Please provide feedback

Please support the ScholarWorks@UMBC repository by emailing scholarworks-group@umbc.edu and telling us what having access to this work means to you and why it's important to you. Thank you.

Discriminative and Generative Transformer-based Models For Situation Entity Classification

Mehdi Rezaee   Kasra Darvish   Gaoussou Youssouf Kebe   Francis Ferraro

Department of Computer Science
University of Maryland Baltimore County
Baltimore, MD 21250 USA
{rezaee1,kasradarvish,mb88814,ferraro}@umbc.edu

Abstract

We re-examine the situation entity (SE) classification task with varying amounts of available training data. We exploit a Transformer-based variational autoencoder to encode sentences into a lower dimensional latent space, which is used to generate the text and learn a SE classifier. Test set and cross-genre evaluations show that when training data is plentiful, the proposed model can improve over the previous discriminative state-of-the-art models. Our approach performs disproportionately better with smaller amounts of training data, but when faced with extremely small sets (4 instances per label), generative RNN methods outperform transformers. Our work provides guidance for future efforts on SE and semantic prediction tasks, and low-label training regimes.

1 Introduction

Semantics has long recognized that a clause that references some event-like *situation* may not actually be referring to any single, *particular* event, but instead a general class of events (Carlson and Pelletier, 1995). For example, the sentence in Fig. 1 provides a statement about general rules or facts (about what should not happen in a volleyball game). The situation that it describes is a general type of event, but it does not necessarily refer to any particular event, such as in a sentence, "The receiving team let the ball be grounded." While linguistically intriguing, such statements can provide important background knowledge to aid various applications such as argumentation mining (Becker et al., 2017a), discourse parsing (Palmer et al., 2007), and characterizing genres (Becker et al., 2017b; Palmer and Friedrich, 2014).

Friedrich et al. (2016) provide a taxonomy of, and associated dataset labeled with, the different types of "situations," such as EVENT, STATE, or a GENERIC statement. Annotating and identifying
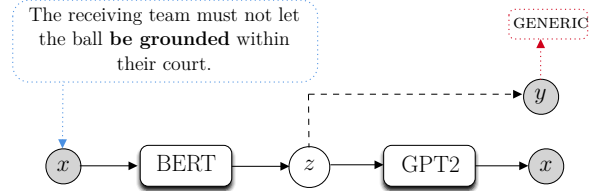

Figure 1: Model structure; → shows pre-training path for text reconstruction using unlabeled data, - - - indicates label prediction during training. The latent representation gathered from the unlabeled data is used for both text reconstruction and SE type prediction jointly.

the appropriate situation type for a clause—called *situation entity (SE) typing*—is not trivial. While efforts to build predictive systems have steadily improved performance (Friedrich et al., 2016; Dai and Huang, 2018), these have only considered *discriminatively* trained models. If labeled data is plentiful, discriminative models can be effective. Discriminative-only architectures may limit the effectiveness of these models when obtaining quality annotated data is difficult—like SE typing.

While generatively-trained classifiers usually provide a reasonable alternative for small training sets, the performance of these generative models can degrade when trained on large training sets (Ding and Gimpel, 2019). However, the expressive capabilities of self-attention mechanism in Transformers (Vaswani et al., 2017), make them a natural choice for both classification and language modeling tasks. Transformers have made it possible to effectively learn strong prior information from large-scale open-domain corpora, which is then fine-tuned on the downstream task. While often effective, a purely discriminative Transformer (BERT) has been shown to underperform the RNN-based models when training data is limited (Ezen-Can, 2020; Phang et al., 2018; Lee et al., 2019).

Our aim in this paper is to handle the SE classification for both low and rich resource settings.

We hypothesize that latent variable models and pre-trained Transformer-based models capture complementary information for this task. To fully make use of these two paradigms, we study a variational autoencoder model (VAE) that uses a conditional model (BERT) to encode the input sentences into a latent space, then employs a generative language model (e.g., GPT-2 or an LSTM) to regenerate the input text from that latent space (Li et al., 2020; Kingma and Welling, 2013). We make the following contributions: (1) We show that when labeled data is plentiful, transformer methods that make local, independent predictions are able to outperform the current SOTA on SE typing that make global, joint predictions. (2) We demonstrate that our variational approach is adaptable to lower-resource settings: when the number of samples per label is small, simpler decoders (e.g., BERT+BOW, BERT+LSTM) outperform a more complex one (i.e., BERT+GPT2). (3) We identify approaches that can nearly double performance when the number of samples per label is very small: we find generative RNN-based models outperform transformer-based models.

