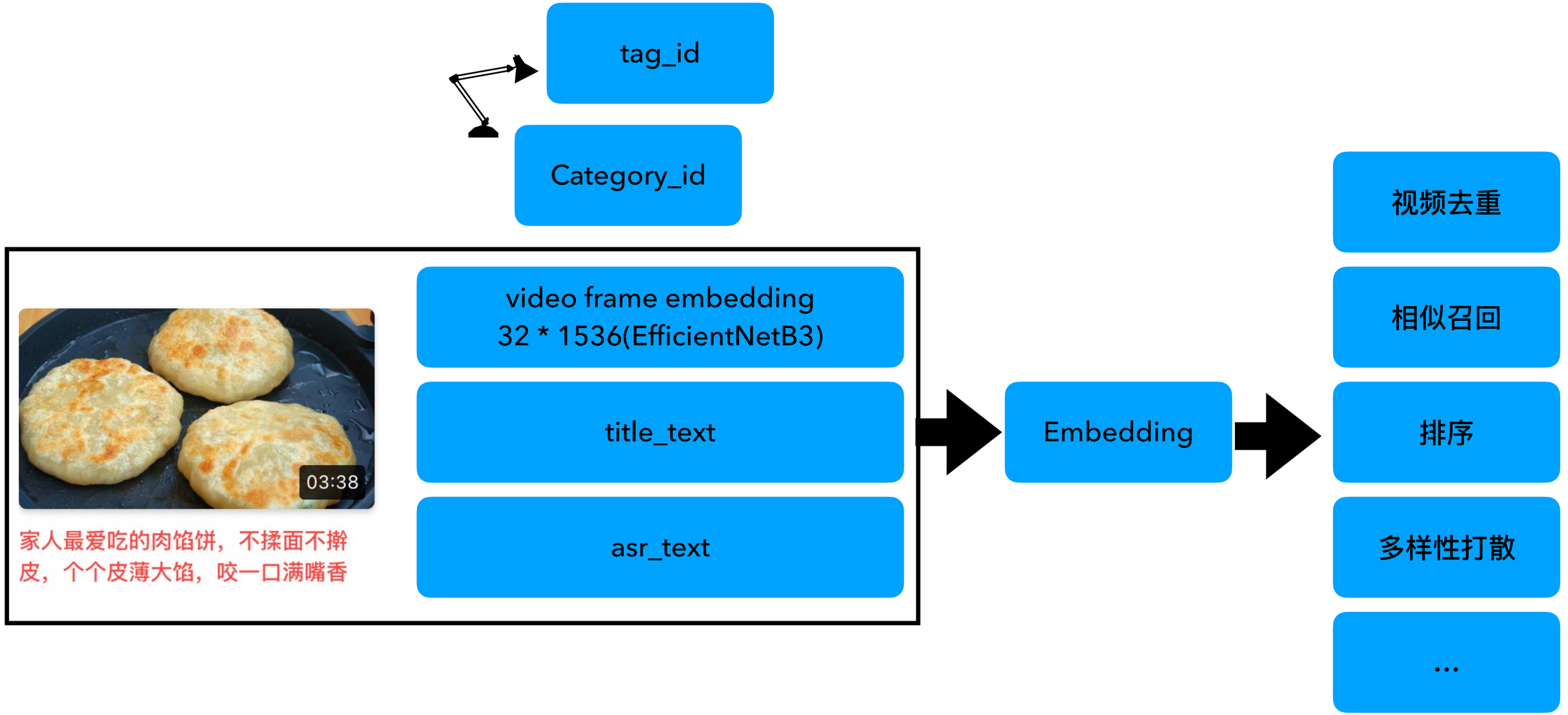
多模态视频检索


tag_id
Category_id
video frame embedding
32 * 1536(EfficientNetB3)
title_text
asr_text
03:38
家人最爱吃的肉馅饼，不揉面不擀皮，个个皮薄大馅，咬一口满嘴香
Embedding
视频去重
相似召回
排序
多样性打散
...

tag_id	Int64List	[41319, 3331787, 81457, 72361, 528121, 5200265, 74106]	人工标注的视频标签ID
category_id	Int64List	11007	人工标注的视频分类ID。 category_id可以拆解为两部分： 前三位为一级分类，后两位对应一级分类下二级分类。-1缺失

