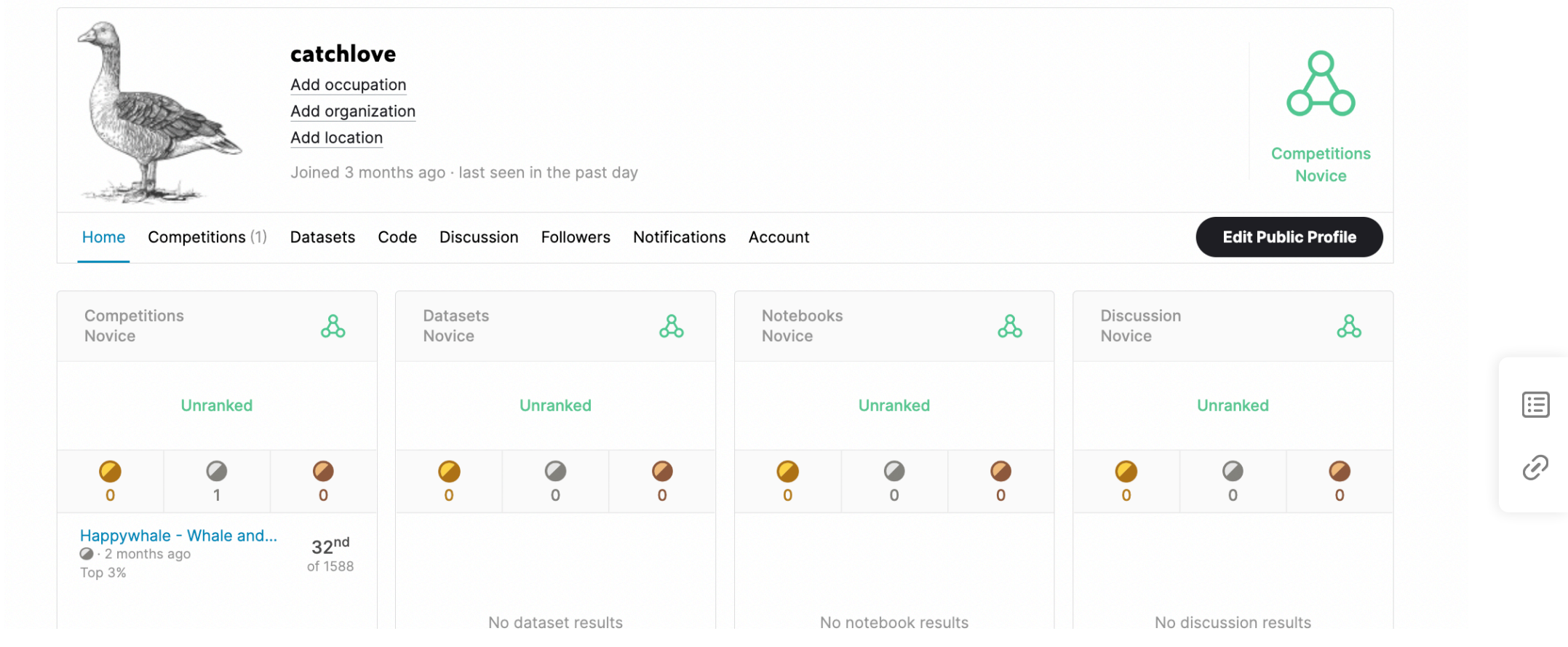


比赛介绍

<https://www.kaggle.com/competitions/happy-whale-and-dolphin>



工作一直较忙，所以现在才更新github。在这次竞赛中，参赛者将建立一个鲸鱼和海豚图像识别的自动化模型，研究人员可以将两者的识别时间减少99%以上。使研究者可以实现以前负担不起或不可能实现的研究规模。

- 竞赛类型：本次竞赛属于计算机视觉/图像检索，所以推荐使用的模型或者库：**EfficientNet/NFNet/Arcface/GeM**
- 赛题数据：本次竞赛的数据包含了从28个不同研究机构收集的30个不同物种的15000多个独特的海洋哺乳动物个体的图像。官方提供的训练集大约有**51033张图片**，测试集大约有**约28000张图片**，总数据集**62GB**。海洋研究人员已经对个体进行了人工识别并赋予了individual_id，而我们的任务是在图像中正确识别这些个体。
- 评估标准：**MAP@5**。对于测试集中的每张图片，最多可以预测5个individual_id标签。测试集中有一些个体在训练数据中没有出现，这些个体应该被预测为new_individual。
- 推荐阅读 Kaggle 内的一篇 EDA（探索数据分析）来获取一些预备知识：[🚡 EDA + Visualization + Augmentation 🔥 | Kaggle](#)

数据说明

本次竞赛的数据包含了从28个不同研究机构收集的30个不同物种的15000多个独特的海洋哺乳动物个体的图像。官方提供的训练集大约有**51033张图片**，测试集大约有**约28000张图片**，总数据集**62GB**。海洋研究人员已经对个体进行了人工识别并赋予了individual_id，而我们的任务是在图像中正确识别这些个体。