2 Task Overview

We consider the task of SE type classification using the publicly available MASC+Wiki dataset (Friedrich et al., 2016).[1] In this task, an English clause, containing an event-like predicate, is classified into different SE types: states, events, reports, generic sentences, generalizing sentences, questions, and imperative sentences. Fig. 1 shows an example of a GENERIC. See Table 3 in the appendix for high-level statistics about the dataset, and Friedrich et al. (2016) for an in-depth explanation of each of the SE types. This dataset also segments the documents into their various genres, such as "email," "fiction," and "technical."

2.1 Related Work on SE Classification

Drawing inspiration from Smith (2003) and Palmer et al. (2007), Friedrich et al. (2016) introduced the MASC+Wiki corpus, consisting of more than $40,000$ SE-labeled clauses from 13 different genres, and a non-neural, linear chain CRF. They employed a linear chain CRF based on a feature set including the POS tags and the main verb of each clause to predict the labels. Becker et al. (2017b)

proposed a model that combines a GRU and attention layer to capture the dependency between the tokens, labels and genres, in order to better predict the SE types. Dai and Huang (2018), the current SOTA, used a hierarchical LSTM-based structure followed by a CRF mechanism to go beyond clause-level to achieve paragraph-wide dependencies.

While SE type prediction has been an understudied area, as Friedrich et al. (2016) describe, one possible outcome of being able to identify the type of situation a clause is describing is an improved ability to analyze the different types of discourse (Smith, 2003; Palmer et al., 2007; Friedrich and Palmer, 2014) or narration that can occur when one situation type is predominant over others. SE types have also been used to perform temporal analysis of situations (Vempala et al., 2018), analyze participants in a reported event (Sanagavarapu et al., 2017), and has helped seed deep, nuanced views of lexical semantics (Govindarajan et al., 2019).

3 Method

We study a simple yet effective encoder-decoder architecture, shown in Fig. 1. For text x and label y, we use neural variational inference (NVI) to compute a latent representation z that is jointly trained to generate x ($p(x|z)$) and accurately predict y ($p(y|z)$). This yields the model $p(x, y, z) = p(x|z)p(y|z)p(z)$. NVI learns a variational distribution $q(z|x)$ to be close to the posterior $p(z|x, y)$.

For training, we use the latent representation, computed from $q(z|x)$, for both clause reconstruction ($\mathbb{E}_{q(z|x)} \log p(x|z)$) and label prediction ($\mathbb{E}_{q(z|x)} \log p(y|z)$). To reduce the KL vanishing issue (Bowman et al., 2016; Shao et al., 2020), we optimize the annealed ELBO,

$$\mathcal{L} = \mathbb{E}_{q(z|x)}[\log p(y|z)p(x|z)] - \beta \text{KL}[q(z|x)||p(z)], \quad (1)$$

where β is the annealing coefficient fixed to 0.5. To predict, we use a MAP estimation, $\mathbb{E}_{q(z|x)} \log p(y|z) \approx \log p(y|\mu_z)$, where $p(y|z)$ is a single dense layer followed by softmax.

The latent variable z, drawn from a neural-parametrized Gaussian distribution, captures the high-level representation of the sentence's content. In this setting, $q(z|x)$ is a multivariate Gaussian, whose mean μ_{z_l} is computed via the CLS embedding from BERT, and $p(x|z)$ is computed by a generative text decoder seeded by z—for example, GPT2 or an LSTM seeded by the sampled z. Two

Model	F1	Acc
Context Aware (Dai and Huang, 2018)	77.4	80.7
Discriminative	58.3	69.1
Generative	59.7	66.7
Latent	60.0	67.0
Par BERT (Cohan et al., 2019)	78.3	81.2
BERT+BOW	77.0	80.0
BERT+LSTM	77.4	80.1
BERT	78.8	81.1
BERT+GPT2	**79.1**	**81.9**

Table 1: Classification performance of our methods and baselines against the current SOTA Dai and Huang (2018), trained with full training data and evaluated on the test set. BERT and BERT+GPT2 predict each clause's SE type individually; Dai and Huang (2018) and Par BERT predict all clauses in a paragraph jointly.

linear functions map the latent vector to the GPT2's memory and the embedding.

