On the paper "Sum-Product Networks: A New Deep Architecture"

Jhonatan S. Oliveira oliveira@cs.uregina.ca
Department of Computer Science
University of Regina
Regina, Canada

Editor: CS807 - Advanced Architecture - Dr. Gerhard and Trevor Tomesh

Abstract

The paper "Sum-Product Networks: A New Deep Architecture" introduces a new mathematical model for inference and learning, called *sum-product networks*. The new model is represented as directed acyclic graph with multiple layers, thus the classification as a deep architecture. The authors are clear and concise with their comments. The paper is well written and excels in presenting the considerable big content in the constrain of a limited space.

Keywords: sum-product network, learning, deep architecture

1. Introduction

Sum-product networks (SPNs) (Poon and Domingos, 2011) are a new kind of deep architecture to graphically model real world problems. The paper (Poon and Domingos, 2011) won the award for best paper at the 27th Uncertainty in Artificial Intelligence (UAI) conference in 2011. The UAI is a respected conference in the field of artificial intelligence. Thus, the paper must be well written and with strong impact in its research area. Indeed, my critic on this paper is very positive, as reviewed in this paper. Few negative comments are due to the lack of space, a limitation imposed by the conference, but this does not compromise the high quality of the work.

The key idea in (Poon and Domingos, 2011) is to propose a solution for learning and inference in probabilistic graphical models (PGMs). It is known (Koller and Friedman, 2009) that inference in PGMs are exponential in worse case. Moreover, learning is also hard to manage given two constrains: the size of the graphical component and the size of the probability information (Zhao et al., 2015). Thus, SPNs draws inspiration from the work of Darwiche (2009) to solve this issues. The graphical component is a rooted directed acyclic graph (DAG) and the probability information is saved on some nodes and edges of the DAG. The DAG has two types of nodes: sum and product, which values are computed by summing or multiplying its children, respectively. Inference is done by computing all nodes' values going up on this DAG until the root is reached.

My critic for the paper is positive. The work is overall well described and the presentation is logically well done. Relating the novel work to related current ones is also well done by the authors, which has a whole section on it. The only negative critics goes to the brief discussion on background information and the quick passage between sections, without further details on the presented ideas. But this is acceptable, since it is known that the paper conference has space limitations.

The remainder of this critic follows. In Section 2 we give a brief introduction to the work presented in the analyzed paper. The critic is then given in Section 3. Section 4 draws conclusions.

2. Background

Sum-product networks (SPNs) are based on the ideas of a *network polynomial*. These are polynomial expression that encodes probability distributions. They make use of indicator variables attached to each variable of the distribution in order to activate (indicator equals 1) or deactivate (indicator equals 0) each correspondent probability distribution variable. When all indicators in a network polynomial are set to 1, that is they are all activated, the evaluation of the polynomial yields the unnormalized probability of the evidence. In other words, if the probability distribution is normalized this evidence value will be 1, otherwise it will be any other real number. The network polynomial is exponential in the number of variables, but it can be compactly represented in a graph as a SPN.

Formally, a SPN over a set of variables is a rooted directed acyclic graph with leaves being indicator variables and internal nodes being sum or products. All edges from a sum node has non-negative weights attached to it. The value of a product node is the product of its children and the value of a sum node is the sum of the product of the value of its children and the weights. For example, Figure 1 illustrated a SPN for a naive Bayes model presented in (Poon and Domingos, 2011). The SPN represents a network polynomial. Thus, by changing the indicator variable values we set different states of the probability distribution represented by the SPN and, consequently, the network polynomial. A SPN is then classified as a probabilistic graphical model (PGM), since it is defined by a graph with probability informations encoded.

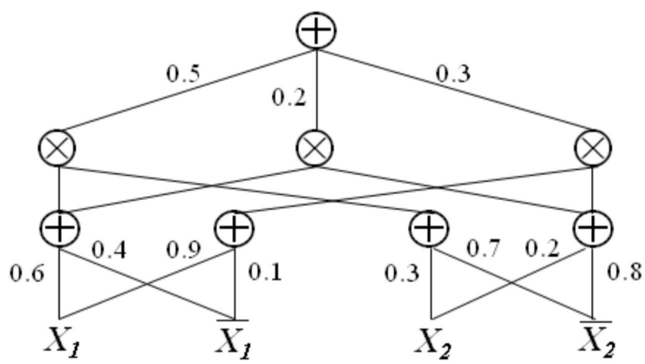

Figure 1: A SPN representing a naive Bayes, as shown in (Poon and Domingos, 2011).

