A Theoretical Analysis of Pooling Operation Using Information Theory

1st Christoforos Nalmpantis
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
christofn@csd.auth.gr

2nd Athanasios Lentzas
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
alentzas@csd.auth.gr

3rd Dimitris Vrakas
School of Informatics
Aristotle University of Thessaloniki
Thessaloniki, Greece
dvrakas@csd.auth.gr

***Abstract*—Several modern deep learning architectures incorporate the operation of pooling in order to achieve sufficient, minimal and invariant representations. Nevertheless, its importance has only been verified empirically, without solid theoretical evidence. This paper presents a comprehensive theoretical analysis, investigates the mechanism of pooling from the information theory point of view and proposes a novel pooling operation based on entropy. In comparison with other versions of pooling, it automatically adapts to the features and creates more compact representations. The theoretical outcomes are validated utilizing both shallow and deep architectures.**

***Index Terms*—Feature Pooling, Deep Learning, Representation Learning, Information Theory, Entropy, Channel Capacity**

I. Introduction

Feature pooling dates back to the seminal paper about complex cells in the visual cortex [1]. It is used in many hand-crafted feature engineering methods such as SIFT [2] and HOG [3]. Especially max [4] and average pooling [5], [6] are commonly used in convolutional neural networks. Stochastic pooling has also shown state-of-the-art results for regularization of deep convolutional neural networks [7].

Pooling is an integral part of several neural network architectures and there is a plethora of examples predicating its benefits. However, most of the studies are empirical and there is a lack of a complete theoretical framework. The goal of this research is to shed light on the dynamics of pooling operation, using information theoretic concepts. The main contributions of the paper are summarized in the following steps: a) Scrutinizing pooling operation as an information channel. b) Understanding max, average and stochastic pooling. c) Reduction of pooling to a special case of the problem of max entropy sampling. d) Proposing a novel pooling operation, named entropy pooling. e) Empirical evaluation of the theoretical outcomes.

II. Related Work

To the best of the authors' knowledge, there is only one theoretical analysis of pooling operation by Boureau et al. [8]. It describes the statistical properties of two basic pooling operations for a two-class categorization problem.

The approach that is followed is simplified considering a two-class classification problem and assuming the features are i.i.d. Bernoulli random variables. According to the authors, the assumption of independence is invalid, since neighbouring image features are highly correlated. Despite its shortcomings, the outcomes of the study are still relevant and they are verified empirically. The two pooling operations under examination are max and average. The analysis evaluates how the statistical properties of the two methods affect the capability of a model to separate two different classes. The underlying reasons that justify this performance are obscured and are contributed to several factors, such as the sparsity of the features and the sample cardinality.

III. Pooling Operation and Information Theory

Recent studies have advanced many deep learning techniques using information theoretic principles, including the generalization of rate distortion theory, named information bottleneck principle [9]. A generalization of dropout technique, named information dropout, is proposed and achieves similar performance to binary dropout [10]. A matrix-based Rényi's α-entropy is used to understand the information flow in stacked autoencoders [11]. A framework, named partial information decomposition, is introduced to analyze the learning phase of convolutional neural networks [12].

This paper establishes a connection between pooling operator and information theory. With this in mind, the function of pooling is revisited. Although there is no formal definition, it is widely accepted that the objective is to downsample a given feature map, while retaining relevant information. Downsampling the data is easy and makes the overall architecture computationally lighter. However, determining which features are relevant is hard. It not only depends on the task, but also on the characteristics of the data.

