Generalizing Natural Language Analysis through Span-relation Representations

Zhengbao Jiang[1], **Wei Xu**[2], **Jun Araki**[3], **Graham Neubig**[1]
Language Technologies Institute, Carnegie Mellon University[1]
Department of Computer Science and Engineering, Ohio State University[2]
Bosch Research North America[3]
{zhengbaj,gneubig}@cs.cmu.edu[1]
xu.1265@osu.edu[2], jun.araki@us.bosch.com[3]

Abstract

Natural language processing covers a wide variety of tasks predicting syntax, semantics, and information content, and usually each type of output is generated with specially designed architectures. In this paper, we provide the simple insight that a great variety of tasks can be represented in a single unified format consisting of labeling spans and relations between spans, thus a single task-independent model can be used across different tasks. We perform extensive experiments to test this insight on 10 disparate tasks spanning dependency parsing (syntax), semantic role labeling (semantics), relation extraction (information content), aspect based sentiment analysis (sentiment), and many others, achieving performance comparable to state-of-the-art specialized models. We further demonstrate benefits of multi-task learning, and also show that the proposed method makes it easy to analyze differences and similarities in how the model handles different tasks. Finally, we convert these datasets into a unified format to build a benchmark, which provides a holistic testbed for evaluating future models for generalized natural language analysis.

1 Introduction

A large number of natural language processing (NLP) tasks exist to analyze various aspects of human language, including syntax (e.g., constituency and dependency parsing), semantics (e.g., semantic role labeling), information content (e.g., named entity recognition and relation extraction), or sentiment (e.g., sentiment analysis). At first glance, these tasks are seemingly very different in both the structure of their output and the variety of information that they try to capture. To handle these different characteristics, researchers usually use specially designed neural network architectures. In this paper we ask the simple questions: are the task-specific architectures really necessary? Or with the appropriate representational methodology, can we devise *a single model that can perform — and achieve state-of-the-art performance on — a large number of natural language analysis tasks*?

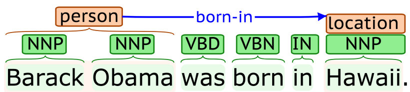


Figure 1: An example from BRAT, consisting of POS, NER, and RE.

Interestingly, in the domain of *efficient human annotation interfaces*, it is already standard to use unified representations for a wide variety of NLP tasks. Figure 1 shows one example of the BRAT (Stenetorp et al., 2012) annotation interface, which has been used for annotating data for tasks as broad as part-of-speech tagging, named entity recognition, relation extraction, and many others. Notably, this interface has a single unified format that consists of spans (e.g., the span of an entity), labels on the spans (e.g., the variety of entity such as "person" or "location"), and labeled relations between the spans (e.g., "born-in"). These labeled relations can form a tree or a graph structure, expressing the linguistic structure of sentences (e.g., dependency tree). We detail this BRAT format and how it can be used to represent a wide number of natural language analysis tasks in Section 2.

The simple hypothesis behind our paper is: *if humans can perform natural language analysis in a single unified format, then perhaps machines can as well*. Fortunately, there already exist NLP models that perform span prediction and prediction of relations between pairs of spans, such as the end-to-end coreference model of Lee et al. (2017). We extend this model with minor architectural modifications (which are *not* our core contributions) and pre-trained contextualized representations (e.g.,

	Information Extraction				POS	Parsing		SRL	Sentiment	
	NER	RE	Coref.	OpenIE		Dep.	Consti.		ABSA	ORL
Different Models for Different Tasks										
ELMo (Peters et al., 2018)	✓	✗	✓	✗	✗	✗	✗	✗	✓	✗
BERT (Devlin et al., 2019)	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗
SpanBERT (Joshi et al., 2019)	✗	✓	✓	✗	✗	✗	✗	✗	✗	✗
Single Model for Different Tasks										
Guo et al. (2016)	✗	✓	✗	✗	✗	✗	✗	✓	✗	✗
Swayamdipta et al. (2018)	✗	✗	✓	✗	✗	✗	✓	✓	✗	✗
Strubell et al. (2018)	✗	✗	✗	✗	✓	✓	✗	✓	✗	✗
Clark et al. (2018)	✓	✗	✗	✗	✓	✓	✗	✗	✗	✗
Luan et al. (2018, 2019)	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗
Dixit and Al-Onaizan (2019)	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗
Marasović and Frank (2018)	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓
Hashimoto et al. (2017)	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗
This Work	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: A comparison of the tasks covered by previous work and our work.

BERT; Devlin et al. (2019)[1]) then demonstrate the applicability and versatility of this single model on 10 tasks, including named entity recognition (NER), relation extraction (RE), coreference resolution (Coref.), open information extraction (OpenIE), part-of-speech tagging (POS), dependency parsing (Dep.), constituency parsing (Consti.), semantic role labeling (SRL), aspect based sentiment analysis (ABSA), and opinion role labeling (ORL). While previous work has used similar formalisms to *understand* the representations learned by pre-trained embeddings (Tenney et al., 2019a,b), to the best of our knowledge this is the first work that uses such a unified model to actually *perform analysis*. Moreover, we demonstrate that despite the model's simplicity, it can achieve comparable performance with special-purpose state-of-the-art models on the tasks above (Table 1). We also demonstrate that this framework allows us to easily perform multi-task learning (MTL), leading to improvements when there are related tasks to be learned from or data is sparse. Further analysis shows that dissimilar tasks exhibit divergent attention patterns, which explains why MTL is harmful on certain tasks. We have released our code and the **G**eneral **L**anguage **A**nalysis **D**atasets (GLAD) benchmark with 8 datasets covering 10 tasks in the BRAT format at `https://github.com/neulab/cmu-multinlp`, and provide a leaderboard to facilitate future work on generalized models for NLP.

[1]In contrast to work on pre-trained contextualized representations like ELMo (Peters et al., 2018) or BERT (Devlin et al., 2019) that learn unified *features* to represent the *input* in different tasks, we propose a unified *representational methodology* that represents the *output* of different tasks. Analysis models using BERT still use special-purpose output predictors for specific tasks or task classes.