The key advantages of SPNs are in the learning process, when compared to other PGMs. Learning a PGM usually involves two steps: first learning the graph structure and later the probability informations. This method is problematic when trying to optimize the learning process by limiting the size of learning informations Zhao et al. (2015). If a restriction is imposed on the graph size, for optimization reasons, it is not guaranteed that the probability informations will also be restricted in size. The other way around is also true: if a restriction is applied to the size of the learned probability information, it is not guaranteed that the graph will be restricted in size. This is a practical issue when learning PGMs. On the other hand, SPNs has the intrinsic feature which guarantees that if the graph is restricted in size the probability information will be as well.

3. Critic

We start by first commenting on the proposed work itself. SPNs are a very promising PGM. Its feature to improve learning is of practical value, but the model was mathematically introduced with satisfactory rigour. The model has internal characteristics which are feasible to test in practice. These internal characteristics tells when the model will be able or not to run inference in reasonable time. Thus, one can learn a SPN while watching for these characteristics to be controlled. In this way, the learned model is guaranteed to do inference.

Besides that, the authors based their work on the arithmetic circuits (ACs) from Darwiche (2009). This credit is stated right at the beginning of the paper and later one when comparing their proposed model with related works. The relation between ACs and SPNs though could be further clarified. It is possible to detect the similarities from the definitions, but there is a lack of clear motivation that would differentiate an AC model from a SPN. Indeed, in Zhao et al. (2015) the authors emphasize the relation between ACs and SPNs. In that work, it is clear that SPNs are defined in a way to be a standalone model. It is worth mentioning that ACs are formally defined to be a compilation of a given BN. That is, one could see an AC as an alternative way of telling the BN. In this sense, defining a SPN as a standalone model implies in having the same expressivity of an AC with the extended feature of modelling any other possible probability distribution that fits into the internal characteristics of a SPN. This is a very nice point to make in the paper, although it is obfuscated by technical discussion on the model itself.

A personal opinion on the proposed model would be with the lack of semantic on it. Consider other PGMs, for instance a BN. One can tell the relationship between variables and even infer other ones by using a BN. On the contrary, a SPN does not have enough semantic to describe these relations neither a easy way of inferring them. That is, learning a SPN might be beneficial but the user loses the ability of understanding the learned variables and its interactions.

Now, we comment on the paper itself. The work is careful described. This can be seen by the good division and flow of ideas. The introduction is well motivated and makes an overview of the whole paper. It also comments on related works, which gives a mor current context to the proposed work. The Sum-Product Networks section describes the model with formal definitions and common examples. This makes the new work more easy to follow and show its soundness. We notice though a small lack of background information on this section. For instance, the new model is introduced in page 2 of the paper, after a brief paragraph of definitions and contextualization. The missing background can be of course caused by the lack of space in a conference paper, but this does not affect the work enough to stop the reader of understanding the proposed ideas. The proofs are short and could have been more smooth, but again this can be due to the lack of space. Section 3 makes a good comparison with relates models, which emphasize the relevance of the new work and how it overcome the other models' issues. The following section on Learning Sum-Product networks is just a brief overview of the method, but enough to describe how one can learn a SPN from historical data. The most impressive section is the experiment results, which shows a SPN beating famous deep architecture models in a practical application of facial completing. This ends the paper with a very good impression.

The authors also have one section before the conclusion to talk about one possible relation between SPNs and the cortex. They associate sum and product nodes of a SPN with two different types of cortex neurones. This serve as a motivation for neuroscience applications that could flourish from the usage of SPNs. Besides that, it also motivate the study of how this relationship can be beneficial for the SPNs itself.

4. Conclusions

SPNs are a new deep architecture to represent probability distributions. They are better than other PGMs when learning from historical data. The proposed model, though, lacks semantic when telling the relationship between variables, when compared to PGMs. The work is carefully described and mathematically sound. The lack of space in this conference paper made some parts of the report be brief, but not enough to affect the understanding. The flow of ideas is well done and the presentation well conducted.

References

Adnan Darwiche. *Modeling and Reasoning with Bayesian Networks.* Cambridge University Press, 1 edition, 2009.

D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques.* MIT Press, 2009.

Hoifung Poon and Pedro M Domingos. Sum-Product Networks: A New Deep Architecture. *UAI*, pages 337–346, 2011.

Han Zhao, Mazen Melibari, and Pascal Poupart. On the Relationship between Sum-Product Networks and Bayesian Networks. *ICML*, pages 116–124, 2015.