gradients for  $\nabla_\theta C$  using backpropagation and optimize with SGD.

<sup>4</sup>For multiplicative  $g$ ,  $\log g$  can be additively absorbed into the potentials. For additive  $g$ , the property follows from the linearity and independence of expectations:  $\mathbb{E}_y[\sum_i g_i(y_i)] = \sum_i \mathbb{E}_y[g_i(y_i)] = \sum_i \mathbb{E}_{y_i}[g_i(y_i)]$ .

<sup>5</sup>This is an extension of the Gumbel-Softmax trick (Jang et al., 2017; Maddison et al., 2017) for categorical distributions to combinatorial distributions. However, a further approximation is introduced by using a polynomial number of  $\gamma_i$ ’s (one per potential) instead of exponentially many (one per structure).

<sup>6</sup>Often the modification is converting a discrete index into a sum over a one-hot indicator vector. The indicator representation may then be relaxed to a soft probability vector.

## 4.5 Oracle Unsupervised Experiments

We begin with oracle unsupervised transfer experiments that evaluate the potential of many types of statistics and some ablations. In this setting, we do not assume any labeled data in the target language, but do assume accurate target statistics and margins, calculated from held-out training data using the method of Section 4.3. This allows us to study the potential of our proposed ESR regularization term  $C$  on its own and without the confounds of supervised data or inaccurate targets.

### 4.5.1 Experimental Setup

Next we describe setup details for the experiments. These settings additionally apply to the rest of the experiments unless otherwise stated.

#### Datasets

In all experiments, the models are first initialized from mBERT, then trained using the UDify code (Kondratyuk, 2019) on 13 diverse treebanks, following Kulmizev et al. (2019); Ustun et al. (2020). This model, further referred to as **UDPRE**, is used as the foundation for all approaches.

As discussed in Kulmizev et al. (2019), these 13 training treebanks were selected to give a diverse sample of languages, taking into account factors such as language families, scripts, morphological complexity, and annotation quality.

We evaluate all proposed methods on 5 held-out languages, similarly selected for a diversity in language typologies, but with the additional factor of transfer performance of the **UDPRE** baseline.<sup>7</sup>

A summary table of these training and evaluation treebanks is given in Table 4.1.

---

<sup>7</sup>While we would like to evaluate on as many UD treebanks as possible, budgetary constraints required that we restrict the number of test languages when experimenting with settings that combinatorially vary in other dimensions. We do however experiment with more languages in Section 4.6.2.

Language	Code	Treebank	Family	Train Sents
Arabic	ar	PADT	Semitic	6.1k
Basque	eu	BDT	Basque	5.4k
Chinese	zh	GSD	Sino-Tibetan	4.0k
English	en	EWT	IE, Germanic	12.5k
Finnish	fi	TDT	Uralic	12.2k
Hebrew	he	HTB	Semitic	5.2k
Hindi	hi	HDTB	IE, Indic	13.3k
Italian	it	ISDT	IE, Romance	13.1k
Japanese	ja	GSD	Japanese	7.1k
Korean	ko	GSD	Korean	4.4k
Russian	ru	SynTagRus	IE, Slavic	15.0k*
Swedish	sv	Talbanken	IE, Germanic	4.3k
Turkish	tr	IMST	Turkic	3.7k
German	de	HDT	IE, Germanic	153.0k
Indonesian	id	GSD	Austronesian	4.5k
Maltese	mt	MUDT	Semitic	1.1k
Persian	fa	PerDT	IE, Iranian	26.2k
Vietnamese	vi	VTB	Austro-Asiatic	1.4k

Table 4.1: Treebanks details for train (top) and test (bottom) sets in UD-13.

\*: the original Russian-SynTagRus dataset has 48.8k sentences, which we downsample to the same 15k sentences as Ustun et al. (2020) to reduce training time and balance the data.

## Approaches

We compare our approach to two strong baselines in all experiments, based on recent advances in the literature for cross-lingual parsing. These baselines are implemented in our code so that we may fairly compare them in all of our experiments.

- **UDPRE:** The first baseline is the UDify (Kondratyuk, 2019) model-transfer approach. Multilingual model-transfer alone is currently one of the state-of-the-art approaches to cross-lingual parsing and is a strong baseline in its own right.
- **UDPRE-PPT:** We also apply the Parsimonious Parser Transfer (PPT) approach from Kurniawan et al. (2021). PPT is a nuanced self-training approach, extending Täckström et al. (2013), that encourages the model to concentrate its mass on its most likely predicted parses for the target treebank. We use their loss implementation, but apply it to our UDPRE base

model (instead of their weaker base model) for a fair comparison, so this approach combines UDify with PPT.

- **UDPRE-ESR:** Our proposed approach, Expected Statistic Regularization (ESR), applied to UDPRE as an unsupervised-only objective. In individual experiments we will specify the statistics used for regularization.

## Training and Evaluation Details

For metrics, we report accuracy for POS tagging, coarse-grained labeled attachment score (LAS) for dependency trees, and their average as a single summary score. The metrics are computed using the official CoNLL-18 evaluation script.<sup>8</sup> For all scenarios, we use early-stopping for model selection, measuring the POS-LAS average on the specified development sets.

We tune learning rates and  $\alpha$  for each proposed loss variant at the beginning of the first experiment with a low-budget grid search, using the settings that achieve best validation metric on average across the 5 language validation sets for all remaining experiments with that variant. We find generally that a base learning rate of  $2 \times 10^{-5}$  and  $\alpha = 0.01$  worked well for all variants of our method. We train all models using AdamW (Loshchilov and Hutter, 2019) on a slanted triangular learning rate schedule (Devlin et al., 2019) with 500 warmup steps. Also, since the datasets vary in size, we normalize the training schedule to 25 epochs at 1000 steps per epoch. We use a batch size of 8 sentences for training and estimating statistic targets. When bootstrapping estimates for  $t$  and  $\sigma$  we use  $B = 1000$  samples.

### 4.5.2 Assessing the Proposed Statistics

In this experiment we evaluate 32 types of statistics from Section 4.4.3 for transfer of the UDPRE model (pretrained on 13 languages) to the target languages. The purpose of this experiment is to get a sense of the effectiveness of each statistic for improving model-based cross-lingual

---

<sup>8</sup><https://universaldependencies.org/conll18/evaluation.html>

Statistic	POS					LAS					avg
	de	id	fa	vi	mt	de	id	fa	vi	mt	
UDPRE	89.3	80.3	83.0	64.7	41.4	82.7	50.4	57.0	48.1	20.9	61.8
UDPRE-PPT	+0.4	+5.6	-1.5	-0.1	+3.1	+0.2	+8.1	-5.5	-0.3	+4.6	+1.5
<b>Child, Label</b>	+3.3	+5.7	+8.0	+4.0	+14.0	+3.5	+10.1	+18.4	+0.5	+10.2	+7.8
*Child, Label, Grand-label	+1.5	+2.8	+5.3	+5.9	+15.7	+2.5	+9.2	+16.2	+3.2	+12.5	+7.5
Head, Child, Label	+2.5	+4.9	+7.2	+4.1	+14.2	+2.8	+9.9	+17.2	+1.3	+10.2	+7.4
Head, Label	+1.7	+3.2	+5.2	+6.3	+14.2	+2.7	+9.0	+16.4	+3.8	+11.0	+7.3
Head, Label   Child	+3.0	+5.2	+5.8	+5.7	+10.9	+2.8	+9.7	+15.3	+2.1	+5.6	+6.6
Label	-0.2	+4.4	+5.1	+0.0	+11.3	+2.9	+8.4	+17.1	+4.0	+9.5	+6.2
Label, Distance	-0.2	+3.7	+3.9	-0.1	+11.7	+2.8	+8.9	+16.3	+4.1	+9.4	+6.0
*Head, Sibling Child Tags	+2.2	+5.1	+5.6	+7.8	+14.0	+1.6	+2.7	+11.3	-3.0	+11.1	+5.8
Head, Child	+2.3	+5.3	+5.9	+3.7	+14.0	+1.9	+3.3	+12.2	-0.7	+10.1	+5.8
Label   Child	+1.2	+4.3	+4.8	-0.5	+10.0	+3.3	+9.5	+16.6	+2.0	+5.6	+5.7
*Head, Sibling Arc Labels	+1.3	+1.8	+4.8	+0.0	+13.0	+1.0	+7.6	+14.5	+3.1	+8.8	+5.6
†Tag   Previous Tag	+2.4	+4.8	+6.3	+3.2	+14.6	+0.8	+1.2	+13.5	-2.2	+10.9	+5.5
Child   Label	+2.4	+3.7	+6.8	+7.7	+10.1	+2.7	+3.4	+14.2	-1.9	+4.5	+5.4
Head, Child   Label	+1.8	+3.3	+7.5	+7.6	+10.3	+2.6	+3.6	+15.0	-1.9	+3.8	+5.4
*Head, Child, Grandchild	+1.8	+4.6	+5.5	+4.0	+14.4	+1.8	+2.1	+10.5	-0.7	+9.3	+5.3
†Tag, Previous Tag	+2.5	+4.8	+6.0	+3.7	+14.9	+0.4	+1.4	+10.8	-2.5	+10.5	+5.3
Child	+2.5	+4.9	+5.8	+3.6	+13.0	+0.4	+2.0	+10.6	-0.7	+8.7	+5.1
Head   Child	+1.3	+5.3	+4.4	+3.6	+11.8	+0.8	+3.9	+11.7	-0.1	+6.6	+4.9
*†Head Valency	+2.2	+4.3	+5.7	+3.4	+13.0	+0.1	+1.9	+11.6	-0.9	+7.8	+4.9
Head	+1.6	+3.1	+4.6	+4.1	+11.4	-0.2	+2.8	+13.5	-1.4	+6.1	+4.6
Distance   Label	-0.4	-0.1	+4.7	+0.6	+9.2	+1.7	+4.7	+14.8	+0.9	+6.8	+4.3
Head   Label	+1.0	+1.6	+4.3	+2.5	+9.9	+0.9	+4.9	+14.0	-0.4	+4.1	+4.3
†Tag	+2.4	+3.8	+4.9	+3.5	+12.0	+0.1	+2.0	+7.5	-1.7	+6.9	+4.1
*Head, Grandchild   Child	+0.8	+1.4	+1.2	+4.1	+12.1	+0.8	+1.7	+6.1	-0.8	+8.5	+3.6
† <b>Universal Arc</b>	+2.4	+4.7	+1.4	+1.4	+8.7	+2.1	+8.1	+4.0	-3.1	+3.9	+3.4
Child, Label   Head	+0.2	+5.2	+0.1	+1.2	+13.2	+0.1	+5.1	+2.1	-1.7	+8.5	+3.4
Child   Head	+0.1	+3.6	+2.6	+3.1	+12.2	+0.2	+2.9	+1.5	-1.7	+7.7	+3.2
*Sibling Labels   Head	+0.4	+0.4	+3.3	-0.2	+9.1	+0.3	+1.3	+8.6	+0.5	+3.3	+2.7
*Label, Grand-label   Child	+0.5	-0.8	-3.2	+1.1	+8.8	-0.1	+2.6	+7.2	+0.7	+7.2	+2.4
Label   Head	-0.1	-0.4	-1.6	-1.0	+9.9	+0.4	+4.3	+2.8	+1.0	+7.8	+2.3
*Sibling Children   Head	+0.0	+1.6	+0.2	-0.9	+10.2	+0.0	+1.6	+1.2	-0.5	+6.7	+2.0
*†Valency   Head	+1.1	+0.8	-2.4	+5.6	+10.3	-0.7	+1.1	-0.2	-2.2	+5.6	+1.9

Table 4.2: *Unsupervised Constraint Variant Results.* (Top): Baseline methods that do not use ESR. (Bottom): Various statistics used by ESR as unsupervised loss on top of UDPRE. Scores are measured on target treebank development (not test) sets. Bold rows mark statistics used in later experiments. (\*): All statistics with \* are intractable and utilize the SST relaxation of Paulus et al. (2020). (†): All statistics *except* those with † also include left/right directional information – those with a † do not have directional information.

transfer.<sup>9</sup> To prevent overfitting to the test sets for later experiments, all metrics for this experiment are calculated on the development sets.

<sup>9</sup>While it would also be possible to try out different combinations of the various statistics, due to cost considerations we leave these experiments to future work.

**Results:** The results of the experiment are presented in Table 4.2, ranked from best to worst. Generally we find that all of the 32 proposed statistics improve upon the UDPRE and UDPRE-PPT models on average, with many exhibiting large boosts. The best performing statistic concerns (Child Tag, Label, Direction) substructures, yielding an average improvement of +7.0 POS and +8.5 LAS, an average relative error rate reduction of 23.5%. Many other statistics are not far behind, and overall statistics that bear on the child tag and dependency label had the highest impact. This indicates that, with accurate target estimates, the proposed statistics are highly complementary to multilingual parser pretraining (UDPRE) and substantially improve transfer quality in the unsupervised setting. By comparison, the PPT approach provides marginal gains to UDPRE of only +1.4 average POS and +1.5 average LAS.

Another interesting result is that several of the intractable two-arc statistics were among the best statistics overall, indicating that the use of the differentiable SST approximation does not preclude the applicability of intractable statistics. For example the directed grandchild statistic of cooccurrences of incoming and outgoing edges for certain tags was the second highest performing, with an average improvement of +7.0 POS accuracy and +8.5 LAS (21.3% average error rate reduction).

Results for the conditional variants were less positive. Generally, conditional variants were worse than their full joint counterparts (e.g., "Child | Label" and "Label | Child" are worse than "Child, Label"), performing worse in 15/16 cases. This makes sense, as we are using accurate statistics and full joints are strictly more expressive.