官方数据页面 [Happywhale - Whale and Dolphin Identification | Kaggle](#)

- 文件
 - train_images/ - 一个包含训练图像的文件夹
 - train.csv - 提供每个训练图像的**species**和**individual_id**
 - test_images/ - 一个包含测试图像的文件夹；
 - sample_submission.csv - 一个正确格式的样本提交文件
- train.csv 字段：
 - image – 图片ID
 - species – 物种
 - individual_id – 个体ID

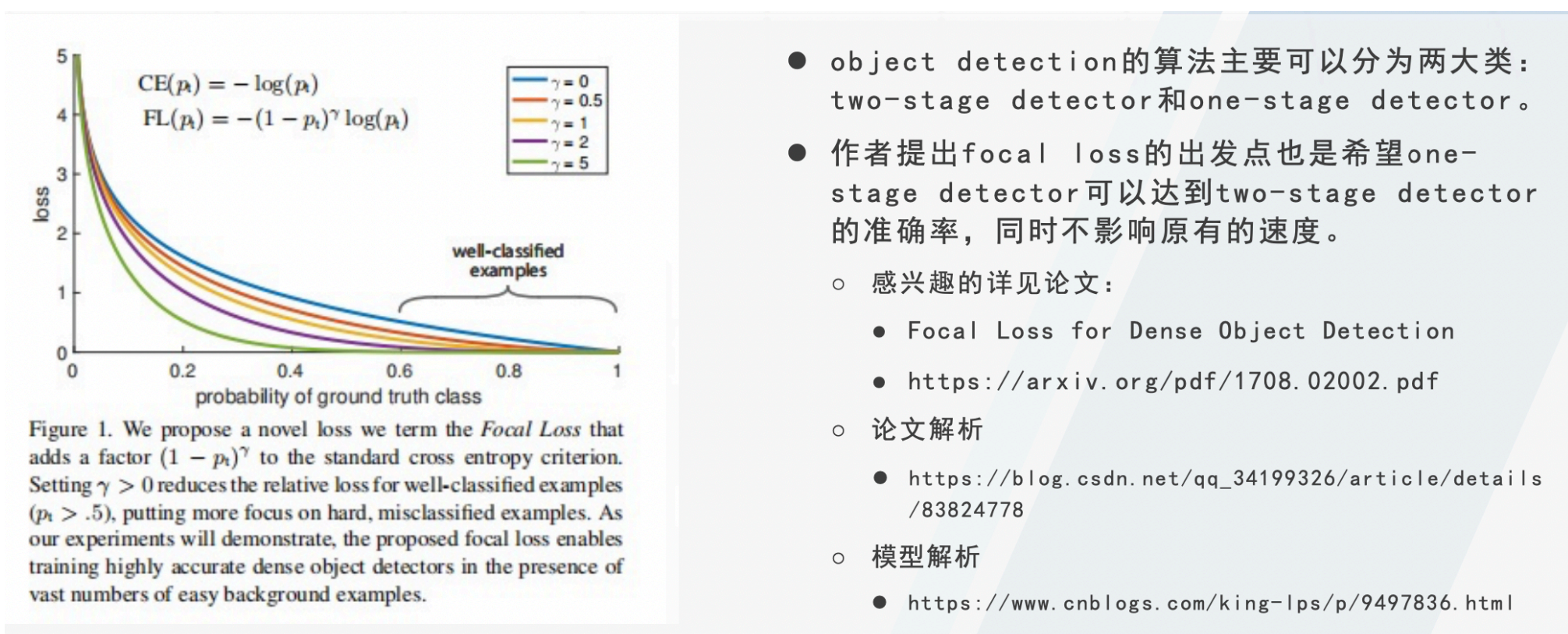
解决方案思路

本次竞赛我们的方案采用了 **EfficientNet_B6 / EfficientNet_V2_L / NFNet_L2** 的多模型融合，由于我们重点只需要关注鲸鱼的fullbody和backfins，所以我们使用Yolov5对原始图片进行裁切fullbody部位和backfins部位，在裁切后的图片上我们训练了tf_efficientnet_b6_ns, tf_efficientnetv2_l_in21k, eca_nfnet_l2 三个模型，并且都加上了GeM Pooling 和 Arcface head，所有模型都会输出一个**512维的Embedding**，我们将他们 concatenated 获得了一个 **512×9=4608** 维的向量，计算距离并选择最邻近的结果，我们还使用new_individual替换进行后处理，最后我们得到了**Private score 0.86+**。

常见数据增强

- RandomResizedCrop** 随机裁剪后固定尺寸
- VerticalFlip** 图片垂直翻转
- HorizontalFlip** 图片水平翻转
- RandomBrightnessContrast** 随机亮度、对比度、明度调整
- GaussianBlur** 高斯模糊
- Normalize** 归一化 让RGB三色的数值分布接近，可以使得训练更稳定