The fundamental formal model we study—$p(x, y, z) = p(z)p(y|z)p(x|z)$—is a fairly straightforward latent variable model. While using a contextualized model to encode text and then a generative model to reconstruct it is a fairly understudied area, Li et al. (2020) demonstrated the effectiveness on standard classification tasks of stitching together BERT and GPT2 via a latent variable z, and then further pre-training the combination on English Wikipedia.[2] The core novelty of our work lies not in the precise model/inference formulation, but rather in the application and analysis of this fairly general approach to both the SE task, and the SE task at varying levels of label availability.

4 Experiments

We first compare the Transformer-based models against previously proposed approaches in terms of classification metrics. We then evaluate the generative models with limited training data samples per SE type. We report the average accuracy over 5 runs and compare the accuracy of all models.[3]

Models The central instantiation of the section 3 method uses BERT to encode and GPT-2 to regenerate the text. We study the effectiveness of the generative model, by presenting two variants of our model with different decoders. Specifically, we redefine $p(x|z)$ in two, simpler, light-weight ways: (1) **BERT+BOW**, where we use the simple

bag-of-words method for the reconstruction part. (2) **BERT+LSTM**, where we use a single LSTM decoder layer as a much simpler autoregressive alternative to GPT2. In BERT+BOW, z is passed to a single, linear softmax, while BERT+LSTM uses z to help compute each token's hidden state.

Baselines We compare our variational method from section 3 with the following approaches: (1) The current SOTA **Context Aware** (Dai and Huang, 2018) for SE prediction: A paragraph is fed to a word-level Bi-LSTM with 300 hidden units. Max pooling over the Bi-LSTM hidden vectors extracts clause representations and another Bi-LSTM uses the clause representations to predict SE types. (2) **BERT**: We perform clause-level classification by adding a fully connected layer followed by a softmax classifier on top of the pretrained uncased BERT-Base model. (3) **Par BERT** (Cohan et al., 2019): Instead of processing clauses of a paragraph one-by-one, the entire paragraph is provided to BERT. Clauses are separated by [SEP], and an MLP uses each of the [SEP] embeddings to predict a clause's SE type. These baselines allow use to compare to the SOTA, basic transformer methods, and transformer methods specifically designed for longer text sequences (such as paragraphs).

However, based on recent work in low-resource text classification (Ding and Gimpel, 2019), we additionally consider three new baselines: (4) **Discriminative Model** (Yogatama et al., 2017): A one-layer LSTM model encodes the sentences and a softmax layer over the average of hidden vectors predicts the labels. (5) **Generative Model** (Yogatama et al., 2017): Each label has an embedding. The tokens are fed to a one-layer LSTM model, and concatenated label embeddings and hidden vectors are used to reconstruct the tokens. (6) **Latent Model** (Ding and Gimpel, 2019): This LSTM-based method computes $p(y, x) = \sum_c p(y, c, x)$ via a discrete latent variable c. We let c be a 30 dimensional variable.

4.1 Full Training

The comparison of all of the baselines and the proposed models are shown in Table 1. Most of the Transformer-based models outperform the previous state-of-the-art approach from Dai and Huang (2018). We want to draw particular attention to the fact that Dai and Huang (2018)'s best model predicted clause SE types jointly for every clause in a paragraph, while BERT and BERT+GPT2 predic-

---

[2]The pre-trained model can be found here: https://github.com/ChunyuanLI/Optimus

[3]See Appendix B for more details on our implementation, models, and baselines.