Data

视频数据分为三种：

1. pointwise dataset包含100w个视频样本
2. pairwise dataset包含67899个视频pair，共63613个视频；每个pair有人工标注的相似度得分
3. test dataset 包含初赛test_a和复赛test_b两份测试集，分别包含31514和43027个视频。测试集只提供视频原始特征，不包含任何人工标注信息

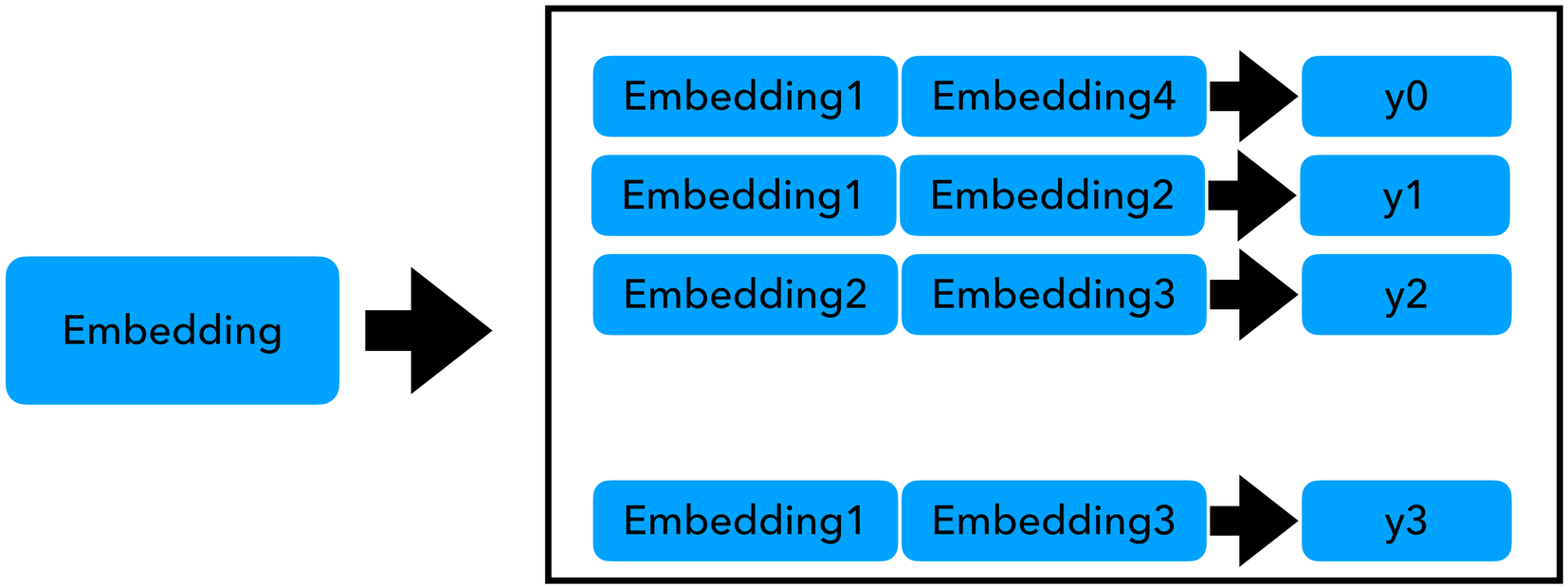
pairwise dataset的pair相似度得分以额外的tsv文件提供，具体包括id1, id2, score三列，以TAB做分隔符。

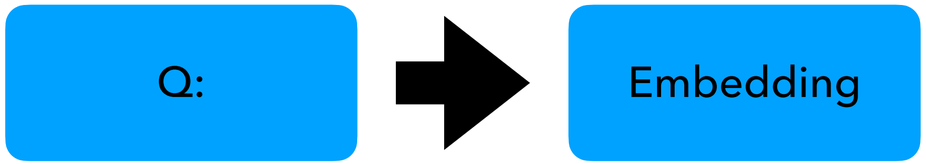
注：test dataset中所有的视频没有在pairwise dataset以及pointwise dataset中出现过

