Open PhD position:
Deep multitask learning with latent structured prediction for natural language processing

Scientific context

The aim of multi-task learning is to improve the performance of a model for some task by exploiting signals available in training data and domain knowledge available for a related task (Caruana, 1998). Most tasks in natural language processing (NLP) are best modeled as structured prediction where output contains multiple variables, possibly with rich and complex interdependencies and constraints. Tasks including morphological, syntactic and semantic analysis as well as applications including question answering and machine translation, involve discrete combinatorial structures such as sequences, trees and graphs. Uncovering such hidden linguistic structure even as intermediate steps is important to inform downstream tasks and inject domain knowledge (He et al., 2017).

Deep neural networks are particularly suited for multi-task scenarios and have been consistently showing positive results since early work by (Collobert and Weston, 2008). The basic and most-studied mechanism is based on modeling each task separately while sharing parameters of general-purpose hidden layers between models. Different tasks are then trained separately in a supervised way. Shared parameters adapt to all tasks and predictions are made separately or from the same layer. Søgaard and Goldberg (2016) showed that sharing parameters is effective especially when the tasks are not similar or no overlapping data is available for training. Tasks are thus organised in a hierarchy or a pipeline with different layers corresponding to different tasks and supervision provided at the right level. This idea is easily generalized to architectures connecting many tasks such as POS tagging, chunking, dependency parsing, semantic relatedness, and textual entailment, etc. The tasks are organized in a predefined architecture based on linguistic hierarchies where increasingly complex tasks happen at successively deeper layers (Hashimoto et al., 2017). Learning may use separate objectives for each task with regularisation to avoid catastrophic forgetting. This approach can be further generalized as in Sluice Networks (Ruder et al., 2017). A more involved mechanism is based on joint modeling of the output from multiple tasks with similar structures which are scored jointly and learned using a joint objective from parallel annotations (Peng et al, 2017).

Creating annotated resources for all tasks is prohibitively expensive and supervised training reduces the ability of intermediate structures to adapt to the final output. Joint modeling can be extended to divergent structures learned from disjoint datasets by treating the missing structures as latent variables (Peng et al, 2018) thus allowing for unsupervised or partially-supervised settings. In fact, in the absence of annotations for some (possibly intermediate) tasks, dense representations of the corresponding structures may be implicitly learned by the hidden layers of powerful models such as BERT. Interpreting these representations as structures requires training a separate model however (Hewitt and Manning, 2019). Alternatively, one can explicitly model latent structures of different tasks as separate components and building blocks connected together in a deep architecture that is end-to-end trainable. Hidden structured variables are inferred from input and used to dynamically define the network's computation graph. These structure predictors need to be differentiable to allow for learning by backpropagation of error gradients. Furthermore, inference needs to be tractable in order to handle the exponential number of distinct structures. Several approaches can be considered to strike a balance between expressivity, differentiability and tractability. These include (a) reinforcement learning with policy gradient networks (Yogatama et al., 2017) (b) the use of marginal inference in graphical models with softmax layers which is differentiable. Marginalization involves summing over exponentially many structures which is tractable for specific structures such as sequences and trees as in structured attention networks (Kim et al., 2017); (c) the use of structured argmax layers with maximum a posteriori inference which is not differentiable but can be replaced with differentiable optimisation methods as SPIGOT (Peng et al., 2018b); (LP-)SparseMAP (Niculae et al., 2018; Niculae and Martins, 2020) and differentiable dynamic programming (Mensh and Blondel, 2018; Corro and Titov, 2019).

Applicative context

Depending on the selected profile, we will apply the developed ideas from multitasking and latent structured prediction in deep neural networks to one of two application domains: information extraction for knowledge graph construction, or Arabic NLP. In both cases we will build a hierarchy of interdependent tasks with a downstream application The architecture is trained end-to-end with available resources for supervised tasks and latent structured variables for unsupervised ones.

References

- Rich Caruana. 1997. Multitask learning. Machine Learning 28(1):41–75.
- Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In Proc. of ICML.
- Nizar Habash. 2010. Introduction to Arabic Natural Language Processing. Synthesis Lectures on Human Language Technologies Morgan & Claypool Publishers.
- Anders Søgaard and Yoav Goldberg. 2016. Deep multi-task learning with low level tasks supervised at lower layers. In Proceedings of ACL.
- Dima Taji, Nizar Habash, and Daniel Zeman. 2017. Universal Dependencies for Arabic. In WANLP.
- Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and what's next. In Proc. of ACL.
- Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A Joint ManyTask Model: Growing a Neural Network for Multiple NLP Tasks. In Proceedings of EMNLP.
- Hao Peng, Sam Thomson, and Noah A. Smith. 2017. Deep multitask learning for semantic dependency parsing. In Proc. of ACL.
- Dani Yogatama, Phil Blunsom, Chris Dyer, Edward Grefenstette, and Wang Ling. 2017. Learning to Compose Words into Sentences with Reinforcement Learning. In ICLR.
- Hao Peng, Sam Thomson, Swabha Swayamdipta, and Noah A. Smith. 2018. Learning joint semantic parsers from disjoint data. In Proc. of NAACL.

- Hao Peng, Sam Thomson, and Noah A. Smith. 2018. Backpropagating through structured argmax using a spigot. In Proceedings of ACL.
- Niculae, V., Martins, A. F., Blondel, M., and Cardie, C. 2018. SparseMAP: Differentiable sparse structured inference. In Proc. of ICML.
- Arthur Mensch and Mathieu Blondel. 2018. Differentiable dynamic programming for structured prediction and attention. In ICML.
- J. Hewitt and C. D. Manning. A structural probe for finding syntax in word representations. In Proceedings of NAACL 2019.
- Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. Latent multi-task architecture learning. In AAAI, 2019.
- Caio Corro, and Ivan Titov. 2019. Learning latent trees with stochastic perturbations and differentiable dynamic programming. In Proc. of ACL.
- V. Niculae and A. F. T. Martins. LP-SparseMAP: Differentiable relaxed optimization for sparse structured prediction, 2020.