2 Span-relation Representations

In this section, we explain how the BRAT format can be used to represent a large number of tasks. There are two fundamental types of annotations: span annotations and relation annotations. Given a sentence $\mathbf{x} = [w_1, w_2, ..., w_n]$ of n tokens, a span annotation (s_i, l_i) consists of a contiguous span of tokens $s_i = [w_{b_i}, w_{b_i+1}, ..., w_{e_i}]$ and its label l_i ($l_i \in \mathcal{L}$), where b_i/e_i are the start/end indices respectively, and $\mathcal{L}$ is a set of span labels. A relation annotation (s_j, s_k, r_{jk}) refers to a relation r_{jk} ($r_{jk} \in \mathcal{R}$) between the head span s_j and the tail span s_k, where $\mathcal{R}$ is a set of relation types. This span-relation representation can easily express many tasks by defining $\mathcal{L}$ and $\mathcal{R}$ accordingly, as summarized in Table 2a and Table 2b. These tasks fall in two categories: **span-oriented tasks**, where the goal is to predict labeled spans (e.g., named entities in NER) and **relation-oriented tasks**, where the goal is to predict relations between two spans (e.g., relation between two entities in RE). For example, constituency parsing (Collins, 1997) is a span-oriented task aiming to produce a syntactic parse tree for a sentence, where each node of the tree is an individual span associated with a constituent label. Coreference resolution (Pradhan et al., 2012) is a relation-oriented task that links an expression to its mentions within or beyond a single sentence. Dependency parsing (Kübler et al.,

Task	Spans annotated with labels
NER	Barack Obama was born in Hawaii. (person; location)
Consti.	And their suspicions of each other run deep. (NP; NP; ADVP; PP; VP; NP; S)
POS	What kind of memory? (WP NN IN NN)
ABSA	Great laptop that offers many great features! (positive)

Table 2a: Span-oriented tasks. Spans are annotated by underlines and their labels.

Task	Spans and relations annotated with labels
RE	The burst has been caused by pressure. (cause-effect)
Coref.	I voted for Tom because he is clever. (coref.)
SRL	We brought you the tale of two cities. (ARG0; ARG2; ARG1)
OpenIE	The four lawyers climbed out from under a table. (ARG0; ARG1)
Dep.	The entire division employs about 850 workers. (det; amod; nsubj; dobj; advmod; nummod)
ORL	We therefore as MDC do not accept this result. (holder; target)

Table 2b: Relation-oriented tasks. Directed arcs indicate the relations between spans.

2009) is also a relation-oriented task that aims to relate a word (single-word span) to its syntactic parent word with the corresponding dependency type. Detailed explanations of all tasks can be found in Appendix A.

While the tasks above represent a remarkably broad swath of NLP, it is worth mentioning what we have *not* covered, to properly scope this work. Notably, sentence-level tasks such as text classification and natural language inference are not covered, although they can also be formulated using this span-relation representation by treating the entire sentence as a span. We chose to omit these tasks because they are already well-represented by previous work on generalized architectures (Lan and Xu, 2018) and multi-task learning (Devlin et al., 2019; Liu et al., 2019), and thus we mainly focus on tasks using phrase-like spans. In addition, the span-relation representations described here are designed for natural language *analysis*, and cannot handle tasks that require *generation* of text, such as machine translation (Bojar et al., 2014), dialog response generation (Lowe et al., 2015), and summarization (Nallapati et al., 2016). There are also a small number of analysis tasks such as semantic parsing to logical forms (Banarescu et al., 2013) where the outputs are not directly associated with spans in the input, and handling these tasks is beyond the scope of this work.

3 Span-relation Model

Now that it is clear that a very large number of analysis tasks can be formulated in a single format, we turn to devising a single model that can solve these tasks. We base our model on a span-based model first designed for end-to-end coreference resolution (Lee et al., 2017), which is then adapted for other tasks (He et al., 2018; Luan et al., 2018, 2019; Dixit and Al-Onaizan, 2019; Zhang and Zhao, 2019). At the core of the model is a module to represent each span as a fixed-length vector, which is used to predict labels for spans or span pairs. We first briefly describe the span representation used and proven to be effective in previous works, then highlight some details we introduce to make this model generalize to a wide variety of tasks.

Span Representation Given a sentence $\mathbf{x} = [w_1, w_2, ..., w_n]$ of n tokens, a span $s_i = [w_{b_i}, w_{b_i+1}, ..., w_{e_i}]$ is represented by concatenating two components: a *content representation* $\mathbf{z}_i^c$ calculated as the weighted average across all token embeddings in the span, and a *boundary representation* $\mathbf{z}_i^u$ that concatenates the embeddings at the start and end positions of the span. Specifically,

$$\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_n = \text{TokenRepr}(w_1, w_2, ..., w_n), \quad (1)$$

$$\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_n = \text{BiLSTM}(\mathbf{c}_1, \mathbf{c}_2, ..., \mathbf{c}_n), \quad (2)$$

$$\mathbf{z}_i^c = \text{SelfAttn}(\mathbf{c}_{b_i}, \mathbf{c}_{b_i+1}, ..., \mathbf{c}_{e_i}), \quad (3)$$

$$\mathbf{z}_i^u = [\mathbf{u}_{b_i}; \mathbf{u}_{e_i}], \mathbf{z}_i = [\mathbf{z}_i^c; \mathbf{z}_i^u], \quad (4)$$

where TokenRepr could be non-contextualized, such as GloVe (Pennington et al., 2014), or contextualized, such as BERT (Devlin et al., 2019). We refer to Lee et al. (2017) for further details.

Span and Relation Label Prediction Since we extract spans and relations in an *end-to-end* fashion, we introduce two additional labels `NEG_SPAN` and `NEG_REL` in $\mathcal{L}$ and $\mathcal{R}$ respectively. `NEG_SPAN` indicates invalid spans (e.g., spans that are not named entities in NER) and `NEG_REL` indicates invalid span pairs without any relation between them (i.e., no relation exists between two arguments in SRL).