Data

字段	pointwise	pairwise	test
id	√	√	√
tag_id	√	√	×
category_id	√	√	×
title	√	√	√
frame_feature	√	√	√
asr_text	√	√	√

Task





参赛者需要提交所有测试集的视频embedding，具体要求如下：

1. embedding维度不能超过256维
2. 提交结果为dictionary，其中key是video id，格式是string，而value是video embedding，格式是float16的list。
3. 将上述结果用json.dump成.json文件，并用zip压缩成.zip格式

视频相似度度量的是两个视频内容的相似性。我们参考了Semantic Textual Similarity任务的做法，将相似程度分档，然后采用Spearman’s rank correlation指标来做评估。相比于文本，视频包含更丰富的多模态信息，也更难度量相似程度。

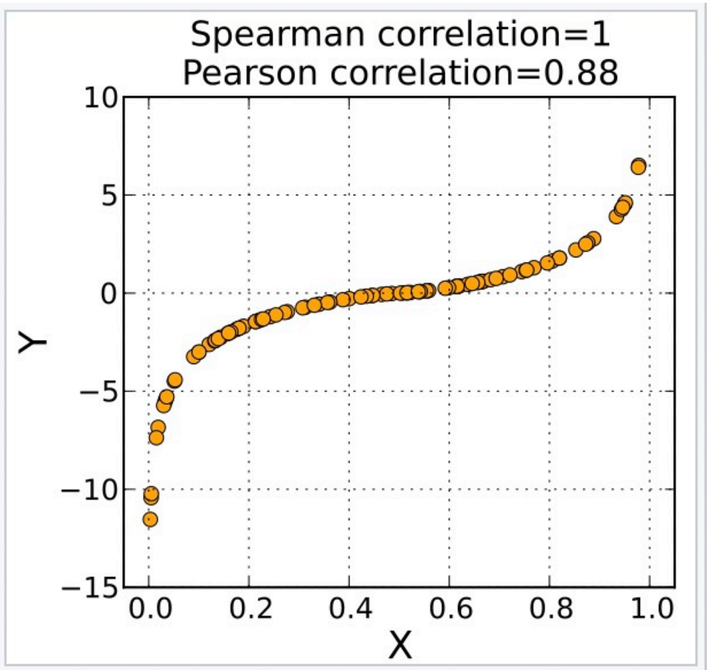
每个视频pair均由10人标注，取平均分作为标注结果。标注人员在标注界面同时播放两个视频（包括标题），标注员观看后，依据标注规范选择不相似、一般相似或强相似。相似强度的标准大体逻辑如下：如果两个视频的主题一致、核心元素（譬如剧名、人物、动作、场景等）相同，则认为是强相似，对应相似得分为1分；如果主题一致，核心元素稍有不同，则认为是一般相似，对应相似得分为0.5分；而如果主题不一致，或者主题一致但核心元素差异巨大，则认为不相似，对应相似得分为0分。我们事先对标注员做培训考核，事中做抽检，确保准确率在90%以上（低于90%的我们会整体打回重标）。

