ICANN19
28th International Conference on Artificial Neural Networks

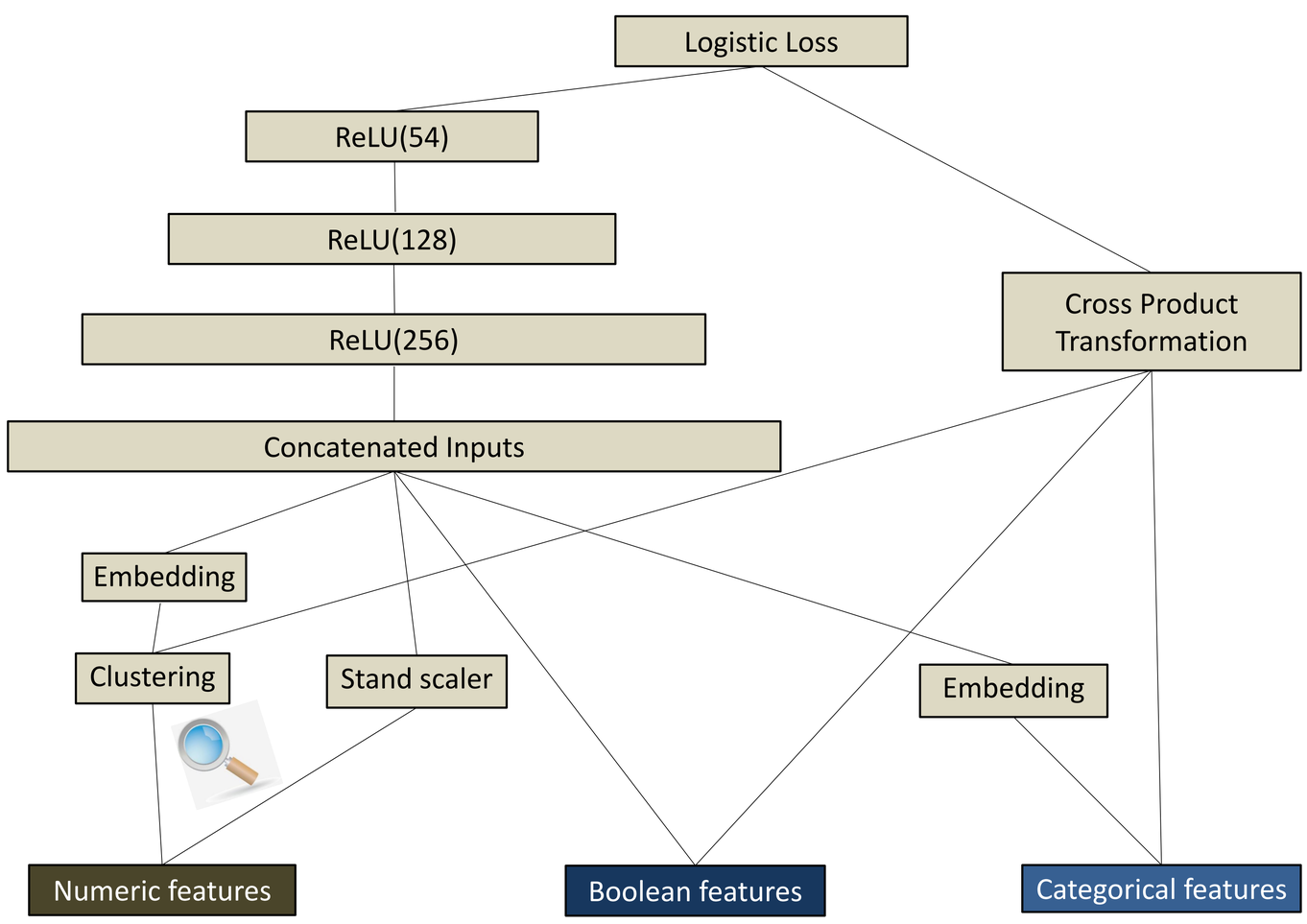
Intrusion Detection via Wide & Deep Model

Zhipeng Li, Zheng Qin, and Pengbo Shen School of Software, Tsinghua University
Contact Author: ZhipengLi Email: lizp14@mails.tsinghua.edu.cn

Abstract:

Intrusion detection system is designed to detect threats and attacks, which are especially important in nowadays' constantly emerging information security incidents. There has been a lot of work devoted to realizing anomaly detection mode of intrusion detection via deep learning since deep learning becomes a research hot spot. However, there is rare work that uses different deep learning networks as hybrid architecture to benefit the advantages of each special part. In this paper, we are inspired by Google's Wide & Deep model which is proposed to combine memorization with generalization via different networks. We propose a framework to use Wide & Deep model for intrusion detection. To get comprehensive categorical representations of continuous features, we use a density-based clustering (DBSCAN) to convert the KDD'99 \NSL_KDD format features into sparse categorical feature representations. A widely used and popular NSL _KDD dataset is used to evaluate the model. A comprehensive empirical evaluation with hypothesis testing demonstrates that the revised Wide & Deep framework outperforms the separated part alone. Compared with other machine learning base line methods and advanced deep learning methods, the proposed model outperforms the baseline results and achieves a steady and promising performance in tests with different levels.

