Table 1: The results for the **full distribution estimation task**. Lower perplexity (PP) is better.

Method	$\frac{PP}{256^3}$
Non-compositional Baseline	0.071
SOWE	**0.075**
CNN	0.078
RNN	0.089

Table 2: The results for the **full point estimation task**. Lower mean squared error (MSE) is better.

Method	MSE
Non-compositional Baseline	0.066
SOWE	**0.067**
CNN	**0.067**
RNN	0.071
Distribution Mean Non-compositional Baseline	0.066
Distribution Mean SOWE	0.068
Distribution Mean CNN	0.069
Distribution Mean RNN	0.077

0.1 Quantitative Results

Overall, we see that our models are able to learn to estimate colors based on sequences of terms. From the consideration of all the results shown in **???????????**, the CNN and SOWE models perform almost as well as the Non-compositional baseline. With the SOWE having a marginal lead for distribution estimation, and the CNN and SOWE being nearly exactly equal for most point estimation tasks. We believe the reason for this is that the SOWE is an easier to learn model from a gradient descent perspective: it is a shallow model with only one true hidden layer. In general the results for the LSTM and GRU were very similar, and both much worse than the non-recurrent models. While it is only marginally behind the SOWE and CNN on the full point estimation task (**??**), on all other tasks for both point estimation and distribution estimation it is significantly worse. This may indicate that it is hard to capture the significant relationships between terms in the sequence. However, as discussed **??** it did learn generally acceptable colors to the human eye, but the quantitative results presented in this section show that it is not as close a match to the population's expectation.

Table 3: The results for the **order distribution estimation task**. Lower perplexity (PP) is better. This is a subset of the full test set containing only tests where the order of the words matters.

Method	$\frac{PP}{256^3}$
Non-compositional Baseline	0.053
SOWE	**0.055**
CNN	0.057
RNN	0.124

Table 4: The results for the **order point estimation task**. Lower mean squared error (MSE) is better. This is a subset of the full test set containing only tests where the order of the words matters.

Method	MSE
Non-compositional Baseline	0.065
SOWE	**0.066**
CNN	**0.066**
RNN	0.096
Distribution Mean Non-compositional Baseline	0.065
Distribution Mean SOWE	**0.066**
Distribution Mean CNN	**0.066**
Distribution Mean RNN	0.095

0.1.1 Ordered Task

The performance of SOWE on the order tasks (**????**) is surprising. For the distribution estimation it outperforms the CNN, and for point estimation it ties with the CNN. The CNN and RNN, can take into account word order, but the SOWE model cannot. The good results for SOWE suggest that the word-order is not very significant for color names. While word order matters, different colors with the same terms in different order are similar enough for most colors that it still performs very well. In theory the models that are capable of using word order have the capacity to ignore it, and thus could achieve a similar result. An RNN can learn to perform a sum of its inputs (the word embeddings), and the CNN can learn to weight all non-unigram filters to zero. In practice we see that for the RNN in particular this clearly did not occur. This can be attributed to the more complex networks being more challenging to train via gradient descent. It seems that color-naming is not a task where word order substantially matters, and thus the simpler SOWE model excels.

0.1.2 Unseen Combinations of Terms

The SOWE and CNN models are able to generalize well to making estimates for combinations of color terms that are not seen in training. **????** show the results of the model on the test set made up of rare combinations of color names (as described in **??**) for the restricted training set (which does not contain those terms). These results on this test set are compared with the same models when trained on the full training set. The Non-compositional baseline are unable to produce estimates from the unseen combinations testing set as they do not process the color names term-wise. Performing well on this task is indicative as to if the models are learning how the terms combine to determine the color, as they cannot be simply matching the full color name (term sequence) against one that occurs in training. This is an important test, as due to the combinatorial nature of language, it is common to encounter term sequences in the real world that never occur during training.

On distribution estimation (**??**) the SOWE results are only marginally worse for the restricted training set as they are for the full training set. The CNN results are worse again, but they are still better than the results on the full test-set. The distribution estimates are good on absolute terms, having low evaluated perplexity.