Dataset	Domain	#Sent.	Task	#Spans	#Relations	Metric
Wet Lab Protocols (Kulkarni et al., 2018)	biology	14,301	NER RE	60,745 60,745	- 43,773	F_1 F_1
CoNLL-2003 (Sang and Meulder, 2003)	news	20,744	NER	35,089	-	F_1
SemEval-2010 Task 8 (Hendrickx et al., 2010)	misc.	10,717	RE	21,437	10,717	Macro F_1 $^\circ$
OntoNotes 5.0 * (Pradhan et al., 2013)	misc.	94,268	Coref. SRL POS Dep. Consti.	194,477 745,796 1,631,995 1,722,571 1,320,702	1,166,513 543,534 - 1,628,558 -	Avg F_1 F_1 Accuracy LAS Evalb F_1 †
Penn Treebank (Marcus et al., 1994)	speech, news	49,208 43,948 43,948	POS Dep. Consti.	1,173,766 1,090,777 871,264	- 1,046,829 -	Accuracy LAS Evalb F_1 †
OIE2016 (Stanovsky and Dagan, 2016)	news, Wiki	2,534	OpenIE	15,717	12,451	F_1
MPQA 3.0 (Deng and Wiebe, 2015)	news	3,585	ORL	13,841	9,286	F_1
SemEval-2014 Task 4 (Pontiki et al., 2014)	reviews	4,451	ABSA	7,674	-	Accuracy $^\circ$

Table 3: Statistics of GLAD, consisting of 10 tasks from 8 datasets. * Following He et al. (2018), we use a subset of OntoNotes 5.0 dataset based on CoNLL 2012 splits (Pradhan et al., 2012). $^\circ$ Previous works use gold standard spans in these evaluations. † We use the bracket scoring program Evalb (Collins, 1997) in constituency parsing.

We first predict labels for *all* spans up to a length of l words using a multilayer perceptron (MLP): $\mathrm{softmax}(\text{MLP}^{\text{span}}(\mathbf{z}_i)) \in \Delta^{|\mathcal{L}|}$, where $\Delta^{|\mathcal{L}|}$ is a $|\mathcal{L}|$-dimensional simplex. Then we keep the top $K = \tau \cdot n$ spans with the lowest NEG_SPAN probability in relation prediction for efficiency, where smaller pruning threshold τ indicates more aggressive pruning. Another MLP is applied to pairs of the remaining spans to produce their relation scores: $\mathbf{o}_{jk} = \text{MLP}^{\text{rel}}([\mathbf{z}_j; \mathbf{z}_k; \mathbf{z}_j \cdot \mathbf{z}_k]) \in \mathbb{R}^{|\mathcal{R}|}$, where j and k index two spans.

Application to Disparate Tasks For most of the tasks, we can simply maximize the probability of the ground truth relation for *all pairs of the remaining spans*. However, some tasks might have different requirements, e.g., coreference resolution aims to cluster spans referring to the same concept and we do not care about which antecedent a span is linked to if there are multiple ones. Thus, we provide two training loss functions:

1. **Pairwise** Maximize the probabilities of the ground truth relations for all pairs of the remaining spans independently: $\mathrm{softmax}(\mathbf{o}_{jk})_{r_{jk}}$, where r_{jk} indexes the ground truth relation.

2. **Head** Maximize the probability of ground truth head spans for a specific span s_j: $\sum_{k\in\text{head}(s_j)} \mathrm{softmax}([o_{j1}, o_{j2}, ..., o_{jK}])_k$, where $\text{head}(\cdot)$ returns indices of one or more heads and $o_{j\cdot}$ is the corresponding scalar from $\mathbf{o}_{j\cdot}$ indicating how likely two spans are related.

We use option 1 for all tasks except for coreference resolution which uses option 2. Note that the above loss functions *only* differ in how relation scores are normalized and the other parts of the model remain the same across different tasks. At test time, we follow previous inference methods to generate valid outputs. For coreference resolution, we link a span to the antecedent with highest score (Lee et al., 2017). For constituency parsing, we use greedy top-down decoding to generate a valid parse tree (Stern et al., 2017). For dependency parsing, each word is linked to exactly one parent with the highest relation probability. For other tasks, we predict relations for all span pairs and use those not predicted as NEG_REL to construct outputs.

Our core insight is that the above formulation is largely *task-agnostic*, meaning that a task can be modeled in this framework as long as it can be formulated as a span-relation prediction problem with properly defined span labels $\mathcal{L}$ and relation labels $\mathcal{R}$. As shown in Table 1, this unified **Span-Rel**ation (SpanRel) model makes it simple to scale to a large number of language analysis tasks, with breadth far beyond that of previous work.

Multi-task Learning The SpanRel model makes it easy to perform multi-task learning (MTL) by sharing all parameters except for the MLPs used for label prediction. However, because different tasks capture different linguistic aspects, they are not equally beneficial to each other. It is expected that jointly training on related tasks is helpful, while forcing the same model to solve unrelated tasks

Category	Task	Metric	Dataset	Setting	SOTA Model	Previous SOTA	Our Model
IE	NER	F_1	CoNLL03	BERT	Devlin et al. (2019)	92.8	92.2
			WLP	ELMo	Luan et al. (2019)	79.5	79.2
	RE	Macro F1	SemEval10	BERT, gold	Wu and He (2019)	89.3	87.4
		F_1	WLP	ELMo	Luan et al. (2019)	64.1	65.5
	Coref.	Avg F_1	OntoNotes	GloVe, CharCNN	Lee et al. (2017)$^{\circ}$	62.0	61.1
	OpenIE	F_1	OIE2016	ELMo	Stanovsky et al. (2018)*	31.1	35.2
SRL		F_1	OntoNotes	ELMo	He et al. (2018)†	82.9	82.4
Parsing	Dep.	LAS	PTB	ELMo	Clark et al. (2018)	94.4	94.7
	Consti.	Evalb F_1	PTB	BERT	Kitaev et al. (2019)	95.6	95.5
Sentiment	ABSA	Accuracy	SemEval14	BERT, gold	Xu et al. (2019)$^{\triangleleft}$	85.0/78.1	85.5/76.6
	ORL	F_1	MPQA 3.0	GloVe, gold	Marasović and Frank (2018)*	56.4	55.6
POS		Accuracy	PTB	ELMo	Clark et al. (2018)	97.7	97.7

Table 4: Comparison between SpanRel models and task-specific SOTA models.[2] Following Luan et al. (2019), we perform NER and RE jointly on WLP dataset. We use gold entities in SemEval-2010 Task 8, gold aspect terms in SemEval-2014 Task 4, and gold opinion expressions in MPQA 3.0 to be consistent with existing works.

might even hurt the performance (Ruder, 2017). Compared to manually choosing source tasks based on prior knowledge, which might be sub-optimal when the number of tasks is large, SpanRel offers a systematic way to examine relative benefits of source-target task pairs by either performing pairwise MTL or attention-based analysis, as we will show in Section 4.3.

