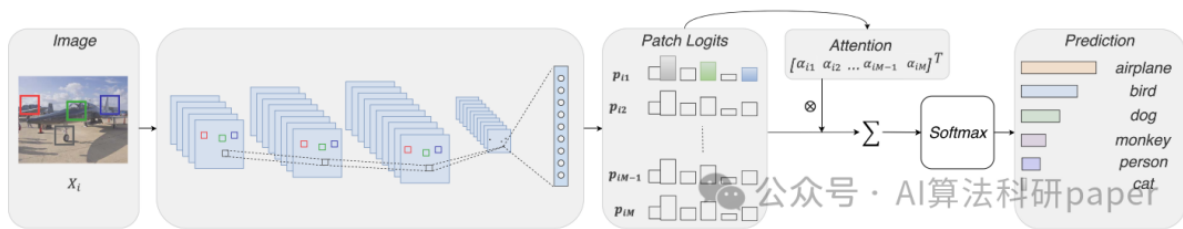


Loss+Attention

Loss-Based Attention for Interpreting Image-Level Prediction of Convolutional Neural Networks

<https://github.com/xssshi2015/Loss-based-Attention-for-Interpreting-Image-level-Prediction-of-Convolutional-Neural-Networks>

方法：论文提出了一种新的通用注意机制，称为基于损失的注意力，通过利用相同的参数来学习图块权重和逻辑（类向量）以及图像预测，从而将注意机制与损失函数相连接，以提高图块精度和召回率。通过修改当前流行的卷积神经网络，作者设计了两种深度架构，分别是卷积架构和胶囊架构，用于挖掘图像中的显著图块，从而解释哪些部分决定了图像的决策。



Loss-Based Attention (LBA) 架构是一种修改后的深度学习网络结构，旨在提高卷积神经网络（CNN）的可解释性。该架构通过挖掘图像中对最终决策有重要影响的区域（patches），来解释图像级预测是如何做出的。以下是LBA架构的关键特点：

- 移除池化操作：**LBA架构移除了传统CNN中的池化（pooling）操作，如最大池化（max-pooling），以保持图像块（patches）之间的空间关系。
- 特征提取：**通过在卷积层中使用步幅（stride）来确定图像块的大小和数量，并提取这些块的特征。
- 注意力机制：**引入了一个基于 $\ell_2, 1$ -范数的注意力机制，用于计算每个图像块的权重。这个机制利用相同的参数同时学习块的权重和类别向量（logits），以及图像预测。
- 加权和：**在图像级决策中，将所有图像块的特征通过注意力权重进行加权求和，以生成最终的图像级预测。
- 损失函数：**LBA架构提出了一种新的损失函数，该函数结合了注意力权重，以鼓励选择与图像类别一致的重要块，并提高块的精确度和召回率。
- 卷积和胶囊网络：**LBA可以应用于不同的深度架构，包括卷积网络和胶囊网络。对于卷积网络，LBA通过卷积层实现；对于胶囊网络，LBA通过胶囊层实现。
- 实验验证：**通过在大规模基准数据库上的实验，LBA架构证明了其在图像分类任务上可以获得与现有流行网络相比更好或有竞争力的性能，并且具有更好的可解释性。
- 开源代码：**论文提供了实现LBA方法的源代码链接，以便其他研究人员可以复现和利用这些方法。

LBA架构的核心在于其能够通过注意力机制突出图像中重要的部分，同时忽略不相关的区域，从而提高了模型的可解释性，并且通过定制的损失函数来提升模型性能。

创新点：

- 提出了一种新的注意力机制，称为Loss-Attention机制。该机制利用相同的参数学习补丁权重和标签，从而将注意力机制与损失函数相连接。这种机制可以挖掘显著的补丁，并通过新的损失函数进一步提高其精确度和召回率。
- 通过修改当前流行的卷积神经网络和胶囊网络，提出了两种新的深层架构。这两种架构都保留了图像补丁的空间关系，使图像级决策成为补丁的加权和。

Backbone	Method	Params ($\times 10^6$)	CIFAR-10	CIFAR-100	SVHN
ConvNet					
	VGG11_bn	28.15	92.14	70.20	96.48
	VGG16_bn	33.65	93.52	72.35	96.83
VGG11_bn	Mean-pooling	30.73	93.99	74.03	96.37
	Attention	30.80	94.17	75.47	96.52
	Gated-Attention	30.86	93.78	74.35	96.88
	Loss-Attention	30.73	94.51	76.81	96.99
	ResNet18	11.17	94.76	76.61	96.98
	ResNet50	58.16	95.01	78.15	97.12
ResNet18	Mean-pooling	32.67	94.59	75.58	96.72
	Attention	32.73	94.83	77.48	97.08
	Gated-Attention	32.80	94.89	76.33	97.07
	Loss-Attention	32.67	95.31	78.53	97.11
CapNet					
	Sabour et al. [10]*	-	89.40	-	95.70
	Nair et al. [47]	-	67.53	-	91.06
	HitNet et al. [48]	-	73.30	-	94.50
	DeepCaps [11] [†]	-	91.01	-	97.16
VGG11_bn	Dynamic Routing	33.41	93.37	68.41	96.08
	Loss-Attention	31.02	94.98	75.84	96.78
ResNet18	Dynamic Routing	35.35	93.88	68.88	96.70
	Loss-Attention	32.96	95.47	76.70	97.24

