EECS 6322 Week 4, Paper 2
Deep Sets

Shivani Sheth (Student #: 218011783)
shivs29@yorku.ca

I. Summary and contributions

The paper proposes a new model in deep learning that takes and operates on variable size inputs, known as sets. As compared to traditional fixed dimensional vectors, the objective function of the model is invariant to the permutations in the set, i.e, it does not change by the order of data in the set. The main theorem proposed by the paper provides a generalized form for permutation invariant functions, which is the main component of the model, and further extends that to derive conditions for permutation equivariance in the model. The model is proved to perform significantly better, both in supervised and unsupervised applications containing large datasets.

The fundamental architecture proposed by the paper, known as DeepSets is a deep network that can operate on variable size sets. It enables a general parameter sharing scheme for sets in supervised models. The main property of the architecture is that the objective function is permutation invariant to the order of elements in the set, which is given by: $\pi : f(x_1, ..., x_M) = f(x_{\pi(1)}, ..., x_{\pi(M)})$ *For supervised learning models, this would mean the objective would be to classify / regress the output variables with respect to the input set while being permutation invariant with respect to the predictors. For unsupervised models, this objective would be to assign high scores to certain valid sets and low scores to uncertain sets. These scores can further be used in set expansion to predict new elements similar to the set.*

The structure of the objective function with a countable set of inputs for the invariant case is given by: $\rho\left(\sum_{x \epsilon X} \phi(x)\right)$ *where* X *is the countable input set. The objective function is further said to be permutation equivariant if all the non-diagonal elements of the weight vector (of a neural network layer) are tied together and all the diagonal elements are equal. This equivariant case is given by* $\Theta = \lambda I + \gamma(11^T)$ *where* Θ *is the weight vector and* I *is the identity matrix.*

The architecture of the framework is composed of the invariant model and the equivariant model. The invariant model transforms and adds up all representations of the input elements and then applies a non linear transformation on the sum of these representations. The working of the model follows the given step-wise approach. First, each input element x_m *is transformed into a representation given by either* $\phi(x_m)$ *for networks that do not have additional meta-information, or* $\phi(x_m|z)$ *for networks that have meta-information where the information is denoted by* z*. Then, the representations of* $\phi(x_m)$ *are added up and the output is processed by the* ρ *network of the layer, that is usually the fully connected layers or the non linearities in a deep network. The equivariant model designs layers in the neural network that are equivariant to the permutation of elements in the input set. A non linearity is applied to the weighted combination of its input and sum of input elements given by:* $\Theta = \lambda I + \gamma(11^T)$*. These layers facilitate parameter sharing across the network. Several equivariant layers can be stacked together in the network since the composition of permutation equivariant functions is also permutation equivariant.*

The architecture has been applied to various domains in supervised learning and some applications discussed in the paper are classification of point clouds, population statistics, sum of digits, and regression using clustering information. In population statistics, the architecture was used to learn the entropy and mutual information of Gaussian distributions without any underlying information about the properties of the distribution. For rotation, a $2x2$ *covariance matrix was randomly chosen and* N *sample sets were generated from* $\mathcal{N}(0, R(\alpha)\sum R(\alpha)^T)$ *for* N *random values of* α *in* 0 *to* π*. Here, the objective was to learn the entropy of the marginal distribution of the first dimension where* $R(\alpha)$ *was the rotation matrix. In correlation, rank 1, and random, a similar setup with different values was used to find the mutual information among the first* d *and last* d *dimensions of the distribution. The architecture used three fully connected layers with ReLU activation and a* L_2 *loss for training. The DeepSet prediction was compared with SDM which produced a significantly lower estimation error for a large number of sets.*

In the 'Sum of Digits' application, the architecture was used to calculate the sum of all numbers input sequentially to the network either in an image or text format. In both cases, a maximum of $M = 10$ *images were sampled from their respective datasets and about* $100k$ *sets of training and test images were used. The architecture was tested on up to 100 digits in text, and up to 50 digits in images. Both cases were compared to LSTM and GRU models, where the DeepSets architecture had a much higher accuracy, especially for larger lengths of sequences. Similarly, in both point cloud classification and regression using clustering information, the DeepSets architecture performed much better than its state of the art counterparts in those domains.*

