Asking without Telling: Exploring Latent Ontologies in Contextual Representations

Julian Michael[1]* **Jan A. Botha**[2] **and Ian Tenney**[2]

[1]Paul G. Allen School of Computer Science & Engineering, University of Washington
[2]Google Research
julianjm@cs.washington.edu
{jabot,iftenney}@google.com

Abstract

The success of pretrained contextual encoders, such as ELMo and BERT, has brought a great deal of interest in what these models learn: do they, without explicit supervision, learn to encode meaningful notions of linguistic structure? If so, how is this structure encoded? To investigate this, we introduce *latent subclass learning* (LSL): a modification to classifier-based probing that induces a latent categorization (or *ontology*) of the probe's inputs. Without access to fine-grained gold labels, LSL extracts *emergent* structure from input representations in an interpretable and quantifiable form. In experiments, we find strong evidence of familiar categories, such as a notion of personhood in ELMo, as well as novel ontological distinctions, such as a preference for fine-grained semantic roles on core arguments. Our results provide unique new evidence of emergent structure in pretrained encoders, including departures from existing annotations which are inaccessible to earlier methods.

1 Introduction

The success of self-supervised pretrained models in NLP (Devlin et al., 2019; Peters et al., 2018a; Radford et al., 2019; Lan et al., 2020) on many tasks (Wang et al., 2018, 2019b) has stimulated interest in how these models work, and what they learn about language. Recent work on model analysis (Belinkov and Glass, 2019) indicates that they may learn a lot about linguistic structure, including part of speech (Belinkov et al., 2017a), syntax (Blevins et al., 2018; Marvin and Linzen, 2018), word sense (Peters et al., 2018a; Reif et al., 2019), and more (Rogers et al., 2020).

Many of these results are based on *predictive methods*, such as probing, which measure how well a linguistic variable can be predicted from intermediate representations. However, the ability of supervised probes to fit weak features makes it difficult to produce unbiased answers about how those representations are structured (Saphra and Lopez, 2019; Voita et al., 2019). *Descriptive methods* like clustering and visualization explore this structure directly, but provide limited control and often regress to dominant categories such as lexical features (Singh et al., 2019) or word sense (Reif et al., 2019). This leaves open many questions: *how* are linguistic features like entity types, syntactic dependencies, or semantic roles represented by an encoder like ELMo (Peters et al., 2018a) or BERT (Devlin et al., 2019)? To what extent do familiar categories like PropBank roles or Universal Dependencies appear naturally? Do these unsupervised encoders learn their own categorization of language?

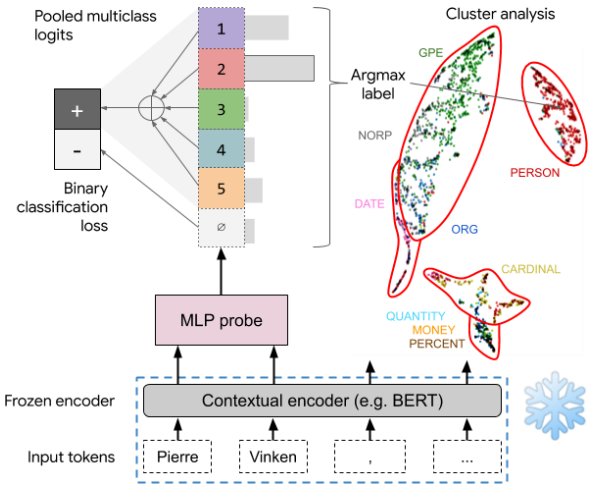


Figure 1: LSL overview. A probing classifier over contextual embeddings produces multi-class *latent logits*, which are marginalized into a single logit trained on binary classification. In this example, "Pierre Vinken" is identified as a named entity and assigned to latent class 2, which aligns well with the PERSON label. We treat the classes as clusters representing a latent ontology that describes the underlying representation space. Figure 2 visualizes latent logits in more detail.

*Work performed while at Google.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 6792–6812,
November 16–20, 2020. ©2020 Association for Computational Linguistics

To tackle these questions, we propose a systematic way to extract *latent ontologies*, or discrete categorizations of a representation space, which we call *latent subclass learning* (LSL); see Figure 1 for an overview. In LSL, we use a binary classification task (such as detecting entity mentions or syntactic dependency arcs) as weak supervision to induce a set of latent clusters relevant to that task (*i.e.*, entity or dependency types). As with *predictive* methods, the choice of task allows us to explore varied phenomena, and induced clusters can be quantified and compared to gold annotations. But also, as with *descriptive* methods, our clusters can be inspected and qualified directly, and observations have high specificity: agreement with external (*e.g.*, gold) categories provides strong evidence that those categories are salient in the representation space.

We describe the LSL classifier in Section 3, and apply it to the edge probing paradigm (Tenney et al., 2019b) in Section 4. In Section 5 we evaluate LSL on multiple encoders, including ELMo and BERT. We find that LSL induces stable and consistent ontologies, which include both striking rediscoveries of gold categories—for example, ELMo discovers *personhood* of named entities and BERT has a notion of *dates*—and novel ontological distinctions—such as fine-grained core argument semantic roles—which are not easily observed by fully supervised probes. Overall, we find unique new evidence of emergent latent structure in our encoders, while also revealing new properties of their representations which are inaccessible to earlier methods.

2 Background

Predictive analysis A common form of model analysis is *predictive*: assessing how well a linguistic variable can be predicted from a model, whether in intrinsic behavioral tests (Goldberg, 2019; Marvin and Linzen, 2018; Petroni et al., 2019) or extrinsic *probing tasks*.

Probing involves training lightweight classifiers over features produced by a pretrained model, and assessing the model's knowledge by the probe's performance. Probing has been used for low-level properties such as word order and sentence length (Adi et al., 2017; Conneau et al., 2018), as well as phenomena at the level of syntax (Hewitt and Manning, 2019), semantics (Tenney et al., 2019b; Liu et al., 2019b; Clark et al., 2019), and discourse structure (Chen et al., 2019). Error analysis on probes has been used to argue that BERT may simulate sequential decision making across layers (Tenney et al., 2019a), or that it encodes its own, soft notion of syntactic distance (Reif et al., 2019).

Predictive methods such as probing are *flexible*: Any task with data can be assessed. However, they only track predictability of pre-defined categories, limiting their descriptive power. In addition, a powerful enough probe, given enough data, may be insensitive to differences between encoders, making it difficult to interpret results based on accuracy (Saphra and Lopez, 2019; Zhang and Bowman, 2018). So, many probing experiments appeal to the *ease of extraction* of a linguistic variable (Pimentel et al., 2020). Existing work has measured this by controlling for probing model capacity, either using relative claims between layers and encoders (Belinkov et al., 2017b; Blevins et al., 2018; Tenney et al., 2019b; Liu et al., 2019a) or using explicit measures to estimate and trade off capacity with accuracy (Hewitt and Liang, 2019; Voita and Titov, 2020). An alternative is to control *amount of supervision*, by restricting training set size (Zhang and Bowman, 2018), comparing learning curves (Talmor et al., 2019), or using description length with online coding (Voita and Titov, 2020).

We extend this further by removing the distinction between gold categories in the training data and reducing the supervision to binary classification, as explained in Section 3. This extreme measure makes our test *high specificity*, in the sense that positive results—*i.e.*, when comprehensible categories are recovered by our probe—are much stronger, since a category must be essentially *invented* without direct supervision.

Descriptive analysis In contrast to predictive methods, which assess an encoder with respect to particular data, *descriptive* methods analyze models on their own terms, and include clustering, visualization (Reif et al., 2019), and correlation analysis techniques (Voita et al., 2019; Saphra and Lopez, 2019; Abnar et al., 2019; Chrupała and Alishahi, 2019). Descriptive methods produce high-specificity tests of what structure is present in the model, and facilitate discovery of new patterns that were not hypothesized prior to testing. However, they lack the flexibility of predictive methods. Clustering results tend to be dominated by principal components of the embedding space, which correspond to only some salient aspects of linguistic knowledge, such as lexical features (Singh et al., 2019) and word sense (Reif et al., 2019). Alterna-