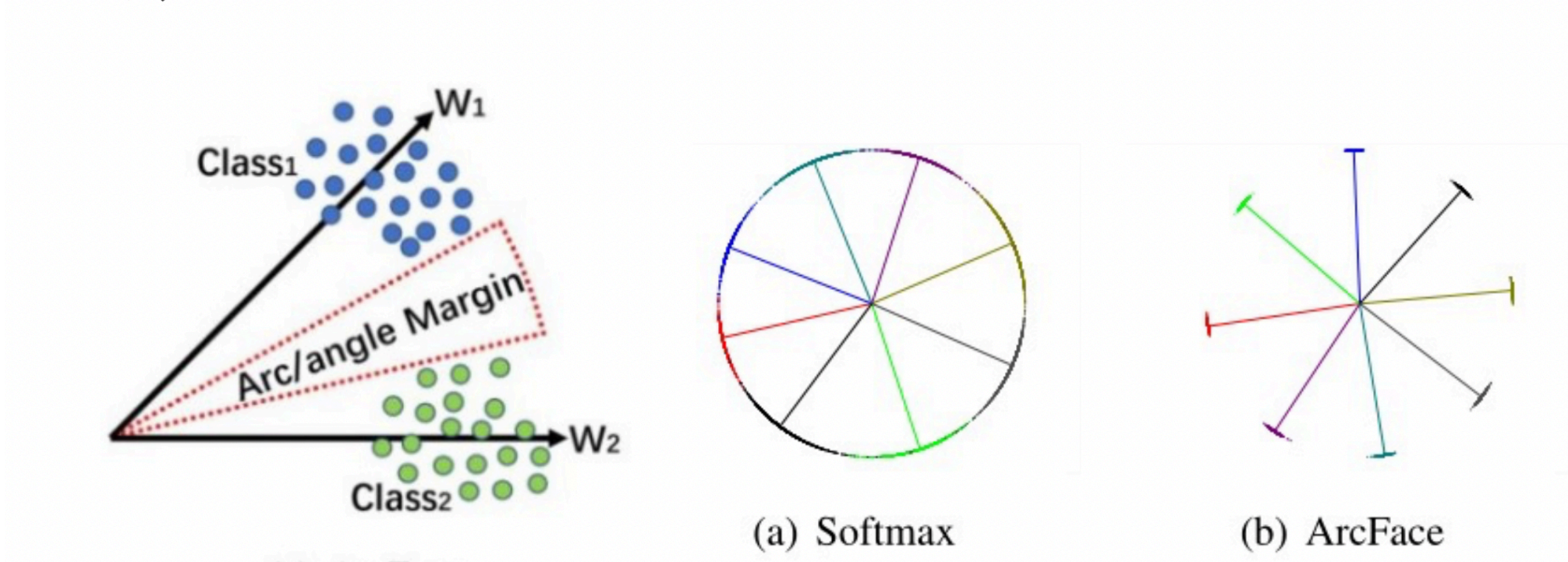
Focal loss（本次比赛没时间常识了）



ArcheFace

普通Softmax和ArcFace的区别

类内紧，类间宽



Gem pooling

- 介绍GeM Pooling之前，先要介绍一下Average Pooling 和 Max Pooling 。
- 一般来说，当Feature Map中的所有信息都应该有所贡献的时候用Average Pooling，例如图像分割中常用Global Average Pooling来获取全局上下文关系，再例如224*224图像分类将最后的7*7map进行Average Pooling，是因为网络深层的高级语义信息一般来说都能帮助分类器分类。
- 反之，为了减少无用信息的影响时用Max Pooling，比如网络浅层常常见到Max Pooling，因为开始几层对图像而言包含较多的无关信息。
- 另外Average Pooling与Max Pooling输出值会有不同的幅度区间，有些时候会遇到Max Pooling输出值幅度波动大，此时可以加一些归一化操作。二者的具体使用场景只有在具体任务上观察，实际效果炼丹后才知道。

比赛上分历程

- 使用Yolov5切分 fullbody数据和 backfins数据；
- 使用 tf_efficientnet_b0_ns + ArcFace 作为 Baseline，训练fullbody 512size, 使用kNN 搜寻，Public LB：0.729；
- 加入new_individual后处理，Public LB：742；
- 训练backfins 512size, Public LB：0.702；
- 使用fullbody 768size图像，并调整了数据增强，Public LB：0.770；
- 将fullbody 768size的模型和 backfins 768size模型两个模型融合，Public LB：0.791；
- 训练 tf_efficientnet_b6_ns，以及上述所有功能微调，Public LB：0.834；
- 训练 tf_efficientnetv2_l_in21k，以及上述所有功能微调，Public LB：0.856；
- 训练 eca_nfnet_l2，以及上述所有功能微调，Public LB：0.854；
- 将上述三个模型的5Fold，挑选cv高的，进行融合，Public LB：0.872；
- kNN 搜寻调参，Public LB：0.885+；