4 GLAD Benchmark and Results

We first describe our **G**eneral **L**anguage **A**nalysis **D**atasets (GLAD) benchmark and evaluation metrics, then conduct experiments to (1) verify that SpanRel can achieve comparable performance across all tasks (Section 4.2), and (2) demonstrate its benefits in multi-task learning (Section 4.3).

4.1 Experimental Settings

GLAD Benchmark and Evaluation Metrics As summarized in Table 3, we convert 8 widely used datasets with annotations of 10 tasks into the BRAT format and include them in the GLAD benchmark. It covers diverse domains, providing a holistic testbed for natural language analysis evaluation. The major evaluation metric is span-based F_1 (denoted as F_1), a standard metric for SRL. Precision is the proportion of extracted spans (spans not predicted as NEG_SPAN) that are consistent with the ground truth. Recall is the proportion of ground truth spans that are correctly extracted. Span F_1 is also applicable to relations, where an extracted relation (relations not predicted as NEG_REL) is correct iff both head and tail spans have correct boundaries and the predicted relation is correct. To make fair comparisons with existing works, we also compute standard metrics for different tasks, as listed in Table 3.

Implementation Details We attempted four token representation methods (Equation 1), namely GloVe (Pennington et al., 2014), ELMo (Peters et al., 2018), BERT (Devlin et al., 2019), and SpanBERT (Joshi et al., 2019). We use $\text{BERT}_{\text{base}}$ in our main results and report $\text{BERT}_{\text{large}}$ in Appendix B. A three-layer BiLSTM with 256 hidden units is used (Equation 2). Both span and relation prediction MLPs have two layers with 128 hidden units. Dropout (Srivastava et al., 2014) of 0.5 is applied to all layers. For GloVe and ELMo, we use Adam (Kingma and Ba, 2015) with learning rate of 1e-3 and early stop with patience of 3. For BERT and SpanBERT, we follow standard fine-tuning with learning rate of 5e-5, $\beta_1 = 0.9$, $\beta_2 = 0.999$, L2 weight decay of 0.01, warmup over the first 10% steps, and number of epochs tuned on development set. Task-specific hyperparameters maximal span length and pruning ratio are tuned on development set and listed in Appendix C.

4.2 Comparison with Task-specific SOTA

We compare the SpanRel model with state-of-the-art task-specific models by training on data from a

[2] $^{\circ}$ The small version of Lee et al. (2017)'s method with 100 antecedents and no speaker features. * For OpenIE and ORL, we use span-based F_1 instead of syntactic-head-based F_1 and binary coverage F_1 used in the original papers because they are biased towards extracting long spans. † For SRL, we choose to compare with He et al. (2018) because they also extract predicates and arguments in an end-to-end way. $^{\triangleleft}$ We follow Xu et al. (2019) to report accuracy of restaurant and laptop domain separately in ABSA.

single task. By doing so we attempt to answer the research question "can a single model with minimal task-specific engineering achieve competitive or superior performance to other models that have been specifically engineered?" We select competitive SOTA models mainly based on settings, e.g., single-task learning and end-to-end extraction of spans and relations. To make fair comparisons, token embeddings (GloVe, ELMo, BERT) and other hyperparameters (e.g., the number of antecedents in Coref. and the maximal span length in SRL) in our method are set to match those used by SOTA models, to focus on differences brought about by the model architecture.

As shown in Table 4, the SpanRel model achieves comparable performances as task-specific SOTA methods (regardless of whether the token representation is contextualized or not). This indicates that the span-relation format can generically represent a large number of natural language analysis tasks and it is possible to devise a single unified model that achieves strong performance on all of them. It provides a strong and generic baseline for natural language analysis tasks and a way to examine the usefulness of task-specific designs.

4.3 Multi-task Learning with SpanRel

To demonstrate the benefit of the SpanRel model in MTL, we perform single-task learning (STL) and MTL across all tasks using end-to-end settings.[3] Following Liu et al. (2019), we perform MTL+fine-tuning and show the results in separate columns of Table 5. Contextualized token representations yield significantly better results than GloVe on all tasks, indicating that pre-training on large corpora is almost universally helpful to NLP tasks. Comparing the results of MTL+fine-tuning with STL, we found that performance with GloVe drops on 8 out of 15 tasks, most of which are tasks with relatively sparse data. It is probably because the capacity of the GloVe-based model is too small to store all the patterns required by different tasks. The results of contextualized representations are mixed, with some tasks being improved and others remaining the same or degrading. We hypothesize that this is because different tasks capture different linguistic aspects, thus are not equally helpful to each other. Reconciling these seemingly different tasks in the same model might be harmful to some tasks. Notably, as the contextualized representations become stronger, the performance of MTL+FT becomes more favorable. 5 out of 15 tasks (NER, RE, OpenIE, SRL, ORL) observe statistically significant improvements (p-value < 0.05 with paired bootstrap re-sampling) with SpanBERT, a contextualized embedding pre-trained with span-based training objectives, while only one task degrades (ABSA), indicating its superiority in reconciling spans from different tasks. The GLAD benchmark provides a holistic testbed for evaluating natural language analysis capability.

Task Relatedness Analysis To further investigate how different tasks interact with each other, we choose five source tasks (i.e., tasks used to improve other tasks, e.g., POS, NER, Consti., Dep., and SRL) that have been widely used in MTL (Hashimoto et al., 2017; Strubell et al., 2018) and six target tasks (i.e., tasks to be improved, e.g., OpenIE, NER, RE, ABSA, ORL, and SRL) to perform pairwise multi-task learning.