In the point estimation task (**??**) the order is flipped with the CNN outperforming the SOWE model. In-fact the CNN actually performs better with the restricted training set for predicting the unseen test colors, than it does for predicting those colors when they are included in the full training set; though the difference is only marginal. Unlike for distribution estimates, the unseen color point estimates are worse than the overall results from the full task (**??**), though the errors are still small on an absolute scale.

Over all the performance of the SOWE and CNN remain strong on the unseen combination tasks. The RNN models continue to perform poorly on the unseen combination of terms task for both point and distribution estimation. The SOWE and CNN perform sufficiently well on the unseen combinations that the color estimates they produce would be practically useful. The unseen combination results are comparable to the full dataset results discussed (shown in **????**), and have very small errors on an absolute scale.

0.1.3 Extracting the mean from the distribution estimates

In the point estimation results discussed so far have been from models trained specifically for point estimation (as described by **??**). However, it is also possible to derive the mean from the distribution estimation models. Those results are also presented in **??????**. In general these results perform marginally worse (using the MSE metric) than their corresponding modules

Table 5: The results for the **unseen combinations distribution estimation task**. Lower perplexity (PP) is better. This uses the unseen test set: a subset of the full test set contain only rare word combinations. In the restricted training set results these rare word combinations were removed from the training and development sets. In the full training set results the whole training and development stet was used, including the rare words that occur in the test set.

Method	Full Training Set $\frac{PP}{256^3}$	Restricted Training Set $\frac{PP}{256^3}$
Non-compositional Baseline	0.050	–
SOWE	**0.050**	**0.055**
CNN	0.052	0.065
RNN	0.117	0.182

Table 6: The results for the **unseen combinations point estimation task**. Lower mean squared error (MSE) is better. This uses the unseen test set: a subset of the full test set contain only rare word combinations. In the restricted training set results these rare word combinations were removed from the training and development sets. In the full training set results the whole training and development stet was used, including the rare words that occur in the test set.

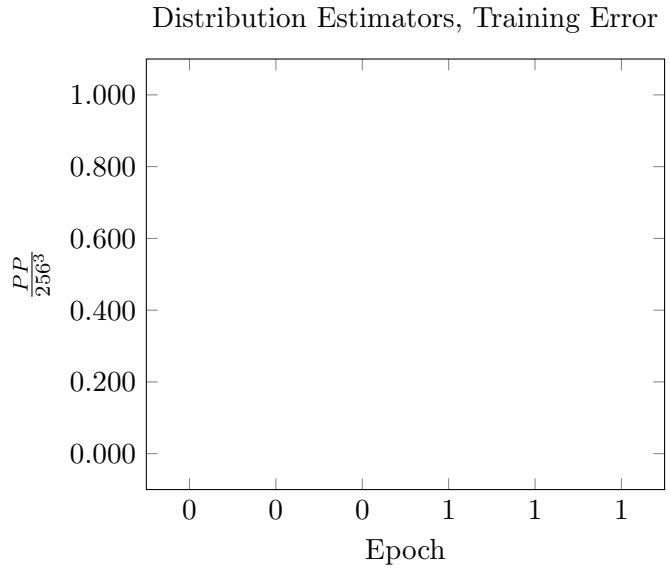
Method	Full Training Set MSE	Restricted Training Set MSE
Non-compositional Baseline	0.062	–
SOWE	**0.065**	0.079
CNN	0.072	**0.070**
RNN	0.138	0.142
Distribution Mean Non-compositional Baseline	0.062	–
Distribution Mean SOWE	0.073	0.076
Distribution Mean CNN	0.073	0.084
Distribution Mean RNN	0.105	0.152

using the point estimation output module. The only exception to this is the LSTM for both the unseen combination tasks and the order task, for which it was notably better to use the mean from the distribution rather than one directly trained. We note that for the Non-compositional baseline, the distributions mean is almost identical to the true mean of points, as expected.