A. Pooling as an Information Channel

Let's define the objective of pooling from the perspective of information theory. Assume a deep neural network with one pooling operation after hidden layer i. Denote the output features of hidden layer i as a random variable X. Let X be

This research has been co-financed by the European Regional Development Fund of the European Union and Greek national funds through the Operational Program Competitiveness, Entrepreneurship and Innovation, under the call RESEARCH–CREATE–INNOVATE (project code:95699 - Energy Controlling Voice Enabled Intelligent Smart Home Ecosystem)

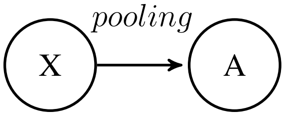

Fig. 1. Markov chain of pooling operation.

the input of the pool and A its output.Then pooling operation can be considered as an information channel, which can be represented by the Markov chain in Fig. 1.

$$H(A|X) = \sum_{x} p(x)H(A|X = x) \tag{1}$$

$$H(A|X) = \sum_{x} p(x)H(1) = 0 \tag{2}$$

Thereupon, for the case of a deterministic function, the mutual information depends only on the entropy of the output features and (4) becomes:

$$I(X; A) = H(A) \tag{3}$$

The mutual information I, between the two random variables X and A and the capacity C of the channel, are defined by the following equations respectively [13]:

$$I(X; A) = H(A) - H(A|X) \tag{4}$$

$$C = \max_{p(x)} I(X; A) \tag{5}$$

, where H(A) is the entropy and H(A|X) the conditional entropy. The ideal channel would allow all the information to be transferred, without loss. In the case of pooling, the goal is to maintain as much relevant information as possible. Measuring the relevance of information would require prior knowledge of the task, otherwise our best estimation would be just a guess. Consequently the best channel is the one that makes no assumptions which is the channel with maximum capacity. From (4) it is obvious that I(X;A) is max when H(A) is maximized and H(A|X) is minimized.

The conditional entropy H(A|X), which is non-negative, can be equal to zero when pooling is a deterministic function, such as max and average ones. H(A|X) can be developed as follows:

B. Understanding Pooling

The outcomes about channel capacity of pooling operation can be used to understand the mechanism of max, average and stochastic pooling. Using the equations derived in the previous section and the statistical properties of the variables X and A, it will be explained how these operations work.

The statistical properties of the input X are the same for each case of pooling and they are described along these lines. Let $\overline{x}$ denote the mean of the joint feature representation X_N with size N and σ_X its standard deviation. The respective formulas are:

$$\overline{x} = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{6}$$

$$\sigma_X = \frac{1}{N}\sum_{i=1}^{N} (x_i - \overline{x})^2 \tag{7}$$

1) Max pooling: Max pooling is a function that given a set of values, it chooses the largest one. Pooling is applied using kernels, so let X_r be a joint feature representation of a kernel with r elements, then the function is:

$$f_{max}(X_r) = max(x_1, x_2, ..., x_r) \tag{8}$$

It is a deterministic choice function and the mutual information between the input and the output, according to (3), is equal to H(A).

Examining the statistical properties of max pooling will now clarify how it affects H(A). By definition, it is obvious that the mean will be increased, thus $\overline{x}_A \geq \overline{x}$. In order to evaluate the standard deviation, more elaboration is needed. By using kernels , the standard deviation in (7) is developed as follows:

$$\sigma_X = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{r}\sum_{j=1,x_{ij}\neq a_i}^{r} (x_{ij} - \overline{x})^2 + \frac{1}{M * r}\sum_{i=1}^{M} (a_i - \overline{x})^2 \tag{9}$$

where M is the number of kernels, r the number of elements of each kernel and a_i the maximum value of each kernel. It is assumed that there is no overlap during pooling and so $N = r * M$. The proof with overlap is similar. Next, the standard deviation of max pooling is given by:

$$\sigma_A = \frac{1}{M}\sum_{i=1}^{M} (a_i - \overline{x}_A)^2 \tag{10}$$

Comparing (9) and (10), it is concluded that the change of standard deviation depends on the distribution of the values of the input features. The input also determines the cardinality of A, which also affect the entropy. From the definition of entropy, higher cardinality means higher entropy. Thus, max operation doesn't control the entropy of the output and there is no guarantee that it will maximize the capacity of pooling. The capacity of the channel is proved that it depends on the input feature distribution and the cardinality of the output A.