We hypothesize that although language modeling pre-training is theoretically orthogonal to MTL (Swayamdipta et al., 2018), in practice their benefits tends to overlap. To analyze these two factors separately, we start with a weak representation GloVe to study task relatedness, then move to BERT to demonstrate how much we can still improve with MTL given strong and contextualized representations. As shown in Table 6 (GloVe), tasks are not equally useful to each other. Notably, (1) for OpenIE and ORL, multi-task learning with SRL improves the performance significantly, while other tasks lead to less or no improvements. (2) Dependency parsing and SRL are generic source tasks that are beneficial to most of the target tasks. This unified SpanRel makes it easy to perform MTL and decide beneficial source tasks.

Next, we demonstrate that our framework also provides a platform for analysis of similarities and differences between different tasks. Inspired by the intuition that the attention coefficients are somewhat indicative of a model's internal focus (Li et al., 2016; Vig, 2019; Clark et al., 2019), we hypothesize that the similarity or difference between attention mechanisms may be correlated with similarity between tasks, or even the success or failure of MTL. To test this hypothesis, we extract the attention maps of two BERT-based SpanRel models (trained on a source t' and a target task t separately)

[3] Span-based F_1 is used as the evaluation metric in SemEval-2010 Task 8 and SemEval-2014 Task 4 as opposed to macro F_1 and accuracy reported in the original papers because we aim at end-to-end extractions.

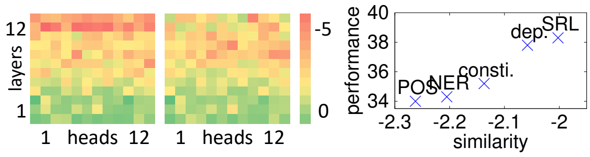
				GloVe			ELMo			$\text{BERT}_{\text{base}}$			$\text{SpanBERT}_{\text{base}}$		
Category	**Task**	**Metric**	**Dataset**	**STL**	**MTL**	**+FT**	**STL**	**MTL**	**+FT**	**STL**	**MTL**	**+FT**	**STL**	**MTL**	**+FT**
IE	NER	F_1	CoNLL03	88.4	86.2↓	87.5↓	91.9	91.6	91.6	91.0	88.6↓	90.2↓	91.3	90.4↓	91.2
			WLP	77.6	71.5↓	76.5↓	79.2	77.4↓	78.2↓	78.1	78.2	78.5	77.9	**78.6**↑	**78.5**↑
	RE	F_1	SemEval10	50.7	15.2↓	33.0↓	61.8	30.6↓	42.9↓	61.7	55.1↓	59.8↓	62.1	54.6↓	61.8
			WLP	64.9	38.5↓	53.9↓	65.5	52.0↓	55.1↓	64.7	**65.9**↑	**66.5**↑	64.1	**67.2**↑	**67.2**↑
	Coref	Avg F_1	OntoNotes	56.3	50.3↓	53.0↓	62.2	**62.9**↑	**63.3**↑	66.2	65.5↓	65.8	70.0	68.9↓	69.7
	OpenIE	F_1	OIE2016	28.3	6.8↓	19.6↓	35.2	30.0↓	32.9↓	36.7	37.1	**38.5**↑	36.5	**37.3**↑	**38.6**↑
SRL		F_1	OntoNotes	78.0	77.9	**78.6**↑	82.4	82.3	82.4	83.3	82.9	83.4	83.1	83.3	**83.8**↑
Parsing	Dep.	LAS	PTB	92.9	93.2	**93.5**↑	94.7	94.9	94.9	94.9	94.8	95.0	95.1	95.1	95.1
			OntoNotes	90.4	90.5	90.5	92.3	**93.2**↑	**92.8**↑	94.1	93.8	94.0	94.2	94.1	94.2
	Consti.	Evalb F_1	PTB	93.4	-	93.8	95.3	-	95.3	95.5	-	95.2	95.8	-	95.5
			OntoNotes	91.0	-	**91.5**↑	93.2	-	**93.7**↑	93.6	-	93.8	94.3	-	94.2
Sentiment	ABSA	F_1	SemEval14	63.5	48.5↓	59.0↓	69.2	57.0↓	59.0↓	70.8	63.1↓	67.0↓	70.0	63.5↓	69.5↓
	ORL	F_1	MPQA 3.0	38.2	18.4↓	31.6↓	42.9	24.7↓	32.4↓	44.5	38.1↓	**45.6**↑	45.2	40.2↓	**47.5**↑
POS		Accuracy	PTB	96.8	96.8	96.8	97.7	97.7	97.8	97.6	97.3	97.3	97.6	97.6	97.6
			OntoNotes	97.0	97.0	97.1	98.2	98.2	98.3	97.7	97.8	97.8	98.3	98.3	98.3

Table 5: Comparison between STL and MTL+fine-tuning across all tasks. **blue**↑ indicates results better than STL, red↓ indicates worse, and black means almost the same (i.e., a difference within 0.5). Constituency parsing requires more memory than other tasks so we restrict its span length to 10 in MTL, and thus do not report results.

over sentences $\mathcal{X}_t$ from the target task, and compute their similarity using the Frobenius norm:

$$\text{sim}_k(t, t') = -\frac{1}{|\mathcal{X}_t|} \sum_{\mathbf{x} \in \mathcal{X}_t} \left\| A_k^t(\mathbf{x}) - A_k^{t'}(\mathbf{x}) \right\|_F,$$

where $A_k^t(\mathbf{x})$ is the attention map extracted from the k-th head by running the model trained from task t on sentence $\mathbf{x}$. We select OpenIE as the target task because it shows the largest performance variation when paired with different source tasks (34.0 - 38.8) in Table 6. We visualize the attention similarity of all heads in BERT (12 layers $\times$ 12 heads) between two mutually harmful tasks (OpenIE/POS on the left) and between two mutually helpful tasks (OpenIE/SRL on the right) in Figure 2a. A common trend is that heads in higher layers exhibit more divergence, probably because they are closer to the prediction layer, thus easier to be affected by the end task. Overall, it can be seen that OpenIE/POS has much more attention divergence than OpenIE/SRL. A notable difference is that almost *all* heads in the last two layers of the OpenIE/POS models differ significantly, while *some* heads in the last two layers of the OpenIE/SRL models still behave similarly, providing evidence that failure of MTL can be attributed to the fact that dissimilar tasks requires different attention patterns. We further compute average attention similarities for all source tasks in Figure 2b, and we can see that there is a strong correlation (Pearson correlation



(a) Attention similarity between OpenIE/POS (left), and between OpenIE/SRL (right) for all heads.