0.1.4 On the differences between the distribution estimation and point estimation training procedure

Beyond the output module there are a few key differences between the point estimation modules and the distribution estimate modules. When training distribution estimation models, all examples of a particular color name is grouped into a single high information training observation using the histogram as the output. Whereas when training for point estimation, each example is processed individually (using minibatches). This means that the distribution estimating models fit to all color names with equal priority. Whereas for point estimates, more frequently used color names have more examples, and so more frequent color names are fit with priority over rarer ones. Another consequence of using training per example using random minibatches, rather than aggregating and training with full batch, is increased resilience to to local minima lecun2012efficient. One of the upsides of the aggregated training used in distribution estimation is that it trains much faster as only a small number of high-information training examples are processed, rather than a much larger number of individual observations.

It may be interesting in future work to consider training the distribution estimates per example using one-hot output representations; thus making the process similar to that used in the point estimate training. It is possible that such a method may have trouble learning the smoothness of the output space (as discussed in **??**), as it would not see demonstration of the partial activation of adjacent bins in the training examples. However, this is not certain, much like the point estimation trained on one-hot learns a representation that minimises mean squared error outputting a point between all the training examples, it is reasonable to expect that the distribution estimates will output a smooth histogram as this is near to a minimum for the cross-entropy. With the current model the presence of partial activation of adjacent bins in all examples may be causing the smoothness to be learned primarily in the output layer, and with little respect for the inputs. Such would explain the difficulties in capturing subtler features of the output distribution, such as the depth of the valley between the two peaks in the hue of greenish shown in **??**. Using one-hot examples for training, may help force encoding the knowledge of the nature of continuous distributions deeper into the network allowing the input color name to have a more pronounce effect.



Final Training Performance

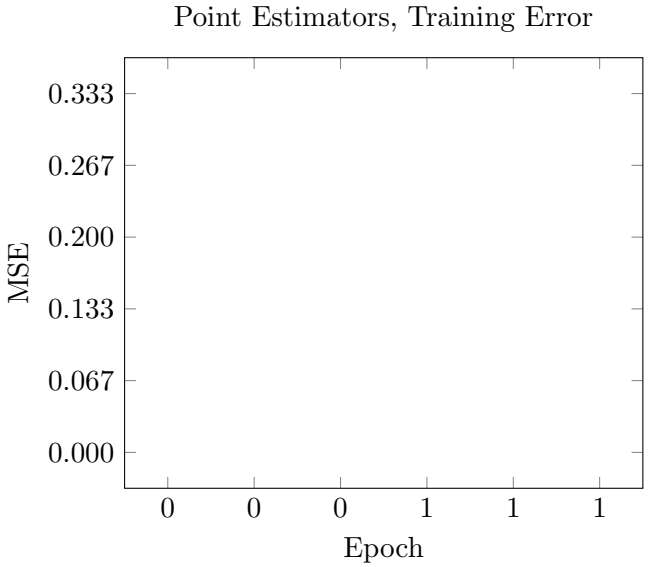
Method	$\frac{PP}{256^3}$
SOWE	0.057
CNN	0.070
GRU	0.081
LSTM	0.094

Figure 1: The training set error of the distribution estimation models, when trained on the full dataset. Note that the plots stop when the model ceased training due to the development set error rising (early stopping).

.

0.2 Training set results

To investigate our supposition that the SOWE, is a much easier function to fit via gradient descent, as compared to the CNN or the RNNs, we consider the error rate on the full training set during the training of the models. These plots are shown in **??** and **??**. These plots seem to support the supposition, as the SOWE training error decreases notably faster (it is a steeper curve) in both cases. This corresponds to a easier error surface in network parameter (weights and biases) space, with fewer points of low gradient, or near local minima. If we compare the final loss of each method on the training set (before it was stopped due to early stopping) against the test set results in **????** we find they are similar, particularly for distribution estimation (**??** and **??**). While for point estimation (**????**), on the test set CNN and SOWE perform similarly, while RNNs perform much worse, despite the fact that in training the performance of CNN is roughly midway between SOWE and the RNNs. In all cases, the absolute error in training has become small relative the to the natural variation in the training set by the time early stopping terminates training. Note that perfect fit is not possible as the training data varies.



Final Training Performance

Method	MSE
SOWE	0.132
CNN	0.135
GRU	0.149
LSTM	0.148

Figure 2: The training set error of the point estimation models, when trained on the full dataset. Note that the plots stop when the model ceased training due to the development set error rising (early stopping).