2) Average pooling:: Average pooling is analyzed in the same fashion, because calculating the average is a deterministic function.

$$f_{avg}(X_r) = \frac{1}{r}\sum_{j=1}^{r} x_j \tag{11}$$

Therefore, the mutual information is described by 3. The mean of the original X_N and the mean of the output of pooling A_M are equal and will be symbolized as $\overline{x}$. Equation (7) is developed using kernels as in:

$$\sigma_X = \frac{1}{M}\sum_{i=1}^{M}\frac{1}{r}\sum_{j=1}^{r} (x_{ij} - \overline{x})^2 \tag{12}$$

Using (11), let $a_i = f_{avg}(X_i)$. Then the formula of standard deviation for the features A_M is:

$$\sigma_A = \frac{1}{M}\sum_{i=1}^{M} (a_i - \overline{x})^2 = \frac{1}{M}\sum_{i=1}^{M} [\frac{1}{r}\sum_{j=1}^{r} (x_{ij} - \overline{x})]^2 \tag{13}$$

From (12) and (13) it is concluded that $\sigma_A \leq \sigma$. Consequently, the distribution of the output A will have smaller standard deviation and the values of its elements will tend to the expected value $\overline{x}$. These two properties of average pooling, lead to a more uniform distribution and make it robust to feature invariance. However, the distribution still depends on the input data. The maximum unique elements of the final distribution is M and it happens when each kernel gives a unique value. The mechanism of average pooling doesn't affect the number of unique output values and therefore it doesn't guarantee maximization of the entropy.

3) Stochastic pooling:: The case of stochastic pooling is different, since it is a non deterministic choice function. Without loss of generality, let the stochastic function be the equal probability of selecting one of the features. Therefore, if N is the cardinality of features, the probability of selection of each feature is $a = 1/N$. Next, let's consider the transition matrix p(A|X), with rows representing A and columns representing X. Let $\overrightarrow{r}$ be a row of the transition matrix, then (4) becomes:

$$I(X; A) = H(A) - H(\overrightarrow{r}) \tag{14}$$

$$I(X; A) = H(A) - f(a) \tag{15}$$

The entropy of the row vector is a function depending on the probability a. The mutual information of stochastic pooling depends on both H(A) and H(A|X). The first term is still free of any parameter of the channel and depends on the data. The difference against max and average pooling is that stochastic pooling has some control over the mutual information via the parameter a.

The outcome of the previous analysis is that current pooling solutions don't have the properties to handle the entropy H(A) and their performance is affected by the distribution of the data. This problem can be addressed by introducing a novel pooling operation based on max entropy sampling.

IV. Entropy Pooling

As it was shown previously, conventional pooling functions don't have the properties to manage the entropy of A in (4). A more advanced approach is required, to accomplish complete control of the entropy H(A) and minimize the conditional entropy H(A|X). The latter requirement can be met by defining a deterministic function. For the former one, we need to understand the problem in depth and reformulate it.

Maximizing the mutual information in (3) can be reduced to the problem of maximum entropy sampling [14]. It is defined as a design problem of selecting a subset T from a set S of N random variables, with regard to retain as much information as possible. In the context of pooling, the problem is to choose the most informative subset, subject to spatial constraints.

Following the variable definitions of the Markov chain in Fig. 1, let f_{entr} be the required function. According to the principle of maximum entropy. the probability distribution that best describes A, is the uniform distribution. Assuming only that the size of A is predefined and equal to M, the required function has lower bound equal to zero and upper bound the entropy of the equivalent uniform distribution.

$$0 \leq f_{entr}(X) \leq log|M| \tag{16}$$

In line with (16), the output of f_{entr} should approximate a uniform distribution. Existing solutions for the problem of maximum entropy sample can be adopted [15]. However, this would be computationally inefficient for large neural networks because it is an NP-Hard problem [16]. For the purpose of this study, a novel algorithm is proposed, named entropy pooling. The proposed version of entropy pooling is non-optimal, but computationally efficient and extendable.