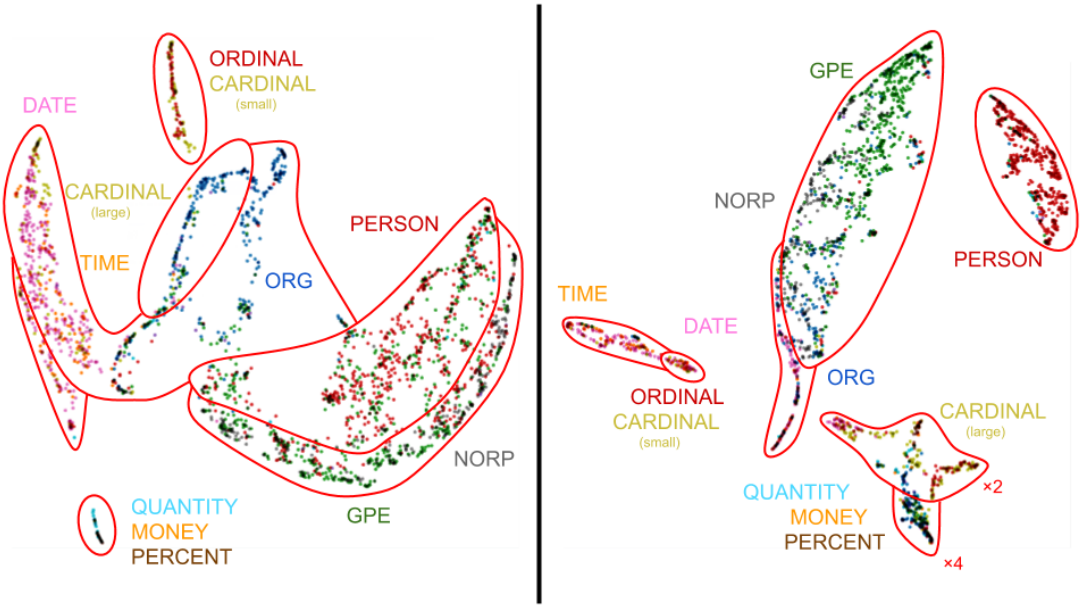


Figure 2: Latent logit vectors from BERT (left) and ELMo (right) for a sample from the Named Entities development set visualized in the Embedding Projector (Smilkov et al., 2016) using UMAP (McInnes et al., 2018), which is designed to preserve local clustering structure in a low dimensional visualization. Points are colored by gold label, and induced clusters are outlined in red. ELMo has a clear notion of personhood (PERSON), while BERT groups people with geopolitical entities (GPE) and nationalities (NORP). BERT strongly identifies dates (DATE) and organizations (ORG), and both models group numeric/quantitative entities together. Both models separate small CARDINAL numbers (roughly, seven or less) and group them with ORDINALs, separate from larger CARDINALs. The outlined areas in the bottom-right of the ELMo visualization include 2 and 4 induced clusters.

tively, more targeted analysis techniques generally have a restricted inventory of inputs, such as layer mixing weights (Peters et al., 2018b), transformer attention distributions (Clark et al., 2019), or pairwise influence between tokens (Wu et al., 2020). As a result of these issues, it is more difficult to discover the underlying structure corresponding to rich, layered ontologies. Our approach retains the advantages of descriptive methods, while admitting more control as the choice of binary classification targets can guide the LSL model to discover structure relevant to a particular linguistic task.

Linguistic ontologies Questions of what encoders learn about language require well-defined *linguistic ontologies*, or meaningful categorizations of inputs, to evaluate against. Most analysis work uses formalisms from the classical NLP pipeline, such as part-of-speech and syntax from the Penn Treebank (Marcus et al., 1993) or Universal Dependencies (Nivre et al., 2015), semantic roles from PropBank (Palmer et al., 2005) or Dowty (1991)'s Proto-Roles (Reisinger et al., 2015), and named entities, which have a variety of available ontologies (Pradhan et al., 2007; Ling and Weld, 2012; Choi et al., 2018). Work on ontology-free, or *open*, representations suggests that the linguistic structure captured by traditional ontologies may be encoded in a variety of possible ways (Banko et al., 2007; He et al., 2015; Michael et al., 2018) while being annotatable at large scale (FitzGerald et al., 2018). This raises the question: when looking for linguistic knowledge in pretrained encoders, what exactly should we expect to find? Predictive methods are useful for fitting an encoder to an existing ontology; but do our encoders latently hold their own ontologies as well? If so, what do they look like? That is the question we investigate in this work.

3 Approach

We propose a way to extract latent linguistic ontologies from pretrained encoders and systematically compare them to existing gold ontologies. We use a classifier based on *latent subclass learning* (Section 3.1), which is applicable in any binary classification setting.[1] We propose several quantitative metrics to evaluate the induced ontologies (Section 3.2), providing a starting point for qualitative analysis (Section 5) and future research.

[1] A similar classifier was concurrently developed and presented for use in model distillation by Müller et al. (2020).

3.1 Latent Subclass Learning

Consider a logistic regression classifier over inputs $\mathbf{x} \in \mathbb{R}^d$. It outputs probabilities according to the following formula:

$$\mathrm{P}(y \mid \mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x}),$$

where $\mathbf{w} \in \mathbb{R}^d$ is a learned parameter. Instead, we propose the *latent subclass learning* classifier:

$$\mathrm{P}_{\mathrm{LSL}}(y \mid \mathbf{x}) = \sigma\left(\log \sum_i^N e^{\mathbf{W}_i \mathbf{x}}\right),$$

where $\mathbf{W} \in \mathbb{R}^{N \times d}$ is a parameter matrix, and N is a hyperparameter corresponding to the number of latent classes.

This corresponds to N+1-way multiclass logistic regression with a fixed 0 baseline for a null class, but trained on binary classification by marginalizing over the N non-null classes (Figure 1). The vector $\mathbf{W}\mathbf{x} \in \mathbb{R}^N$ may then be treated as a set of *latent logits* for a random variable $C(\mathbf{x}) \in \{1, \ldots, N\}$ defined by the softmax distribution. Taking the hard maximum of $\mathbf{W}\mathbf{x}$ assigns a latent class $\hat{C}(\mathbf{x})$ to each input, which may be viewed as a *weakly supervised clustering*, learned on the basis of external supervision but not explicitly optimized to match prior gold categories.

For the loss $\mathcal{L}_{\mathrm{LSL}}$, we use the cross-entropy loss on $\mathrm{P}_{\mathrm{LSL}}$. However, this does not necessarily encourage a diverse, coherent set of clusters; an LSL classifier may simply choose to collapse all examples into a single category, producing an uninteresting ontology. To mitigate this, we propose two *clustering regularizers*.

Adjusted batch-level negative entropy We wish for the model to induce a diverse ontology. One way to express this is that the expectation of C has high entropy, *i.e.*, we wish to maximize

$$\mathrm{H}(\mathbb{E}_{\mathbf{x}}\, C(\mathbf{x})).$$

In practice, we use the expectation over a batch. The maximum value this can take is the entropy of the uniform distribution over N items, or $\log N$. Therefore, we wish to minimize the *adjusted batch-level negative entropy loss*:

$$\mathcal{L}_{\mathrm{be}} = \log N - \mathrm{H}(\mathbb{E}_{\mathbf{x}}\, C(\mathbf{x})),$$

which takes values in $[0, \log N]$.

Instance-level entropy In addition to using all latent classes in the expected case, we also wish for the model to assign a single coherent class label to each input example. This can be done by minimizing the *instance-level entropy loss*:

$$\mathcal{L}_{\mathrm{ie}} = \mathbb{E}_{\mathbf{x}}\, \mathrm{H}(C(\mathbf{x})).$$

This also takes values in $[0, \log N]$, and we compute the expectation over a batch.

Loss We optimize the regularized LSL loss

$$\mathcal{L}_{\mathrm{LSL}} + \alpha \mathcal{L}_{\mathrm{be}} + \beta \mathcal{L}_{\mathrm{ie}},$$

where α and β are hyperparameters, via gradient descent. Together, the regularizers encourage a balanced solution where the model uses many clusters yet gives each input a distinct assignment. Note that if $\alpha = \beta$, the this objective maximizes the mutual information between $\mathbf{x}$ and C, encouraging the ontology to encode as much information as possible about the training data while still supporting the binary classification objective.

3.2 Metrics

Since our interest is in descriptively analyzing encoders' latent ontologies, there are no normatively 'correct' categories. However, we can leverage existing gold ontologies—such as PropBank role labels or Universal Dependencies—to quantify our results in terms of well-understood categories. For the following metrics, we consider only points in the gold positive class.

B^3 B-cubed (or B^3) is a standard clustering metric (Bagga and Baldwin, 1998; Amigó et al., 2009) which calculates the precision and recall of each point's predicted cluster against its gold cluster, averaging over points. It allows for label-wise scoring by restricting to points with specific gold labels, allowing for fine-grained analysis, *e.g.*, of whether a gold label is concentrated in few predicted clusters (high recall) or well-separated from other labels (high precision).

Normalized PMI Pointwise mutual information (PMI) is commonly used as an association measure reflecting how likely two items (such as tokens in a corpus) are to occur together relative to chance (Church and Hanks, 1989). *Normalized* PMI (nPMI; Bouma, 2009) is a way of factoring out the effect of item frequency on PMI. Formally,

the nPMI of two items x and y is

$$\left(\log \frac{\mathrm{P}(x,y)}{\mathrm{P}(x)\,\mathrm{P}(y)}\right) \Big/ -\log(\mathrm{P}(x,y)) \ ,$$

taking the limit value of -1 when they never occur together, 1 when they only occur together, and 0 when they occur independently. We use nPMI to analyze the co-occurrence of *gold labels* in *predicted clusters*: A pair of gold labels with high nPMI are preferentially grouped together by the induced ontology, whereas two labels with low nPMI are preferentially distinguished.

Plotting pairwise nPMI of gold labels allows us to see specific ways the induced clustering agrees or disagrees with a gold reference (Section 5, Figure 3). Since nPMI is information-theoretic and chance-corrected, it is a reliable indicator of the degree of information about gold labels contained in a set of predicted clusters. However, it is relatively insensitive to cluster granularity (*e.g.*, the total number of predicted categories, or whether a single gold category is split into many different predicted clusters), which is better understood through our other metrics.

Diversity We desire fine-grained ontologies with many meaningful classes. Number of attested classes may not be a good measure of this, since it could include classes with very few members and no broad meaning. So we propose *diversity*:

$$\exp(\mathrm{H}(\mathbb{E}_{\mathbf{x}}\,\hat{C}(\mathbf{x}))).$$

This increases as the clustering becomes more fine-grained and evenly distributed, with a maximum of N when $\mathrm{P}(\hat{C})$ is uniform. More generally, exponentiated entropy is sometimes referred to as the *perplexity* of a distribution, and corresponds (softly) to the number of classes required for a uniform distribution of the same entropy. In that sense, it may be regarded as the effective number of classes in an ontology. We use the predicted class $\hat{C}$ rather than its distribution C because we care about the diversity of the model's clustering, and not just uncertainty in the model.