This experiment gives a broad but shallow view into the effectiveness of the various proposed statistics. In the rest of the experiments, we evaluate the following two variants in more depth:

1. **ESR-CLD**, which supervises target proportions for (Child Tag, Label, Direction) triples. This is the "Child, Label" row in Table 4.2.
2. **ESR-UNIARC**, which supervises the 9,966 universally impossible (Head Tag, Child Tag, Label) arcs that do not require labeled data to estimate. All of these combinations have targets values of  $t = 0$  and margins  $\sigma = 0$ . This is the "Universal Arc" row in Table 4.2.

We choose these two because ESR-CLD is the best performing statistic overall and ESR-UNIARC is unique in that it does not require labeled data to estimate; we do not evaluate others because of cost considerations.

### 4.5.3 Ablation Studies

Next, we perform two ablation experiments to evaluate key design choices of the proposed approach. First, we evaluate the use of batch-level aggregation in the statistics before the loss, versus the more standard approach of loss-per-sentence. In the second, we evaluate the proposed form of  $\ell$ .

We compare the two aggregation variants using the CLD (Child Tag, Label, Direction) statistic (ESR-CLD). We report test set results averaged over all 5 languages. We use the same hyperparameters selected in Section 4.5.2.

#### Batch-level Loss Ablation

In this ablation, we evaluate a key feature of our proposal—the aggregation of the statistic over the batch before loss computation Equation 4.5 versus the more standard approach, which is to apply the loss per-sentence. The former, “Loss per batch”, has the form:  $\ell(t, \sigma, f(\mathcal{D}_U, p_\theta))$  while the latter, “Loss per sentence”, has the form:  $\sum_{x \in \mathcal{D}_U} \ell(t, \sigma, f(x, p_\theta))$ .

The significance of this difference is that “Loss per batch” allows for the variation in individual sentences to somewhat average out and hence is less noisy, while “Loss per sentence” requires that each sentence individually satisfy the targets.

**Results:** The results are presented in Table 4.3. From the table we can see that “Loss per batch” has an average POS of 79.9 and average LAS of 60.4, compared to “Loss per sentence” with average POS of 77.1 and LAS of 58.5, which amount to +2.8 POS and +1.9 LAS improvements. This indicates that applying the loss at the batch level confers an advantage over applying per sentence.

Aggregation Variant	POS avg	LAS avg	avg
Loss per sentence	77.1	58.5	67.8
Loss per batch (ESR)	<b>79.9</b>	<b>60.4</b>	<b>70.1</b>

Table 4.3: *Loss Aggregation Ablation Results.* Loss per batch outperforms loss per sentence for both POS and LAS on average.

$\ell$ Variant	POS avg	LAS avg	avg
L2 ( $\sigma = 0$ )	78.0	58.2	68.1
L1 ( $\sigma = 0$ )	78.5	60.3	69.5
Hard L1 (max-margin)	78.4	59.9	69.2
Smooth L1 (ESR)	<b>79.9</b>	<b>60.4</b>	<b>70.1</b>

Table 4.4: *Loss Function Ablation Results.* The Smooth L1 loss outperforms the other simpler loss variants for both POS and LAS, averaged over 5 languages.

### Smooth Hinge-Loss Ablation

Next, we evaluate the efficacy of the proposed smoothed hinge-loss distance function  $\ell$ . We compare to using just L1 or L2 uninterpolated and with no margin parameters ( $\sigma = 0$ ). We also compare to the ‘‘Hard L1’’, which is the max-margin hinge  $\ell(t, \sigma, x) = \max\{0, |t - x| - \sigma\}$ . We use the same experimental setup as the previous ablation.

**Results:** The results are presented in Table 4.4. From the table we can see that the Smooth L1 loss outperforms the other variants.

## 4.6 Realistic Semi-Supervised Experiments

The previous experiments considered an unsupervised transfer scenario without labeled data. In these next experiments we turn to a realistic semi-supervised application of our approach where we have access to limited labeled data for the target treebank.

### 4.6.1 Learning Curves

In this experiment we present learning curves for the approaches, varying the amount of labeled data  $|\mathcal{D}_L^{\text{train}}| \in \{50, 100, 500, 1000\}$ . To make experiments realistic, we calculate the target statistics  $t$  and margins  $\sigma$  from the small subsampled labeled training datasets using Equations 4.11 and

4.12.

We study two distinct settings. First, we study the multi-source domain-adaptation transfer setting, UDPRE. Second, we study our approach in a more standard semi-supervised scenario where we cannot utilize intermediate on-task pretraining and domain-adaption, instead learning on the target dataset starting “from scratch” with the pretrained PLM (MBERT).

We use the same baselines as before, but augment each with a supervised fine-tuning loss on the supervised data in addition to any unsupervised losses. We refer to these models as **UDPRE-FT**, **UDPRE-FT-PPT**, and **UDPRE-FT-ESR**. That is, models with **FT** in the name have some supervised fine-tuning in the target language.

In these experiments, we subsample labeled training data 3 times for each setting. We report averages over all 5 languages, 3 supervised subsample runs each, for a total of 15 runs per method and dataset size. We also use subsampled development sets so that model selection is more realistic.<sup>10</sup> For development sets we subsample the data to a size of  $|\mathcal{D}_L^{\text{dev}}| = \min(100, |\mathcal{D}_L^{\text{train}}|)$ , which reflects a 50/50 train/dev split until  $|\mathcal{D}_L| \geq 200$ , at which point we maximize training data and only hold out 100 sentences for validation.

We use the same hyperparameters as before, except we use 40 epochs with 200 steps per epoch as the training schedule, mixing supervised and unsupervised data at a rate of 1:4.

## UDPRE Transfer

In this experiment, we evaluate in the multilingual transfer scenario by initializing from UDPRE. In addition to the two chosen realistic ESR variants, we also experiment with an “oracle” version of ESR-CLD, called ESR-CLD\*, that uses target statistics estimated from the full training data. This allows us to see if small-sample estimates cause a degradation in performance compared to accurate large-sample estimates.

**Results:** Learning curves for the different approaches, averaged over all 3 runs for all 5 languages (15 total), are given in Figure 4.2. Detailed results are given in Table B.1 in the Appendix. From

---

<sup>10</sup>As is argued by Oliver et al. (2018), using a realistically-sized development set is overlooked in much of the semi-supervised literature, leading to inappropriately strong model selection and overly optimistic results.

the figure we can discern several encouraging results.

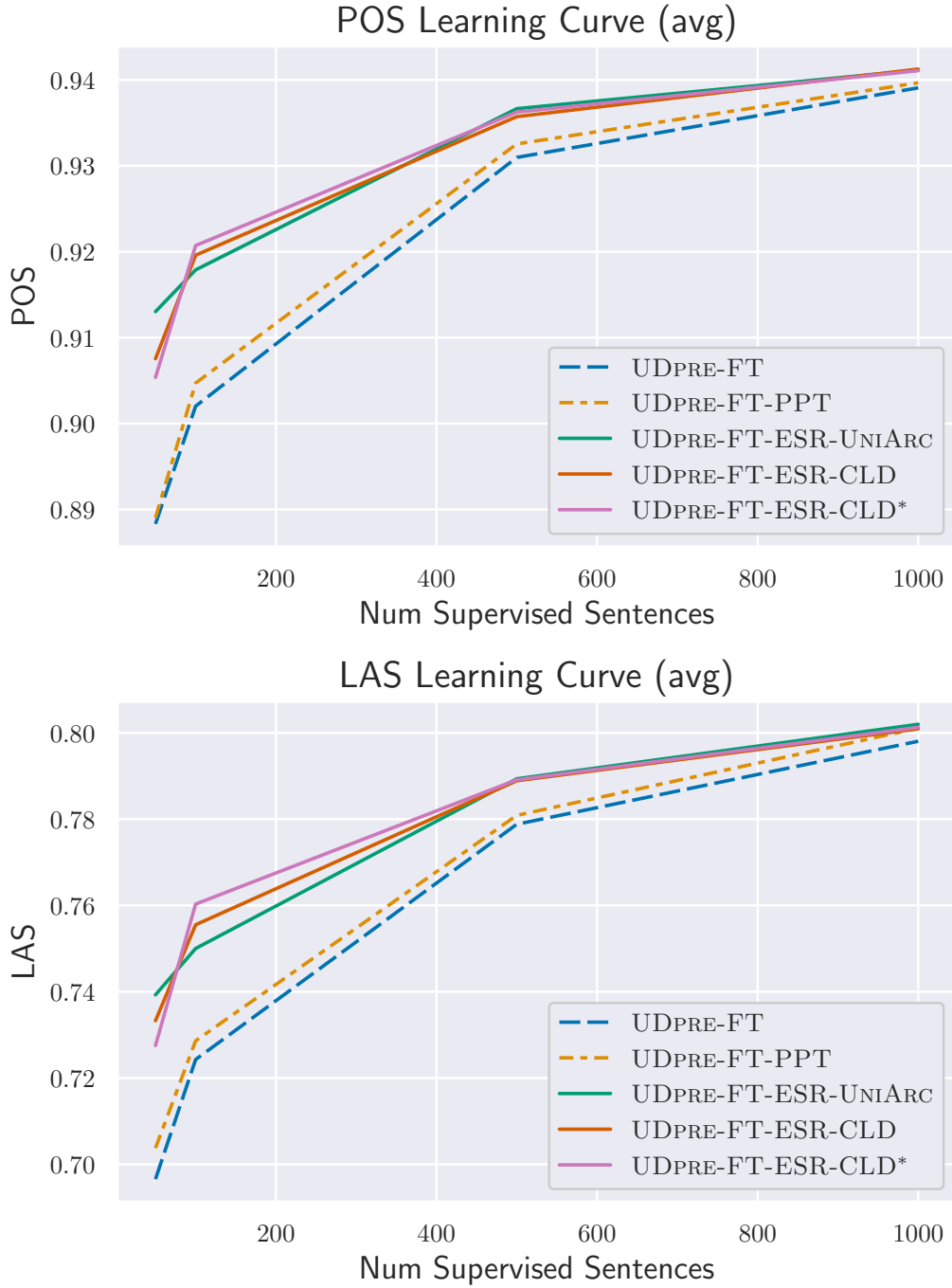


Figure 4.2: *Multi-Source UDPRE Transfer Learning Curves*. Baseline approaches are dotted, while ESR variants are solid. All curves show the average of 15 runs across 5 different languages with 3 randomly sampled labeled datasets per language. The plots indicate a significant advantage of ESR over the baselines in low-data regions.

**ESR-CLD and ESR-UNIARC add significant benefit to fine-tuning for small data.** Both variants significantly outperform the baselines at 50 and 100 labeled examples. For example, relative to UDPRE-FT, the ESR-CLD model yielded gains of +2 POS, +3.6 LAS at 50 examples and +1.8 POS, +3.2 LAS at 100 labeled examples. At 500 and 1000 examples, however, we begin to see diminishing benefits to ESR on top of fine-tuning.

**ESR-UNIARC is much more effective in conjunction with fine-tuning.** Compared to the unsupervised experiment in Section 4.5.2 where it ranked 25/32, the ESR-UNIARC statistic is much more competitive with the more detailed ESR-CLD statistics. One potential explanation is that without labeled data (as in Section 4.5.2) the ESR-UNIARC statistic is under-specified (the 727 allowed arcs are all free to take any value), whereas the inclusion of some labeled data in this experiment fills this gap by implicitly indicating target proportions for the allowed arcs. This suggests that an approach which combines UniArc constraints with elements of self-training (like PPT) that supervise the “free” non-zero combinations could potentially be a useful approach to zero-shot transfer. However, we leave this to future work.

**Small-data estimates for ESR-CLD are as good as accurate estimates.** Comparing ESR-CLD to the unrealistic ESR-CLD\*, we find no significant difference between the two, indicating that, at least for the CLD statistic, using target estimates from small samples is as good as large-sample estimates. This may be due in part to the margin estimates  $\sigma$ , which are wider for the small samples and somewhat mitigate their inaccuracies.

**PPT adds little benefit to fine-tuning.** Relative to UDPRE-FT, the UDPRE-FT-PPT baseline does not yield much gain, with a maximum average improvement of +0.3 POS and +0.7 LAS over all dataset sizes. This indicates that fine-tuning and PPT-style self-training may be redundant.

## **MBERT Transfer**

In this experiment, we consider a counterfactual setting: what if the UD data was not a massively multilingual dataset where we can utilize multilingual model-transfer, and instead was an isolated dataset with no related data to transfer from? This situation reflects the more standard

semi-supervised learning setting, where we are given a new task, some labeled and unlabeled data, and must build a model “from scratch” on that data.

For this experiment, we repeat the learning curve setting from Section 4.6.1, but initialize our model directly with **MBERT**, skipping the intermediate **UDPRE** training.

**Results:** Learning curves for the different approaches, averaged over all 3 runs for all 5 languages (15 total), are given in Figure 4.3 with detailed results in Table B.2 in the Appendix. From the figure we can discern several encouraging results.

**ESR has even greater benefits when fine-tuning directly from MBERT.** Similar to Section 4.6.1, we find that both ESR approaches significantly outperform the baselines on average. Moreover, without UDPRE transfer, the effect is more pronounced and is evident at all amounts of labeled data. In particular, relative to the standard baseline of MBERT-FT, the MBERT-FT-ESR-CLD model achieved the following average improvements: +3.3 POS, +8.7 LAS at 50 examples; +3 POS, +7.7 LAS at 100 examples; +1.2 POS, +3.2 LAS at 500 examples; and +0.4 POS, +1.1 LAS at 1000 examples. This result lends evidence that our general proposed approach from Section 4.2 may be applicable to more standard semi-supervised structured prediction problems.