(b) Correlation between attention similarity and MTL performance.

Figure 2: Attention-based task relatedness analysis.

of 0.97) between the attentions similarity and the performance of pairwise MTL, supporting our hypothesis that attention pattern similarities can be used to predict improvements of MTL.

MTL under Different Settings We analyze how token representations and sizes of the target dataset affect the performance of MTL. Comparing BERT and GloVe in Table 6, the improvements become smaller or vanish as the token representation becomes stronger, e.g., improvement on OpenIE with SRL reduces from 5.8 to 1.6. This is expected because both large-scale pre-training and MTL aim to learn general representations and their benefits tend to overlap in practice. Interestingly, some helpful source tasks become harmful when we shift from GloVe to BERT, such as OpenIE paired with POS. We conjecture that the gains of MTL might have already been achieved by BERT, but the task-specific characteristics of POS hurt the performance of OpenIE. We did not observe many tasks benefitting

Target \ Source	GloVe STL	GloVe POS	GloVe NER	GloVe Consti.	GloVe Dep.	GloVe SRL	BERT$_{base}$ STL	BERT$_{base}$ POS	BERT$_{base}$ NER	BERT$_{base}$ Consti.	BERT$_{base}$ Dep.	BERT$_{base}$ SRL
OpenIE	28.3	**29.9**↑	27.0↓	**31.2**↑	**32.9**↑	**34.1**↑	36.7	34.0↓	34.3↓	35.2↓	**37.8**↑	**38.3**↑
NER (WLP)	77.6	77.8	**78.3**↑	77.9	**78.6**↑	**78.1**↑	78.1	78.0	78.1	78.1	77.7	**78.8**↑
RE (WLP)	64.9	**65.5**↑	**65.6**↑	64.9	**66.5**↑	**65.9**↑	64.7	64.4	64.7	64.3	64.9	**65.3**↑
RE (SemEval10)	50.7	**52.3**↑	**52.8**↑	49.6↓	**52.9**↑	**52.8**↑	61.7	61.9	60.2↓	59.2↓	62.1	59.9↓
ABSA	63.5	63.4	62.8↓	59.8↓	63.5	60.2↓	70.8	68.9↓	**71.4**↑	70.4	69.9↓	69.6↓
ORL	38.2	35.7↓	37.9	36.1↓	38.6	**41.0**↑	44.5	**45.8**↑	44.2	44.8	**45.1**↑	**46.6**↑
SRL (10k)	68.8	**69.6**↑	68.9	**70.7**↑	**71.3**↑	-	78.7	**79.4**↑	**79.5**↑	**79.6**↑	**79.8**↑	-

Table 6: Performance of pairwise multi-task learning with GloVe and BERT$_{base}$. **blue**↑ indicates results better than STL, red↓ indicates worse, and black means almost the same (i.e., a difference within 0.5). We show the performance after fine-tuning. Dataset of source tasks POS, Consti., Dep. is PTB and dataset of NER is CoNLL-2003.

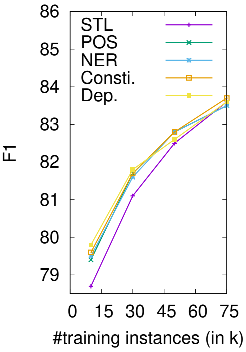


Figure 3: MTL Performance of SRL wrt. the data size.

from MTL for the GloVe-based model in Table 5 because it is trained on *all* tasks (instead of *two*), which is beyond its limited model capacity. The improvements of MTL shrink as the size of the SRL datasets increases, as shown in Figure 3, indicating that MTL is useful when the target data is sparse.

Time Complexity Analysis Time complexities of span and relation prediction are $\mathcal{O}(l \cdot n)$ and $\mathcal{O}(K^2) = \mathcal{O}(\tau^2 \cdot n^2)$ respectively for a sentence of n tokens (Section 3). The time complexity of BERT is $\mathcal{O}(L \cdot n^2)$, dominated by its L self-attention layers. Since the pruning threshold τ is usually less than 1, the computational overhead introduced by the span-relation output layer is much less than BERT. In practice, we observe that the training/testing time is mainly spent by BERT. For SRL, one of the most computation-intensive tasks with long spans and dense span/relation annotations, 85.5% of the time is spent by BERT. For POS, a less heavy task, the time spent by BERT increases to 98.5%. Another option for span prediction is to formulate it as a sequence labeling task, as in previous works (Lample et al., 2016; He et al., 2017), where time complexity is $\mathcal{O}(n)$. Although slower than token-based labeling models, span-based models offer the advantages of being able to model overlapping spans and use span-level information for label prediction (Lee et al., 2017).

5 Related Work

General Architectures for NLP There has been a rising interest in developing general architectures for different NLP tasks, with the most prominent examples being sequence labeling framework (Collobert et al., 2011; Ma and Hovy, 2016) used for tagging tasks and sequence-to-sequence framework (Sutskever et al., 2014) used for generation tasks. Moreover, researchers typically pick related tasks, motivated by either linguistic insights or empirical results, and create a general framework to perform MTL, several of which are summarized in Table 1. For example, Swayamdipta et al. (2018) and Strubell et al. (2018) use constituency and dependency parsing to improve SRL. Luan et al. (2018, 2019); Wadden et al. (2019) use a span-based model to jointly solve three information-extraction-related tasks (NER, RE, and Coref.). Li et al. (2019) formulate both nested NER and flat NER as a machine reading comprehension task. Compared to existing works, we aim to create an output representation that can solve *nearly every* natural language analysis task in one fell swoop, allowing us to cover a far broader range of tasks with a single model.

In addition, NLP has seen a recent burgeoning of contextualized representations pre-trained on large corpora (e.g., ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019)). These methods focus on learning generic *input* representations, but are agnostic to the *output* representation, requiring different predictors for different tasks. In contrast, we present a methodology to formulate the output of different tasks in a unified format. Thus our work is orthogonal to those on contextualized embeddings. Indeed, in Section 4.3, we demonstrate that the SpanRel model can benefit from stronger contextualized representation models, and even provide a testbed for their use in natural language analysis.

