Research Statement

Jay-Yoon Lee,
Ph.D., Carnegie Mellon University,
Postdoctoral associate, University of Massachusetts Amherst

Machine learning models have shown remarkable advancements in the last decade, setting new state-of-the-art performance on core natural language processing (NLP) tasks. We are finally one step closer toward NLP models that can perform knowledge extraction at scale across millions of documents and communicate this knowledge to guide humans in decision-making. However, **important piece is missing for the safe and practical use of NLP systems: the injection of human knowledge *into* the model.**

Currently, **even the best performing NLP models violate simple constraints**, such as grammar rules (Linzen et al., 2016), syntactic constraints (**Lee** et al., 2019), and first-order logic (Hwang et al., 2022). In question answering, for example, the questions "Is A *larger* than B?" and "Is B *smaller* than A?" should have the same answer. However, current NLP models cannot easily satisfy these simple logical constraints. (Asai & Hajishirzi, 2020). When NLP models cannot even guarantee to follow such simple constraints, it is difficult to trust them in production, where models are required to analyze multiple documents from various domains, especially for safety-critical domains such as medicine and law.

My research goal is to create a coherent communication and analysis methods between humans and machines across multiple instances, domains, and tasks. I aim to build NLP models that domain experts can rely on for scientific and sociological research, and that can be deployed to safety-critical applications such as medical diagnosis and prescription. **Toward this end, my research provides a mechanism for injecting human knowledge that can enforce the models to be consistent and safe**. Furthermore, for NLP models to be applicable across millions of data sources at scale, I am interested in **generalizing models to multiple domains** and **building computationally efficient models**.

To date, my work has provided an **algorithm for enforcing arbitrary constraints at inference time** (§1), showing a significant improvement in performance at a faster speed (higher constraint satisfaction rate, 5x more improvements, at 1.4x speed compared to the previous best method). I have also worked on **injecting constraints at training time, directly embedding human knowledge into the model parameters** (§1). I further showed that benefits from the two methods are complementary rather than exclusive: the performance is best when training-time and inference-time methods are combined. These constraint injection methods mentioned above further demonstrated their **effectiveness for low-resource, and domain adaptation (§2) as constraints provide a way to utilize unlabeled data**, showing performance gains ranging from 6% to 24% depending on the task and domains. I also explored models that could **automatically learn implicit constraints** without explicitly injecting them (§3).

1 Injecting constraints into machine learning models

Injecting human knowledge into neural networks is a crucial but non-trivial task as **neural networks are often treated as a black-box function, making their inner workings uninterpretable**. Ironically, while this black-box treatment of neural networks made them famous for the ease of modeling, it gave us less controllability on the output space as its relation to the model parameters is unclear. Furthermore, human knowledge on a task about logical rules, common sense, or structural **constraints is not explicit in the training data** itself. Thus models are only trained on constraint-satisfying instances; they are thus not trained to discriminate constraint violating outputs. Lastly, symbolic constraints are often non-differentiable.

To overcome the unclear dependency between output variables and model parameters in neural networks, I first take a model-agnostic approach of injecting knowledge through loss functions (*constraint loss*). Building separate model structures per task and constraint would be too expensive and less applicable. My works (**Lee** et al., 2019; **Lee*** et al., 2018, 2020; Xu et al., 2021) take a similar approach to reinforcement learning, where a

scoring function is used to promote or penalize a model's behavior. Specifically, evaluation of constraint error rate on the output of a model is used as a scoring function, a simple way of expressing abstract knowledge of constraint.

Within the above framework, I first developed a **gradient-based inference (GBI) algorithm (Lee et al., 2019) *for arbitrary constraints* that can enforce constraints at test time**. Existing methods for enforcing constraints consist of distinct post-processing with heuristic rules or efficiently searching the output space with dynamic programming (e.g. A* algorithm). Unfortunately, the output is a discrete sequence, making the search combinatorial. Instead, I showed that my constraint loss approach can search for continuous model parameters to "produce" constraint-satisfying output. This directly utilizes the modern neural network framework that is optimized for efficient gradient-based search in the continuous space and **showed better (*5x more gain*) and faster (*1.4x faster*) performance while converting more instances to become constraint satisfying,** compared to the best previous method (A*). Additionally, **this GBI method showed robust performance to noisy constraint** information as it utilizes constraint in a soft manner rather than directly searching on discrete space.