Overview of Deep & Shallow model for intrusion detection



Clustering Algorithm:

- DBSCAN (NSL_KDD dataset)
- float rate features (ϵ=0.01， MinPts =50)
- Integer features (ϵ=1， MinPts =50)

Feature type	Feature name	Category conversion
Numeric features	same_srv_rate	[0.0,0.003] 0.04 [0.05,0.06]…0.5 [0.13,0.99]
	duration	[0,5] [6,42908]
…		
Boolean features	land	0 1
	logged_in	0 1
…		
Categorical features	protocol_type	tcp udp icmp
	flag	SF S0 REJ RSTR SH RSTO S1 RSTO S0 S3 S2 OTH
…		

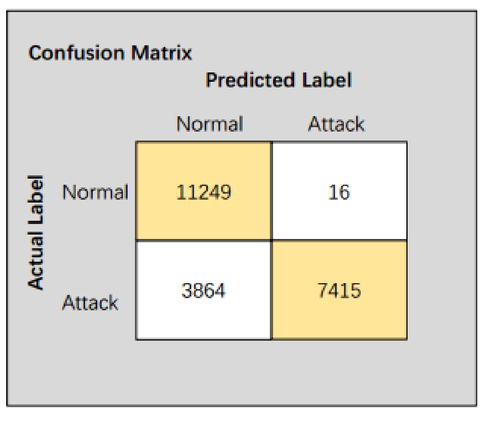
$$Deep\ part: a^{(l+1)} = f(W^{(l)}a^{(l)} + b^{(l)})$$

$$Cross\ product\ transformation: \varphi_k(x) = \prod_{i=1}^{d} x_i^{c_{ki}}\ c_{ki} \in 0,1$$

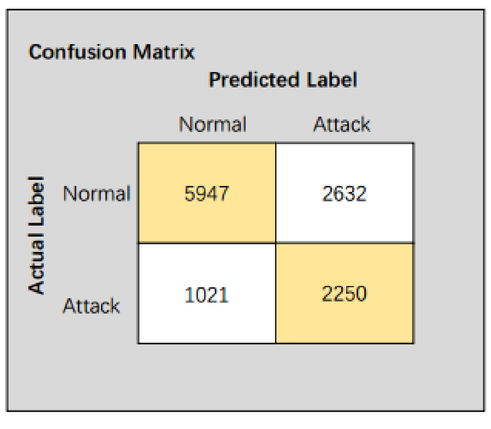
$$Joint\ training: P(Y=1|x) = \sigma(w_{wide}^T[x, \varphi(x)] + w_{deep}^T a^{lf} + b)$$

Binary classification metrics of different models

	Accuracy	Precision	Recall	F1 score
Wide model on Test^{+}	76.12%	80.42%	**75.09%**	77.66%
Deep model on Test^{+}	77.68%	81.47%	60.31%	69.31%
Wide & Deep model on Test^{+}	**82.79%**	**92.16%**	74.43%	**82.34%**
Wide model on Test^{-21}	66.74%	67.13%	**86.74%**	75.69%
Deep model on Test^{-21}	67.23%	67.77%	75.56%	71.45%
Wide & Deep on Test^{-21}	**69.17%**	**69.32%**	85.34%	**76.50%**



NSL_KDD Test^{+}



NSL_KDD Test^{-21}

Accuracy comparison of different models

Model	KDD Test^{+}	KDD Test^{-21}
J48 [20]	81.05%	63.97%
Naive Bayes [20]	76.56%	55.77%
NB Tree [20]	82.02%	66.16%
Random Forest [20]	80.67%	63.26%
Random Tree [20]	81.59%	58.51%
Muti-layer Perceptron [20]	77.41%	57.34%
SVM [20]	69.52%	42.29%
RNN [23]	**83.28%**	**68.55%**
Semantic LSTM [10]	82.21%	66.10%
CNN (ResNet50) [11]	79.14%	81.57%
CNN (GoogLeNet) [11]	**77.04%**	**81.84%**
Wide & Deep	**82.79%**	**69.17%**

Other contact Information: Wechat Number: THUpeterpan

ManuscriptID:351