The algorithm of entropy pooling computes the probabilities p for each feature map with size N. Next, the map of probabilities is divided into regions, according to the specified kernel size and strides, in the same way as in classic pooling operations. For each region the element with the smallest probability is selected. The mathematical formula of entropy pooling, for a region of size r, is:

$$f_{entr}(X_r) = X_r[g(P_r)], \tag{17}$$

$$g(P_r) = \underset{1 \leq i \leq r}{\arg\min}\, p_i \tag{18}$$

, where X_r is the input feature map and P_r the constructed map of probabilities. Consequently, (3) gives:

$$I(X; A) = H(f_{entr}(X_r)) \tag{19}$$

So far, the desired function is deterministic and selects features with high sparsity, handling the amount of information that will be propagated via the neural network. It remains to explain why choosing sparse features increases the entropy. The intuition is that rare features cannot be selected several times and as a result the output will have a flatten distribution with high cardinality. This is in good agreement with the finding of Boureau et al. [8], that max pooling is well adopted to rare activated features.

A more rigorous proof is given considering the bounds in (16), the property that entropy is concave and the fact that the final feature map A has no element with $p < 1/M$. Observing the graph of a random entropy function such as Fig. 2, the peak is at $p = 1/M$ and every solution to the right has lower entropy. As a conclusion, selecting features that are activated rarely, increases the output entropy of pooling.

V. Experiments and Discussion

The aim of the experiments is twofold. The first goal is to validate the theoretical outcomes and the second one to demonstrate that entropy pooling is robust and can perform on par with other pooling operations. For the purpose of the experiments, two versions of entropy pooling are used. One which gives an output with high entropy and one with low entropy. The former one works as it was described in the previous section. The latter one selects the most frequent features instead of the sparse ones. An optimal solution is not examined because it is out of the scope of this study and it would be computationally intractable. Thus,

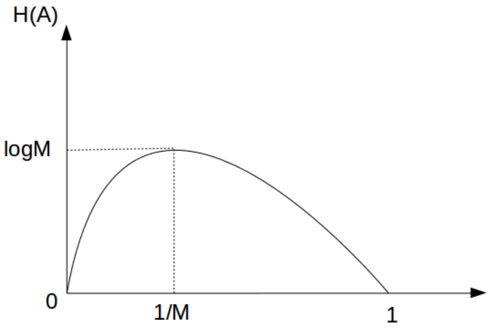

Fig. 2. Diagram of an entropy function.

the experiments are not expected to demonstrate better than state-of-the-art results. The code of the experiments can be found at https://github.com/ChristoferNal/pooling-operations-and-information-theory.

A. Validation of Theoretical Outcomes

1) The effect of pooling operation on images: Considering pooling as an information channel, pure pooling is applied on images. Max, average and entropy pooling are examined using random images from the dataset Cifar10 [17]. The goal is to observe how the distribution of an image is changed by the three operators. The pipeline is very simple, a single channel is introduced to each pooling and then the output is plotted. Then Shannon entropy, mean and standard deviation are calculated. Tables I and II present two representative examples with the respective results of a dark image and a bright one.

The experiments verify the theoretical outcomes. The most robust operators are entropy and average pooling. The output of entropy pooling tends to be uniform and its Shannon entropy is the first or second highest. The output of average pooling, also tends to be uniform while standard deviation is always lower than the original and the mean equal to the mean of the original image. Shannon entropy of average pooling is very high for the majority of the images of this dataset. This is explained by the high cardinality of the output due to the unique results of the function of average.