Second, I have also shown that this **constraint loss can be successfully utilized at training time: to improve performance and reduce the constraint violation rate (Lee* et al., 2018; Xu et al., 2021)**. Furthermore, through ablation studies, I show that improvements from **training-time and test-time constraint injections are complementary: the performance is best when two efforts are combined (Lee* et al., 2018)**.

In these previous experiments, I have also portrayed that **improving coherence across different NLP tasks can also result in better model performance**. Many NLP tasks are related such as word chunking, named entity detection, syntactic parsing, and semantic role labeling, where all these tasks require consistent span segmentations. Previously, the research community tried leveraging task similarity through multi-task learning (MTL) in learning a common representation for multiple tasks. Nonetheless, they have not ensured the coherence of multi-task outputs, which is crucial for reliability. In my work, I have enforced semantic-role-labeling models to conform to syntactic information (Lee* et al., 2018) and syntactic parsing models to abide to previously known span information, e.g., named entity recognition (Xu et al., 2021). In addition to this, with collaborators, I am studying whether an information-extraction task can benefit from enforcing semantic consistency to the similar task of abstract meaning representation (AMR) parsing. All of these examples showed **improved inter-task consistency and improved model performances** after injecting coherence constraint.

In addition to the model-agnostic approach, with collaborators, I have also worked on reflecting human knowledge about a task to the model formulation (Balachandran et al., 2021; Hwang et al., 2022). For instance, **I proposed a relation extraction model that can satisfy first-order logic constraints such as symmetry and transitivity, by utilizing the spatial box (Vilnis et al., 2018) model that can capture antisymmetric relationships**. Suppose we classify an event A occurred *before* event B, then by symmetry, it should be trivial that event B occurred *after* event A. Surprisingly, the state-of-the-art event relation extraction models do not satisfy these constraints ($>$ 30% error). The problem resides in the vector representation's incapability to express antisymmetric relationships. To overcome this challenge, collaborators and I have proposed to work with **spatial box representation that can enforce symmetry constraint errors to 0% and also transitivity constraint violation to less than 1%** (Hwang et al., 2022).

2 Generalizing to low-resource and new domains

For the widespread deployment of NLP models in production, we require models that can perform well on diverse domains, with different text distributions and/or limited annotation. For example, we want NLP models to understand both conversational dialog well and be able to extract information from a scientific text. In addition, most of the languages in the world (7,000 of them) do not enjoy as rich annotation as other popular languages such as English or other European languages. Ultimately, I want the global community to have access to confident NLP models.

Through various examples, I have shown that injecting human knowledge as a constraint can provide a learning signal on unlabeled data, which is valuable in low-resource setups with very scarce training data. (1) I have shown that **injecting syntactic information as a constraint** (as discussed in §1) can increase the performance of low-resource (1% train set) semantic role labeling by more than 5% while reducing nearly 60% of constraint violation (**Lee** et al., 2019; **Lee**, 2020). (2) Rather than injecting task-specific knowledge, I've injected a general constraint that **outputs of multi-view models should be coherent** (**Lee*** et al., 2020) to improve extremely low-resource (50, 100 labeled examples per each language) dependency parsing. I examined this approach over 9 languages which showed more than 10% average improvement. (3) For low-resource models to truly improve, the role of unsupervised learning that can utilize extensive unlabeled data is vital. With collaborators, I have shown that **constraint injection can work on top of an unsupervised learning regime** with the application of syntactic parsing (Xu et al., 2021). Annotating a whole syntactic parse tree or semantic structure (SRL, AMR, etc.) is difficult as it requires the work of linguistic experts. Nonetheless, span information such as 'Niagara Falls,' 'New York,' and 'Incheon airport' is easy to collect when such information does not already exist. Injecting a constraint that the model's output should conform to these spans (Figure 1), we improved the syntactic parsing performance by more than 20%.

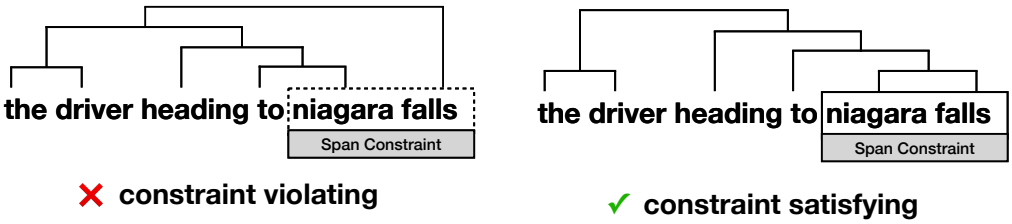


Figure 1: An example of a parse tree that is incorrect, where the 'Niagara falls' part fails the span constraint. The figure on the right shows a gold parse tree which was recovered after applying span constraint. By learning to mimic constraint-satisfying examples and by penalizing constraint-violating examples, the parsing model can learn from light span annotation.