Uncertainty In order for our learned classes to be meaningful, we desire distinct and coherent clusters. To measure this, we propose *uncertainty*:

$$\mathbb{E}_{\mathbf{x}} \exp(\mathrm{H}(C(\mathbf{x}))).$$

This is also related to perplexity, but unlike diversity, it takes the expectation over the input after calculating the perplexity of the distribution. This reflects how many classes, on average, the model is confused between when provided with an input. Low values correspond to coherent clusters, with a minimum of 1 when every latent class is assigned with full confidence. As with diversity, we take the expectation over the evaluation set.

4 Experimental Setup

We adopt a similar setup to Tenney et al. (2019b) and Liu et al. (2019a), training probing models over several contextualizing encoders on a variety of linguistic tasks.

4.1 Tasks

We cast several structure labeling tasks from Tenney et al. (2019b) as binary classification by adding negative examples, bringing the positive to negative ratio to 1:1 where possible.

Named entity labeling requires labeling noun phrases with entity types, such as person, location, date, or time. We randomly sample non-entity noun phrases as negatives.

Nonterminal labeling requires labeling phrase structure constituents with syntactic types, such as noun phrases and verb phrases. We randomly sample non-constituent spans as negatives.

Syntactic dependency labeling requires labeling token pairs with their syntactic relationship, such as a subject, direct object, or modifier. We randomly sample non-attached token pairs as negatives.

Semantic role labeling requires labeling predicates (usually verbs) and their arguments (usually syntactic constituents) with labels that abstract over syntactic relationships in favor of more semantic notions such as *agent*, *patient*, modifier roles involving, *e.g.*, time and place, or predicate-specific roles. We draw the closest non-attached predicate-argument pairs as negatives.

We use the English Web Treebank part of Universal Dependencies 2.2 (Silveira et al., 2014) for dependencies, and the English portion of Ontonotes 5.0 (Weischedel et al., 2013) for other tasks.

4.2 Encoders

We run experiments on the following encoders:

ELMo (Peters et al., 2018a) is the concatenation of representations from 2-layer LSTMs (Hochreiter and Schmidhuber, 1997) trained with forward and

	Named Entities				Universal Dependencies			
	P / R / F1	**Acc.**	**Div↑**	**Unc↓**	**P / R / F1**	**Acc.**	**Div↑**	**Unc↓**
Gold	1.0 / 1.0 / 1.0	1.0	9.71	1.00	1.0 / 1.0 / 1.0	1.0	22.91	1.00
Multi	.86 / .88 / .87	.94	8.58	1.88	.86 / .83 / .84	.93	21.94	1.77
LSL	.28 / .80 / .41	.96	2.85	1.45	.10 / .60 / .18	.94	3.50	2.07
+be	.20 / .43 / .27	.96	4.78	31.23	.18 / .13 / .15	.94	**29.83**	12.33
+ie	.13 / 1.0 / .23	.93	1.00	**1.00**	.09 / .79 / .15	.94	2.00	**1.01**
+be +ie	.43 / .54 / **.48**	.88	**7.00**	1.10	.18 / .27 / **.22**	.86	14.96	1.35
Single	.13 / 1.0 / .23	-	1.00	1.00	.06 / 1.0 / .11	-	1.00	1.00

Table 1: Model selection results over BERT-large. **Multi** is the standard multi-class model trained directly on gold labels, and **Single** is the degenerate single-cluster baseline. Our clustering regularizers (**b**atch and/or **i**nstance-level **e**ntropy), when taken together, yield a good tradeoff between **div**ersity and **unc**ertainty, though at some expense to binary classification **acc**uracy.

backward language modeling objectives. We use the publicly available instance[2] trained on the One Billion Word Benchmark (Chelba et al., 2014).

BERT (Devlin et al., 2019) is a deep Transformer stack (Vaswani et al., 2017) trained on masked language modeling and next sentence prediction tasks. We use the 24-layer BERT-large instance[3] trained on about 2.3B tokens from English Wikipedia and BooksCorpus (Zhu et al., 2015).

BERT-lex is a lexical baseline, using only BERT's context-independent wordpiece embedding layer.

4.3 Probing Model

We reimplement the model of Tenney et al. (2019b),[4] which gives a unified architecture that works for a wide range of probing tasks. Specifically, it classifies single spans or pairs of spans in the following way: 1) construct token representations by pooling across encoder layers with a learned scalar mix (Peters et al., 2018a), 2) construct span representations from these token representations using self-attentive pooling (Lee et al., 2017), and 3) concatenate those span representations and feed the result through a fully-connected layer to produce input features for the classification layer. We follow Tenney et al. (2019b) in training a probing model over a frozen encoder, while using our LSL classifier (Section 3) as the final output layer in place of the usual softmax.

[2] tfhub.dev/google/elmo/2

[3] github.com/google-research/bert

[4] Publicly available at https://jiant.info

4.4 Model selection

We run initial studies to determine hidden layer sizes and regularization coefficients. For all LSL probes, we use $N = 32$ latent classes.[5]

Probe capacity To mitigate the influence of probe capacity on the results, we follow the best practice recommended by Hewitt and Liang (2019) and use a single hidden layer with the smallest size that does not sacrifice performance. For each task, we train binary logistic regression probes with a range of hidden sizes and select the smallest yielding at least 97% of the best model's performance. Details are in Appendix A.

Mitigating variance To decrease variance across random restarts, we use a consistency-based model selection criterion: train 5 models, compute their pairwise B^3 F1 scores, and choose the one with the highest average F1. (However, as we find in Section 5, the qualitative patterns that emerged were consistent between runs.)

Regularization coefficients We run preliminary experiments using BERT-large on Universal Dependencies and Named Entity Labeling with ablations on our clustering regularizers. For each ablation, we choose hyperparameters with the best F1 against gold.

Results Results are shown in Table 1. As expected, the batch-level entropy loss drives up both diversity and uncertainty, while the instance-level entropy loss drives them down. In combination,

[5] Preliminary experiments found similar results for larger N, with similar diversity in the full setting.

	BERT-lex		ELMo		BERT-large		Gold
Task	P / R / F1	Div	P / R / F1	Div	P / R / F1	Div	Div
Dependencies	.06 / .86 / .11	1.33	.23 / .42 / **.29**	11.11	.14 / .33 / .19	**11.22**	22.91
Named Entities	.19 / .39 / .26	4.33	.40 / .66 / **.50**	5.07	.47 / .53 / **.50**	**7.50**	9.71
Nonterminals	.22 / .80 / .34	1.47	.36 / .25 / .30	**10.16**	.35 / .34 / **.35**	7.80	7.15
Semantic Roles	.19 / .39 / **.26**	2.81	.40 / .17 / .24	**22.35**	.37 / .17 / .24	18.70	8.73

Table 2: Results by task for three pretrained encoding methods. All probing models were trained with the LSL loss and cluster regularization coefficients $\alpha = \beta = 1.5$, and chosen by the best-of-5 consistency criterion and detailed in Section 4.4. Uncertainty for all models was close to 1 and is omitted for space.

however, they produce the right balance, with uncertainty near 1 while retaining diversity.

Notably, the Named Entity model with the batch-level loss has *higher* diversity when the instance-level loss is added. This happens because batch-level entropy can be increased by driving up instance-level entropy without changing the entropy of the expected distribution of predictions $\mathrm{H}(\mathbb{E}_{\mathbf{x}}\,\mathrm{P}(\hat{C}(\mathbf{x})))$. So by keeping the uncertainty down on each input, the instance-level entropy loss helps the batch-level entropy loss promote diversity in the induced ontology.

Based on these results, we set $\alpha = \beta = 1.5$ for $\mathcal{L}_{be}$ and $\mathcal{L}_{ie}$ for the main experiments.

5 Results and Analysis

Table 2 shows aggregate results for the tasks and encoders described in Section 4.[6] Taking all metrics into account, contextualized encodings produce richer ontologies that agree more with gold than the lexical baseline does. In fact, BERT-lex has normalized PMI scores very close to zero across the board (plots are provided in Appendix C), encoding virtually no information about gold categories. For this reason, we omit it from the rest of the analysis.

Named entities As shown in Table 3, neither BERT nor ELMo are sensitive to categories that are related to specialized world knowledge, such as languages, laws, and events. However, they are in tune with other types: ELMo discovers a clear PERSON category, whereas BERT has distinguished DATEs. Visualization of the clusters (Figure 2) corroborates this, furthermore showing that the models have a sense of scalar values and measurement; indeed, instead of the gold distinction between ORDINAL and CARDINAL numbers, both models distinguish between *small* and *large* (roughly, seven or greater) numbers. See Appendix C for detailed nPMI scores.

	DATE	PERCENT	ORG	PERSON	⋯	EVENT	LAW	LANG.
BERT	.70	.60	.54	.48	·	.03	.02	.01
ELMo	.38	.28	.35	.81	·	.02	.01	.01

Table 3: Label-wise B^3 F1 scores for Named Entities, sorted by decreasing BERT-large F1. Induced ontologies capture some labels surprisingly well, but are indifferent to more specialized categories which may require more world knowledge to distinguish.

Nonterminals Patterns in nPMI (Figure 3a) suggest basic syntactic notions: complete clauses (S, TOP, SINV) form a group, as do phrase types which take subjects (SBAR, VP, PP), and wh-phrases (WHADVP, WHPP, WHNP).