**PPT hurts fine-tuning from MBERT.** When fine-tuning from MBERT, we find that the self-training PPT approach is detrimental at all amounts of labeled data.

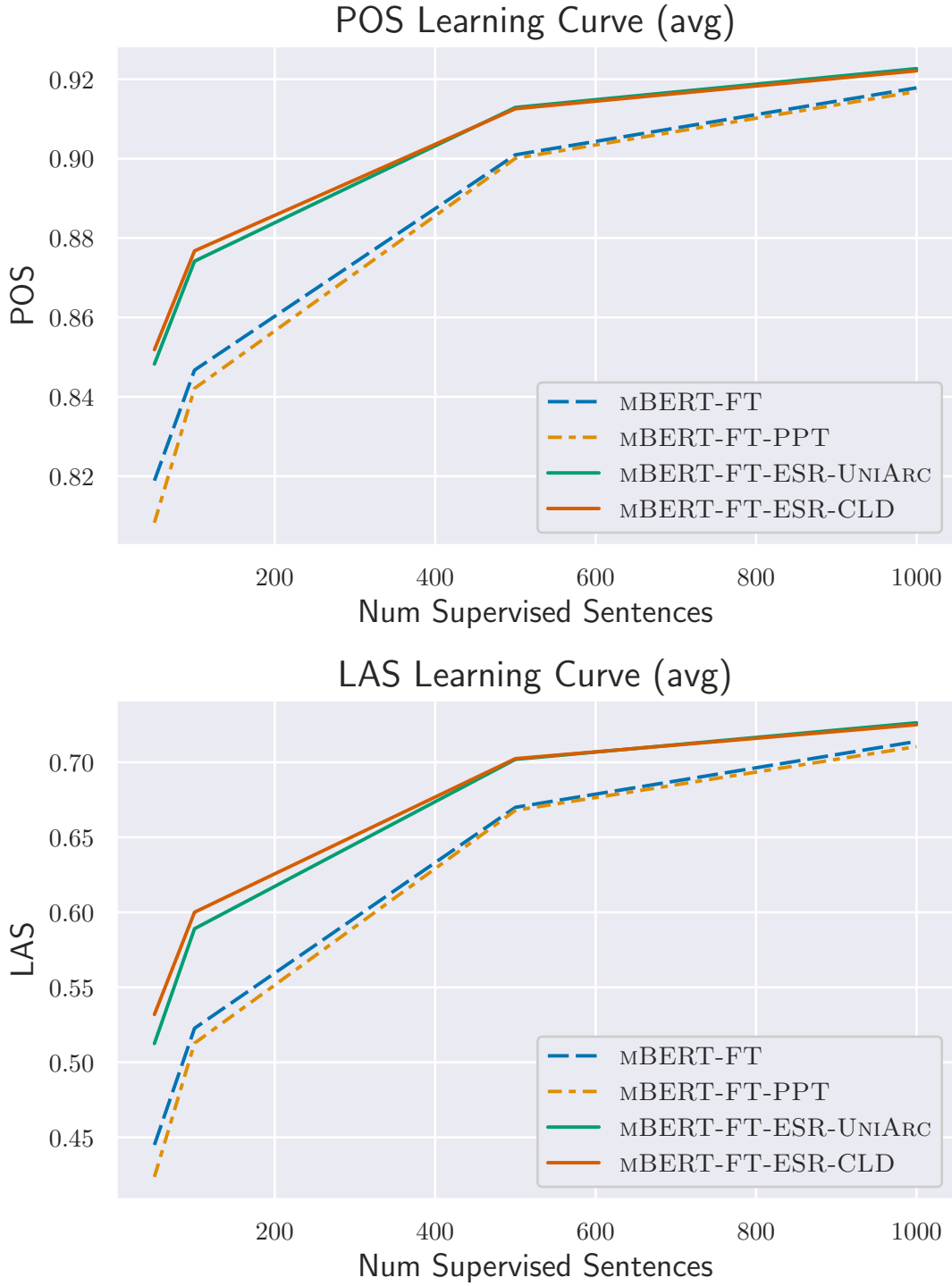


Figure 4.3: “From Scratch” MBERT *Transfer Learning Curves*. Baseline approaches are dotted, while ESR variants are solid. All curves show the average of 15 runs across 5 different languages with 3 randomly sampled labeled datasets per language. The plots indicate a significant advantage of ESR over the baselines in low-data regions.

#### 4.6.2 Low-Resource Transfer

In previous experiments, we limited the number of evaluation treebanks to 5 to allow for variation in other dimensions (i.e., constraint types, loss types, differing amounts of labeled data). In this experiment, we expand the number of treebanks and evaluate transfer performance in a low-resource setting with only  $|\mathcal{D}_L^{\text{train}}| = 50$  labeled sentences in the target treebank, comparing UDPRE, UDPRE-FT, and UDPRE-FT-ESR-CLD. As before, we subsample 3 small datasets per treebank and calculate the target statistics  $t$  and margins  $\sigma$  from these to make transfer results realistic. We select evaluation treebanks according to the following criteria. For each unique language in UD v2.8 that is not one of the 13 training languages, we select the largest treebank, and keep it if has at least 250 train sentences and a development set, so that we can get reasonable variability in the subsamples. This process yields 44 diverse evaluation treebanks.

**Results:** The results of this experiment are given in Table 4.5. From the table we can see that our approach ESR (UDPRE-FT-ESR-CLD) outperformed supervised fine-tuning (UDPRE-FT) in many cases, often by a large margin. On average, UDPRE-FT-ESR-CLD outperformed UDPRE-FT by +2.6 POS and +2.3 LAS across the 44 languages. Further, UDPRE-FT-ESR-CLD outperformed zero shot transfer, UDPRE, by +10.0 POS and +14.7 LAS on average.

Interestingly, we found that there were several cases of large performance gains while there were no cases of large performance declines. For example, ESR improved LAS scores by +17.3 for Wolof, +16.8 for Maltese, and +12.5 for Scottish Gaelic, and 9/44 languages saw LAS improvements  $\geq +5.0$ , while the largest decline was only  $-2.5$ . Additionally, ESR improved POS scores by +20.9 for Naija, +11.2 for Welsh, and 9/44 languages saw POS improvements  $\geq +5.0$ .

The cases of performance decline for LAS merit further analysis. Of the 20 languages with negative  $\Delta$  LAS, 18 of these are modern languages spoken in continental Europe (mostly Slavic and Romance), while only 5 of the 24 languages with positive  $\Delta$  LAS meet this criteria. We hypothesize that this tendency is due to the training data used for pretraining MBERT, which was heavily skewed towards this category (Devlin et al., 2019). This suggests that ESR is particularly helpful in cases of transfer to domains that are underrepresented in pretraining.

Treebank	Family	POS				LAS			
		UDPRE	FT	ESR	$\Delta$	UDPRE	FT	ESR	$\Delta$
Wolof-WTB	Northern Atlantic	40.6	79.5	<b>85.4</b>	+5.9	12.7	55.9	<b>73.3</b>	+17.3
Maltese-MUDT	Semitic	35.1	82.6	<b>91.8</b>	+9.2	16.0	57.5	<b>74.2</b>	+16.8
Scottish_Gaelic-ARCOSG	Celtic	45.7	66.0	<b>75.9</b>	+9.9	24.4	56.4	<b>68.9</b>	+12.5
Faroese-FarPaHC	Germanic	74.7	86.2	<b>87.2</b>	+1.1	43.0	71.4	<b>80.7</b>	+9.3
Gothic-PROIEL	Germanic	30.1	67.6	<b>71.7</b>	+4.1	12.6	45.8	<b>54.6</b>	+8.8
Welsh-CCG	Celtic	71.9	74.7	<b>85.8</b>	+11.2	54.8	69.4	<b>77.6</b>	+8.1
Western_Armenian-ArmTDP	Armenian	80.6	84.9	<b>87.1</b>	+2.2	60.4	67.0	<b>72.7</b>	+5.7
Telugu-MTG	Dravidian	82.0	<b>81.6</b>	<b>81.6</b>	0.0	70.9	74.6	<b>80.1</b>	+5.5
Vietnamese-VTB	Viet-Muong	67.0	85.6	<b>88.5</b>	+2.9	46.3	55.3	<b>60.8</b>	+5.5
Turkish_German-SAGT	Code Switch	76.8	84.4	<b>85.8</b>	+1.4	48.0	58.0	<b>62.1</b>	+4.1
Afrikaans-AfriBooms	Germanic	90.7	88.0	<b>91.3</b>	+3.3	62.0	79.4	<b>83.4</b>	+3.9
Hungarian-Szeged	Ugric	87.9	79.9	<b>89.7</b>	+9.7	74.0	77.8	<b>81.7</b>	+3.9
Galician-CTG	Romance	91.8	89.0	<b>91.2</b>	+2.2	60.5	74.3	<b>77.8</b>	+3.6
Marathi-UFAL	Marathi	71.4	81.1	<b>82.3</b>	+1.1	44.9	59.5	<b>62.5</b>	+3.0
Naija-NSC	Creole	46.5	68.0	<b>88.9</b>	+20.9	27.9	71.1	<b>73.4</b>	+2.3
Greek-GDT	Greek	87.1	<b>92.8</b>	92.5	-0.3	78.7	86.3	<b>88.0</b>	+1.8
Tamil-TTB	Dravidian	72.3	72.4	<b>79.6</b>	+7.2	46.7	64.9	<b>66.4</b>	+1.5
Indonesian-GSD	Austronesian	82.3	89.8	<b>90.2</b>	+0.5	58.3	72.9	<b>74.3</b>	+1.4
Uyghur-UDT	Turkic	23.7	59.8	<b>65.5</b>	+5.6	14.0	38.0	<b>39.2</b>	+1.3
Old_French-SRCMF	Romance	65.3	74.2	<b>76.2</b>	+2.0	44.0	56.7	<b>57.8</b>	+1.2
Old_Church_Slavonic-PROIEL	Slavic	37.3	54.7	<b>61.0</b>	+6.3	19.2	39.0	<b>40.1</b>	+1.1
Portuguese-GSD	Romance	92.1	89.6	<b>92.8</b>	+3.3	74.4	84.1	<b>84.5</b>	+0.4
Danish-DDT	Germanic	92.0	<b>92.7</b>	92.1	-0.6	71.0	75.5	<b>75.7</b>	+0.2
Armenian-ArmTDP	Armenian	84.7	<b>88.1</b>	88.0	-0.1	64.1	69.0	<b>69.2</b>	+0.1
Spanish-AnCora	Romance	94.5	95.2	<b>95.4</b>	+0.2	77.8	<b>83.0</b>	82.9	-0.1
Catalan-AnCora	Romance	92.9	94.4	<b>94.6</b>	+0.3	75.8	<b>82.5</b>	82.4	-0.1
Serbian-SET	Slavic	91.2	90.7	<b>93.1</b>	+2.4	81.6	<b>86.5</b>	86.4	-0.1
Slovak-SNK	Slavic	91.5	91.5	<b>92.0</b>	+0.5	81.6	<b>84.0</b>	83.9	-0.1
Romanian-Nonstandard	Romance	79.2	83.3	<b>85.0</b>	+1.7	54.5	<b>63.6</b>	63.4	-0.2
Polish-PDB	Slavic	89.7	90.4	<b>90.9</b>	+0.5	76.0	<b>79.7</b>	79.4	-0.3
German-HDT	Germanic	89.6	<b>94.4</b>	94.2	-0.2	83.0	<b>88.2</b>	87.7	-0.5
Lithuanian-ALKSNIS	Baltic	87.0	<b>87.4</b>	<b>87.4</b>	0.0	65.4	<b>69.2</b>	68.6	-0.6
Latin-ITTB	Italic	73.8	80.9	<b>81.7</b>	+0.8	51.7	<b>64.3</b>	63.7	-0.6
Bulgarian-BTB	Slavic	91.9	<b>94.7</b>	94.6	-0.1	78.0	<b>84.4</b>	83.7	-0.7
Czech-PDT	Slavic	90.6	92.1	<b>92.7</b>	+0.6	78.1	<b>81.9</b>	81.1	-0.8
Persian-PerDT	Iranian	79.1	<b>91.0</b>	90.8	-0.2	48.4	<b>74.6</b>	73.7	-0.9
Slovenian-SSJ	Slavic	89.2	90.9	<b>91.2</b>	+0.3	79.6	<b>84.5</b>	83.5	-0.9
Croatian-SET	Slavic	91.4	91.7	<b>92.1</b>	+0.4	80.0	<b>84.1</b>	83.1	-1.0
Urdu-UDTB	Indic	86.9	<b>90.0</b>	88.2	-1.8	68.7	<b>75.7</b>	74.4	-1.3
Ukrainian-IU	Slavic	91.5	92.0	<b>92.4</b>	+0.3	79.6	<b>81.2</b>	80.0	-1.3
Dutch-Alpino	Germanic	90.0	<b>90.6</b>	<b>90.6</b>	0.0	78.9	<b>81.6</b>	80.3	-1.3
Norwegian-Bokmaal	Germanic	91.7	91.8	<b>92.1</b>	+0.3	80.8	<b>82.5</b>	81.0	-1.5
Belarusian-HSE	Slavic	91.5	91.6	<b>91.9</b>	+0.3	78.9	<b>79.8</b>	78.1	-1.8
Estonian-EDT	Finnic	89.1	<b>89.6</b>	89.2	-0.4	70.4	<b>71.4</b>	68.9	-2.5
Average		77.3	84.7	<b>87.3</b>	+2.6	59.0	71.4	<b>73.7</b>	+2.3

Table 4.5: *Low-Resource Semi-Supervised Transfer Results*. “FT” refers to the UDPRE-FT fine-tuning baseline, “ESR” refers to our UDPRE-ESR-CLD approach, and  $\Delta$  refers to the absolute difference of ESR minus FT. Best performing methods are in bold.