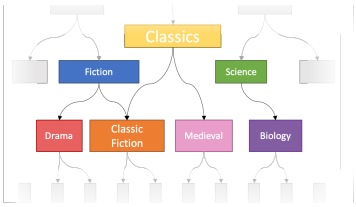
Domain adaptation is another technique often used to mitigate the low-resource problem. It learns a confident model in a rich-resource domain and tries to transfer this model to another domain. The technique requires mitigating distributional differences between domains. In my Ph.D. thesis (**Lee**, 2020), I also exhibit that **enforcing constraints utilizing unlabeled data from the target domain can reduce the gap between the target and source domain**. For instance, the SRL example mentioned above shows that the performance gap reduces by 63% when we apply learning with constraints. Enforcing constraints at inference time also shows that it can reduce the performance gap even further (by 80%).

3 Capturing implicit knowledge and constraints in an efficient manner

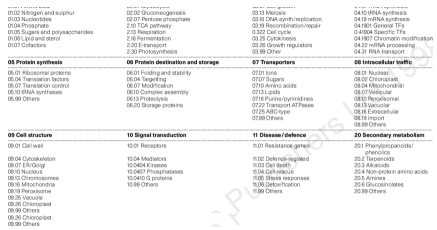
In addition to the structural and logical constraints we discussed in §1 and §2, various dependencies exist on the label space. For example, as displayed in Figure 2, the constraint can be simple, such as "a book genre cannot be classified as '*science fiction*' and '*non-fiction*' (Aly et al., 2019) at the same time", or complicated, such as enforcing the fact that '*lactate fermentation*' is not under the category of '*respiratory*' with respect to the taxonomy of *functional proteins* (Mewes et al., 2002).

When thousands of constraints exist, it would be nearly impossible to acquire them explicitly for two reasons. First, annotating label relationships for thousands of labels is a costly process, especially when the process requires domain expertise. Even if annotation is possible for one or two datasets, it would be unrealistic to expect similar information for every dataset. Second, existing methods for injecting constraints on such a large scale have not yet been well studied. For example, traditional probabilistic methods have shown limitations for

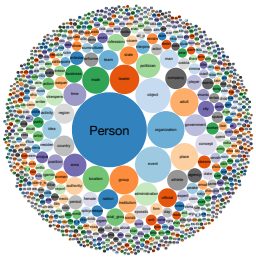
modeling large-scale probability distributions containing thousands of random variables. As these constraints are rarely provided to us, we need to build a model that can automatically capture dependencies between output variables as much as possible – this requires both computational and statistical efficiency.



(a) Book genre collection (Aly et al., 2019)



(b) Functional proteins (Mewes et al., 2002)



(c) Fine-grained entity types (Choi et al., 2018)

Figure 2: The label taxonomy for various domains.

To automatically and efficiently capture hidden label dependencies, I have worked on **utilizing energy network models as a teacher network** (Lee et al., 2021). Energy networks (Belanger & McCallum, 2016; Gygli et al., 2017) have shown that they can capture arbitrary dependencies amongst the output variables without explicitly injecting them in a statistically efficient manner – displaying better performance than standard probabilistic models. Despite their success, energy networks need to traverse complex energy surfaces at inference time, leading to unstable and slower inference. To take advantage of the expressivity of energy networks without incurring the high inference cost, I have proposed Structured Energy As Loss (SEAL): **a general framework that utilizes energy networks as a teacher network to teach standard student networks**. By utilizing SEAL on multi-label classification, in Lee et al. (2021), I have shown that this method can capture most of the correct label dependencies, enjoy a fast inference of standard feedforward network, and perform better than both models (individual energy network and feedforward model). In extreme cases, we saw performance improvement of 100% when particular patterns exist in the label space.