Genre	Context Aware	BERT	Par BERT	BERT+GPT2	Humans
blog	70.3	72.03	**74.14**	72.37	72.9
email	71.5	73.84	**75.88**	74.53	67.0
essays	64.1	66.99	**67.49**	66.15	64.6
ficlets	68.8	**75.14**	73.11	73.86	81.7
fiction	72.1	77.52	75.42	**78.99**	76.7
gov-docs	68.9	**72.52**	72.31	71.08	72.6
jokes	75.0	**77.11**	74.46	76.73	82.0
journal	66.4	68.75	68.81	**71.96**	63.7
letters	71.2	72.01	**75.64**	71.93	68.0
news	72.7	**75.20**	74.58	73.11	78.6
technical	60.5	61.72	53.38	**62.72**	54.7
travel	53.6	**68.54**	54.74	58.18	48.9
wiki	60.6	63.04	64.58	**67.86**	69.2

Table 2: Cross-genre F1 Classification Results. Models are trained on all other genres (e.g., not-blog) and then evaluated on the target genre (blog). All Transformer methods surpass the current SOTA.

tions are made independently for each clause. That is, the previous SOTA method relied on global context, whereas the Transformer methods are able to do localized prediction without the explicit global context. In addition, when we made joint predictions for all clauses in a paragraph (Par BERT), performance slightly decreased. These results suggest that Transformers are able to implicitly capture enough background lexical knowledge to be performant when presented with limited/reduced context, though the lack of improvement of Par BERT over BERT or BERT+GPT2 suggests taking advantage of broader context remains a challenge.

We carried out cross-genre classification experiments on the training dataset (Table 2). We only compare Context Aware, BERT, Par BERT, and BERT+GPT2, as those had the highest accuracy results in Table 1; we include human scores (from Dai and Huang (2018)) for context. For all genres, BERT and BERT+GPT2 outperform the Context Aware model. For more than half of the genres, each of these two models are performing better than the human predictions. These results reaffirm the notion that while transformer-based approaches can be very beneficial, there exist sizable differences in performance across domains. Fully understanding these domain-differences is out-of-scope for this work and is an area for future study.

4.2 Medium-data and Small-data Training

As the expertise and overall cost for quality annotation is high, we take random subsets of the training data, having 64, 100, 400, 600, and 1000 instances (clauses) per label; while sizable, 1000 is an order of magnitude less than the full set.[4]

---
[4]The current SOTA model is paragraph-level, which reduces to an LSTM model for the clause-level task.
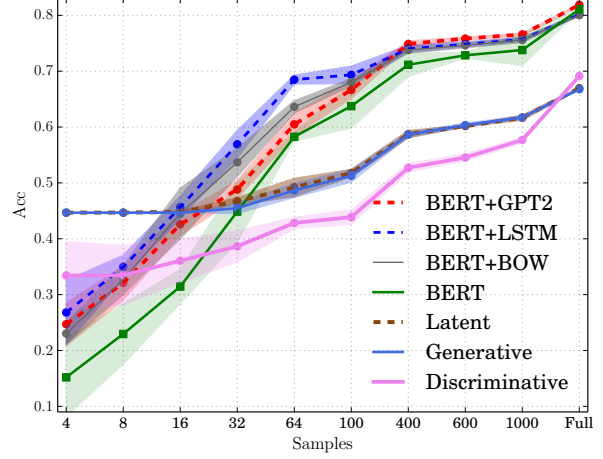

Figure 2: Performance, based on the number of instances per label during training.

In Fig. 2, with at least 64 examples per label, we observe a large gap between the Transformer-based and RNN-based models (Lat, Gen, Disc). With more than 100 training instances, the transformer methods are fairly close, though at 64 and 100 they begin to separate, with BERT underperforming the variational methods, and BERT+LSTM surpassing all others. These gaps show the effectiveness of the pre-trained models, and the effectiveness of the variational training at lower levels of supervision.

4.3 Extreme Low-Label Learning

As the number of instances decreases, the accuracy of BERT becomes disproportionately lower than the other transformer methods. With 16-100 samples/label, BERT rapidly declines, yet the variational transformer methods provide some mitigation, e.g., at 32 samples BERT+GPT2 is 10% higher than BERT. At even lower levels, the RNN-based Latent and Generative methods outperform the transformer methods. While variationally-trained transformer methods can be effective in low-resource setting, RNN methods may be better in extremely low label settings.