## 4.7 Related Work

Related work generally falls into two categories: weak supervision and cross-lingual transfer.

### 4.7.1 Weak Supervision

Supervising models with signals weaker than fully labeled data has and continues to be a popular topic of interest. Current trends in weak supervision focus on generating instance-level supervision, using weak information such as: relations between multiple tasks (Greenberg et al., 2018b; Ratner et al., 2018; Noach and Goldberg, 2019); labeled features (Druck et al., 2008; Ratner et al., 2016b; Karamanolakis et al., 2019a); coarse-grained labels (Angelidis and Lapata, 2018; Karamanolakis et al., 2019b); dictionaries and distant supervision (Bellare and McCallum, 2007; Carlson et al., 2009; Liu et al., 2019b; Ustun et al., 2020); or some combination of thereof (Ratner et al., 2016b; Karamanolakis et al., 2019a).

In contrast, our work is more closely related to older work on population-level supervision. These techniques include Constraint-Driven Learning (CODL) (Chang et al., 2007b), posterior regularization (PR) (Ganchev et al., 2010b), the measurements framework of Liang et al. (2009), and the generalized expectation criteria (GEC) (Mann and McCallum, 2007; Druck et al., 2008, 2009; Mann and McCallum, 2010).

Our work can be seen as an extension of GEC to more expressive combinations of expectations and to modern mini-batch SGD training. There are a couple of more recent works that utilize these ideas, but both have significant downsides compared to our approach. Meng et al. (2019) use a PR approach inspired by Ganchev and Das (2013) for cross-lingual parsing, but must use very simple constraints and require a slow inference procedure at test time – the model parameters cannot be trained with this loss. Noach and Goldberg (2019) utilize GEC with minibatch training, but focus on using related tasks for computing simple constraints and do not adapt their targets to small batch sizes in a principled way.

#### 4.7.2 Cross-Lingual Transfer

Earlier trends in cross-lingual transfer for parsing were based on dexicalized parsers (Zeman and Resnik, 2008; McDonald et al., 2011; Täckström et al., 2013) and then aligned multilingual word vector-based approaches (Guo et al., 2015; Ammar et al., 2016; Rasooli and Collins, 2017; Ahmad et al., 2019). With the rapid rise of transformers and language-model pretraining (Peters et al., 2018a; Devlin et al., 2019; Liu et al., 2019c), recent research in cross-lingual parsing has focused on multilingual pretraining and multitask fine-tuning to achieve generalization in transfer. Wu and Dredze (2019) demonstrated that a multilingual pretrained language model (PLM) afforded surprisingly effective cross-lingual transfer using only English as the fine-tuning language. Kondratyuk (2019) extended this approach by fine-tuning a PLM on the concatenation of all treebanks. Tran and Bisazza (2019), however, demonstrate that transfer to languages that are distant or poorly-represented in either pretraining or fine-tuning benefit less.

Other recent successes have been found by using linguistic side-information (Meng et al., 2019; Ustun et al., 2020), careful methodology for source-treebank selection (Tiedemann and Agic, 2016; Tran and Bisazza, 2019; Lin et al., 2019; Glavas and Vulic, 2021), self-training (Kurniawan et al., 2021), and paired bilingual text for annotation projection (Rasooli and Tetreault, 2015; Rasooli and Collins, 2019; Liu et al., 2020; Shi et al., 2021). Our approach can most closely be associated with Meng et al. (2019) in that we use structural side information, but also is significantly different in that we estimate this information ourselves at a fine-grained level and our approach is flexibly handles many more cross-task constraints and incorporates these during training instead of test time.

### 4.8 Conclusion

We have presented Expected Statistic Regularization, a general approach to weak supervision for structured prediction, and studied it in the context of modern cross-lingual multi-task syntactic parsing. We evaluated a wide range of expressive cross-task statistics in idealized and realistic

transfer scenarios and have shown that the proposed approach is effective and complementary to the state-of-the-art model-transfer approaches.

We have also given a principled method for estimating loss targets using small amounts of labeled data and shown that realistic semi-supervised transfer with our method massively improves over zero-shot parsing even with only 50 labeled sentences, indicating the proposed approach is highly applicable to building parsers in low-resource languages with at least some labeled data.

The holy grail for most of the cross-lingual parsing community, however, is zero-shot transfer and we have shown in idealized experiments that our approach also leads to significant gains in this regime, *if* we have good estimates of the target statistics for the target language. Future work then will focus on methods for accurately inferring these statistic targets without labeled data in the target language. There are several potential paths: hierarchical bayesian approaches that incorporate prior knowledge about UD in general and in nearby language families; labeled data in related languages; estimating the statistics directly from PLMs or outside knowledge source such as WALS (Dingemanse, 2008) or URIEL (Littell et al., 2017); and most likely some combination of these information sources.

In the broader context of this thesis, ESR is an exciting development toward expanding the family of possible approaches that can be used to increase annotation-efficiency in biased settings. Because ESR allows for the expression of statistics that bear on marginal model over a whole sample of texts, it can allow us to use weak supervision functions that do not depend on the input, as opposed to most prior work on weak supervision. Further, it does not require that the target statistic value be known apriori by the expert, as we can estimate it from small amounts of labeled data. Finally, it can be used to counteract known biases in the model (or those in the dataset that will be reflected in the induced model), simply by regularizing away from these specific behaviors. Together, we believe that ESR can be applicable to many more NLP problems that require annotation-efficiency and/or have biased data.

## Chapter 5: Conclusions

In this thesis, we have contributed approaches for improving annotation-efficiency in biased learning settings for NLP. Our motivation to work on this problem stems from the desire to improve the accessibility of using NLP for a wider variety of applications, so that experts can use NLP in more diverse and niche use-cases with lower financial and time costs to obtain suitable accuracy. We believe that continued work in this direction is critical, as it has the potential to democratize access to NLP for generating knowledge from unstructured text and expand human knowledge overall.

A central theme of this thesis has been that intervention at the loss-function level is a promising path forward toward injecting domain and problem expertise as inductive bias into modern machine learning models. This is because it is compatible with recent advances in deep learning that prescribe complex mathematical architectures with uninterpretable dense vector representations. In contrast, it is less clear how to inject expert knowledge at the input or model level.

Beyond this, we would like to stress our advocacy for a utilizing a combination of learning paradigms to produce applications. Unsupervised learning, weakly supervised learning, and traditional supervised learning can all be mutually beneficial to each other in resource-constrained scenarios. This is because they each make a different trade-off of data acquisition cost versus information they carry about the target task. Unsupervised data is plentiful and can be used to pre-train and initialize models with effective generalization capabilities, but they do not yield directly usable models. Then both weak supervision and traditional supervised learning can be used to train the model for the end task, with weak supervision providing broad and cheap but incomplete signal about the target task while strong supervision provides complete but costly signal.

Another core idea in this thesis is that expert knowledge can be used to counteract dataset biases in addition to improving annotation-efficiency, and that these two uses can be mutually beneficial:

high-level expert knowledge can allow for biased labeling processes that are even less costly for the experts while still providing key strong supervised learning signal that cannot be not easily abstracted into weak supervision signals such as rules. We have shown this in Chapter 2, where we were able to improve the model without requiring additional manual annotations by using the fact that the unreviewed predictions of the previous model would have overwhelming bias towards the negative class. In Chapter 3 we showed this by proposing an annotation scheme that trades off labeling bias for increased context coverage given a fixed annotation budget, allowing us to train better models with modest budgets compared to exhaustive annotation. Lastly, in Chapter 4, we used our knowledge that pretrained multilingual parsers are trained with data that is biased toward certain high-resource languages and that this would cause erratic but simple grammatical divergences on “distant” low-resource target languages to derive cheap and simple but effective statistic functions that explicitly discourage this behavior.

We have explored these themes through empirical studies across three core NLP tasks: text classification, named entity recognition, and syntactic parsing. By working on different problem types, we have hopefully demonstrated that these themes are more generally applicable across a range of NLP problems. Further, we have addressed variants of the problem settings that are more constrained than traditional supervised settings, and demonstrated the real-world applicability of these settings.

All of the data, code, and journal papers that make up the chapters of this thesis can be found at <https://github.com/teffland>. Questions about this thesis or any of the related materials should be directed to the author.

Next, in Section 5.1 we again summarize our key contributions. Finally, we discuss limitations of our approaches and implications for future work in Section 5.2.

## **5.1 Contributions**

Our key contributions can be summarized as follows:

1. We propose a novel approach to improving a deployed rare-event classification with biased

incidental feedback (Effland et al., 2018). Specifically we contribute:

- (a) A method for improving and debiasing the deployed system using a combination of importance weighting and data imputation that exploits the selection bias of the previous system iteration without requiring additional labels from domain experts.
  - (b) A detailed evaluation and error analysis of the method for two applied problems in rare-event text classification. Our evaluation shows that our method improves precision and recall of the resulting model and counteracts the training data bias.
  - (c) Considerable improvements in performance of a real-world deployed rare-event text classification system with immediate impact.
2. We propose a novel approach for learning named entity recognition models using biased, partially labeled data (Effland and Collins, 2021). Specifically we contribute:
- (a) A principled method that utilizes a weak expert prior about the relative occurrence rate of entities in the text to train accurate NER models using low-recall data.
  - (b) Theory justifying the statistical consistency of the approach, proving that our approach recovers the true tagging distribution in the limit of infinite data under mild conditions.
  - (c) Extensive benchmark comparisons showing that our method equals or outperforms previous state-of-the-art approaches across 7 corpora, 6 languages, and 2 diverse low-recall annotation scenarios.
  - (d) A novel partial annotation scheme that we call “Exploratory Expert” (EE) annotation, which allows experts to inexhaustively skim and annotate documents, generating more varied example contexts for a fixed time budget.
  - (e) A user study, showing that EE is as fast as exhaustive annotation.
  - (f) Learning curve experiments that show EE annotation can outperform exhaustive annotation for modest annotation budgets.

3. We propose a novel approach for improving cross-lingual syntactic parsing in low-resource scenarios by using expected typological statistics in the target language as weak supervision. Specifically we contribute:

- (a) A novel and general regularization framework, “Expected Statistic Regularization” (ESR), that can be used to regularize models on unlabeled target datasets with a broad class of functions that describe expected model behavior. These statistics allow for the incorporation of various forms of high-level expert knowledge as supervision.
- (b) A method for estimating target statistic values using small amounts of labeled data.
- (c) An application of the method that improves state-of-the-art cross-lingual parsing on low-resource languages. We contribute seven families of descriptive statistics that bear on parser behavior and extensively evaluate their impact on transfer, showing most to be useful.
- (d) An extensive benchmark evaluation on transfer to 44 languages showing that ESR leads to significant improvements over state-of-the-art approaches on many low-resource languages.
- (e) Learning curve experiments that demonstrate the impact of the approach is largest for target datasets with 500 or fewer annotated sentences.
- (f) Ablation studies justifying key design choices for the proposed loss function.

## 5.2 Limitations and Future Work

In this thesis we have only begun to scratch the surface of possible ways to create annotation-efficiency and turn dataset bias to our advantage in low-resource settings, both in terms of technical directions and application areas. Here we focus on current limitations and future iterations of ESR, as it is our most recent and general work and as such we view it as the most promising direction.

One limitation of our work to date with ESR is the class of explored statistic functions. So far we have primarily explored a families of statistics that calculate various entropies and marginal

substructure probabilities under the parsing model. Some of these, specifically the conditional substructure marginals, were not as beneficial as we had hoped. One can easily imagine working with other types of statistics. Some possibilities for these are:

- **Input-conditional statistics.** Statistics that depend on the input data itself, as in:

$$f_i(x, p_\theta) = \mathbb{1}\{x \text{ has some feature}\} h_i(p_\theta(y|x))$$

where  $h_i$  is intentionally abstract. Akin to “labeling functions” in popular weak-supervision (Ratner et al., 2016a; Karamanolakis et al., 2019b) functions that activate when looking at things like particular pairs of words within certain distances could be used to create higher precision weak supervision at lower recall. An interesting advantage of ESR over labeling functions in this case is that we could still optimize the expected values of these over batches of data instead of using them for labeling particular instances, and we would not need to rely on an additional model to denoise the labels.

- **“Deconfusion” statistics based on error-analysis.** The idea behind these statistics would be that we can use labeled training sets to identify particular categories of confusion and directly correct for these cases. For example, when transferring the UDify model to Persian, we found that it often confused VERB-compound->NOUN edges, which are particularly common in Persian, for VERB-object->NOUN edges that are much more typical across the training languages. It could potentially be quite useful to specifically supervise the model with the signal that says “often when the initial model predicted a dependency to be VERB-object->NOUN, it is more likely a VERB-compound->NOUN.” We could do this by first checking the model on labeled training data for these error types, then locate these positions in the unlabeled training data where its likely the model will make this type of mistake, and apply the statistic just at these locations instead of in expectation over all positions in the batch. Further, one can imagine statistic functions that are adaptive, changing over the course of training based on the evolution of error types. Mathematically these

statistics could look something like:

$$f_i(x, p_{\theta_t}) = \mathbb{1}\{\arg \max p_{\theta_{t-1}}(y|x) \text{ contains a known error type at position } k\} h_i^t(p_{\theta_t}(y|x))$$

where  $t$  is the training epoch and  $h_i^t$  selects the marginals at position  $k$  and penalizes them for not matching the most likely error correction for  $p_{\theta_{t-1}}$  calculated using validation data after the last iteration.