Dependencies Patterns in nPMI (Figure 3b) indicate several salient groups: verb arguments (nsubj, obj, obl, xcomp), left-heads (det, nmod:poss, compound, amod, case), right-heads (acl, acl:relcl, nmod[7]), and punct.

Semantic roles Patterns in nPMI (Figure 3c) roughly match intuition: primary core arguments (ARG0, ARG1) are distinguished, as well as modals (ARGM-MOD) and negation (ARGM-NEG), while trailing arguments (ARG2–5) and modifiers (ARGM-TMP, LOC, etc.) form a large group. On one hand, this reflects surface patterns: primary core arguments are usually close to the verb, with ARG0 on the left and ARG1 on the right; trailing arguments and modifiers tend to be prepositional phrases or subordinate clauses; and modals

[6] Results for more tasks and encoders are in Appendix B.

[7] Often the object in a prepositional phrase modifying a noun.

Gold Label	P / R / F1
ARGM-MOD	.62 / .41 / .49
ARG0	.52 / .17 / .26
ARG1	.50 / .09 / .15
ARGM-NEG	.36 / .60 / .45
ARG2	.28 / .13 / .18

Table 4: Top semantic role labels by BERT-large B^3 precision. Core arguments ARG0–2 are most preferentially split, with high precision but low recall.

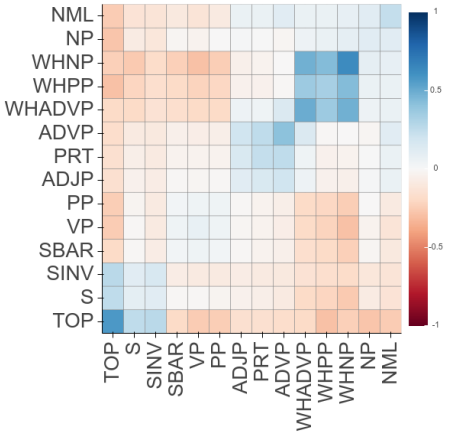
and negation are identified by lexical and positional cues. On the other hand, this also reflects error patterns in state-of-the-art systems, where label errors can sometimes be traced to ontological choices in PropBank, which distinguish between arguments and adjuncts that have very similar meaning (He et al., 2017; Kingsbury et al., 2002).

While the number of induced classes roughly matches gold for most tasks, induced ontologies for semantic roles are considerably more diverse, with a diversity measure close to 20 for ELMo and BERT (Table 2). Even though the alignment of predicted clusters with gold is dominated by a few patterns (Figure 3), the induced clustering contains more information than just these patterns. To locate this information, we examine the gold classes exhibiting the highest B^3 precision, shown in Table 4. Among these, core arguments ARG0, ARG1, and ARG2 have very low recall, indicating that the ontology splits them into finer-grained labels.

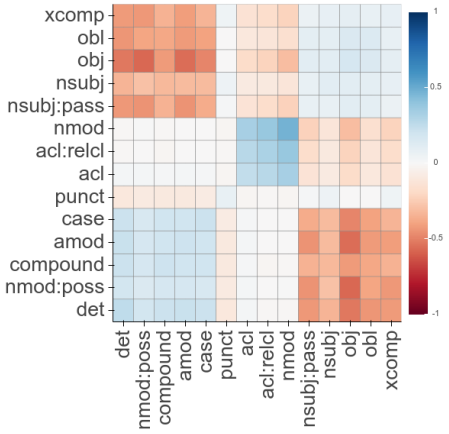
This follows intuition for PropBank core argument labels, which have predicate-specific meanings. Other approaches based on Frame Semantics (Baker et al., 1998; Fillmore et al., 2006), Proto-Roles (Dowty, 1991; Reisinger et al., 2015), or Levin classes (Levin, 1993; Schuler, 2005) have more explicit fine-grained roles. Concurrent work (Kuznetsov and Gurevych, 2020) shows that the choice of semantic role formalism meaningfully affects the behavior of supervised probes; further comparisons using LSL probing may help shed light on the origins of such differences.

6 Discussion

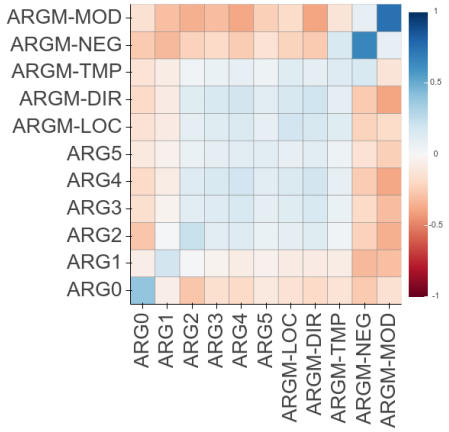
Our exploration of latent ontologies has yielded some surprising results: ELMo knows people, BERT knows dates, and both sense scalar and measurable values, while distinguishing between small and large numbers. Both models preferentially split core semantic roles into many fine-grained



(a) Nonterminals.



(b) Universal dependencies.



(c) Semantic roles.

Figure 3: Pairwise gold label nPMIs on selected categories for ontologies induced from BERT-large on selected tasks. Blue is positive nPMI, representing that gold labels are preferentially grouped together (*i.e.*, conflated by the model) relative to chance. Red is negative nPMI, representing that gold labels are well-separated. Perfectly matching ontologies would be 1 (blue) along the diagonal and -1 (red) in all off-diagonal cells. Counts are summed over all 5 runs to better reflect the underlying representations, though variance was low and our observed trends hold across all runs.

categories, and seem to encode broad notions of syntactic and semantic structure. These findings contrast with those from fully-supervised probes, which produce strong agreement with existing annotations (Tenney et al., 2019b) but can also report false positives by fitting to weak patterns in large feature spaces (Zhang and Bowman, 2018; Voita and Titov, 2020). Instead, agreement of latent categories with known concepts can be taken as strong evidence that these concepts (or similar ones) are present as important, salient features in an encoder's representation space.

This issue is particularly important when looking for *deep, inherent understanding* of linguistic structure, which by nature must generalize. For supervised systems, generalization is often measured by out-of-distribution objectives like out-of-domain performance (Ganin et al., 2016), transferability (Wang et al., 2018), targeted forms of compositionality (Geiger et al., 2020), or robustness to adversarial inputs (Jia and Liang, 2017). Recent work also advocates for counterfactual learning and evaluation (Qin et al., 2019; Kaushik et al., 2020) to mitigate confounds, or contrastive evaluation sets (Gardner et al., 2020) to rigorously test local decision boundaries. Overall, these techniques target discrepancies between salient features in a model and causal relationships in a task. In this work, we extract such features directly and investigate them by comparing induced and gold ontologies. This identifies some very strong cases of transferability from the binary detection task to detection tasks over gold subcategories, such as ELMo's *people* and BERT's *dates* (Table 3). Future work may investigate *cross-task* ontology matching to identify other transferable features, the emergence of categories signifying pipelined reasoning (Tenney et al., 2019a), surface patterns, or new, perhaps unexpected distinctions which can appear when going beyond existing schemas (Michael et al., 2018).

Our results point to a paradigm of **probing with latent variables**, for which LSL is one potential technique. We have only scratched the surface of what may emerge with such methods: while our probing test is high specificity, it is low power; extant latent structure may still be missed. LSL probing may produce different ontologies due to many factors, such as tokenization (Singh et al., 2019), encoder architecture (Peters et al., 2018b), probe architecture (Hewitt and Manning, 2019), data distribution (Gururangan et al., 2018), pretraining task (Liu et al., 2019a; Wang et al., 2019a), or pretraining checkpoint. Any such factors may be at work in the differences we observe between ELMo and BERT: for example, BERT's tokenization method may not as readily induce *personhood* features due to splitting of rare words (like names) in byte-pair encoding. Furthermore, concurrent work (Chi et al., 2020) has already found qualitative evidence of syntactic dependency types emergent in the special case of multilingual structural probes (Hewitt and Manning, 2019). With LSL, we provide a method that can be adapted to a variety of probing settings to both quantify and qualify this kind of structure.

7 Conclusion

We introduced a new model analysis method based on *latent subclass learning*: by factoring a binary classifier through a forced choice of latent subclasses, latent ontologies can be coaxed out of input features. Using this approach, we showed that encoders such as BERT and ELMo can be found to hold stable, consistent latent ontologies on a variety of linguistic tasks. In these ontologies, we found clear connections to existing categories, such as *personhood* of named entities. We also found evidence of ontological distinctions beyond traditional gold categories, such as distinguishing large and small numbers, or preferring fine-grained semantic roles for core arguments. In latent subclass learning, we have shown a general technique to uncover some of these features discretely, providing a starting point for descriptive analysis of our models' latent ontologies. The high specificity of our method opens doors to more insights from future work, which may include investigating how LSL results vary with probe architecture, developing intrinsic quality measures on latent ontologies, or applying the technique to discover new patterns in settings where gold annotations are not present.

Acknowledgments

We would like to thank Tim Dozat, Kenton Lee, Emily Pitler, Kellie Webster, other members of Google Research, Sewon Min, and the anonymous reviewers, who all provided valuable feedback on this paper. We also thank Rafael Müller, Simon Kornblith, and Geoffrey Hinton for discussion on the LSL classifier, and Alessandro Sordoni for pointing out the connection between the clustering regularizers and mutual information.

References

Samira Abnar, Lisa Beinborn, Rochelle Choenni, and Willem Zuidema. 2019. Blackbox meets blackbox: Representational similarity and stability analysis of neural language models and brains. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 191–203.