In unsupervised learning applications, DeepSets was tested on text concept set retrieval, image tagging, and set anomaly detection. In text concept set retrieval, the architecture had to

predict new words similar to the words in a given set. It was tested on three datasets namely LDA-$1k$, LDA-$3k$, and LDA-$5k$ and used 3 fully connected layers with ReLU activation for ϕ and ρ transformations. It was compared against six baseline architectures, and it outperformed all baselines in LDA-$3k$, and LDA-$5k$ datasets in all 3 Recall %. In image tagging, DeepSets tested relevant tags retrieval for an image and was conditioned using predict tags during training. It was tested on three datasets namely ESPGame, IAPRTC-12.5, and COCO-Tag, and was compared against several baselines such as MBRM, JEC, FastTag. The model performs significantly better in all metrics on the COCO-Tag dataset, while it had a comparable performance in the other two datasets. In Set Anomaly, the architecture was tested to differentiate an anomaly from the rest of the images in a set that shared the same 2 features.The model was able to achieve an accuracy of 75% on test sets as compared to the baseline model which achieved an accuracy of 6.3%.

In conclusion, permutation equivariance and invariance gives the DeepSet architecture a huge advantage over the previous baselines compared in the paper. The model works very well in different domains of deep learning and performs better than the state of the art baselines designed for the domain.

II. Strengths

The DeepSets architecture proposed by the paper proves to be a versatile/ general model. The architecture works on different application domains and outperforms the state of the art models, specialized for that particular domain, in most cases. The proposed model also has a strong theoretical grounding as it satisfies the 'de Finetti theorem' defined for exchangeable models, the Representer theorem used for support distribution machines, and the Spectral methods which can be viewed as a special case of the general theorem of permutation invariance proposed by the paper. The permutation equivariance and invariance enable the DeepSet architecture to operate directly on representations that were not feasible before, such as the point cloud representations. It also shows significantly higher accuracies in applications such as Set Anomaly and Sum of Digits, especially for large datasets.

III. Weaknesses

The model requires large datasets for optimal performance and does not perform very well on small datasets. This is because the model requires large amounts of data to understand the underlying variable representations and hence would give a lower accuracy in applications where less data is available.

IV. Correctness

The claims proposed by the paper are correct and are backed by strong theoretical grounding, as well as successful experiments. The permutation equivariance and invariance as proposed by the paper satisfies the previous theorems for exchangeable models, and in some cases also prove that they are a specific case of the general theorem proposed by the paper. In addition, the authors have also implemented the theory into deep neural networks which outperformed many of the state of the art baselines in various domains. Thus, the results prove that permutation equivariance and invariance give a huge advantage to the DeepSets architecture over the other architectures discussed in the paper.

V. Clarity

The paper is well written in general. The format of the paper is straightforward and easy to read with the theories explained briefly while connecting with their implementation and are further elaborated upon in the appendix for stronger theoretical grounding. However, it also has a few typing and grammatical errors. For example, in the second last line of the subtopic 4.1.2, under the "image" section, the sentence looks incomplete. It could mean 'the probability that at least one image will be misclassified is $1-(1-p)^N$', but due to the missing words it creates a little ambiguity.

VI. Relation to prior work

Previous work related to invariance and equivariance in neural networks are conducted based on a general group of transformations. One such example is where deep permutation invariant features are constructed through a pairwise coupling of features at a previous layer given by: $f_{i,j}([x_i, x_j]) = [|x_i - x_j| x_i + x_j]$. The concept of pooling functions has also been used across several domains such as binary classification tasks for causality on a given set of samples. Certain applications in multi-agent settings and sensor networks have used a specialized case of the equivariant model as proposed by the paper.

The framework proposed by the paper differs from the previous works since it introduces a novel and generalized structure of the permutation invariant objective function. This relates DeepSets, the proposed framework, to various other applications in machine learning. The permutation equivariant layers used in the neural networks of the architecture also promote parameter sharing across the network.

VII. Reproducibility

There are sufficient theoretical details to implement the major results of the work proposed by the authors. However, no code snippets/ links relevant to the architecture have been provided by the authors, hence making it difficult to implement the finer architectural details of the model.

VIII. Additional comments

A few examples of ambiguous statements in the paper that could be improved for easier readability are as follows. Under subtopic 3.2, the first line of the second paragraph is missing the words "in a ". Hence the sentence could be modified to: " In [19], pooling was used in a binary classification task ... ". Similarly, in the second bullet point under subtopic 4.1.1, the last sentence does not seem absolutely correct. It could be modified to 'Goal was to learn the mutual information among the first d and last d dimensions'.