GME: A Mixture of Experts Model for Software Fault Prediction

AMAN OMER, SANTOSH SINGH RATHORE, and SANDEEP KUMAR

I. Mixture of Experts (MoE)

Combining classifiers approaches are promising ones as they aim to improve the performance of the classification models [1], [2], [3], specifically for complex problems, which involve the limited number of training samples, high-dimensional features, and highly overlapped classes [4], [5]. Many previous works have shown that combining the classifiers approach is the most effective when the used underlying experts' (base learners) are negatively correlated or uncorrelated [6]. Therefore, more improved approaches need to be developed, which can produce accurate and negatively correlated experts that can be combined and produced improved performance [7]. The MoE method is a type of combining approach, which uses individual learning techniques as experts who are specialized in its particular subspace of input space [8]. Jacobs et al. originally proposed this technique in 1991 as an "Adaptive mixture of local experts" [9]. It uses the idea of dividing the input space into the number of subspaces, trains experts in each subspace, and combines the learning of experts using a gating function. The work of Jacobs et al. [10] has shown that in comparison to the common combining approaches, which produce unbiased experts with uncorrelated estimated errors, the MoE method produces biased experts with negatively correlated errors. This special feature of MoE compared to common combining approaches helped in achieving improved model performance. The working of the MoE method is based on the divide-and-conquer principle, where the input space is partitioned randomly into a number of subspaces using a special employed function, the experts become specialized on each generated subspace. After that, with the help of a gating function, the weights of the experts are computed dynamically according to the local efficiency of each expert. The experts are performing supervised learning in that their individual outputs are combined with modeling the desired output [8]. There is, however, the experts are also performing self-organized learning. That is, they self-organize to find a good partitioning of the input space so that each expert does well at modeling its own subspace, and as a whole group, they model the input space well. The working overview of the MoE is explained in Figure 1.

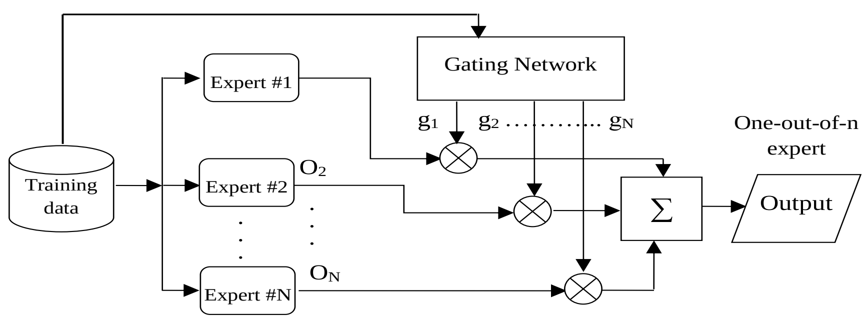

Fig. 1. Overview of the MoE method (based on [11])

The MoE method mainly has three components: (1) several intermediate experts, which are either regression or classifiers depending upon the given problem domain, (2) a gating function that partitions the input space into a number of subspaces using soft partition boundaries, and (3) and a probabilistic model to combine the experts and the gate. The final prediction output of the MoE method is a weighted sum of experts, where weights are dynamically updated via an input-dependent gating function. These properties of MoE help in representing non-stationary or piecewise continuous data in a prediction process and identification of the nonlinearities in a classification process. One of the main advantages of MoE is that it is flexible to combine a variety of different learning techniques. The simulation of the MoE model as presented by R. Jacobs [10] showed that MoE leads to the negatively correlated experts, which is the foremost requirement of any combining approach. Further, the author stated that the negative correlations come because the MoE method adaptively partitions the input space into regions so that the target function has different properties in each region. The experts learn from these different regions and different experts provided different "basis" functions.

References

[1] Yun Zhang, David Lo, Xin Xia, and Jianling Sun. Combined classifier for cross-project defect prediction: an extended empirical study. *Frontiers of Computer Science*, 12(2):280–296, 2018.

[2] Chubato Wondaferaw Yohannese, Tianrui Li, Macmillan Simfukwe, and Faisal Khurshid. Ensembles based combined learning for improved software fault prediction: A comparative study. In *2017 12th International Conference on Intelligent Systems and Knowledge Engineering (ISKE)*, pages 1–6. IEEE, 2017.

[3] Fatih Yucalar, Akin Ozcift, Emin Borandag, and Deniz Kilinc. Multiple-classifiers in software quality engineering: Combining predictors to improve software fault prediction ability. *Engineering Science and Technology, an International Journal*, 23(4):938–950, 2020.

[4] Sushant Kumar Pandey, Ravi Bhushan Mishra, and Anil Kumar Tripathi. Bpdet: An effective software bug prediction model using deep representation and ensemble learning techniques. *Expert Systems with Applications*, 144:113085, 2020.

[5] Xinli Yang, David Lo, Xin Xia, and Jianling Sun. Tlel: A two-layer ensemble learning approach for just-in-time defect prediction. *Information and Software Technology*, 87:206–220, 2017.

[6] Isobel Claire Gormley and Sylvia Frühwirth-Schnatter. Mixture of experts models. *Handbook of Mixture Analysis*, pages 271–307, 2019.

[7] Pierre Geurts. Bias vs variance decomposition for regression and classification. In *Data mining and knowledge discovery handbook*, pages 733–746. Springer, 2009.

[8] Saeed Masoudnia and Reza Ebrahimpour. Mixture of experts: a literature survey. *Artificial Intelligence Review*, 42(2):275–293, 2014.

[9] Robert A Jacobs, Michael I Jordan, Steven J Nowlan, Geoffrey E Hinton, et al. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87, 1991.

[10] Robert A Jacobs. Bias/variance analyses of mixtures-of-experts architectures. *Neural computation*, 9(2):369–383, 1997.

[11] Saeed Reza Kheradpisheh, Fatemeh Sharifizadeh, Abbas Nowzari-Dalini, Mohammad Ganjtabesh, and Reza Ebrahimpour. Mixture of feature specified experts. *Information Fusion*, 20:242–251, 2014.