Benchmarks for Evaluating Natural Language Understanding Due to the rapid development of NLP models, large-scale benchmarks, such as SentEval (Conneau and Kiela, 2018), GLUE (Wang et al., 2019b), and SuperGLUE (Wang et al., 2019a) have been proposed to facilitate fast and holistic evaluation of models' understanding ability. They mainly focus on sentence-level tasks, such as natural language inference, while our GLAD benchmark focuses on token/phrase-level analysis tasks with diverse coverage of different linguistic structures. New tasks and datasets can be conveniently added to our benchmark as long as they are in the BRAT standoff format, which is one of the most commonly used data format in the NLP community, e.g., it has been used in the BioNLP shared tasks (Kim et al., 2009) and the Universal Dependency project (McDonald et al., 2013).

6 Conclusion

We provide the simple insight that a large number of natural language analysis tasks can be represented in a single format consisting of spans and relations between spans. As a result, these tasks can be solved in a single modeling framework that first extracts spans and predicts their labels, then predicts relations between spans. We attempted 10 tasks with this SpanRel model and show that this generic task-independent model can achieve competitive performance as state-of-the-art methods tailored for each tasks. We merge 8 datasets into our GLAD benchmark for evaluating future models for natural language analysis. Future directions include (1) devising hierarchical span representations that can handle spans of different length and diverse content more effectively and efficiently; (2) robust multitask learning or meta-learning algorithms that can reconcile very different tasks.

Acknowledgments

This work was supported by gifts from Bosch Research. We would like to thank Hiroaki Hayashi, Bohan Li, Pengcheng Yin, Hao Zhu, Paul Michel, and Antonios Anastasopoulos for their insightful comments and suggestions.

References

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, LAW-ID@ACL 2013, August 8-9, 2013, Sofia, Bulgaria*, pages 178–186.

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007*, pages 2670–2676.

Ondrej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Ales Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation, WMT@ACL 2014, June 26-27, 2014, Baltimore, Maryland, USA*, pages 12–58.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1925, Brussels, Belgium. Association for Computational Linguistics.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pages 16–23, Madrid, Spain. Association for Computational Linguistics.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel P. Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.*, 12:2493–2537.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, Miyazaki, Japan. European Languages Resources Association (ELRA).

Lingjia Deng and Janyce Wiebe. 2015. MPQA 3.0: An entity/event-level sentiment corpus. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1323–1328.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Kalpit Dixit and Yaser Al-Onaizan. 2019. Span-level model for relation extraction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5308–5314, Florence, Italy. Association for Computational Linguistics.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Comput. Linguist.*, 28(3):245–288.

Jiang Guo, Wanxiang Che, Haifeng Wang, Ting Liu, and Jun Xu. 2016. A unified architecture for semantic role labeling and relation classification. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1264–1274, Osaka, Japan. The COLING 2016 Organizing Committee.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple NLP tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1923–1933.

Luheng He, Kenton Lee, Omer Levy, and Luke Zettlemoyer. 2018. Jointly predicting predicates and arguments in neural semantic role labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 364–369, Melbourne, Australia. Association for Computational Linguistics.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 473–483, Vancouver, Canada. Association for Computational Linguistics.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. Semeval-2010 task 8: Multi-way classification of semantic relations between pairs of nominals. In *Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval@ACL 2010, Uppsala University, Uppsala, Sweden, July 15-16, 2010*, pages 33–38.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2019. Spanbert: Improving pre-training by representing and predicting spans. *CoRR*, abs/1907.10529.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. 2009. Overview of bionlp'09 shared task on event extraction. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task*, BioNLP '09, pages 1–9, Stroudsburg, PA, USA. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Nikita Kitaev, Steven Cao, and Dan Klein. 2019. Multilingual constituency parsing with self-attention and pre-training. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3499–3505, Florence, Italy. Association for Computational Linguistics.

Sandra Kübler, Ryan McDonald, and Joakim Nivre. 2009. Dependency parsing. *Synthesis Lectures on Human Language Technologies*, 1(1):1–127.

Chaitanya Kulkarni, Wei Xu, Alan Ritter, and Raghu Machiraju. 2018. An annotated corpus for machine reading of instructions in wet lab protocols. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 97–106, New Orleans, Louisiana. Association for Computational Linguistics.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 260–270.

Wuwei Lan and Wei Xu. 2018. Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3890–3902, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 188–197.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. Understanding neural networks through representation erasure. *CoRR*, abs/1612.08220.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2019. A unified MRC framework for named entity recognition. *CoRR*, abs/1910.11476.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4487–4496.

Ryan Lowe, Nissan Pow, Iulian Serban, and Joelle Pineau. 2015. The ubuntu dialogue corpus: A large dataset for research in unstructured multi-turn dialogue systems. In *Proceedings of the SIGDIAL 2015 Conference, The 16th Annual Meeting of the Special Interest Group on Discourse and Dialogue, 2-4 September 2015, Prague, Czech Republic*, pages 285–294.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3219–3232.

Yi Luan, Dave Wadden, Luheng He, Amy Shah, Mari Ostendorf, and Hannaneh Hajishirzi. 2019. A general framework for information extraction using dynamic span graphs. 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 3036–3046, Minneapolis, Minnesota. Association for Computational Linguistics.

Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Ana Marasović and Anette Frank. 2018. SRL4ORL: Improving opinion role labeling using multi-task learning with semantic role labeling. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 583–594, New Orleans, Louisiana. Association for Computational Linguistics.

Mitchell P. Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The penn treebank: Annotating predicate argument structure. In *Human Language Technology, Proceedings of a Workshop held at Plainsboro, New Jerey, USA, March 8-11, 1994*.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 280–290.

Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. 2018. A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. Semeval-2014 task 4: Aspect based sentiment analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014.*, pages 27–35.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Hwee Tou Ng, Anders Björkelund, Olga Uryupina, Yuchen Zhang, and Zhi Zhong. 2013. Towards robust linguistic analysis using ontonotes. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, August 8-9, 2013*, pages 143–152.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. 2012. Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning - Proceedings of the Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes, EMNLP-CoNLL 2012, July 13, 2012, Jeju Island, Korea*, pages 1–40.

Adwait Ratnaparkhi. 1996. A maximum entropy model for part-of-speech tagging. In *Conference* *on Empirical Methods in Natural Language Processing*.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *CoRR*, abs/1706.05098.