TABLE I
EFFECT OF POOLING ON A BRIGHT IMAGE

Pooling Type	**Properties**			
	Cardinality	***Mean***	***SD***	***Entropy***
Original Img	221	176.06	60.35	7.3
Max	115	192.14	54.14	6.56
Average	199	176.06	57.01	7.51
High Entropy	154	168.25	60.9	7.06
Low Entropy	86	182.5	57.37	5.85

According to average function, average pooling is not just selecting features, it generates new ones and the output space is not any more integers in [0, 255]. In order to make more direct the comparison with the other two pooling operators, which act as selectors, an alternative version of average pooling is also taken into account. The result of average is rounded to the closest integer. Then, the output features belong to the same group with the original ones, e.g. integers in the range [0, 255]. This new version of average pooling behaves in the same way. The output features still have high entropy, but at this time it is lower than the Shannon entropy coming from entropy pooling. The cardinality is also lower as expected.

TABLE II
EFFECT OF POOLING ON A DARK IMAGE

Pooling Type	**Properties**			
	Cardinality	***Mean***	***SD***	***Entropy***
Original Img	240	81.19	73.18	7.38
Max	164	105.69	75.43	7.15
Average	221	81.19	68.51	7.71
High Entropy	165	94.38	72.67	7.2
Low Entropy	79	69.66	73.43	5.75

TABLE III
COMPARISON OF LOW AND HIGH ENTROPY

Dataset	***Low Entropy***	***High Entropy***
Cifar10	53.48 +-0.52	**59.07 +-0.25**
Cifar100	22.76 +-0.32	**26.80 +-0.95**
MNIST	71.73 +-12.98	**72.93 +-16.64**
FMNIST	74.54 +-34.19	**77.88 +-10.41**

The output of max pooling has higher mean value than the original image. The standard deviation doesn't show a consistent behaviour, which confirms the theoretical analysis. Regarding the entropy it is empirically verified that it depends on the data. More specifically, it is observed that high entropy and max pooling have similar behaviour for dark images. The equivalent entropies have almost the same value. On the other hand bright images are transformed into a lower entropy feature representation via max pooling. These observations are shown in tables I and II.

TABLE IV
IMAGE CLASSIFICATION RESULTS

Neural Network	**Image Dataset**	**Pooling Operation**		
		Max	***Average***	***High Entr***
LeNet	MNIST	**98.93 +-0.01**	98.64 +-0.01	98.51 +-0.05
	FMNIST	**99.32 +-0.07**	99.18 +-0.04	99.16 +-0.05
ResNet20	Cifar10	49.25 +-39.29	**91.87 +-0.11**	90.57 +-0.04
	Cifar100	64.81 +-0.44	**67.77 +-0.16**	62.44 +-0.48
ResNet20 2P	cifar10	85.05 +-0.06	87.15 +-0.24	**88.57 +-0.22**
	cifar100	56.31 +-0.43	59.41 +-0.81	**62.75 +-0.08**

2) Increasing information flow in a shallow neural network: Concerning the complexity of a deep neural network, the simplest way to investigate how information flows is to build a very shallow neural network. With this in mind, the simple neural network consists of a convolutional layer, followed by a pooling operation and a fully connected layer. The datasets that are used are: Cifar10, Cifar100, MNIST and FMNIST.

Each of the datasets corresponds to a ten class classification task, apart from Cifar100 which has one hundred labels. The model is trained and tested each time with one of the following pooling operations: high entropy and low entropy. The goal is to verify that pooling features with higher entropy benefit classification accuracy regardless of the dataset or the task.

The intuition that lies in this statement is that pooling is a bottleneck and should maximize the amount of information that passes to deeper layers. By looking at table III and comparing low and high entropy operations, it is confirmed that high entropy pooling always achieves better performance.

B. Comparative Analysis of Pooling Operations

There are many open questions on how a neural network is trained and it is very difficult to isolate and measure the impact of a single layer. In order to validate the robustness of entropy pooling two popular architectures are used.

The first one is LeNet [6] and consists of two sets of convolutional and pooling layers, followed by two fully-connected layers and finally a softmax classifier. LeNet is evaluated on MNIST and FMNIST. Table IV shows that the best accuracy is achieved by max pooling, whereas average and high entropy have equivalent accuracy.