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. In *International Conference on Learning Representations*.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12(4):461–486.

Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 79–85, Montreal, Quebec, Canada. Association for Computational Linguistics.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet project. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*, pages 86–90. Association for Computational Linguistics.

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matt Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 2670–2676, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017a. What do neural machine translation models learn about morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872. Association for Computational Linguistics.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017b. Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10. Asian Federation of Natural Language Processing.

Terra Blevins, Omer Levy, and Luke Zettlemoyer. 2018. Deep RNNs encode soft hierarchical syntax. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 14–19. Association for Computational Linguistics.

Gerlof Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction. In *GSCL*.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2014. One billion word benchmark for measuring progress in statistical language modeling. In *Proceedings of Interspeech*.

Mingda Chen, Zewei Chu, and Kevin Gimpel. 2019. Evaluation benchmarks and learning criteria for discourse-aware sentence 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 649–662, Hong Kong, China. Association for Computational Linguistics.

Ethan A. Chi, John Hewitt, and Christopher D. Manning. 2020. Finding universal grammatical relations in multilingual BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5564–5577. Association for Computational Linguistics.

Eunsol Choi, Omer Levy, Yejin Choi, and Luke Zettlemoyer. 2018. Ultra-fine entity typing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 87–96, Melbourne, Australia. Association for Computational Linguistics.

Grzegorz Chrupała and Afra Alishahi. 2019. Correlating neural and symbolic representations of language. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2952–2962, Florence, Italy. Association for Computational Linguistics.

Kenneth Ward Church and Patrick Hanks. 1989. Word association norms, mutual information, and lexicography. In *27th Annual Meeting of the Association for Computational Linguistics*, volume 16, pages 76–83. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties.

In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136. Association for Computational Linguistics.

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.

David Dowty. 1991. Thematic proto-roles and argument selection. *Language*, 67:547–619.

Charles J Fillmore et al. 2006. Frame semantics. *Cognitive linguistics: Basic readings*, 34:373–400.

Nicholas FitzGerald, Julian Michael, Luheng He, and Luke Zettlemoyer. 2018. Large-scale QA-SRL parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2051–2060. Association for Computational Linguistics.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hanna Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating NLP models via contrast sets.

Atticus Geiger, Kyle Richardson, and Christopher Potts. 2020. Modular representation underlies systematic generalization in neural natural language inference models. *arXiv preprint arXiv:2004.14623*.

Yoav Goldberg. 2019. Assessing bert's syntactic abilities. *arXiv preprint arXiv:1901.05287*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, volume 2, pages 107–112. 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.

Luheng He, Mike Lewis, and Luke Zettlemoyer. 2015. Question-answer driven semantic role labeling: Using natural language to annotate natural language. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653. Association for Computational Linguistics.

John Hewitt and Percy Liang. 2019. Designing and interpreting probes with control tasks. 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 2733–2743, Hong Kong, China. Association for Computational Linguistics.

John Hewitt and Christopher D. Manning. 2019. A structural probe for finding syntax in word representations. 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 4129–4138, Minneapolis, Minnesota. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031, Copenhagen, Denmark. Association for Computational Linguistics.

Divyansh Kaushik, Eduard Hovy, and Zachary Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *International Conference on Learning Representations*.

Paul Kingsbury, Martha Palmer, and Mitch Marcus. 2002. Adding semantic annotation to the penn treebank. In *In Proceedings of the Human Language Technology Conference*.

Ilia Kuznetsov and Iryna Gurevych. 2020. A matter of framing: The impact of linguistic formalism on probing results. *arXiv preprint arXiv:2004.14999*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

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*,

pages 188–197. Association for Computational Linguistics.

B. Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, pages 94–100. AAAI Press.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. 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 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019b. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

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

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202. Association for Computational Linguistics.

Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*.

Julian Michael, Gabriel Stanovsky, Luheng He, Ido Dagan, and Luke Zettlemoyer. 2018. Crowdsourcing question-answer meaning representations. 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 560–568. Association for Computational Linguistics.

Rafael Müller, Simon Kornblith, and Geoffrey Hinton. 2020. Subclass distillation. *arXiv preprint arXiv:2002.03936*.

Joakim Nivre, Željko Agić, Maria Jesus Aranzabe, Masayuki Asahara, Aitziber Atutxa, Miguel Ballesteros, John Bauer, Kepa Bengoetxea, Riyaz Ahmad Bhat, Cristina Bosco, Sam Bowman, Giuseppe G. A. Celano, Miriam Connor, Marie-Catherine de Marneffe, Arantza Diaz de Ilarraza, Kaja Dobrovoljc, Timothy Dozat, Tomaž Erjavec, Richárd Farkas, Jennifer Foster, Daniel Galbraith, Filip Ginter, Iakes Goenaga, Koldo Gojenola, Yoav Goldberg, Berta Gonzales, Bruno Guillaume, Jan Hajič, Dag Haug, Radu Ion, Elena Irimia, Anders Johannsen, Hiroshi Kanayama, Jenna Kanerva, Simon Krek, Veronika Laippala, Alessandro Lenci, Nikola Ljubešić, Teresa Lynn, Christopher Manning, Cătălina Mărănduc, David Mareček, Héctor Martínez Alonso, Jan Mašek, Yuji Matsumoto, Ryan McDonald, Anna Missilä, Verginica Mititelu, Yusuke Miyao, Simonetta Montemagni, Shunsuke Mori, Hanna Nurmi, Petya Osenova, Lilja Øvrelid, Elena Pascual, Marco Passarotti, Cenel-Augusto Perez, Slav Petrov, Jussi Piitulainen, Barbara Plank, Martin Popel, Prokopis Prokopidis, Sampo Pyysalo, Loganathan Ramasamy, Rudolf Rosa, Shadi Saleh, Sebastian Schuster, Wolfgang Seeker, Mojgan Seraji, Natalia Silveira, Maria Simi, Radu Simionescu, Katalin Simkó, Kiril Simov, Aaron Smith, Jan Štěpánek, Alane Suhr, Zsolt Szántó, Takaaki Tanaka, Reut Tsarfaty, Sumire Uematsu, Larraitz Uria, Viktor Varga, Veronika Vincze, Zdeněk Žabokrtský, Daniel Zeman, and Hanzhi Zhu. 2015. Universal dependencies 1.2. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018a. 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. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Luke Zettlemoyer, and Wen-tau Yih. 2018b. Dissecting contextual word embeddings: Architecture and representation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. 2020. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*,

pages 4609–4622. Association for Computational Linguistics, Association for Computational Linguistics.

Sameer S Pradhan, Eduard Hovy, Mitch Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2007. Ontonotes: A unified relational semantic representation. *International Journal of Semantic Computing*, 1(04):405–419.

Lianhui Qin, Antoine Bosselut, Ari Holtzman, Chandra Bhagavatula, Elizabeth Clark, and Yejin Choi. 2019. Counterfactual story reasoning and generation. 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 5043–5053, Hong Kong, China. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *https://blog.openai.com/better-language-models*.

Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. 2019. Visualizing and measuring the geometry of BERT. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8592–8600. Curran Associates, Inc.

Drew Reisinger, Rachel Rudinger, Francis Ferraro, Craig Harman, Kyle Rawlins, and Benjamin Van Durme. 2015. Semantic proto-roles. *Transactions of the Association for Computational Linguistics*, pages 475–488.

Anna Rogers, Olga Kovaleva, and Anna Rumshisky. 2020. A primer in BERTology: What we know about how BERT works. *arXiv preprint arXiv:2002.12327*.

Naomi Saphra and Adam Lopez. 2019. Understanding learning dynamics of language models with SVCCA. 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 3257–3267, Minneapolis, Minnesota. Association for Computational Linguistics.

Karin Kipper Schuler. 2005. *VerbNet: A broad-coverage, comprehensive verb lexicon*. Ph.D. thesis, University of Pennsylvania.

Natalia Silveira, Timothy Dozat, Marie-Catherine de Marneffe, Samuel Bowman, Miriam Connor, John Bauer, and Chris Manning. 2014. A gold standard dependency corpus for English. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2897–2904. European Language Resources Association (ELRA).

Jasdeep Singh, Bryan McCann, Richard Socher, and Caiming Xiong. 2019. BERT is not an interlingua and the bias of tokenization. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 47–55, Hong Kong, China. Association for Computational Linguistics.

Daniel Smilkov, Nikhil Thorat, Charles Nicholson, Emily Reif, Fernanda B Viégas, and Martin Wattenberg. 2016. Embedding projector: Interactive visualization and interpretation of embeddings. In *NIPS 2016 Workshop on Interpretable Machine Learning in Complex Systems*.

Alon Talmor, Yanai Elazar, Yoav Goldberg, and Jonathan Berant. 2019. oLMpics – on what language model pre-training captures. *arXiv preprint arXiv:1912.13283*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019a. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Sam Bowman, Dipanjan Das, and Ellie Pavlick. 2019b. What do you learn from context? probing for sentence structure in contextualized word representations. In *International Conference on Learning Representations*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of NIPS*.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. 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 4396–4406. Association for Computational Linguistics.

Elena Voita and Ivan Titov. 2020. Information-theoretic probing with minimum description length. *arXiv preprint arXiv:2003.12298*.

