Research Statement

Ph.D. applicant *Sidak Pal Singh*

In my opinion, there are three essential prerequisites for successfully bringing artificial intelligence to the real world, via machine learning (ML). (♠) Learning representations of entities across the data modalities that effectively captures their meaning, especially in domains or environments with scarce data. (♠) Continual transfer of the knowledge gained from learning and optimizing on past tasks (or objectives), while being robust to settings that might be adversarial. (♠) Doing all of this in a resource-efficient manner so that these systems can actually be built, rather than remain lofty dreams in the GPU fairyland!

My aim is to develop a practical toolkit, backed by strong theoretical guarantees, to address these central problems that underly ML, particularly in the context of deep neural networks. In short, this is why I want to pursue a PhD. Below, I describe the key research projects that have shaped my interests, as well as give an outlook on some other questions that I would like to investigate in the future.

(♠) Representation learning. One of the biggest factors behind the recent success in ML has been the use of representations learned by deep neural networks. In particular, this ranges from text embeddings learned via LSTMs or Transformers to image representations obtained via CNNs. A common idea across these methods is to map each entity (or object) whose representation is being learned, to a single vector embedding. However, an important issue with this approach is the inability to capture the various semantics or contexts of an entity within this vector embedding itself. This has mainly led to works that either maintain multiple embeddings or learn embeddings that capture some specific information. Nevertheless, a general-purpose way to obtain powerful unsupervised representations remains to be found.

Together with my colleagues at EPFL, we proposed an alternative called, *Context Mover's Distance & Barycenters* (Singh et al., 2019a). We represent each entity (like words) as a distribution over its contexts, and the contexts themselves are vector embeddings in a low-dimensional space. This allows us to cast the problem of comparing two entities as an instance of the Optimal Transport problem. Intuitively, it implies that the entities are similar if the contexts of one entity can be cheaply transported to the contexts of the other. We show how this can be extended to represent a composition of entities (like sentences) by Wasserstein barycenters and discuss its flexibility to utilize a problem-specific transportation cost between contexts (like for measuring entailment). Further, this framework is not specific to natural language, but can be employed in any problem with a co-occurrence structure, such as movie-recommendations or nodes in a social network graph.

Next, during my internship at Facebook AI Research (FAIR), I became interested in the question: what if we learned this composition straight from the data, rather than assuming any particular composition strategy (such as averaging or using LSTM-based encoders) to aggregate the representations. In this way of non-compositional learning, we directly optimized the representation of the entity along with a decoder model, so as to be able to reconstruct the entity. This resulting technique (Singh et al., 2019b) shows strong performance on unsupervised text similarity tasks, while requiring the least amount of data in comparison to other state-of-the-art methods, and thus highlights the potential of such an approach in low-resource settings.

(♠) Robust–continual learning. Learning effective representations is, nonetheless, just one part of the equation. One of the important characteristics of human intellect is our ability to apply what we have learned from similar scenarios in the past to efficiently solve a new task. Hence, this characteristic to 'learn continually' is a fundamental requirement for an intelligent system. Therefore, instead of training independent learners from scratch every time, our goal is to retain the knowledge gained while doing the tasks and utilize it in the future. Further, even for a particular task like image recognition, its nature can vary between the agents as the input data distribution changes. Still combining the knowledge across them would benefit the overall task performance. A commonly used strategy is to form an ensemble of models and then average their individual predictions. However, this approach soon becomes infeasible, as the cost involved in maintaining the models and performing inference grows linearly with their number.

In a hope of doing this in a *one-shot manner*, we instead look at averaging the parameters (i.e., weights in a neural network). But, the issue with a vanilla parameter averaging is that there is no one-to-one correspondence between the parameters of a network and another. In other words, there is a permutation invariance with respect to the ordering of neurons in a layer. Hence, the key idea of our work (Singh and Jaggi, 2019) is to first align the layers of the two neural networks via the Optimal Transport map (which behaves like a soft-permutation matrix up to scaling) and then average their parameters. We demonstrate how our method can be used to perform efficient one-shot averaging of popular CNNs like VGG11 and also MLPs, which might have been initialized differently or trained on different data distributions, while enjoying performance benefits in skill-transfer settings.