5 Conclusion

We investigated the performance of discriminative and generative transformer models on the SE classification task, reporting new SOTA results. We showed that generative language modeling can be leveraged via latent variable learning into large improvements in the low-resource setting. Our work provides guidance and the foundation for future SE and low-label classification work.

References

Maria Becker, Alexis Palmer, and Anette Frank. 2017a. Semantic clause types and modality as features for argument analysis 1. *Argument & Computation*, 8(2):95–112.

Maria Becker, Michael Staniek, Vivi Nastase, Alexis Palmer, and Anette Frank. 2017b. Classifying semantic clause types: Modeling context and genre characteristics with recurrent neural networks and attention. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (* SEM 2017)*, pages 230–240.

Samuel Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21.

Gregory N Carlson and Francis Jeffry Pelletier. 1995. *The Generic Book*. University of Chicago Press.

Arman Cohan, Iz Beltagy, Daniel King, Bhavana Dalvi, and Daniel S Weld. 2019. Pretrained language models for sequential sentence classification. 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 3684–3690.

Zeyu Dai and Ruihong Huang. 2018. Building context-aware clause representations for situation entity type classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3305–3315, Brussels, Belgium. Association for Computational Linguistics.

Xiaoan Ding and Kevin Gimpel. 2019. Latent-variable generative models for data-efficient text classification. 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 507–517.

Aysu Ezen-Can. 2020. A comparison of lstm and bert for small corpus. *arXiv preprint arXiv:2009.05451*.

Annemarie Friedrich and Alexis Palmer. 2014. Situation entity annotation. In *Proceedings of LAW VIII - The 8th Linguistic Annotation Workshop*, pages 149–158, Dublin, Ireland. Association for Computational Linguistics and Dublin City University.

Annemarie Friedrich, Alexis Palmer, and Manfred Pinkal. 2016. Situation entity types: automatic classification of clause-level aspect. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1757–1768.

Venkata Govindarajan, Benjamin Van Durme, and Aaron Steven White. 2019. Decomposing generalization: Models of generic, habitual, and episodic statements. *Transactions of the Association for Computational Linguistics*, 7:501–517.

Nancy Ide, Collin Baker, Christiane Fellbaum, Charles Fillmore, and Rebecca Jane Passonneau. 2008. Masc: The manually annotated sub-corpus of american english. In *6th International Conference on Language Resources and Evaluation, LREC 2008*, pages 2455–2460. European Language Resources Association (ELRA).

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Cheolhyoung Lee, Kyunghyun Cho, and Wanmo Kang. 2019. Mixout: Effective regularization to fine-tune large-scale pretrained language models. *arXiv preprint arXiv:1909.11299*.

Chunyuan Li, Xiang Gao, Yuan Li, Baolin Peng, Xiujun Li, Yizhe Zhang, and Jianfeng Gao. 2020. Optimus: Organizing sentences via pre-trained modeling of a latent space. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4678–4699, Online. Association for Computational Linguistics.

Alexis Palmer and Annemarie Friedrich. 2014. Genre distinctions and discourse modes: Text types differ in their situation type distributions. In *Workshop on Frontiers and Connections between Argumentation Theory and NLP*.

Alexis Palmer, Elias Ponvert, Jason Baldridge, and Carlota Smith. 2007. A sequencing model for situation entity classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 896–903, Prague, Czech Republic. Association for Computational Linguistics.

Jason Phang, Thibault Févry, and Samuel R Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint arXiv:1811.01088*.

Krishna Chaitanya Sanagavarapu, Alakananda Vempala, and Eduardo Blanco. 2017. Determining whether and when people participate in the events they tweet about. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Huajie Shao, Shuochao Yao, Dachun Sun, Aston Zhang, Shengzhong Liu, Dongxin Liu, Jun Wang, and Tarek Abdelzaher. 2020. Controlvae: Controllable variational autoencoder. In *International Conference on Machine Learning*, pages 8655–8664. PMLR.