The second architecture is ResNet20 [18]. Two variations of ResNet20 are utilized, one with one pooling operation before the last fully connected layer and one with one extra pooling operation after the first convolutional layer. In this paper, the former one is called ResNet20 1P and the latter one ResNet20 2P. ResNet is trained and validated using Cifar10 and Cifar100.

The accuracy results of ResNet20 1P are presented in table IV. Among the various pooling operations, average fits the best. This is not a surprise as the original model is proposed with average pooling. When it comes to max pooling, the model doesn't always converge efficiently. Specifically for Cifar10, the final accuracy can fluctuate a lot, which can be attributed to the deficiency that max pooling is sensitive to feature variability. High entropy pooling demonstrates descent results for both datasets.

It is worth mentioning that the place of pooling inside this neural network might be the reason that shows the weakness of max pooling. Being at the end of the model means that all features are important and well defined by previous layers. In accordance to our theoretical conclusions max pooling misses important information that have small absolute values, whereas average and high entropy preserve the most important features.

Regarding ResNet20 2P, the pooling operation before the last fully connected layer is the average one, across all experiments. The first pooling is replaced with max, average and high entropy. The maximum accuracy is achieved with high entropy pooling and the rest of the results are as expected.

VI. Conclusion and Future Work

This study strengthens the understanding of deep learning, by scrutinizing pooling operation from the information theory perspective. Using fundamental information theoretic principles it is evident how pooling operators enhance the performance of neural networks. Rigorous mathematical proofs show the strengths and the vulnerabilities of existing pooling functions, emphasizing the need of a new property of these functions that controls the information flow. Thereupon, pooling is revisited as a special case of the problem of max entropy sampling, suggesting a novel robust solution, named entropy pooling. The theoretical outcomes are validated thoroughly via practical experiments and the proposed method is empirically compared to existing approaches.

These findings add to a growing body of literature on developing a complete theory of deep learning. Further work is suggested on investigating the behaviour of pooling during training of a neural network, paying attention to the order of pooling inside the network. Researchers are highly encouraged to use entropy pooling, as it can be swapped into to any existing neural network architecture.

References

[1] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *The Journal of physiology*, vol. 160, no. 1, pp. 106–154, 1962.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[4] K. Jarrett, K. Kavukcuoglu, Y. LeCun *et al.*, "What is the best multi-stage architecture for object recognition?" in *Computer Vision, 2009 IEEE 12th International Conference on.* IEEE, 2009, pp. 2146–2153.

[5] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, 1990, pp. 396–404.

[6] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[7] M. D. Zeiler and R. Fergus, "Stochastic pooling for regularization of deep convolutional neural networks," *arXiv preprint arXiv:1301.3557*, 2013.

[8] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 111–118.

[9] N. Tishby, F. C. Pereira, and W. Bialek, "The information bottleneck method," *arXiv preprint physics/0004057*, 2000.

[10] A. Achille and S. Soatto, "Information dropout: Learning optimal representations through noisy computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018.

[11] S. Yu and J. C. Principe, "Understanding autoencoders with information theoretic concepts," *arXiv preprint arXiv:1804.00057*, 2018.

[12] s. Yu, R. Jenssen, and J. C. Principe, "Understanding convolutional neural network training with information theory," *arXiv preprint arXiv:18804.06537*, 2018.

[13] T. M. Cover and J. A. Thomas, *Elements of information theory.* John Wiley & Sons, 2012.

[14] M. C. Shewry and H. P. Wynn, "Maximum entropy sampling," *Journal of applied statistics*, vol. 14, no. 2, pp. 165–170, 1987.

[15] A. H. El-Shaarawi and W. W. Piegorsch, *Encyclopedia of environmetrics.* John Wiley and Sons, 2001, vol. 1.

[16] C.-W. Ko, J. Lee, and M. Queyranne, "An exact algorithm for maximum entropy sampling," *Operations Research*, vol. 43, no. 4, pp. 684–691, 1995.

[17] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Citeseer, Tech. Rep., 2009.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.