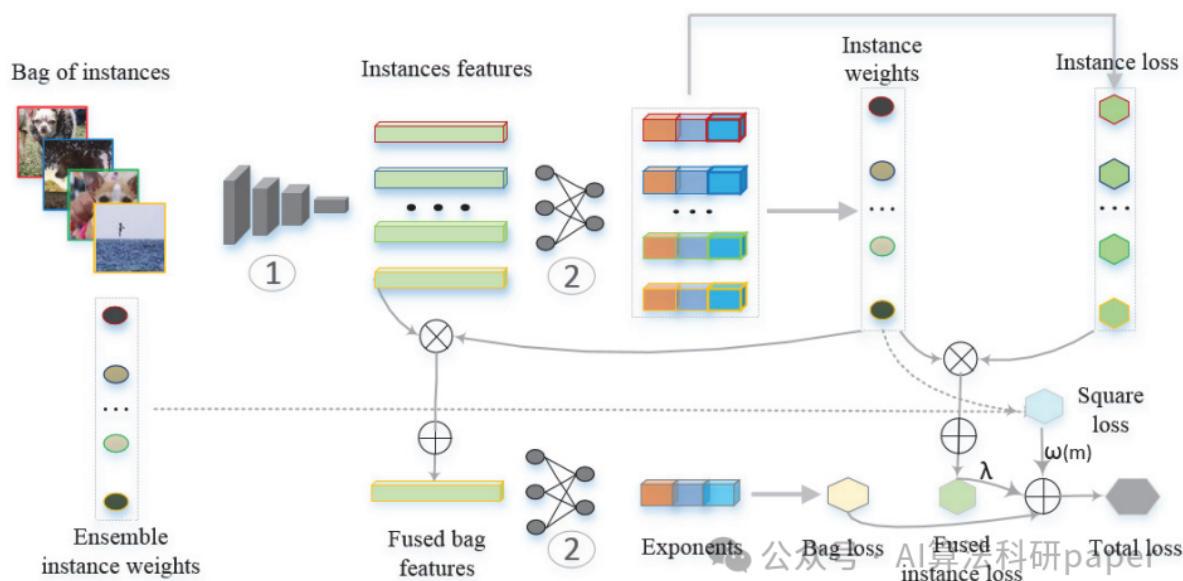
Loss-Based Attention for Deep Multiple Instance Learning

<https://github.com/xsshi2015/Loss-Attention>

方法：本文提出了一种新颖的基于损失的注意机制，通过将注意机制与softmax和交叉熵损失函数相连接，同时学习实例权重和预测结果，以及深度多实例学习中的包预测。提出的注意机制使用全连接层的参数来学习实例权重，并根据损失函数直接计算实例权重。

主要流程如下图：

- 1) 数字1表示用于获取实例特征的传统或卷积神经网络；
- 2) 数字2表示接指数函数的全连接层；
- 3) 实例权重和实例损失点积后求和作为融合的实例损失；
- 4) 包损失基于softmax加交叉熵计算获得；
- 5) 平方损失是训练过程的一致性代价。



创新点:

- 提出了一种新颖的基于损失的注意力机制，用于同时学习实例权重和预测以及包的预测。该注意力机制将注意力机制与softmax和交叉熵损失函数相连接，通过使用全连接层的参数来学习实例权重，并直接基于损失函数计算实例权重。
- 引入了一个由学习的权重和交叉熵函数组成的正则化项，以进一步提高实例的召回率，并引入了一致性成本来平滑神经网络的训练过程。

Method	MUSK1	MUSK2	FOX	TIGER	ELEPHANT
mi-SVM	0.874±N/A	0.836±N/A	0.582±N/A	0.784±N/A	0.822±N/A
MI-SVM	0.779±N/A	0.843±N/A	0.578±N/A	0.840±N/A	0.843±N/A
MI-Kernel	0.880±N/A	0.893±N/A	0.603±N/A	0.842±N/A	0.843±N/A
EM-DD	0.849±0.098	0.869±0.108	0.609±0.101	0.730±0.096	0.771±0.097
mi-Graph	0.889±0.073	0.903±0.086	0.620±0.098	0.860±0.083	0.869±0.078
miVLAD	0.871±0.097	0.872±0.095	0.620±0.098	0.811±0.087	0.850±0.080
miFV	0.909±0.089	0.884±0.094	0.621±0.109	0.813±0.083	0.852±0.081
mi-Net	0.889±0.088	0.858±0.110	0.613±0.078	0.824±0.076	0.858±0.083
MI-Net	0.887±0.091	0.859±0.102	0.622±0.084	0.830±0.072	0.862±0.077
MI-Net with DS	0.894±0.093	0.874±0.097	0.630±0.080	0.845±0.087	0.872±0.072
MI-Net with RC	0.898±0.097	0.873±0.098	0.619±0.104	0.836±0.083	0.857±0.089
Attention	0.892±0.090	0.858±0.106	0.615±0.096	0.839±0.054	0.868±0.054
Gated-Attention	0.900±0.088	0.863±0.094	0.603±0.068	0.845±0.046	0.857±0.064
Loss-Attention	0.917±0.066	0.911±0.063	0.712±0.074	0.897±0.065	0.900±0.069