- **Learned/parameterized statistic functions.** In our current work, we employed only statistic functions that were apriori known and hypothesized to be useful, much like in the feature engineering era of machine learning before deep learning came to dominate. One of the most appealing aspects of deep learning is that it automatically learns the features (representations) of the data in the lower layers and does a better job at this than human intuition. So it is indeed natural to ask if we can similarly learn the statistic functions that we use as auxiliary supervision for our models. One obvious case of this is generative modeling. If we view the statistic function  $f$  as the conditional likelihood of a sample of text given the annotations, and the inference model  $p$  as a parameterized posterior, i.e.,

$$f(\mathcal{D}, p_{\theta}) = q_{\phi}(\mathcal{D}|y), \quad y \sim p_{\theta}(y|x), x \in \mathcal{D}$$

then we have something similar to a variational autoencoder (Kingma et al., 2014), and could potentially learn the statistic parameters  $\phi$  using unlabeled data that way.<sup>1</sup>

Another clear direction of future work is expansion into different NLP tasks and application areas. In particular, we believe there is tremendous opportunity for usage of ESR in information extraction (IE) and knowledge base construction problems because the semantics of their label do-

---

<sup>1</sup>For this direction it is worth noting, however, that using generative models to do semi-supervised learning of the structured inference model is difficult in practice. We attempted this in a previous project for semi-supervised NER, but found the results to be suboptimal compared to the discriminative supervision of the EER loss. It is entirely possible though that the previous suboptimal performance of this approach was due to technical optimization complications, and not the idea itself.

mains typically imply many useful constraints. For one, the ontologies and database schemas that define the entities and relations in the target domains of these applications present an opportunity to automatically derive constraints and expectations for “type checking” of IE or slot filling models, similar to type-checking constraints in NELL (Carlson et al., 2010). This would be analogous to how we use the universal dependencies formalism to find grammatically invalid substructures of POS and dependency arc labels. Further, there are opportunities to use domain expertise or the current state of the knowledge bases to estimate priors over numerical semantic substructure likelihoods and use these as statistic targets, similarly to how we estimated substructure target distributions from small amounts of training data in ESR. Finally, and perhaps more loftily, we could potentially pair the current state of knowledge in the database with other more complex “validation” statistic functions, such as theorem provers (Rocktäschel and Riedel, 2017), to regularize models away from extractions that semantically contradict our current hypotheses about relevant global domain state.

Continued developments in these directions could be powerful for building domain-specific knowledge base construction systems more quickly and easily, and would continue to push the boundaries of the feasibility of using NLP for generating knowledge in diverse and niche domains, the primary goal of this thesis.

## Bibliography

- Eugene Agichtein and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In Proceedings of the fifth ACM conference on Digital libraries, pages 85–94.
- Wasi Uddin Ahmad, Zhisong Zhang, Xuezhe Ma, Kai-Wei Chang, and Nanyun Peng. 2019. Cross-lingual dependency parsing with unlabeled auxiliary languages. In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 372–382, Hong Kong, China. Association for Computational Linguistics.
- Hernán Ahumada, Guillermo L. Grinblat, Lucas C. Uzal, Pablo M. Granitto, and H. Alejandro Ceccatto. 2008. Repmac: A new hybrid approach to highly imbalanced classification problems. 2008 Eighth International Conference on Hybrid Intelligent Systems, pages 386–391.
- Rehan Akbani, Stephen Kwek, and Nathalie Japkowicz. 2004. Applying support vector machines to imbalanced datasets. In ECML.
- Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. Pooled contextualized embeddings for named entity recognition. 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 724–728, Minneapolis, Minnesota. Association for Computational Linguistics.
- Firoj Alam, Shafiq R. Joty, and Muhammad Imran. 2018. Graph based semi-supervised learning with convolution neural networks to classify crisis related tweets. In ICWSM.
- Nikolaos Aletras, Dimitrios Tsarapatsanis, Daniel Preotiuc-Pietro, and Vasileios Lampsos. 2016. Predicting judicial decisions of the european court of human rights: a natural language processing perspective. PeerJ Comput. Sci., 2:e93.
- Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2016. Many languages, one parser. Transactions of the Association for Computational Linguistics, 4:431–444.
- Rangachari Anand, Kishan G. Mehrotra, Chilukuri Krishna Mohan, and Sanjay Ranka. 1993. An improved algorithm for neural network classification of imbalanced training sets. IEEE transactions on neural networks, 4 6:962–9.
- Stefanos Angelidis and Mirella Lapata. 2018. Multiple instance learning networks for fine-grained sentiment analysis. Transactions of the Association for Computational Linguistics, 6:17–31.
- Isabelle Augenstein, Leon Derczynski, and Kalina Bontcheva. 2017. Generalisation in named entity recognition: A quantitative analysis. Computer Speech & Language, 44:61–83.
- Abhijeet Awasthi, Sabyasachi Ghosh, Rasna Goyal, and Sunita Sarawagi. 2020. Learning from rules generalizing labeled exemplars. In International Conference on Learning Representations.

- Stephen H Bach, Daniel Rodriguez, Yintao Liu, Chong Luo, Haidong Shao, Cassandra Xia, Souvik Sen, Alex Ratner, Braden Hancock, Houman Alborzi, et al. 2019. Snorkel drybell: A case study in deploying weak supervision at industrial scale. In Proceedings of the 2019 International Conference on Management of Data, pages 362–375.
- Alexei Baevski, Sergey Edunov, Yinhan Liu, Luke Zettlemoyer, and Michael Auli. 2019. Cloze-driven pretraining of self-attention 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 5360–5369, Hong Kong, China. Association for Computational Linguistics.
- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.
- Chi Y Bahk, David A Scales, Sumiko R Mekar, John S Brownstein, and Clark C Freifeld. 2015. Comparing timeliness, content, and disease severity of formal and informal source outbreak reporting. BMC infectious diseases, 15(1):1–6.
- Kedar Bellare and Andrew McCallum. 2007. Learning extractors from unlabeled text using relevant databases. In Sixth international workshop on information integration on the web.
- 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.
- Leo Breiman. 2001. Random forests. Machine learning, 45(1):5–32.
- Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. 1992. Class-based  $n$ -gram models of natural language. Computational Linguistics, 18(4):467–480.
- Vannevar Bush et al. 1945. As we may think. The atlantic monthly, 176(1):101–108.
- Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka, and Tom M Mitchell. 2010. Toward an architecture for never-ending language learning. In Twenty-Fourth AAAI conference on artificial intelligence.
- Andrew Carlson, Scott Gaffney, and Flavian Vasile. 2009. Learning a named entity tagger from gazetteers with the partial perceptron. In AAAI Spring Symposium: Learning by Reading and Learning to Read, pages 7–13.
- Xavier Carreras. 2007. Experiments with a higher-order projective dependency parser. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), pages 957–961, Prague, Czech Republic. Association for Computational Linguistics.
- Rich Caruana. 2004. Multitask learning. Machine Learning, 28:41–75.
- Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. Legal-bert: The muppets straight out of law school. ArXiv, abs/2010.02559.

- Ming-Wei Chang, Lev Ratinov, and Dan Roth. 2007a. Guiding semi-supervision with constraint-driven learning. In Proceedings of the 45th annual meeting of the association of computational linguistics, pages 280–287.
- Ming-Wei Chang, Lev-Arie Ratinov, and Dan Roth. 2007b. Guiding semi-supervision with constraint-driven learning. In ACL.
- N. Chawla, K. Bowyer, Lawrence O. Hall, and W. Philip Kegelmeyer. 2002. Smote: Synthetic minority over-sampling technique. J. Artif. Intell. Res., 16:321–357.
- Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc V. Le. 2018. Semi-supervised sequence modeling with cross-view training. In EMNLP.
- Michael Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10, pages 1–8. Association for Computational Linguistics.
- Michael Collins and Yoram Singer. 1999. Unsupervised models for named entity classification. In 1999 Joint SIGDAT conference on empirical methods in natural language processing and very large corpora.
- Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. Machine learning, 20(3):273–297.
- David R Cox. 1958. The regression analysis of binary sequences. Journal of the Royal Statistical Society: Series B (Methodological), 20(2):215–232.
- Daniel Dahlmeier. 2017. On the challenges of translating NLP research into commercial products. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 92–96, Vancouver, Canada. Association for Computational Linguistics.
- Jesse Davis and Mark H. Goadrich. 2006. The relationship between precision-recall and roc curves. Proceedings of the 23rd international conference on Machine learning.
- Stephen Della Pietra, Vincent Della Pietra, and John Lafferty. 1997. Inducing features of random fields. IEEE transactions on pattern analysis and machine intelligence, 19(4):380–393.
- Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the wnut2017 shared task on novel and emerging entity recognition. In NUT@EMNLP.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In NAACL.
- Mark Dingemanse. 2008. Wals online. Elanguage.
- Pedro M. Domingos. 1999. Metacost: a general method for making classifiers cost-sensitive. In KDD '99.

- Timothy Dozat and Christopher D. Manning. 2017. Deep biaffine attention for neural dependency parsing. ArXiv, abs/1611.01734.
- Gregory Druck, Gideon S. Mann, and Andrew McCallum. 2008. Learning from labeled features using generalized expectation criteria. In SIGIR '08.
- Gregory Druck, Gideon S. Mann, and Andrew McCallum. 2009. Semi-supervised learning of dependency parsers using generalized expectation criteria. In ACL.
- Harris Drucker, Chris J.C. Burges, Linda Kaufman, Alex Smola, and Vladimir Vapnik. 1996. Support vector regression machines. In NIPS.
- Thomas Effland and Michael Collins. 2021. Partially supervised named entity recognition via the expected entity ratio loss. Transactions of the Association for Computational Linguistics, 9:1320–1335.
- Thomas Effland, Anna Lawson, Sharon Balter, Katelynn Devinney, Vasudha Reddy, HaeNa Waechter, Luis Gravano, and Daniel Hsu. 2018. Discovering foodborne illness in online restaurant reviews. Journal of the American Medical Informatics Association, 25(12):1586–1592.
- Bradley Efron. 1979. Bootstrap methods: Another look at the jackknife. Annals of Statistics, 7:1–26.
- Bradley Efron and Robert J Tibshirani. 1994. An introduction to the bootstrap. CRC press.
- Johannes C. Eichstaedt, Robert J. Smith, Raina M. Merchant, Lyle H. Ungar, Patrick Crutchley, Daniel Preotiuc-Pietro, David A. Asch, and H. A. Schwartz. 2018. Facebook language predicts depression in medical records. Proceedings of the National Academy of Sciences of the United States of America, 115:11203 – 11208.
- Charles Elkan and Keith Noto. 2008. Learning classifiers from only positive and unlabeled data. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 213–220.
- Kazuo J Ezawa, Moninder Singh, and Steven W Norton. 1996. Learning goal oriented bayesian networks for telecommunications risk management. In ICML, pages 139–147.
- Tom Fawcett and Foster J Provost. 1996. Combining data mining and machine learning for effective user profiling. In KDD, volume 96, pages 8–13.
- Titus Hei Yeung Fong, Shahryar Sarkani, and John M. Fossaceca. 2021. Auto defect detection using customer reviews for product recall insurance analysis. In Frontiers in Applied Mathematics and Statistics.
- Clark C Freifeld, Kenneth D Mandl, Ben Y Reis, and John S Brownstein. 2008. Healthmap: global infectious disease monitoring through automated classification and visualization of internet media reports. Journal of the American Medical Informatics Association, 15(2):150–157.
- Kuzman Ganchev and Dipanjan Das. 2013. Cross-lingual discriminative learning of sequence models with posterior regularization. In EMNLP.

- Kuzman Ganchev, Joao Graça, Jennifer Gillenwater, and Ben Taskar. 2010a. Posterior regularization for structured latent variable models. The Journal of Machine Learning Research, 11:2001–2049.
- Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2010b. Posterior regularization for structured latent variable models. J. Mach. Learn. Res., 11:2001–2049.
- Francesco Gesualdo, Giovanni Stilo, Eleonora Agricola, Michaela V. Gonfiantini, Elisabetta Pandolfi, Paola Velardi, and Alberto Eugenio Tozzi. 2013. Influenza-like illness surveillance on twitter through automated learning of naïve language. PLoS ONE, 8.
- Rayid Ghani, Katharina Probst, Yan Liu, Marko Krema, and Andrew E. Fano. 2006. Text mining for product attribute extraction. SIGKDD Explor., 8:41–48.
- Ross Girshick. 2015. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision, pages 1440–1448.
- Goran Glavas, Federico Nanni, and Simone Paolo Ponzetto. 2019. Computational analysis of political texts: Bridging research efforts across communities. In ACL.
- Goran Glavas and Ivan Vulic. 2021. Climbing the tower of treebanks: Improving low-resource dependency parsing via hierarchical source selection. In FINDINGS.
- L Hannah Gould, Kelly A Walsh, Antonio R Vieira, Karen Herman, Ian T Williams, Aron J Hall, and Dana Cole. 2013. Surveillance for foodborne disease outbreaks—united states, 1998–2008. Morbidity and Mortality Weekly Report: Surveillance Summaries, 62(2):1–34.
- Yves Grandvalet and Yoshua Bengio. 2004a. Semi-supervised learning by entropy minimization. In CAP.
- Yves Grandvalet and Yoshua Bengio. 2004b. Semi-supervised learning by entropy minimization. Advances in neural information processing systems, 17.
- Edouard Grave. 2014. Weakly supervised named entity classification. In Workshop on Automated Knowledge Base Construction (AKBC).
- Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018).
- Nathan Greenberg, Trapit Bansal, Patrick Verga, and Andrew McCallum. 2018a. Marginal likelihood training of BiLSTM-CRF for biomedical named entity recognition from disjoint label sets. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2824–2829.
- Nathan Greenberg, Trapit Bansal, Patrick Verga, and Andrew McCallum. 2018b. Marginal likelihood training of BiLSTM-CRF for biomedical named entity recognition from disjoint label sets. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2824–2829, Brussels, Belgium. Association for Computational Linguistics.

