Jacob Andreas

Research Statement

My research uses language to inform and explain the computational structure of machine learning models. Computation is the core of language—ask a linguist how to represent the meaning of a sentence like *There are three eggs left in the refrigerator*, and she is likely to respond with a computation: a λ-calculus expression that takes the state of the refrigerator as input and outputs whether the sentence is true. When we design question-answering assistants and instruction-following agents, what we're really building are systems that translate from natural language to computations: one-off programs that can be executed against a database or on a robot to produce behavior. The structure of this computation is closely related to the structure of the language that generates it. Individual words like *three* or *in* correspond to reusable primitive functions. Local syntactic coherence in language (like *three eggs*) provides information about the algorithmic structure it induces. What does this tell us about how to build learning systems that interact with humans through language? I'm interested in a few different ways of asking this question:

- How can linguistic structure guide the structure of models for language processing?
- How can language help explain the computations performed by black-box learned models?
- How can language data supply reusable knowledge about the world for general learning problems?

Below I describe some first steps toward answers.

Model structure from linguistic structure

Suppose we want to build a tool to answer natural language queries about images—to support large-scale data analysis of photographic archives, or perhaps to provide navigation assistance to the visually impaired. Users might ask things like *Is the building red?* or *How many arches are between the two red doors?* At some level of representation, these two questions specify fundamentally different computations. But they also share common substructure (e.g. the ability to evaluate *red*ness). Effective language understanding depends on *compositionality*: the ability to reuse this substructure in new contexts. General-purpose question answering is challenging because it requires models to support this kind of flexible compositionality, but also to draw inferences from unstructured inputs like photographs as well as structured knowledge sources like databases.

Most of my work on question answering has been based around a class of deep models called neural module networks ("NMNs") that I introduced in 2016. NMN training induces an inventory of shallow network fragments or "modules" that can be freely assembled into new, input-specific structures. NMNs provide discrete compositionality as a first-class operation, while supporting the kind of end-to-end learning that we've discovered is essential for tasks like computer vision. They can be generated from natural language syntax or predicted directly given appropriate supervision. We were excited to see NMNs produce one of the early successes on a large-scale visual question answering task [Andreas et al., 2016b]. NMN-based approaches remain at or near the state of the art for a wide variety of question answering problems featuring structured knowledge bases [Andreas et al., 2016a], natural images [Hu, Andreas, et al., 2017] and complex reasoning

[Johnson et al., 2017, Suhr et al., 2017]. Other researchers have found that NMNs specifically exhibit better generalization on complex question categories [Agrawal et al., 2017] and improved robustness to adversarial attacks [Xu et al., 2017].

I think one of the key things we've lost in the deep learning era is a compositional language for specifying models. Many other machine learning model families, including factor graphs and probabilistic programs, are inherently compositional: because their variables have known semantics, we can easily take fragments of models we've trained previously, stitch them together at their common interfaces, and obtain sensible answers from the new models without retraining. By design, deep learning approaches can never produce variables with the same kind of explicit semantics. Results with module networks hint at the possibility that some kind of explicit compositional mechanism for deep learning is still possible, but there's lots of work needed to make this compositionality robust and broadly applicable. Our understanding of natural language syntax provides hints about how this might be done.

Explainable models with natural language

Suppose that a question answering system like the one described above is trained to assist in medical diagnosis. Such a system will occasionally produce incorrect answers; when this happens, physicians need some way of understanding how the model came to its decision. How can we most effectively help users understand the behavior of learned models?

My work on interpretability has focused on producing natural language model explanations. There are deep connections between the explanation problem and linguistic accounts of pragmatics, which attempt to explain (among other things) how human speakers generate informative, accurate, and concise language. Using communication between humans as a reference frame provides a powerful framework for formalizing the notions of adequacy and truthfulness that are core to interpretable machine learning.

My first project in this area investigated models with an explicit communication component. A number of approaches exist for learning multiagent systems in which agents communicate using a kind of "neuralese" language—a vector-valued protocol automatically induced by the training procedure. Because this communication protocol is learned rather than hand-specified, the messages sent by agents cannot be straightforwardly interpreted by humans. My work aimed to make these systems understandable by translating neuralese into natural language. Standard machine translation techniques cannot be applied to the problem because of the absence of parallel training data. I instead developed a technique for learning a translation system using only examples of *unaligned* agent–agent and human–human interaction [Andreas et al., 2017a]. This technique turns out to be useful for analyzing not just multiagent systems, but machine learning models more generally. My most recent work in this area has focused on extending the technique to discover and describe interpretable structure in the representations learned by generic sequence-to-sequence architectures [Andreas and Klein, 2017].