As an orthogonal effort to capture label dependencies automatically, I have worked on a spatial representation model that can efficiently represent hierarchical or graph relationships; in contrast, vector representations typically exhibit difficulty in capturing these asymmetric relationships between labels. With collaborators, I have shown that **box representations can successfully recover the true label taxonomy without utilizing this information at training time** (Patel et al., 2022). The experimental results show that outputs from box representations have far fewer label-taxonomic violations, **making predictions from box models more coherent and trustworthy** than standard vector representations.

4 Future Work

For NLP models to **interactively extract information from massive data coherently while communicating with humans in the loop**, there are many challenges to be resolved. I will establish the essential steps for overcoming these challenges in the following paragraphs.

Evaluation framework for measuring coherence: **I will build an evaluation framework that can assess the coherence of NLP models**. As discussed earlier, there is a slew of evidence that NLP systems lack coherence across multiple instances (e.g. symmetric or transitive relations, see §1) and tasks. However, current evaluations are limited to per-instance metrics, eg., evaluating if the model output is close to annotated output at the sentence level. This is problematic from a user's perspective, because the real usability of a system, *effective accuracy*, is often a function of accuracy and coherence. For example, consider two models with different accuracy and coherence rates: model A (70% accuracy with 50% coherence) and model B (60% accuracy with

100% coherence). While model A has higher accuracy, 50% coherence means that the prediction set will have to be disregarded in one out of two cases as users would not be able to trust conflicting outputs. This results in the effective accuracy of A being less than 50% and B being 60%. **In summary, this example displays a fundamental limitation of current evaluation metrics**. While model A would seem to perform better on the surface when using standard evaluation metrics which measures the "individual correctness", the actual utility indicates that model B (60%) is more useful than A (<50%). The analysis above is simplified since it is a basic abstract example. I will create a framework to analyze coherence and incorporate coherence into a metric for effective accuracy. **Ultimately, by building this framework, I will encourage the research community to focus on evaluating and designing coherent NLP models, leading to greater trust and adoption**.

Identifying faulty information: Blindly enforcing coherence can lead to faulty conclusions when conflicting information exists. Therefore, when compiling multiple sources of information, identifying and screening faulty sentences or documents is essential. One possible way of detecting anomalous information could be conducted by comparing information extracted by natural language understanding (NLU) models across different subsets of documents.[1] The documents that frequently contribute to conflicting NLU output when the rest of the subsets agree are more likely to be faulty. Advancements in internet technology have made it easier for individuals to share and access fake news and computational propaganda, creating confusion. **This line of research will have a high societal impact, as determining whether a piece of information is suspicious or fake is an increasingly important issue in the modern world**.

Coherent multi-task learning: As discussed in §1, while previous multi-task learning (MTL) focused on sharing parameters, they have not enforced the consistency of labels on the joint label space across tasks (e.g., SRL and parse tree, parse tree and NER). To overcome this problem, I have increased inter-task consistency by enforcing primary task output (e.g., SRL) to be consistent with secondary task output (parse tree) of the same input sentence. While effective for inter-task consistency, this solution may limit the learning experience of the model as it requires a separate model per task, removing the opportunity to share common aspects of the tasks, the main benefit of MTL. **Toward this end, I plan to bring the best of both worlds by combining MTL and inter-task-consistency injection (*consistent MTL*). I propose to examine whether this approach can perform self-supervised learning where one task output can guide the learning of another task on unlabeled data**. If successful, consistent MTL would lead to engineering efficiency as there will be fewer models to store, and expensive joint-task inference won't be required as the models will produce outputs with high inter-task consistency.

Multi-modal learning: While we mainly discussed coherence in the label space, it is also crucial to enforce coherence between feature spaces. This will be especially important in solving multi-modal tasks where heterogeneous features such as video, text, acoustic, and multiple sensors are provided as input. I am interested in extending my previous work of injecting agreement constraints between multi-view models (**Lee*** et al., 2020) toward multi-modal learning. There exists many more feature types in multi-modal tasks (e.g., 10 modalities in ego4d dataset) compared to the previous multi-view work, and not all of them are always present. **Thus, the process of effectively selecting meaningful sets of modalities will be required for injecting consistency.** Comparing all combinations of modalities would be computationally infeasible and also would not add much value due to relevance. I have started exploring new aspects of multi-modal tasks through mentoring Master students in collaboration with Meta.