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, CoNLL 2003, Held in cooperation with HLT-NAACL 2003, Edmonton, Canada, May 31 - June 1, 2003*, pages 142–147.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, pages 1929–1958.

Gabriel Stanovsky and Ido Dagan. 2016. Creating a large benchmark for open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2300–2305, Austin, Texas. Association for Computational Linguistics.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 885–895, New Orleans, Louisiana. Association for Computational Linguistics.

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 at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 102–107, Avignon, France. Association for Computational Linguistics.

Mitchell Stern, Jacob Andreas, and Dan Klein. 2017. A minimal span-based neural constituency parser. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 818–827.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5027–5038, Brussels, Belgium. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Swabha Swayamdipta, Sam Thomson, Kenton Lee, Luke Zettlemoyer, Chris Dyer, and Noah A. Smith. 2018. Syntactic scaffolds for semantic structures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3772–3782, Brussels, Belgium. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4593–4601.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R. Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R. Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Kristina Toutanova, Dan Klein, Christopher D. Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 252–259.

Jesse Vig. 2019. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28 - August 2, 2019, Volume 3: System Demonstrations*, pages 37–42.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. 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 5783–5788, Hong Kong, China. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019a. Superglue: A stickier benchmark for general-purpose language understanding systems. *CoRR*, abs/1905.00537.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Shanchan Wu and Yifan He. 2019. Enriching pre-trained language model with entity information for relation classification. *CoRR*, abs/1905.08284.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT post-training for review reading comprehension and aspect-based sentiment analysis. 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 2324–2335, Minneapolis, Minnesota. Association for Computational Linguistics.

Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers*, pages 1640–1649.

Junlang Zhang and Hai Zhao. 2019. Span based open information extraction. *CoRR*, abs/1901.10879.

A Detailed Explanations of 10 Tasks

- **Span-oriented Tasks (Table 2a)**
  - **Named Entity Recognition** (Sang and Meulder, 2003) NER is traditionally considered as a sequence labeling task. We model named entities as spans over one or more tokens.
  - **Constituency Parsing** (Collins, 1997) Constituency parsing aims to produce a syntactic parse tree for each sentence. Each node in the tree is an individual span associated with a constituent label, and spans are nested.
  - **Part-of-speech Tagging** (Ratnaparkhi, 1996; Toutanova et al., 2003) POS tagging is another sequence labeling task, where every single token is an individual span with a POS tag.
  - **Aspect-based Sentiment Analysis** (Pontiki et al., 2014) ABSA is a task that consists of identifying certain spans as aspect terms and predicting their associated sentiments.

- **Relation-oriented Tasks (Table 2b)**
  - **Relation Extraction** (Hendrickx et al., 2010) RE concerns the relation between two entities.
  - **Coreference** (Pradhan et al., 2012) Coreference resolution is to link named, nominal, and pronominal mentions that refer to the same concept, within or beyond a single sentence.
  - **Semantic Role Labeling** (Gildea and Jurafsky, 2002) SRL aims to identify arguments of a predicate (verb or noun) and classify them with semantic roles in relation to the predicate.
  - **Open Information Extraction** (Banko et al., 2007; Niklaus et al., 2018) In contrast to the fixed relation types in RE, OpenIE aims to extract open-domain predicates and their arguments (usually subjects and objects) from a sentence.
  - **Dependency Parsing** (Kübler et al., 2009) Spans are single-word tokens and a relation links a word to its syntactic parent with the corresponding dependency type.
  - **Opinion Role Labeling** (Yang and Cardie, 2013) ORL detects spans that are opinion expressions, as well as holders and targets related to these opinions.

B Results of BERT Large Model

Table 7 shows the performance of single-task learning with different token representations. $\text{BERT}_{\text{large}}$ achieves the best performance on most of the tasks.

Category	Task	Metric	Dataset	GloVe	ELMo	$\text{BERT}_{\text{base}}$	$\text{SpanBERT}_{\text{base}}$	$\text{BERT}_{\text{large}}$
IE	NER	F_1	CoNLL03	88.4	91.9	91.0	91.3	90.9
			WLP	77.6	79.2	78.1	77.9	78.3
	RE	F_1	SemEval10	50.7	61.8	61.7	62.1	64.7
			WLP	64.9	65.5	64.7	64.1	65.1
	Coref	Avg F_1	OntoNotes	56.3	62.2	66.3	70.0	-
	OpenIE	F_1	OIE2016	28.3	35.2	36.7	36.5	36.5
SRL		F_1	OntoNotes	78.0	82.4	83.3	83.1	84.4
Parsing	Dep.	LAS	PTB	92.9	94.7	94.9	95.1	95.3
			OntoNotes	90.4	92.3	94.1	94.2	94.5
	Consti.	Evalb F_1	PTB	93.4	95.3	95.5	95.8	95.8
			OntoNotes	91.0	93.2	93.6	94.3	93.9
Sentiment	ABSA	F_1	SemEval14	63.5	69.2	70.8	70.0	73.8
	ORL	F_1	MPQA 3.0	38.2	42.9	44.5	45.2	47.1
POS		Accuracy	PTB	96.8	97.7	97.6	97.6	97.4
			OntoNotes	97.0	98.2	97.7	98.3	97.9

Table 7: Single-task learning performance of the SpanRel model with different token representations. $\text{BERT}_{\text{large}}$ requires a large amount of memory so we cannot feed the entire document to the model in coreference resolution.

	Information Extraction				POS	Parsing		SRL	Sentiment	
	NER	RE	Coref.	OpenIE		Dep.	Consti.		ABSA	ORL
max span length l	10	5	10	30	1	1	-	30	10	30
pruning ratio τ	-	5	0.4	0.8	-	1.0	-	1.0	-	0.3

Table 8: Task-specific hyperparameters. Span-oriented tasks do not need pruning ratio.

C Task-specific Hyperparameters

As shown in Table 8, a larger maximum span length is used for tasks with longer spans (e.g., OpenIE), and a larger pruning ratio is used for tasks with more spans (e.g., SRL). Constituency parsing does not have span length limit because spans can be as long as the entire sentence. Since relation extraction aims to extract exactly two entities and their relation from a sentence, we keep pruning ratio fixed (top 5 spans in this case) regardless of the length of the sentence.