Alex Wang, Jan Hula, Patrick Xia, Raghavendra Pappagari, R. Thomas McCoy, Roma Patel, Najoung Kim, Ian Tenney, Yinghui Huang, Katherin Yu, Shuning Jin, Berlin Chen, Benjamin Van Durme, Edouard Grave, Ellie Pavlick, and Samuel R. Bowman. 2019a. Can you tell me how to get past sesame street? sentence-level pretraining beyond language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4465–4476, Florence, Italy. Association for Computational Linguistics.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019b. SuperGLUE: A multi-task benchmark and analysis platform for natural language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 3261–3275. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355. Association for Computational Linguistics.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. OntoNotes release 5.0 LDC2013T19. *Linguistic Data Consortium, Philadelphia, PA*.

Zhiyong Wu, Yun Chen, Ben Kao, and Qun Liu. 2020. Perturbed masking: Parameter-free probing for analyzing and interpreting BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4166–4176. Association for Computational Linguistics.

Kelly Zhang and Samuel Bowman. 2018. Language modeling teaches you more than translation does: Lessons learned through auxiliary syntactic task analysis. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 359–361. Association for Computational Linguistics.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Probe capacity tuning

Results from hidden size tuning are shown in Figure 4. We use the accuracy of a binary classifier trained only on binary labels, choosing the smallest hidden size with at least 97% of the maximum performance over all trials. For comparison, we report the accuracy of a fully supervised multi-class model with the same hidden size. Our method sometimes chooses a hidden size where the accuracy of the fully supervised probe is much lower than max. While this suggests limits on the structure that can be produced, it makes our method independent of fine-grained gold labeling. Future work may investigate the role of probe expressiveness in determining induced ontologies.

B More Experimental Results

Results on larger set of encoders and tasks are shown in Tables 5–11. The extra tasks are undirected Universal Dependencies (Nivre et al., 2015), TAC relation classification (Zhang et al., 2017), and OntoNotes coreference (Pradhan et al., 2007). The extra encoders are BERT-base, multilingual BERT (mBERT)[8] and ALBERT (Lan et al., 2020).

C More Analysis Results

We show more comparative nPMI plots for BERT-large and ELMo in Figure 5 and Figure 6. These use co-occurrence counts summed over 5 runs, and exhibit the same overall trends as each run.

Relation classification nPMI plots for BERT-large and ELMo are shown for TAC relation classification in Figure 7. ELMo produces two diffuse groups of gold labels, while BERT seems to more clearly identify several categories of relations. Some of these may seem intuitive, *e.g.*, `org:founded_by` and `per:date_of_birth` relate to the creation of an entity, and are grouped together. However, the model distinguishes these from `per:origin` and `per:parents`, which may also intuitively seem similar. The broad distribution and highly specific semantics of TAC relations makes direct qualitative assessment difficult. Further analysis, perhaps comparing induced clusters more surface-level features (*e.g.*, dependency paths) may shed more light on these results.

Lexical baseline results Normalized PMI plots for the lexical baseline on several tasks are shown in Figure 8. In most cases, these show essentially no relation to gold categories. In the few cases where groups seem to emerge, they are coarser and more diffuse than what we observe with probes over contextual representations.

[8] https://github.com/google-research/bert/blob/master/multilingual.md

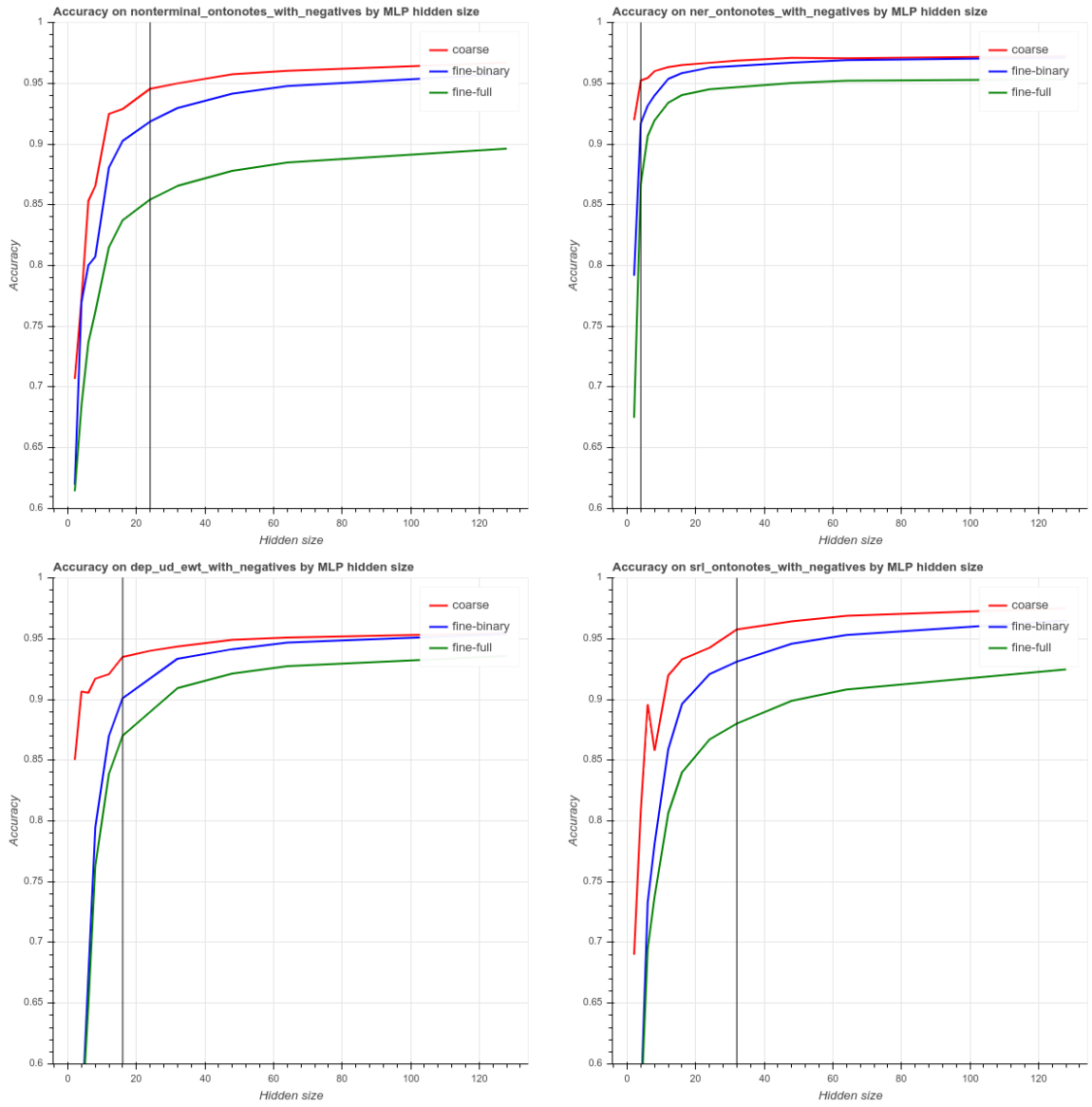


Figure 4: Performance on hidden size tuning experiments for different tasks. Clockwise from top-left, they are nonterminals, named entities, semantic roles, and syntactic dependencies. `coarse` (red) is binary accuracy of a binary classifier, `fine-binary` (blue) is binary accuracy of a full multiclass classifier, and `fine-full` (green) is the full multiclass accuracy of the multiclass classifier. The black vertical line is the smallest hidden size that passes the 97% performance threshold for `coarse`.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	9.71	1.00
ELMo	0.40	0.66	0.50	0.83	5.07	1.08
BERT-base	0.43	0.57	0.49	0.88	6.09	1.11
BERT-large	0.47	0.53	0.50	0.86	7.50	1.10
mBERT	0.25	0.67	0.37	0.84	3.29	1.06
ALBERT-large	0.38	0.53	0.44	0.89	6.00	1.15
BERT-large (lex)	0.19	0.39	0.26	0.74	4.33	1.13

Table 5: Results by encoder for OntoNotes named entity labeling.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	7.15	1.00
ELMo	0.36	0.25	0.30	0.58	10.16	1.12
BERT-base	0.36	0.41	0.38	0.60	5.76	1.06
BERT-large	0.35	0.34	0.35	0.61	7.80	1.06
mBERT	0.36	0.34	0.35	0.59	7.38	1.06
ALBERT-large	0.38	0.28	0.32	0.59	9.07	1.08
BERT-large (lex)	0.22	0.80	0.34	0.50	1.47	1.26

Table 6: Results by encoder for OntoNotes nonterminal labeling.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	22.91	1.00
ELMo	0.23	0.42	0.29	0.67	11.11	1.22
BERT-base	0.13	0.34	0.19	0.76	9.69	1.23
BERT-large	0.14	0.33	0.19	0.77	11.22	1.23
mBERT	0.27	0.51	0.35	0.73	9.40	1.22
ALBERT-large	0.23	0.41	0.29	0.72	9.84	1.20
BERT-large (lex)	0.06	0.86	0.11	0.50	1.33	1.02

Table 7: Results by encoder for Universal Dependency labeling.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	22.91	1.00
ELMo	0.19	0.23	0.21	0.71	19.12	1.14
BERT-base	0.27	0.24	0.25	0.85	22.79	1.20
BERT-large	0.23	0.23	0.23	0.82	18.51	1.17
mBERT	0.24	0.20	0.21	0.83	20.31	1.19
ALBERT-large	0.30	0.27	0.28	0.81	20.53	1.14
BERT-large (lex)	0.09	0.54	0.16	0.50	3.39	1.00