Carlota S Smith. 2003. *Modes of discourse: The local structure of texts*, volume 103. Cambridge University Press.

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 *Advances in neural information processing systems*, pages 5998–6008.

Alakananda Vempala, Eduardo Blanco, and Alexis Palmer. 2018. Determining event durations: Models and error analysis. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*.

Dani Yogatama, Chris Dyer, Wang Ling, and Phil Blunsom. 2017. Generative and discriminative text classification with recurrent neural networks. *arXiv preprint arXiv:1703.01898*.

SE type	MASC	Wiki	Count
STATE	49.8%	24.3%	18337
EVENT	24.3%	18.9%	9688
REPORT	4.8%	0.9%	1617
GENERIC	7.3%	49.7%	7582
GENERALIZING	3.8%	2.5%	1466
QUESTION	3.3%	0.1%	1056
IMPERATIVE	3.2%	0.2%	1046

Table 3: Dataset Statistics, as reported by Friedrich et al. (2016). The Count column shows the number of clauses per SE type.

A Dataset

The MASC+Wiki corpus consists of more than 40,000 sentences with their corresponding SE types (labels) from Wikipedia and MASC (Ide et al., 2008). There are $26,283$ training data , $6,571$ validation data, and $7,937$ test data. The details are shown in Table 3.

B Additional Implementation and Model Details

B.1 Implementation Details

All models were trained and tested on a single RTX-based GPU. No models required more than 11GB of memory and 5 hours of runtime. Models were trained until convergence, as determined by performance on the validation dataset.

B.2 BERT and GPT2 Structures

We use the basic (smaller) BERT model, with 12 layers, 12 self-attention heads, a hidden size of 768 and a total of 110 million parameters.

The GPT2 layer has L layers and the embedding dimension is H. One linear layer maps the latent $z \in \mathbb{R}^P$ layer to $h_{\text{Mem}} = W_m z$ and the other one $h'_{\text{Emb}} = h_{\text{Emb}} + W_D z$, where $W_m \in \mathbb{R}^{LH \times P}$ projects the latent z to the L layers and $W_D \in \mathbb{R}^{H \times P}$ projects the latent to the embedding space. Our latent dimension in these experiments is $P = 30$, $L = 12$ and $H = 768$.

B.3 Comparison of Generative and Discriminative Classifiers

The generative and latent baselines maximize the joint probability of tokens and labels. In the discriminative classifier the conditional probability of labels given documents is maximized $\sum_{\langle x,y \rangle \in \mathcal{D}} \log p(y|x)$ where x is encoded using an LSTM. We mention the generative classifier and

the latent-variable generative model and how they differ from each other and from the discriminative classifier in following subsections.

B.3.1 Class-based Language Model (Gen):

The objective of a generative classifier is to maximize the joint probability $\sum_{\langle x,y \rangle \in \mathcal{D}} \log p(x,y)$ where x and y represent a document of length T and its label, respectively. The factorization of the joint probability is given in equation 2.

$$p(x,y) = p(x \mid y)p(y), \qquad (2)$$

where

$$\log p(x \mid y) = \sum_{t=1}^{T} \log p(x_t \mid x_{<t}, y) \qquad (3)$$

The prediction of the next word x_{t+1} is done by concatenating the LSTM hidden states and the label embeddings and feeding them into a softmax layer. For prediction at inference, the discriminative classifier maximizes $p(y|x)$ with respect to y and the generative classifier maximizes $p(x|y)p(y)$.

B.3.2 Latent-Variable Generative Model (Lat):

Incorporating discrete latent variables into the standard generative classifier can be formulated as shown in equation 4,

$$p(x,y,c) = p_\Theta(x \mid c,y)p_\Phi(c)p_\Psi(y), \qquad (4)$$

where Θ and Φ are the set of parameters of the language model and the set of parameters for the prior distribution of the latent variable, respectively. Same as the generative classifier, $p_\Psi(y)$ is obtained from the empirical label distribution. The prior distribution of the latent variable is parameterized in equation 5:

$$p_\Phi(c) \propto \exp\{w_c^\top v_c + b_c\}. \qquad (5)$$

Same as the generative classifier, the prediction is done by an LSTM and a softmax layer.