Scientific domains for constraint injection: **I am also interested in broadening my research towards applications in scientific domains requiring information extraction and modeling scientific processes.** There are active efforts in chemistry and biology research to utilize AI models by expressing reactions into a sequence of symbols, similar to how text is modeled in NLP. In finding new molecules to new reaction paths, I hope to collaborate with domain experts to provide progress in scientific research. As a preliminary step, I am mentoring MS students with IBM on the related topic: *retrosynthesis*, a task of finding how to synthesize a

[1] For example, given N documents, we could create subsets where each consists of k documents (We can have N-choose-k such subsets). Ideally, all of these subsets will not have any conflict.

target molecule, and have collaborated with PhD students working on information extraction from large corpora of scientific literature in material science.

References

Aly, R., Remus, S., and Biemann, C. Hierarchical multi-label classification of text with capsule networks. In *ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop*, pp. 323–330, 2019. ISBN 9781950737475. doi: 10.18653/v1/p19-2045. URL https://github.

Asai, A. and Hajishirzi, H. Logic-guided data augmentation and regularization for consistent question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5642–5650, 2020.

Balachandran, V., Pagnoni, A., **Lee**, J.-Y., Rajagopal, D., Carbonell, J. G., and Tsvetkov, Y. StructSum: Summarization via Structured Representations. In *EACL (Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume)*, pp. 2575–2585, 2021.

Belanger, D. and McCallum, A. Structured prediction energy networks. In *33rd International Conference on Machine Learning, ICML 2016*, volume 3, pp. 1545–1554, 2016. ISBN 9781510829008.

Choi, E., Levy, O., Choi, Y., and Zettlemoyer, L. Ultra-fine entity typing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 87–96, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1009. URL https://aclanthology.org/P18-1009.

Gygli, M., Norouzi, M., and Angelova, A. Deep value networks learn to evaluate and iteratively refine structured outputs. In *34th International Conference on Machine Learning, ICML 2017*, volume 3, pp. 2160–2170, 2017. ISBN 9781510855144. URL https://goo.gl/8OLufh.

Hwang, E., **Lee**, J.-Y., Yang, T., Patel, D., Zhang, D., and McCallum, A. Event-Event Relation Extraction using Probabilistic Box Embedding. In *ACL (Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics)*, 2022.

Linzen, T., Dupoux, E., and Goldberg, Y. Assessing the ability of lstms to learn syntax-sensitive dependencies. *Transactions of the Association for Computational Linguistics*, 4:521–535, 2016.

Mewes, H. W., Frishman, D., Güldener, U., Mannhaupt, G., Mayer, K., Mokrejs, M., Morgenstern, B., Münsterkötter, M., Rudd, S., and Weil, B. MIPS: a database for genomes and protein sequences. *Nucleic Acids Research*, 30(1):31–34, 01 2002. ISSN 0305-1048. doi: 10.1093/nar/30.1.31. URL https://doi.org/10.1093/nar/30.1.31.

Patel, D., Dangati, P., **Lee**, J.-Y., Boratko, M., and McCallum, A. Modeling Label Space Interactions in Multi-label Classification using Box Embeddings. In *ICLR (International Conference on Learning Representations)*, 2022.

Lee, J.-Y. *Injecting output constraints into neural NLP models*. PhD thesis, Carnegie Mellon University, May 2020.

Lee*, J.-Y., Mehta*, S. V., and Carbonell, J. G. Towards Semi-Supervised Learning for Deep Semantic Role Labeling. In *EMNLP (Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing)*, pp. 4958–4963, 2018.

Lee, J.-Y., Mehta, S. V., Wick, M., Tristan, J.-B., and Carbonell, J. Gradient-based inference for networks with output constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4147–4154, 2019.

Lee*, J.-Y., Lim*, K., Carbonell, J., and Poibeau, T. Semi-supervised learning on meta structure: Multi-task tagging and parsing in low-resource scenarios. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 8344–8351, 2020.

Lee, J.-Y., Patel, D., Goyal, P., and McCallum, A. Structured Energy Network as a dynamic loss function: A case study with multi-label Classification. *Openreview preprint*, 2021.

Vilnis, L., Li, X., Murty, S., and McCallum, A. Probabilistic embedding of knowledge graphs with box lattice measures. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 263–272, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1025. URL https://aclanthology.org/P18-1025.

Xu, Z., Drozdov, A., **Lee**, J.-Y., O'Gorman, T., Rongali, S., Finkbeiner, D., Suresh, S., Iyyer, M., and McCallum, A. Improved Latent Tree Induction with Distant Supervision via Span Constraints. In *EMNLP (Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing)*, 2021.