- Justin Grimmer and Brandon M Stewart. 2013. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. Political Analysis, 21:267 – 297.
- Emil Julius Gumbel. 1954. Statistical Theory of Extreme Values and Some Practical Applications: A Series of Lectures, volume 33. US Government Printing Office.
- Jiang Guo, W. Che, David Yarowsky, H. Wang, and T. Liu. 2015. Cross-lingual dependency parsing based on distributed representations. In ACL.
- Yoni Halpern, Steven Horng, Youngduck Choi, and David A. Sontag. 2016. Electronic medical record phenotyping using the anchor and learn framework. Journal of the American Medical Informatics Association : JAMIA, 23:731 – 740.
- Hui Han, Wenyuan Wang, and Binghuan Mao. 2005. Borderline-smote: A new over-sampling method in imbalanced data sets learning. In ICIC.
- Jenine K Harris, Raed Mansour, Bechara Choucair, Joe Olson, Cory Nissen, and Jay Bhatt. 2014. Health department use of social media to identify foodborne illness—chicago, illinois, 2013–2014. Morbidity and Mortality Weekly Report, 63(32):681.
- Cassandra Harrison, Mohip Jorder, Henri Stern, Faina Stavinsky, Vasudha Reddy, Heather Hanson, HaeNa Waechter, Luther Lowe, Luis Gravano, and Sharon Balter. 2014. Using online reviews by restaurant patrons to identify unreported cases of foodborne illness—new york city, 2012–2013. Morbidity and Mortality Weekly Report, 63(20):441.
- Haibo He, Yang Bai, Edwardo A. Garcia, and Shutao Li. 2008. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. 2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), pages 1322–1328.
- Junxian He, Zhisong Zhang, Taylor Berg-Kirkpatrick, and Graham Neubig. 2019. Cross-lingual syntactic transfer through unsupervised adaptation of invertible projections. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3211–3223, Florence, Italy. Association for Computational Linguistics.
- Xuanli He, Islam Nassar, Jamie Kiros, Gholamreza Haffari, and Mohammad Norouzi. 2022. Generate, annotate, and learn: NLP with synthetic text. Trans. Assoc. Comput. Linguistics, 10:826–842.
- Tim Head, Manoj Kumar, Holger Nahrstaedt, Gilles Louppe, and Iaroslav Shcherbatyi. 2020. scikit-optimize/scikit-optimize.
- Daniel G. Horvitz and D. J. Thompson. 1952. A generalization of sampling without replacement from a finite universe. Journal of the American Statistical Association, 47:663–685.
- Eduard Hovy, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers, NAACL-Short '06, pages 57–60, Stroudsburg, PA, USA. Association for Computational Linguistics.

- Muhammad Imran, Shady Elbassuoni, Carlos Castillo, Fernando Diaz, and Patrick Meier. 2013. Extracting information nuggets from disaster- related messages in social media. In ISCRAM.
- Muhammad Imran, Prasenjit Mitra, and Carlos Castillo. 2016. Twitter as a lifeline: Human-annotated twitter corpora for nlp of crisis-related messages. ArXiv, abs/1605.05894.
- Hugging Face Inc. 2019. PyTorch Pretrained BERT: The Big & Extending Repository of pretrained Transformers.
- Eric Jang, Shixiang Shane Gu, and Ben Poole. 2017. Categorical reparameterization with gumbel-softmax. ArXiv, abs/1611.01144.
- Nathalie Japkowicz and Shaju Stephen. 2002. The class imbalance problem: A systematic study. Intell. Data Anal., 6:429–449.
- Zhanming Jie, Pengjun Xie, Wei Lu, Ruixue Ding, and Linlin Li. 2019. Better modeling of incomplete annotations for named entity recognition. In Proceedings of NAACL.
- Justin M. Johnson and Taghi M. Khoshgoftaar. 2019. Survey on deep learning with class imbalance. Journal of Big Data, 6:1–54.
- Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. Transactions of the Association for Computational Linguistics, 8:64–77.
- Dongyeop Kang, Tushar Khot, Ashish Sabharwal, and Eduard Hovy. 2018. AdvEntuRe: Adversarial training for textual entailment with knowledge-guided examples. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2418–2428, Melbourne, Australia. Association for Computational Linguistics.
- Giannis Karamanolakis, Daniel Hsu, and Luis Gravano. 2019a. Leveraging just a few keywords for fine-grained aspect detection through weakly supervised co-training. 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 4611–4621, Hong Kong, China. Association for Computational Linguistics.
- Giannis Karamanolakis, Daniel Hsu, and Luis Gravano. 2019b. Weakly supervised attention networks for fine-grained opinion mining and public health. In Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019), pages 1–10, Hong Kong, China. Association for Computational Linguistics.
- Giannis Karamanolakis, Daniel J. Hsu, and Luis Gravano. 2019c. Weakly supervised attention networks for fine-grained opinion mining and public health. ArXiv, abs/1910.00054.
- Giannis Karamanolakis, Jun Ma, and Xin Dong. 2020. Textract: Taxonomy-aware knowledge extraction for thousands of product categories. ArXiv, abs/2004.13852.
- Giannis Karamanolakis, Subhabrata Mukherjee, Guoqing Zheng, and Ahmed Hassan Awadallah. 2021. Self-training with weak supervision. In NAACL.

- Chris Kedzie, Kathleen McKeown, and Fernando Diaz. 2015. Predicting salient updates for disaster summarization. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1608–1617.
- Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114.
- Durk P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. 2014. Semi-supervised learning with deep generative models. In Advances in neural information processing systems, pages 3581–3589.
- Leslie Kish. 1965. Survey Sampling. John Wiley and Sons, Inc., New York.
- D. Kondratyuk. 2019. 75 languages, 1 model: Parsing universal dependencies universally. In EMNLP/IJCNLP.
- Miroslav Kubat, Robert C Holte, and Stan Matwin. 1998. Machine learning for the detection of oil spills in satellite radar images. Machine learning, 30(2):195–215.
- Artur Kulmizev, Miryam de Lhoneux, Johannes Gontrum, Elena Fano, and Joakim Nivre. 2019. Deep contextualized word embeddings in transition-based and graph-based dependency parsing - a tale of two parsers revisited. In EMNLP/IJCNLP.
- Kemal Kurniawan, Lea Frermann, Philip Schulz, and Trevor Cohn. 2021. Ppt: Parsimonious parser transfer for unsupervised cross-lingual adaptation. In EACL.
- John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In ICML.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, K. Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In NAACL.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.
- Michael Laver, Kenneth Benoit, and John Garry. 2003. Extracting policy positions from political texts using words as data. American Political Science Review, 97:311 – 331.
- Belinda Z. Li, Gabriel Stanovsky, and Luke Zettlemoyer. 2020. Active learning for coreference resolution using discrete annotation. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8320–8331, Online. Association for Computational Linguistics.
- William Li, Pablo Azar, David Larochelle, and Phil Hill. 2012. Using algorithmic attribution techniques to determine authorship in unsigned judicial opinions. Stan. Tech. L. Rev., 16:503.
- Yangming Li, Lemaou Liu, and Shuming Shi. 2021. Empirical analysis of unlabeled entity problem in named entity recognition. In International Conference on Learning Representations.

- Percy Liang. 2005. Semi-supervised learning for natural language.
- Percy Liang, Michael I. Jordan, and Dan Klein. 2009. Learning from measurements in exponential families. In ICML '09.
- Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing transfer languages for cross-lingual learning. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 3125–3135, Florence, Italy. Association for Computational Linguistics.
- Charles X. Ling and Chenghui Li. 1998. Data mining for direct marketing: Problems and solutions. In KDD.
- Patrick Littell, David R. Mortensen, Ke Lin, Katherine Kairis, Carlisle Turner, and Lori Levin. 2017. URIEL and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 8–14, Valencia, Spain. Association for Computational Linguistics.
- Bing Liu, Yang Dai, Xiaoli Li, Wee Sun Lee, and Philip S Yu. 2003. Building text classifiers using positive and unlabeled examples. In Third IEEE International Conference on Data Mining, pages 179–186. IEEE.
- Bing Liu, Wee Sun Lee, Philip S. Yu, and Xiaoli Li. 2002. Partially supervised classification of text documents. In ICML, volume 2, pages 387–394.
- Lu Liu, Yi Zhou, Jianhan Xu, Xiaoqing Zheng, Kai-Wei Chang, and Xuanjing Huang. 2020. Cross-lingual dependency parsing by pos-guided word reordering. In FINDINGS.
- Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. ArXiv, abs/1903.08855.
- Tianyu Liu, Jin-ge Yao, and Chin-Yew Lin. 2019b. Towards improving neural named entity recognition with gazetteers. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5301–5307.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019c. RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692v1.
- Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In ICLR.
- Jouni Luoma and Sampo Pyysalo. 2020. Exploring cross-sentence contexts for named entity recognition with bert. In COLING.
- Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. ArXiv, abs/1603.01354.

- Zhuang Ma and Michael Collins. 2018. Noise contrastive estimation and negative sampling for conditional models: Consistency and statistical efficiency. In EMNLP.
- Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2017. The concrete distribution: A continuous relaxation of discrete random variables. ArXiv, abs/1611.00712.
- Gideon S. Mann and Andrew McCallum. 2007. Simple, robust, scalable semi-supervised learning via expectation regularization. In ICML '07.
- Gideon S. Mann and Andrew McCallum. 2010. Generalized expectation criteria for semi-supervised learning with weakly labeled data. J. Mach. Learn. Res., 11:955–984.
- Stephen Mayhew, Snigdha Chaturvedi, Chen-Tse Tsai, and Dan Roth. 2019. Named entity recognition with partially annotated training data. In Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL), pages 645–655, Hong Kong, China. Association for Computational Linguistics.
- Andrew McCallum and Wei Li. 2003. Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4, pages 188–191. Association for Computational Linguistics.
- R. McDonald, Slav Petrov, and Keith B. Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In EMNLP.
- Ryan T. McDonald and Fernando C Pereira. 2006. Online learning of approximate dependency parsing algorithms. In EACL.
- Tao Meng, Nanyun Peng, and Kai-Wei Chang. 2019. Target language-aware constrained inference for cross-lingual dependency parsing. ArXiv, abs/1909.01482.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26.
- Kai Ming Ting and Zijian Zheng. 1998. Boosting trees for cost-sensitive classifications. In European conference on machine learning, pages 190–195. Springer.
- Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP, pages 1003–1011.
- Seyed Abolghasem Mirroshandel and Alexis Nasr. 2011. Active learning for dependency parsing using partially annotated sentences. In Proceedings of the 12th International Conference on Parsing Technologies, pages 140–149, Dublin, Ireland. Association for Computational Linguistics.

- Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. 2020. Monte carlo gradient estimation in machine learning. J. Mach. Learn. Res., 21(132):1–62.
- Federico Nanni, Goran Glavas, Simone Paolo Ponzetto, and Heiner Stuckenschmidt. 2021. Political text scaling meets computational semantics. ACM/IMS Transactions on Data Science (TDS), 2:1 – 27.
- Joakim Nivre. 2020. Multilingual dependency parsing from universal dependencies to sesame street. In TDS.
- Matan Ben Noach and Yoav Goldberg. 2019. Transfer learning between related tasks using expected label proportions. In EMNLP.
- Farhad Nooralahzadeh, Jan Tore Lønning, and Lilja Ovrelid. 2019. Reinforcement-based denoising of distantly supervised ner with partial annotation. In DeepLo@EMNLP-IJCNLP.
- Joel Nothman, James R Curran, and Tara Murphy. 2008. Transforming wikipedia into named entity training data. In Proceedings of the Australasian Language Technology Association Workshop 2008, pages 124–132.
- Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R Curran. 2013. Learning multilingual named entity recognition from wikipedia. Artificial Intelligence, 194:151–175.
- Elaine O Nsoesie, Sheryl A Kluberg, and John S Brownstein. 2014. Online reports of foodborne illness capture foods implicated in official foodborne outbreak reports. Preventive medicine, 67:264–269.
- Avital Oliver, Augustus Odena, Colin Raffel, Ekin Dogus Cubuk, and Ian J. Goodfellow. 2018. Realistic evaluation of deep semi-supervised learning algorithms. In NeurIPS.
- IV OliverJ.BearDon’tWalk, Tony Y. Sun, Adler J. Perotte, and Noémie Elhadad. 2021. Clinically relevant pretraining is all you need. Journal of the American Medical Informatics Association : JAMIA.
- Arghya Pal and Vineeth N Balasubramanian. 2018. Adversarial data programming: Using gans to relax the bottleneck of curated labeled data. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 1556–1565.
- Leysia Palen, Kenneth Mark Anderson, Gloria Mark, James H. Martin, Douglas C. Sicker, Martha Palmer, and Dirk Grunwald. 2010. A vision for technology-mediated support for public participation & assistance in mass emergencies & disasters.
- Max B. Paulus, Dami Choi, Daniel Tarlow, Andreas Krause, and Chris J. Maddison. 2020. Gradient estimation with stochastic softmax tricks. ArXiv, abs/2006.08063.
- Minlong Peng, Xiaoyu Xing, Qi Zhang, Jinlan Fu, and Xuanjing Huang. 2019. Distantly supervised named entity recognition using positive-unlabeled learning. In ACL.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. Deep contextualized word representations. In NAACL.

- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018b. Deep contextualized word representations. In NAACL.
- Petar Petrovski and Christian Bizer. 2017. Extracting attribute-value pairs from product specifications on the web. Proceedings of the International Conference on Web Intelligence.
- Sameer Pradhan, Noémie Elhadad, Brett R. South, David Martínez, Lee M. Christensen, Amy Vogel, Hanna Suominen, Wendy W. Chapman, and Guergana K. Savova. 2015. Evaluating the state of the art in disorder recognition and normalization of the clinical narrative. Journal of the American Medical Informatics Association : JAMIA, 22:143 – 154.
- Foster J. Provost and Tom Fawcett. 1998. Robust classification systems for imprecise environments. In AAAI/IAAI.
- Duangmanee Putthividhya and Junling Hu. 2011. Bootstrapped named entity recognition for product attribute extraction. In EMNLP.
- J Ross Quinlan. 2014. C4. 5: programs for machine learning. Elsevier.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. OpenAI Blog, 1:8.
- Anand Rajaraman and Jeffrey David Ullman. 2011. Mining of massive datasets. Cambridge University Press.
- Mohammad Sadegh Rasooli and M. Collins. 2019. Low-resource syntactic transfer with unsupervised source reordering. In NAACL-HLT.
- Mohammad Sadegh Rasooli and Michael Collins. 2017. Cross-lingual syntactic transfer with limited resources. Transactions of the Association for Computational Linguistics, 5:279–293.
- Mohammad Sadegh Rasooli and Joel R. Tetreault. 2015. Yara parser: A fast and accurate dependency parser. Computing Research Repository, arXiv:1503.06733. Version 2.
- Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In Proceedings of the thirteenth conference on computational natural language learning, pages 147–155. Association for Computational Linguistics.
- Alexander Ratner, Stephen H Bach, Henry Ehrenberg, Jason Fries, Sen Wu, and Christopher Ré. 2017. Snorkel: Rapid training data creation with weak supervision. In Proceedings of the VLDB Endowment. International Conference on Very Large Data Bases, volume 11, page 269. NIH Public Access.
- Alexander J Ratner, Christopher M De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016a. Data programming: Creating large training sets, quickly. In Advances in neural information processing systems, pages 3567–3575.
- Alexander J. Ratner, Braden Hancock, Jared A. Dunnmon, Roger E. Goldman, and Christopher Ré. 2018. Snorkel metal: Weak supervision for multi-task learning. Proceedings of the Second Workshop on Data Management for End-To-End Machine Learning.

- Alexander J. Ratner, Christopher De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016b. Data programming: Creating large training sets, quickly. Advances in neural information processing systems, 29:3567–3575.
- Herbert Robbins and Sutton Monro. 1951. A stochastic approximation method. The annals of mathematical statistics, pages 400–407.
- James M Robins, Andrea Rotnitzky, and Lue Ping Zhao. 1994. Estimation of regression coefficients when some regressors are not always observed. Journal of the American statistical Association, 89(427):846–866.
- Tim Rocktäschel and Sebastian Riedel. 2017. End-to-end differentiable proving. ArXiv, abs/1705.11040.
- Adam Sadilek, Henry Kautz, Lauren DiPrete, Brian Labus, Eric Portman, Jack Teitel, and Vincent Silenzio. 2016. Deploying nemesis: Preventing foodborne illness by data mining social media. In Twenty-Eighth IAAI Conference.
- Kuniaki Saito, Y. Ushiku, and Tatsuya Harada. 2017. Asymmetric tri-training for unsupervised domain adaptation. In ICML.
- Mauricio Santillana, André T Nguyen, Mark Dredze, Michael J Paul, Elaine O Nsoesie, and John S Brownstein. 2015. Combining search, social media, and traditional data sources to improve influenza surveillance. PLoS computational biology, 11(10):e1004513.
- Manabu Sassano and Sadao Kurohashi. 2010. Using smaller constituents rather than sentences in active learning for Japanese dependency parsing. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, pages 356–365, Uppsala, Sweden. Association for Computational Linguistics.
- Elaine Scallan, Patricia M Griffin, Frederick J Angulo, Robert V Tauxe, and Robert M Hoekstra. 2011. Foodborne illness acquired in the united states—unspecified agents. Emerging infectious diseases, 17(1):16.
- Haoyue Shi, Kevin Gimpel, and Karen Livescu. 2021. Substructure distribution projection for zero-shot cross-lingual dependency parsing. ArXiv, abs/2110.08538.
- Richard Shin, Christopher H. Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, Dan Klein, Jason Eisner, and Benjamin Van Durme. 2021. Constrained language models yield few-shot semantic parsers. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021, pages 7699–7715. Association for Computational Linguistics.
- Yanchuan Sim, Bryan R. Routledge, and Noah A. Smith. 2016. Friends with motives: Using text to infer influence on scotus. In EMNLP.

- Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. BRAT: A web-based tool for NLP-assisted text annotation. In Proceedings of the Demonstrations Session at EACL 2012, Avignon, France. Association for Computational Linguistics.
- Haitian Sun, William W. Cohen, and Lidong Bing. 2018. Semi-supervised learning with declaratively specified entropy constraints. In NeurIPS, pages 4430–4440.
- Oscar Täckström, R. McDonald, and Joakim Nivre. 2013. Target language adaptation of discriminative transfer parsers. In HLT-NAACL.
- Yuchun Tang, Yanqing Zhang, N. Chawla, and Sven Krasser. 2009. Svms modeling for highly imbalanced classification. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39:281–288.
- Jörg Tiedemann and Zeljko Agić. 2016. Synthetic treebanking for cross-lingual dependency parsing. J. Artif. Intell. Res., 55:209–248.
- Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pages 142–147.
- Kristina Toutanova, Christopher D. Manning, and A. Ng. 2004. Learning random walk models for inducing word dependency distributions. Proceedings of the twenty-first international conference on Machine learning.
- Khai Tran and Arianna Bisazza. 2019. Zero-shot dependency parsing with pre-trained multilingual sentence representations. In DeepLo@EMNLP-IJCNLP.
- Yuta Tsuboi, Hisashi Kashima, Hiroki Oda, Shinsuke Mori, and Yuji Matsumoto. 2008. Training conditional random fields using incomplete annotations. In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1, pages 897–904. Association for Computational Linguistics.
- A. Ustun, Arianna Bisazza, G. Bouma, and Gertjan van Noord. 2020. Uadapter: Language adaptation for truly universal dependency parsing. In EMNLP.
- Vladimir N. Vapnik. 1995. The Nature of Statistical Learning Theory. Springer-Verlag, Berlin, Heidelberg.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. Advances in neural information processing systems, 30.
- Pascal Vincent, H. Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res., 11:3371–3408.

- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In BlackboxNLP@EMNLP.
- William Yang Wang, Kapil Thadani, and Kathleen McKeown. 2011. Identifying event descriptions using co-training with online news summaries. In IJCNLP.
- Yushi Wang, Jonathan Berant, and Percy Liang. 2015. Building a semantic parser overnight. In ACL.
- Ronald J. Williams and Jing Peng. 1991. Function optimization using connectionist reinforcement learning algorithms. Connection Science, 3:241–268.
- Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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 833–844, Hong Kong, China. Association for Computational Linguistics.
- Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. Luke: Deep contextualized entity representations with entity-aware self-attention. In EMNLP.
- Yaosheng Yang, Wenliang Chen, Zhenghua Li, Zhengqiu He, and Min Zhang. 2018. Distantly supervised ner with partial annotation learning and reinforcement learning. In Proceedings of the 27th International Conference on Computational Linguistics, pages 2159–2169.
- David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In 33rd Annual Meeting of the Association for Computational Linguistics, pages 189–196, Cambridge, Massachusetts, USA. Association for Computational Linguistics.
- Pengcheng Yin, John Wieting, Avirup Sil, and Graham Neubig. 2022. On the ingredients of an effective zero-shot semantic parser. ArXiv, abs/2110.08381.
- Daniel Zeman and Philip Resnik. 2008. Cross-language parser adaptation between related languages. In IJCNLP.
- Yu Zhang and Qiang Yang. 2021. A survey on multi-task learning. IEEE Transactions on Knowledge and Data Engineering.
- Shang-Ming Zhou, Muhammad A. Rahman, Mark D. Atkinson, and Sinead Brophy. 2014. Mining textual data from primary healthcare records: Automatic identification of patient phenotype cohorts. 2014 International Joint Conference on Neural Networks (IJCNN), pages 3621–3627.
- Zhi-Hua Zhou and Ming Li. 2005. Tri-training: Exploiting unlabeled data using three classifiers. IEEE Transactions on knowledge and Data Engineering, 17(11):1529–1541.
- Ran Zmigrod, Tim Vieira, and Ryan Cotterell. 2021. Efficient computation of expectations under spanning tree distributions. Transactions of the Association for Computational Linguistics, 9:675–690.

Mustafa Çataltaş, Sevcan Dogramaci, Semih Yumusak, and Kasim Öztoprak. 2020. Extraction of product defects and opinions from customer reviews by using text clustering and sentiment analysis. 2020 IEEE International Conference on Big Data (Big Data), pages 4529–4534.

## Appendix A: Partially Supervised Named Entity Recognition via the Expected Entity Ratio

In this appendix, we present detailed results for our benchmark experiments in both non-native speaker and exploratory expert annotation scenarios.

### A.1 Full Benchmark Results

#### A.1.1 Non-Native Speaker

CoNLL English, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	92.1	93.2	92.7
Raw-BERT	92.1	73.7	81.9
CBL-LSTM	78.6	79.8	79.2
CBL-BERT	86.6	83.1	84.8
EER-BERT	83.7	92.8	88.0

Table A.1: CoNLL English, Non-native Speaker

CoNLL German, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	84.7	83.0	83.9
Raw-BERT	88.1	56.8	69.1
CBL-LSTM	37.3	39.7	38.4
CBL-BERT	76.9	78.2	77.5
EER-BERT	71.4	84.2	77.3

Table A.2: CoNLL German, Non-native Speaker

CoNLL Spanish, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	87.9	88.7	88.3
Raw-BERT	90.8	59.5	71.2
CBL-LSTM	57.5	51.9	54.6
CBL-BERT	81.1	76.4	78.7
EER-BERT	75.5	87.1	80.9

Table A.3: CoNLL Spanish, Non-native Speaker

CoNLL Dutch, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	90.9	91.3	91.1
Raw-BERT	91.0	57.0	70.1
CBL-LSTM	49.1	47.2	48.2
CBL-BERT	71.9	79.1	75.3
EER-BERT	69.3	86.4	76.9

Table A.4: CoNLL Dutch, Non-native Speaker

Ontonotes English, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	90.3	91.1	90.7
Raw-BERT	93.1	53.5	68.0
CBL-LSTM	69.7	66.2	67.9
CBL-BERT	80.5	72.6	76.3
EER-BERT	81.2	88.2	84.5

Table A.5: Ontonotes English, Non-native Speaker

Ontonotes Chinese, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	77.4	81.5	79.4
Raw-BERT	79.5	50.7	61.9
CBL-LSTM	52.8	54.3	53.5
CBL-BERT	71.8	66.3	68.9
EER-BERT	62.8	70.8	66.6

Table A.6: Ontonotes Chinese, Non-native Speaker

Ontonotes Arabic, Non-native Speaker			
	Precision	Recall	F1
Gold-BERT	70.7	75.4	72.9
Raw-BERT	77.1	40.2	52.8
CBL-LSTM	43.5	36.3	39.4
CBL-BERT	62.6	61.2	61.9
EER-BERT	49.4	66.4	56.6

Table A.7: Ontonotes Arabic, Non-native Speaker

### A.1.2 Exploratory Expert

CoNLL English, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	92.1	93.2	92.7
Raw-BERT-all	93.3	0.2	0.4
Raw-BERT-short	78.3	30.7	44.1
Raw-BERT-shortest	86.3	75.8	80.7
CBL-LSTM-all	72.4	51.6	60.2
CBL-LSTM-shortest	69.6	66.1	67.8
CBL-BERT-all	39.8	19.7	36.4
CBL-BERT-short	52.4	37.5	43.7
CBL-BERT-shortest	82.0	79.2	80.6
EER-BERT-all	85.6	87.1	86.3
EER-BERT-short	88.1	89.9	89.0
EER-BERT-shortest	85.0	89.8	87.3

Table A.8: CoNLL English, Exploratory Expert

CoNLL German, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	84.7	83.0	83.9
Raw-BERT-all	78.7	01.3	02.6
Raw-BERT-short	74.4	24.8	37.2
Raw-BERT-shortest	74.1	58.5	65.4
CBL-LSTM-all	24.3	31.7	27.5
CBL-LSTM-shortest	16.7	25.4	20.1
CBL-BERT-all	62.6	45.7	52.8
CBL-BERT-short	63.2	66.3	64.7
CBL-BERT-shortest	64.4	65.9	65.1
EER-BERT-all	67.7	79.6	73.2
EER-BERT-short	65.1	81.0	72.2
EER-BERT-shortest	68.5	79.6	73.6