An inherent assumption usually considered in the continual learning setting, is that all the involved agents or learners are collaborative and not adversarial. However, in practice, there could be attackers who are trying to manipulate the shared information (like the training data). Alternatively, it might be that the data arises from a sensitive domain such as healthcare. Hence, ensuring security and privacy is of the utmost importance. This was in part another motivation for our work above, since our proposed model-fusion mechanism does not involve the exchange of data, which is crucial in *Federated learning* applications. Other alternatives for combining models, like distillation, are thus rendered infeasible in such use cases.

(♠) Resource–efficient learning. Neural networks with millions and billions of parameters are not exceptional today. Many recent works employ GPUs or TPUs with the usage that can be calculated literally in days, months and even years (in the distributed setting). Consequently, the problem that arises is how to deploy such amazing but large models in the real world, where there are resource constraints. One idea is to compress the models by pruning away the less important parameters. A simple but successful strategy has been to remove the low magnitude weights. In my master thesis, which I recently started at IST Austria (advised by Dan Alistarh), our focus is to instead utilize the local curvature of neural networks with respect to their parameters for pruning. This is described by the Hessian matrix, which is again not a small object for the neural networks of interest. We aim to look at its suitable approximations by exploiting the links to the Fisher information matrix and utilizing the discovered empirical structure of Hessians (with respect to the neurons) based on our results.

Hessians play a pivotal role in ML, most noticeably in optimization, but have been side-stepped in deep learning due to their heavy computational cost. As another example, take the problem of *dataset distillation*, where the objective is to find the minimal set of key examples, such that when trained on these examples alone, the model maintains the performance in the usual full data setting. Hessians again show up here in the form of influence functions. On a higher level, the goal for my master thesis is to look at efficient approximations of the Hessian, or specifically in the form of inverse Hessian vector products.

Outlook. ⋄ *Priors faithful to task geometry.* I learned about Optimal Transport (OT) in an internship with Marco Cuturi at Kyoto University, where we applied it to train generative models like GANs. Ever since, OT has been an indispensable source of ideas for my research. This has made me realize not only the importance of utilizing the inherent geometry or structure of the task, like modeling by empirical measures; but in a broader sense, how mathematical tools can be so powerful for ML. Hence, I am keen to further enrich my toolkit, with ideas from concepts such as, negative dependence and submodularity, as well as positive association and graphical structures.

⋄ *Optimization.* It is the workhorse of machine learning, thus developing efficient techniques hold fundamental significance. Some of the many questions that intrigue me include: the over-parameterization puzzle, connections of SGD to Wasserstein gradient flows, non-convexity due to symmetry (note how the alignment via OT can be seen as reducing this non-convexity (Singh and Jaggi, 2019)), robust optimization (for adversarial examples), communication-efficient optimization & aggregation, second-order methods and analyzing optimization trajectories.

⋄ *Applications in Computer Science (CS).* The traditional programming paradigm dominates the computing realm, in comparison to the data-driven inductive paradigm used in ML. Having started off with a bachelor's in CS at IIT Roorkee, I am excited by the immense opportunities where ML can be applied, such as (a) program synthesis and translation (b) compiler optimization (c) learned index structures suitable for operating systems and databases.

References

Sidak Pal Singh, Andreas Hug, Aymeric Dieuleveut, and Martin Jaggi. Context Mover's Distance & Barycenters: Optimal transport of contexts for building representations. ICLR DeepGenStruct Workshop, 2019a.

Sidak Pal Singh, Angela Fan, and Michael Auli. GLOSS: Generative Latent Optimization of Sentence Representations, 2019b.

Sidak Pal Singh and Martin Jaggi. Model Fusion via Optimal Transport. NeurIPS OTML workshop (& submitted to AISTATS), 2019.