$$p_\Theta(x_t \mid x_{<t}, c, y) \propto \exp\{u^\top([h_t; v_y; v_c]) + b\} \qquad (6)$$

Here, v_y and v_c are the embeddings for the label and the latent variable, and $[.;.]$ indicates vertical concatenation. The hidden representation, label, and latent embeddings are concatenated for the text

reconstruction. The training objective of the latent-variable generative model is to maximize the log marginal likelihood which is shown in equation 7.

$$\max_{\Theta,\Phi,V_{\mathcal{C}},V_{\mathcal{Y}}} \sum_{\langle x,y\rangle\in\mathcal{D}} \log \sum_{c\in\mathcal{C}} p(x \mid c, y)p(c)p(y) \quad (7)$$

All the embeddings and hidden representations are 100-dimensional in these baselines. The latent model requires the expensive and time consuming full marginalization over the latent variables c, which is a major obstacle for using this model with Transformers.

C Additional Insights

In this section we provide some additional insights to supplement the core results reported in the main paper.

In Fig. 3, we show confusion matrices for BERT+GPT2, as we vary the number of samples per label. Notice a diagonal pattern emerges relatively quickly, though there are large confusions, especially between STATIVE and GENERIC SE labels.

In the paper we mentioned that the MAP approximation we use when making (and learning to make) the predictions has a nice, qualitatively observed side-effect: that the training results in latent variables z that can be nicely clustered. This can be observed in Fig. 4, where we show t-SNE plots for the mean μ_z from the training set, as we vary the amount of supervision we have per label.
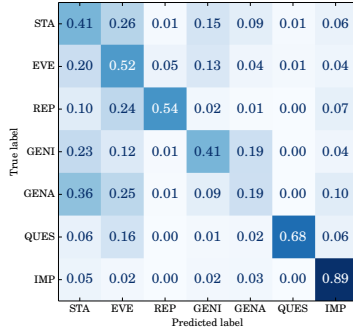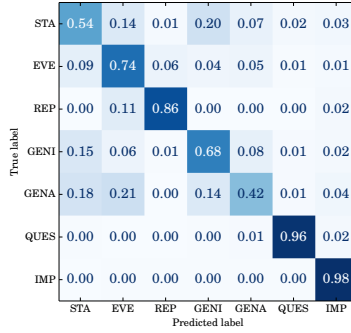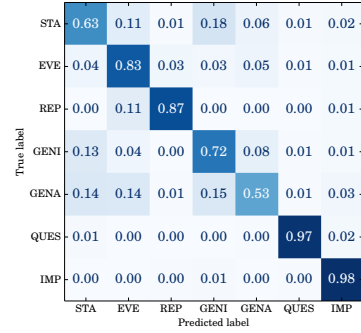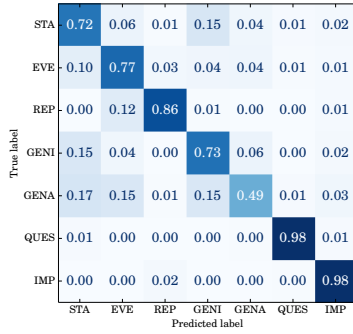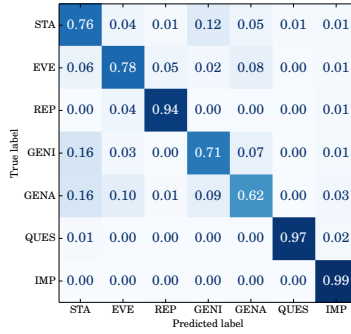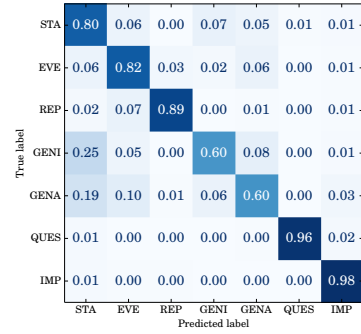
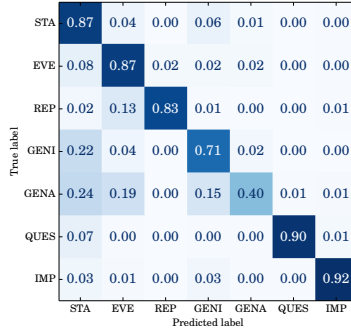Figure 3: Confusion matrices for BERT+GPT2, as we vary the number of samples per label. Notice a diagonal pattern emerges relatively quickly, though there are large confusions, especially between STATIVE and GENERIC SE labels.
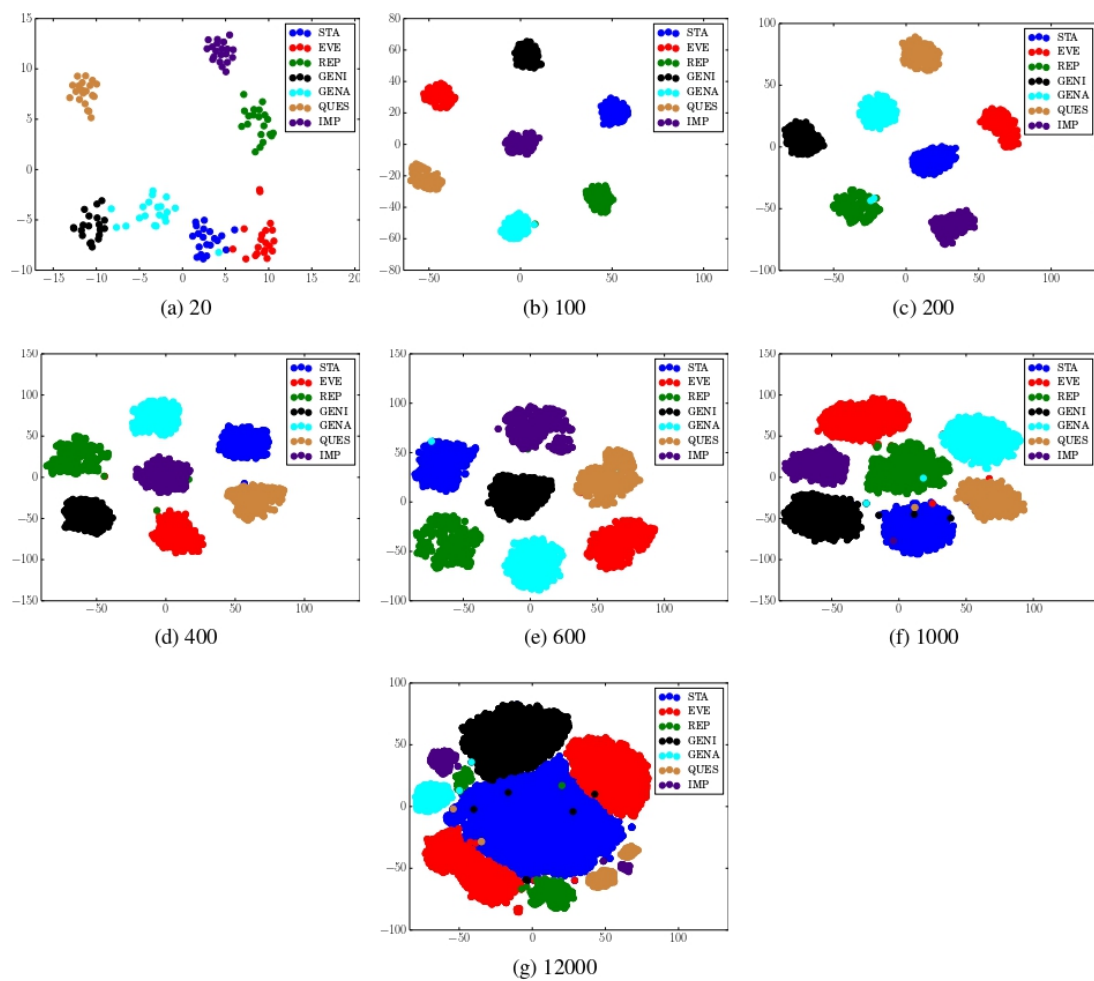
Figure 4: A series of t-SNE representations from training set clauses.