Table A.9: CoNLL German, Exploratory Expert

CoNLL Spanish, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	87.9	88.7	88.3
Raw-BERT-all	92.3	00.3	00.7
Raw-BERT-short	80.6	30.6	44.4
Raw-BERT-shortest	80.2	67.0	73.0
CBL-LSTM-all	51.2	34.5	41.2
CBL-LSTM-shortest	45.6	30.1	36.2
CBL-BERT-all	50.9	34.1	40.9
CBL-BERT-short	59.4	53.8	56.4
CBL-BERT-shortest	74.2	75.3	74.7
EER-BERT-all	78.7	82.0	80.3
EER-BERT-short	71.7	81.9	76.5
EER-BERT-shortest	71.2	82.6	76.5

Table A.10: CoNLL Spanish, Exploratory Expert

CoNLL Dutch, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	90.9	91.3	91.1
Raw-BERT-all	0.0	0.0	0.0
Raw-BERT-short	0.0	0.0	0.0
Raw-BERT-shortest	76.8	62.7	69.1
CBL-LSTM-all	35.2	31.6	33.3
CBL-LSTM-shortest	20.9	37.3	26.7
CBL-BERT-all	56.1	49.3	52.5
CBL-BERT-short	64.5	57.4	60.8
CBL-BERT-shortest	69.5	73.0	71.2
EER-BERT-all	75.8	85.3	80.2
EER-BERT-short	75.6	85.6	80.3
EER-BERT-shortest	64.5	87.3	74.2

Table A.11: CoNLL Dutch, Exploratory Expert

Ontonotes English, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	90.3	91.1	90.7
Raw-BERT-all	79.4	0.4	0.4
Raw-BERT-short	73.8	17.6	28.4
Raw-BERT-shortest	77.0	60.1	67.5
CBL-LSTM	57.4	14.5	23.1
CBL-LSTM-shortest	45.4	39.1	42.0
CBL-BERT	40.6	15.4	22.4
CBL-BERT-short	24.6	11.8	16.0
CBL-BERT-shortest	34.9	24.0	28.4
EER-BERT	63.3	59.2	61.2
EER-BERT-short	74.9	76.9	75.9
EER-BERT-shortest	68.7	80.3	74.0

Table A.12: Ontonotes English, Exploratory Expert

Ontonotes Chinese, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	77.4	81.5	79.4
Raw-BERT-all	89.5	1.2	2.4
Raw-BERT-short	66.3	21.5	32.4
Raw-BERT-shortest	64.9	51.0	57.1
CBL-LSTM	39.1	24.2	29.9
CBL-LSTM-shortest	29.4	21.2	24.6
CBL-BERT	47.6	21.1	29.3
CBL-BERT-short	37.7	27.7	31.2
CBL-BERT-shortest	54.1	53.0	53.6
EER-BERT	57.4	55.0	56.2
EER-BERT-short	55.1	69.3	61.4
EER-BERT-shortest	59.7	69.7	64.3

Table A.13: Ontonotes Chinese, Exploratory Expert

Ontonotes Arabic, Exploratory Expert			
	Precision	Recall	F1
Gold-BERT	70.7	75.4	72.9
Raw-BERT-all	75.0	02.7	05.3
Raw-BERT-short	55.6	9.0	15.4
Raw-BERT-shortest	52.9	34.8	42.0
CBL-LSTM	19.0	12.7	15.3
CBL-LSTM-shortest	14.0	07.4	09.7
CBL-BERT	48.5	13.2	20.8
CBL-BERT-short	43.8	23.1	30.2
CBL-BERT-shortest	43.5	35.6	39.2
EER-BERT	52.3	36.2	42.9
EER-BERT-short	53.1	41.9	46.8
EER-BERT-shortest	36.5	49.6	42.0

Table A.14: Ontonotes Arabic, Exploratory Expert

## **Appendix B: Improving Low-Resource Cross-lingual Parsing with Expected Statistic Regularization**

### **B.1 Detailed Learning Curve Results**

In this appendix we present detailed numerical learning curve results corresponding to Figures 4.2 and 4.3 for completeness.

Method	$ \mathcal{D}_L^{\text{train}} $	POS				LAS			
		50	100	500	1000	50	100	500	1000
German (de)									
UDPRE		89.6	89.6	89.6	89.6	83.0	83.0	83.0	83.0
UDPRE-FT		<b>94.4</b>	<b>95.8</b>	96.9	<b>97.2</b>	87.9	89.3	<b>91.6</b>	<b>92.3</b>
UDPRE-FT-PPT		94.1	95.4	96.7	97.1	<b>88.3</b>	<b>89.4</b>	91.4	<b>92.3</b>
UDPRE-FT-ESR-UNIARC		94.2	95.6	<b>96.9</b>	<b>97.2</b>	87.4	89.0	91.3	<b>92.3</b>
UDPRE-FT-ESR-CLD		94.1	95.0	96.7	<b>97.2</b>	87.4	88.9	91.1	92.2
UDPRE-FT-ESR-CLD*		94.2	95.2	96.7	<b>97.2</b>	87.5	88.8	91.1	92.1
Indonesian (id)									
UDPRE		82.3	82.3	82.3	82.3	58.3	58.3	58.3	58.3
UDPRE-FT		89.7	90.0	92.0	92.2	71.8	74.4	77.5	78.5
UDPRE-FT-PPT		89.9	90.2	92.3	92.6	72.4	74.7	78.0	79.1
UDPRE-FT-ESR-UNIARC		<b>91.3</b>	<b>91.9</b>	92.9	<b>93.1</b>	<b>75.4</b>	76.4	<b>79.3</b>	<b>80.0</b>
UDPRE-FT-ESR-CLD		90.7	91.6	92.8	<b>93.1</b>	74.6	76.5	79.1	<b>80.0</b>
UDPRE-FT-ESR-CLD*		90.6	91.8	<b>93.0</b>	93.0	73.8	<b>76.8</b>	79.1	79.9
Persian (fa)									
UDPRE		79.1	79.1	79.1	79.1	48.4	48.4	48.4	48.4
UDPRE-FT		90.9	91.7	93.7	94.2	<b>74.9</b>	<b>77.0</b>	80.9	83.0
UDPRE-FT-PPT		90.5	<b>91.8</b>	<b>93.9</b>	94.2	74.7	76.9	<b>81.0</b>	82.7
UDPRE-FT-ESR-UNIARC		90.9	91.1	93.5	<b>94.5</b>	73.0	75.6	80.6	83.1
UDPRE-FT-ESR-CLD		90.5	91.3	93.5	94.3	72.9	75.3	80.8	82.5
UDPRE-FT-ESR-CLD*		<b>91.1</b>	91.1	93.5	<b>94.5</b>	74.3	75.9	80.8	<b>83.2</b>
Vietnamese (vi)									
UDPRE		67.0	67.0	67.0	67.0	46.3	46.3	46.3	46.3
UDPRE-FT		86.3	87.1	90.0	91.5	55.8	56.9	63.7	65.9
UDPRE-FT-PPT		86.3	87.3	90.2	<b>91.6</b>	57.1	57.7	64.0	<b>66.5</b>
UDPRE-FT-ESR-UNIARC		<b>87.9</b>	88.7	91.3	91.4	<b>58.8</b>	60.6	<b>65.4</b>	65.7
UDPRE-FT-ESR-CLD		87.0	88.7	<b>91.4</b>	91.5	58.3	60.4	65.3	65.8
UDPRE-FT-ESR-CLD*		87.0	<b>90.5</b>	<b>91.4</b>	91.4	58.5	<b>64.5</b>	65.3	65.6
Maltese (mt)									
UDPRE		35.1	35.1	35.1	35.1	16.0	16.0	16.0	16.0
UDPRE-FT		82.9	86.4	92.9	94.3	57.9	64.5	75.7	79.4
UDPRE-FT-PPT		83.8	87.6	93.2	94.3	59.3	65.6	76.1	<b>80.0</b>
UDPRE-FT-ESR-UNIARC		<b>92.2</b>	91.7	<b>93.7</b>	<b>94.5</b>	<b>75.1</b>	73.3	78.0	79.9
UDPRE-FT-ESR-CLD		91.4	<b>93.2</b>	93.4	<b>94.5</b>	73.4	<b>76.6</b>	78.1	<b>80.0</b>
UDPRE-FT-ESR-CLD*		89.7	91.7	93.5	94.4	69.7	74.3	78.2	79.8
All (avg)									
UDPRE		70.6	70.6	70.6	70.6	50.4	50.4	50.4	50.4
UDPRE-FT		88.8	90.2	93.1	93.9	69.7	72.4	77.9	79.8
UDPRE-FT-PPT		88.9	90.5	93.3	94.0	70.4	72.9	78.1	80.1
UDPRE-FT-ESR-UNIARC		<b>91.3</b>	91.8	<b>93.7</b>	<b>94.1</b>	<b>73.9</b>	75.0	<b>78.9</b>	<b>80.2</b>
UDPRE-FT-ESR-CLD		90.8	92.0	93.6	<b>94.1</b>	73.3	75.6	<b>78.9</b>	80.1
UDPRE-FT-ESR-CLD*		90.5	<b>92.1</b>	93.6	<b>94.1</b>	72.8	<b>76.0</b>	<b>78.9</b>	80.1

Table B.1: UDPRE *Semi-Supervised Learning Curves*.

Method	$ \mathcal{D}_L^{\text{train}} $	POS				LAS			
		50	100	500	1000	50	100	500	1000
German (de)									
UDPRE		89.6	89.6	89.6	89.6	83.0	83.0	83.0	83.0
MBERT-FT		87.9	<b>90.9</b>	95.1	<b>96.2</b>	59.7	66.7	81.5	<b>86.3</b>
MBERT-FT-PPT		86.3	89.6	94.7	96.0	58.0	65.3	80.9	85.8
MBERT-FT-ESR-UNIARC		88.1	90.6	<b>95.2</b>	96.1	60.6	66.4	82.6	85.6
MBERT-FT-ESR-CLD		<b>88.9</b>	<b>90.9</b>	<b>95.2</b>	95.8	<b>62.4</b>	<b>67.1</b>	<b>82.9</b>	85.0
Indonesian (id)									
UDPRE		82.3	82.3	82.3	82.3	58.3	58.3	58.3	58.3
MBERT-FT		86.5	88.3	90.8	92.1	52.1	60.6	71.8	75.2
MBERT-FT-PPT		86.6	87.9	90.9	91.8	51.7	59.5	71.7	75.2
MBERT-FT-ESR-UNIARC		<b>88.4</b>	89.4	<b>92.2</b>	<b>93.0</b>	59.3	66.2	<b>76.2</b>	<b>77.7</b>
MBERT-FT-ESR-CLD		88.3	<b>89.9</b>	<b>92.2</b>	92.9	<b>61.0</b>	<b>67.9</b>	75.7	77.4
Persian (fa)									
UDPRE		79.1	79.1	79.1	79.1	48.4	48.4	48.4	48.4
MBERT-FT		81.1	83.7	89.6	91.6	44.1	52.7	68.7	74.3
MBERT-FT-PPT		78.6	83.0	89.5	91.5	35.6	50.7	68.5	73.6
MBERT-FT-ESR-UNIARC		82.4	85.0	90.6	91.9	45.2	57.4	71.6	74.7
MBERT-FT-ESR-CLD		<b>82.8</b>	<b>85.8</b>	<b>90.7</b>	<b>92.1</b>	<b>50.3</b>	<b>59.5</b>	<b>71.8</b>	<b>75.0</b>
Vietnamese (vi)									
UDPRE		67.0	67.0	67.0	67.0	46.3	46.3	46.3	46.3
MBERT-FT		79.7	81.7	86.8	88.4	31.6	39.9	53.9	57.2
MBERT-FT-PPT		78.3	81.6	86.9	88.6	31.1	39.1	54.2	57.0
MBERT-FT-ESR-UNIARC		83.1	<b>85.7</b>	<b>88.1</b>	88.8	44.3	50.3	56.4	<b>57.9</b>
MBERT-FT-ESR-CLD		<b>83.6</b>	85.6	88.0	88.7	<b>44.7</b>	<b>50.8</b>	56.6	<b>57.9</b>
Maltese (mt)									
UDPRE		35.1	35.1	35.1	35.1	16.0	16.0	16.0	16.0
MBERT-FT		74.3	78.8	88.2	90.5	35.0	41.5	59.0	64.0
MBERT-FT-PPT		74.3	78.9	88.1	90.6	35.4	41.7	58.7	63.7
MBERT-FT-ESR-UNIARC		82.2	<b>86.4</b>	<b>90.3</b>	91.5	47.0	54.4	64.2	<b>67.3</b>
MBERT-FT-ESR-CLD		<b>82.3</b>	86.2	<b>90.3</b>	91.5	<b>47.5</b>	<b>54.7</b>	<b>64.3</b>	67.2
All (avg)									
UDPRE		70.6	70.6	70.6	70.6	50.4	50.4	50.4	50.4
MBERT-FT		81.9	84.7	90.1	91.8	44.5	52.3	67.0	71.4
MBERT-FT-PPT		80.8	84.2	90.0	91.7	42.4	51.3	66.8	71.0
MBERT-FT-ESR-UNIARC		84.8	87.4	<b>91.3</b>	<b>92.3</b>	51.3	58.9	<b>70.2</b>	<b>72.6</b>
MBERT-FT-ESR-CLD		<b>85.2</b>	<b>87.7</b>	<b>91.3</b>	92.2	<b>53.2</b>	<b>60.0</b>	<b>70.2</b>	72.5

Table B.2: mBERT *Semi-Supervised Learning Curves*.