Language as a scaffold for learning

The conceptual scaffolding provided by language reveals something about the structure of the world: for example, the fact that it's easy to communicate the concept *left of the circle* but comparatively difficult to communicate *mean saturation of the first five pixels in the third column* tells us

about the set of abstractions that humans find useful for navigating the visual environment. I'm interested in using this kind of linguistic background knowledge as an informative prior even for learning problems like image classification and policy search that do not directly involve language.

My first research project in this direction focused on learning behaviors for interactive agents, using lightweight "sketches" of the desired behavior [Andreas et al., 2017c]. While past work has attempted to guide learning with demonstrations or detailed supervision about which actions are likely to lead to success (e.g. *move joint #17 five degrees to the left*), the key insight here was that standard learning algorithms are perfectly capable of inferring this low-level structure if only told something about the high-level structure of their tasks (e.g. *collect wood, then carry it to the workbench*). Models trained with sketches learn faster than conventional reinforcement learners; most importantly, they acquire a library of reusable high-level actions that allow them to adapt better to new tasks even when annotation is no longer present. In other words, with language-like guidance, our models were able to learn something reusable and generalizable about the world.

My recent focus has been on developing even more flexible ways of using linguistic side-information to improve the performance of general-purpose machine learning models [Andreas et al., 2017b]. I've found that it's possible to enhance models for a variety of tasks—image classification, programming by demonstration, and policy learning—with learning algorithms that describe their solutions with natural language strings rather than weight matrices. In short, by equipping models with the ability to explain themselves to people while learning, they become both more comprehensible and more accurate.

Next steps

Two broad themes run through the work I've outlined above: language as a source of prior knowledge for machine learning, and language as an explanatory tool for understanding learned models. I believe that the two go hand-in-hand—the kinds of inductive bias imposed by a demand for interpretability are often precisely those that encourage effective generalization.

Even more broadly: for the last two decades or so, natural language processing as a field has largely been concerned with what machine learning can do for language data. Effective practice has involved synthesizing general-purpose learning techniques with linguistic domain knowledge, and has fundamentally been about constructing learning systems that consume language data as input or produce it as output.

With the addition of representation learning techniques to the NLP practitioner's toolkit—and in particular with the availability of components that can effectively serve as black-box transducers between arbitrary input types like strings, images, and parameter vectors—I think we can take a more expansive view of the role of NLP within the larger AI ecosystem. Of course, there are still lots of unsolved problems involving language data! I'm especially interested in continuing to explore the role of structured inference procedures for tasks like parsing and language generation. But after my initial work on approaches that use text as a tool to improve model accuracy and interpretability, I'm equally excited to keep working on problems like few-shot learning and policy search through a language processing lens—for example, using language to discover task hierarchies, exploration strategies, or disentangled feature representations. More generally, I suspect that the next two decades will be defined as much by what language can do for learning as by what learning can do for language.

References

[Agrawal et al., 2017] Agrawal, A., Kembhavi, A., Batra, D., and Parikh, D. (2017). C-VQA: A compositional split of the visual question answering v1.0 dataset. *arXiv preprint arXiv:1704.08243*.

[Andreas et al., 2017a] Andreas, J., Dragan, A., and Klein, D. (2017a). Translating neuralese. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

[Andreas and Klein, 2017] Andreas, J. and Klein, D. (2017). Analogs of linguistic structure in deep representations. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[Andreas et al., 2017b] Andreas, J., Klein, D., and Levine, S. (2017b). Learning with latent language. *arXiv preprint arXiv:1711.00482*.

[Andreas et al., 2017c] Andreas, J., Klein, D., and Levine, S. (2017c). Modular multitask reinforcement learning with policy sketches. In *Proceedings of the International Conference on Machine Learning*.

[Andreas et al., 2016a] Andreas, J., Rohrbach, M., Darrell, T., and Klein, D. (2016a). Learning to compose neural networks for question answering. In *Proceedings of the Annual Meeting of the North American Chapter of the Association for Computational Linguistics*.

[Andreas et al., 2016b] Andreas, J., Rohrbach, M., Darrell, T., and Klein, D. (2016b). Neural module networks. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*.

[Hu et al., 2017] Hu, R., Andreas, J., Rohrbach, M., Darrell, T., and Saenko, K. (2017). Learning to reason: End-to-end module networks for visual question answering. In *Proceedings of the International Conference on Computer Vision*.

[Johnson et al., 2017] Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. (2017). CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. *Proceedings of the Conference on Computer Vision and Pattern Recognition*.

[Suhr et al., 2017] Suhr, A., Lewis, M., Yeh, J., and Artzi, Y. (2017). A corpus of natural language for visual reasoning. In *55th Annual Meeting of the Association for Computational Linguistics, ACL*.

[Xu et al., 2017] Xu, X., Chen, X., Liu, C., Rohrbach, A., Darell, T., and Song, D. (2017). Can you fool AI with adversarial examples on a visual Turing test? *arXiv preprint arXiv:1709.08693*.