Table 8: Results by encoder for undirected Universal Dependency labeling.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	8.73	1.00
ELMo	0.40	0.17	0.24	0.76	22.35	1.08
BERT-base	0.39	0.18	0.25	0.86	21.95	1.15
BERT-large	0.37	0.17	0.24	0.88	18.70	1.15
mBERT	0.41	0.21	0.28	0.88	19.05	1.12
ALBERT-large	0.43	0.21	0.28	0.87	19.90	1.12
BERT-large (lex)	0.19	0.39	0.26	0.46	2.81	1.01

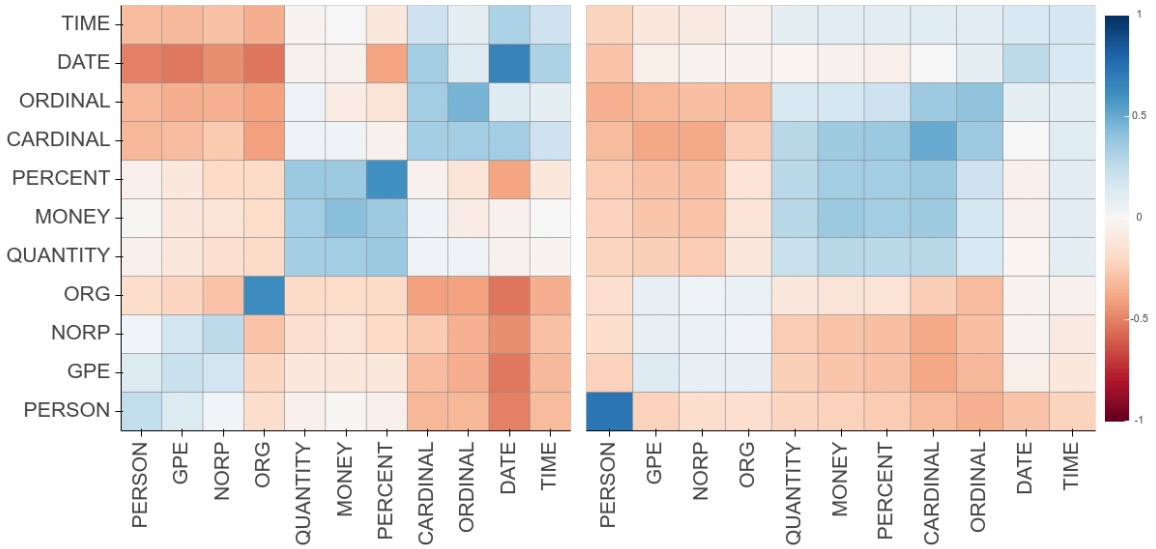
Table 9: Results by encoder for OntoNotes semantic role labeling.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	1.00	1.00
ELMo	1.00	0.09	0.16	0.80	14.22	1.18
BERT-base	1.00	0.09	0.16	0.86	14.67	1.24
BERT-large	1.00	0.09	0.17	0.87	15.57	1.27
mBERT	1.00	0.09	0.16	0.83	13.86	1.24
ALBERT-large	1.00	0.09	0.16	0.86	13.56	1.26
BERT-large (lex)	1.00	0.78	0.87	0.78	1.60	1.03

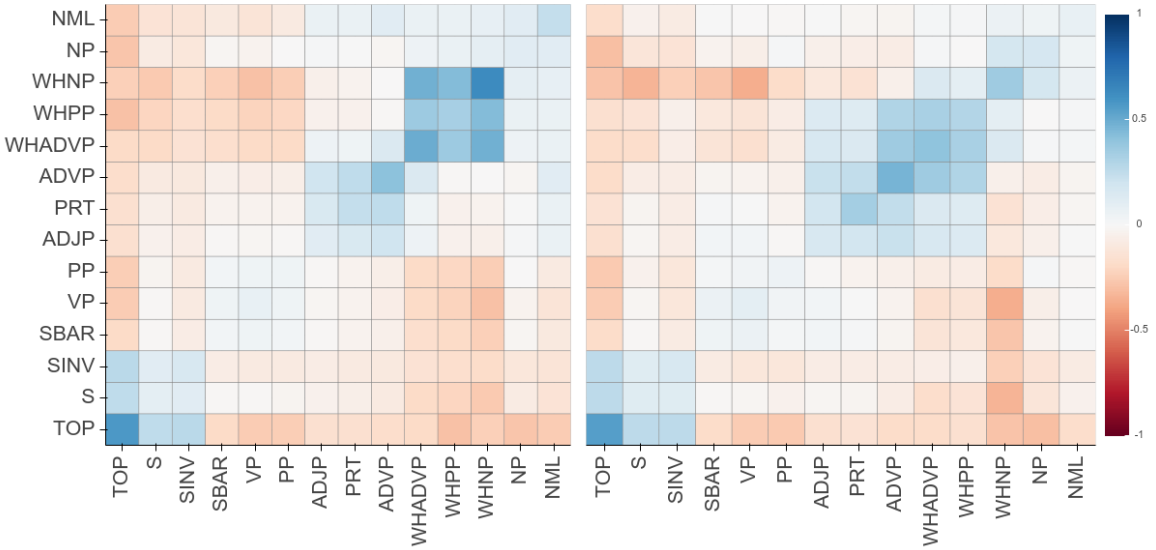
Table 10: Results by encoder for OntoNotes coreference. Note the high diversity scores, showing that the LSL model can find fine-grained structure even in the case of binary labels.

	P	R	F1	Acc.	Diversity	Uncertainty
Gold	1.00	1.00	1.00	1.00	24.78	1.00
ELMo	0.11	0.78	0.20	0.77	2.38	1.05
BERT-base	0.11	0.90	0.20	0.76	1.94	1.05
BERT-large	0.16	0.63	0.25	0.80	3.87	1.11
mBERT	0.15	0.87	0.26	0.76	2.21	1.05
BERT-large (lex)	0.07	0.97	0.13	0.76	1.11	1.02

Table 11: Results by encoder for TAC relation classification. Note that the diversity scores are much lower than gold for most encoders. This accords with Tenney et al. (2019b)'s findings that ELMo and BERT have middling performance on the task; it seems unlikely that the highly specific relations in TACRED are salient in their feature spaces.

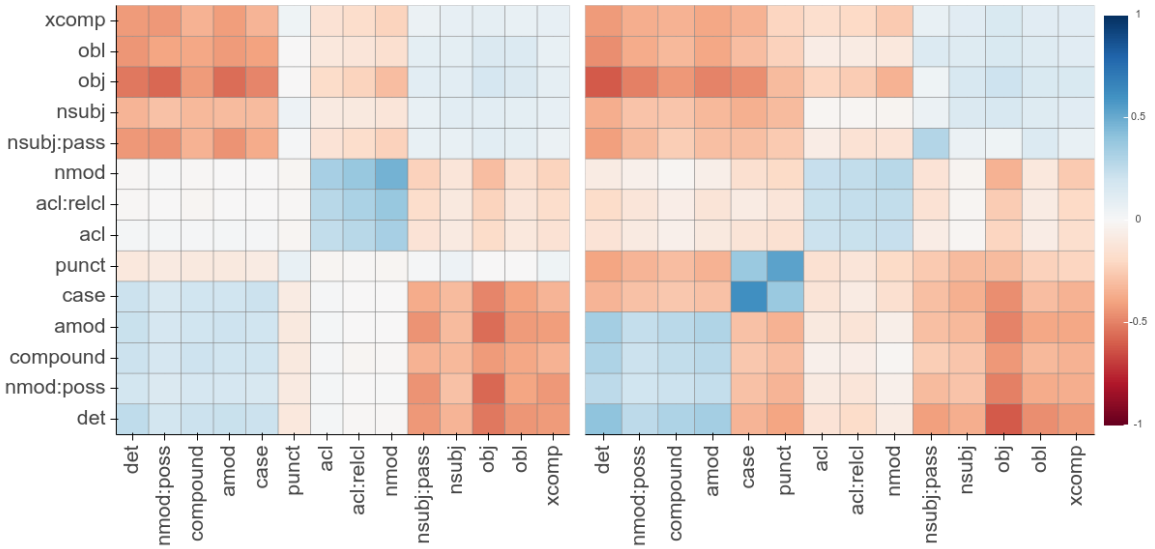


(a) Pairwise nPMIs for selected named entity classes in ontologies induced on BERT-large (left) and ELMo (right).

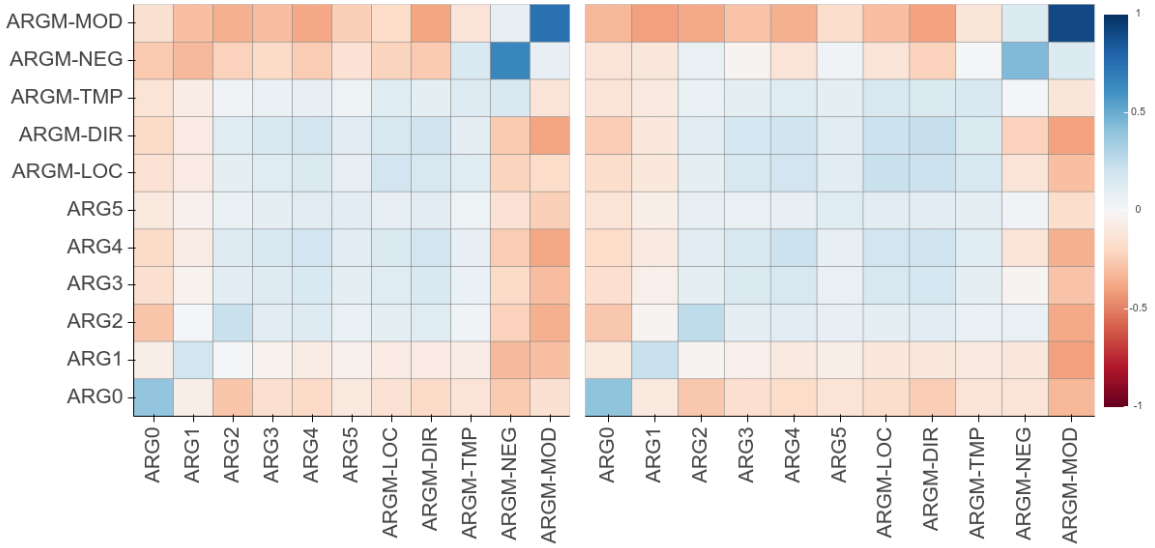


(b) Pairwise nPMIs for selected nonterminal classes in ontologies induced on BERT-large (left) and ELMo (right).

Figure 5: Pairwise nPMI charts for named entities and nonterminals.



(a) Pairwise nPMIs for selected named universal dependency labels in ontologies induced on BERT-large (left) and ELMo (right).



(b) Pairwise nPMIs for selected semantic roles in ontologies induced on BERT-large (left) and ELMo (right).

Figure 6: Pairwise nPMI charts for syntactic dependencies and semantic roles.

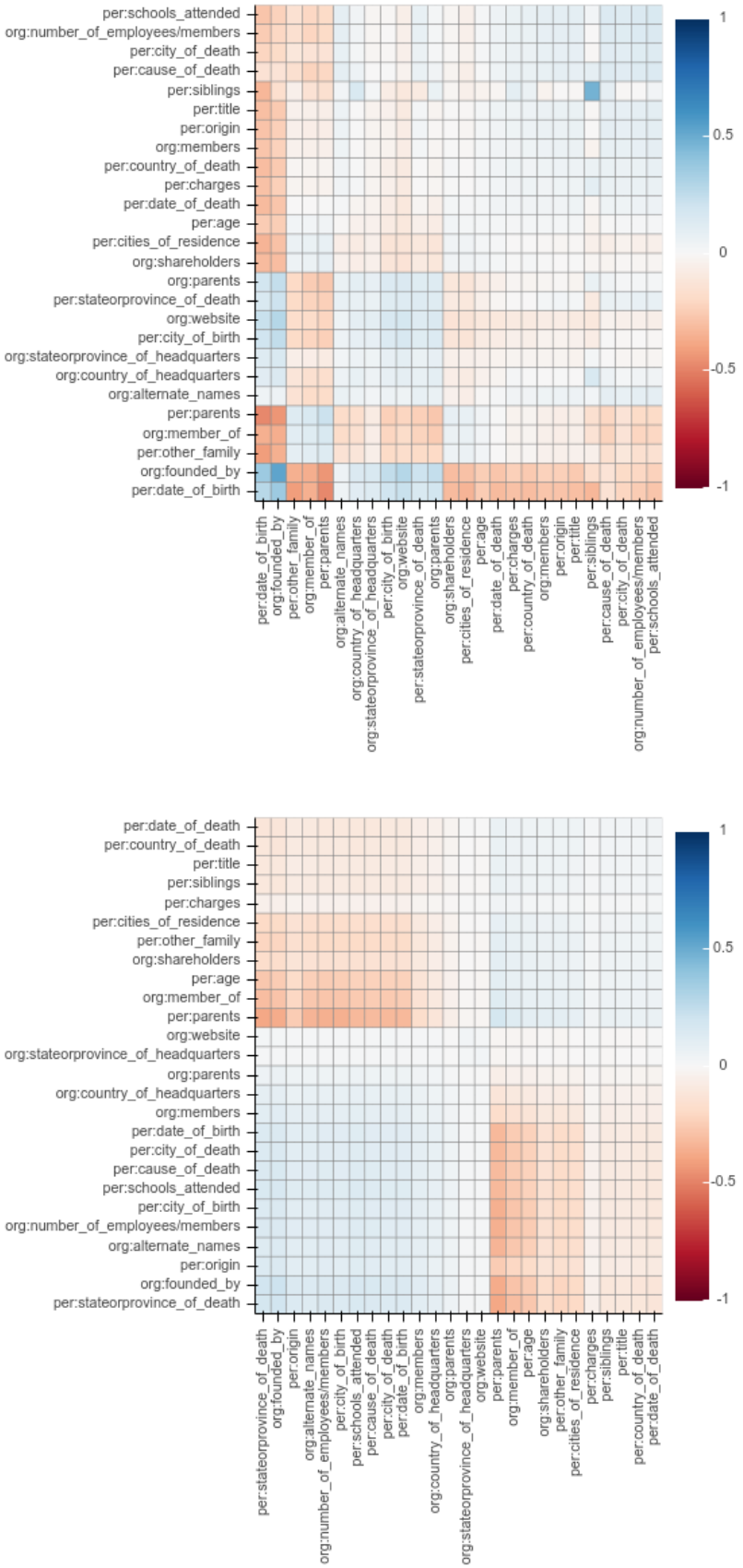


Figure 7: Pairwise nPMIs for TAC relations in ontologies induced on BERT-large (top) and ELMo (bottom).

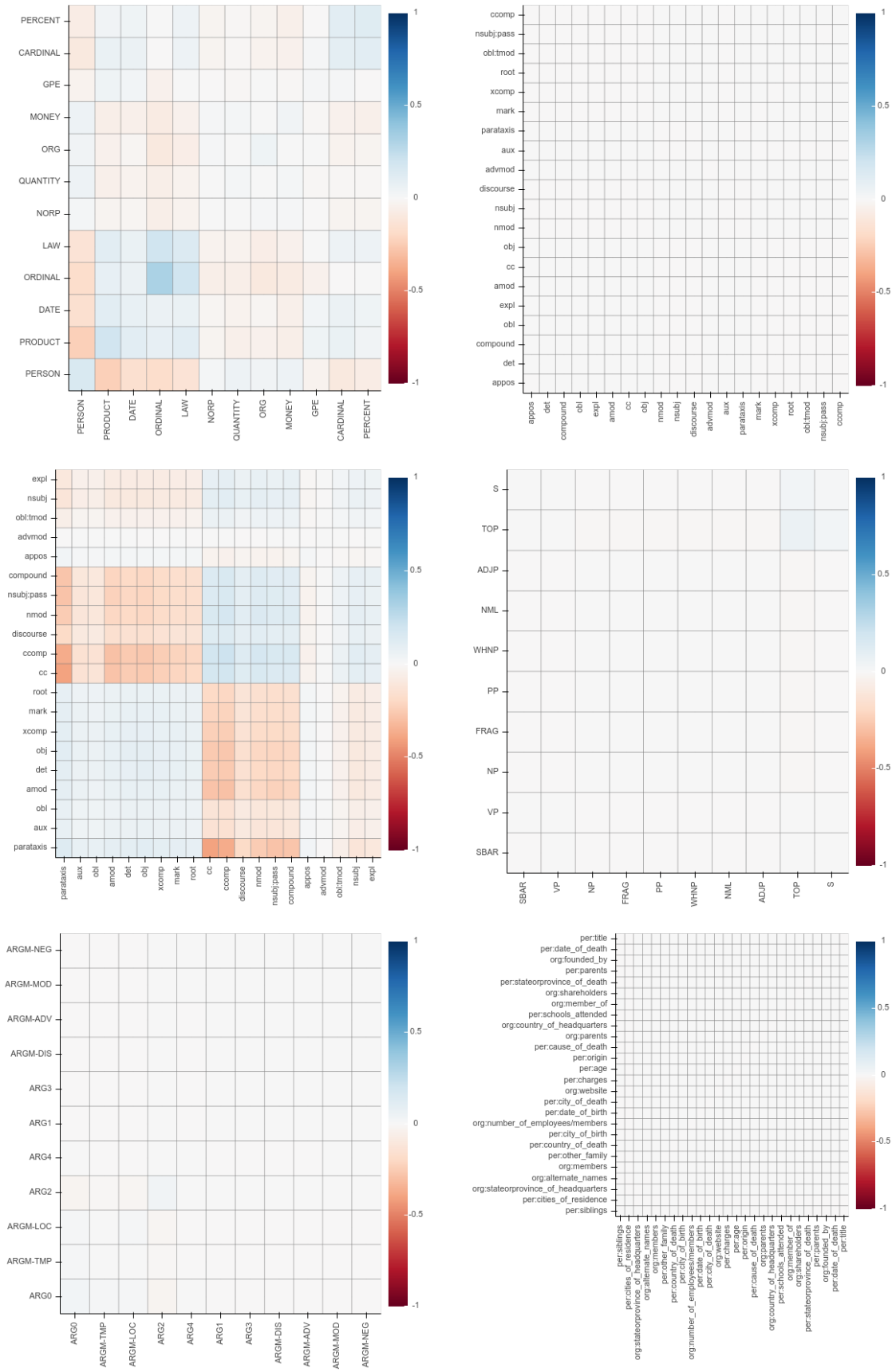


Figure 8: Pairwise nPMI charts for the lexical baseline using non-contextual embeddings from BERT-large. Clockwise from top-left, they are named entities, Universal Dependencies, nonterminals, TAC relations, semantic roles, and undirected Universal Dependencies. In most cases this model seems to have no relation to gold labels, and in the few cases with interesting structure, this structure is weaker and coarser than with contextual embeddings.