Evaluation

Spearman Correlation

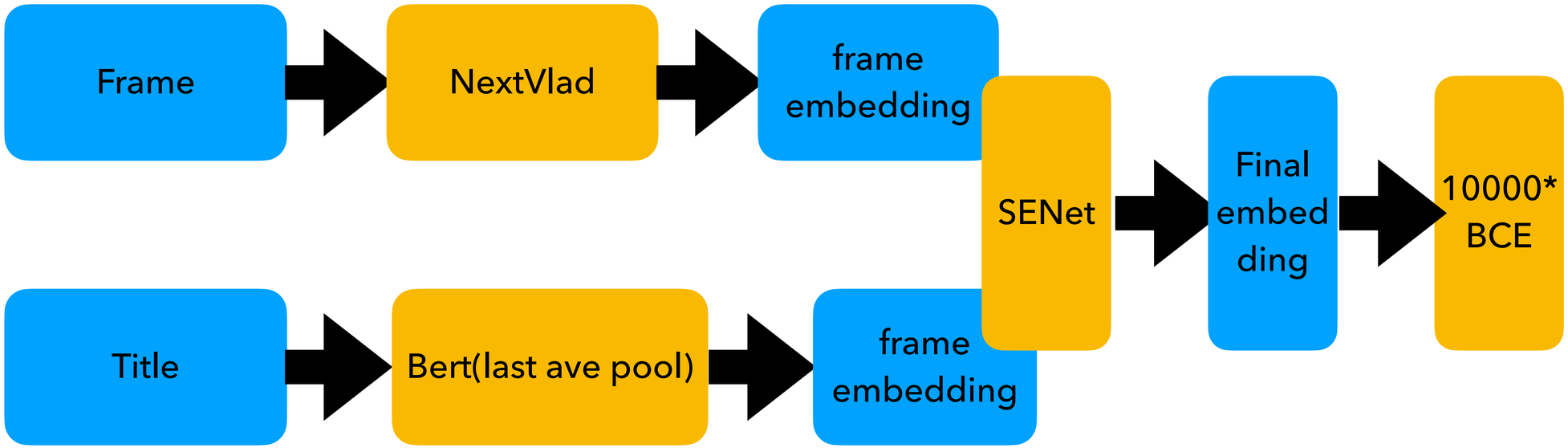
斯皮尔曼相关系数被定义成等级变量之间的皮尔逊相关系数。[1]对于样本容量为n的样本，n个 原始数据X_i, Y_i被转换成等级数据x_i, y_i，相关系数ρ为

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}.$$



斯皮尔曼等级相关系数为1表明两个被比较的变量是相关的，即使它们之间的关系并非线性的。相较而言，它并未给出完整的皮尔逊相关系数。

Official Baseline



NextVlad

3.1 NetVLAD Aggregation Network for Video Classification

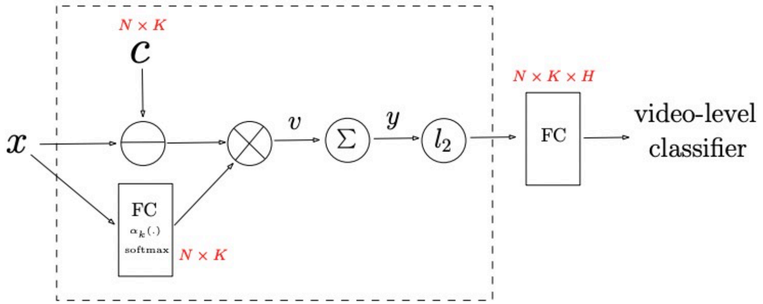


Fig. 1. Schema of NetVLAD model for video classification. Formulas in red denote the number of parameters (ignoring biases or batch normalization). FC means fully-connected layer.

3.2 NeXtVLAD Aggregation Network

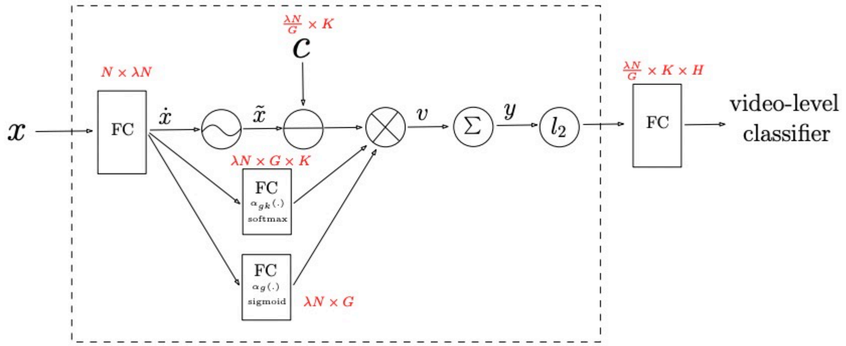


Fig. 2. Schema of our NeXtVLAD network for video classification. Formulas in red denote the number of parameters (ignoring biases or batch normalization). FC represents a fully-connected layer. The wave operation means a reshape transformation.

Related Task

- semantic textual similarity
- retrieval
- recognition or identity

- Imgae-Text Retrieval
- Text-Image Retrieval
- Video-Text Retrieval
- Text-Video Retrieval

CLIP2Video: Mastering Video-Text Retrieval via Image CLIP

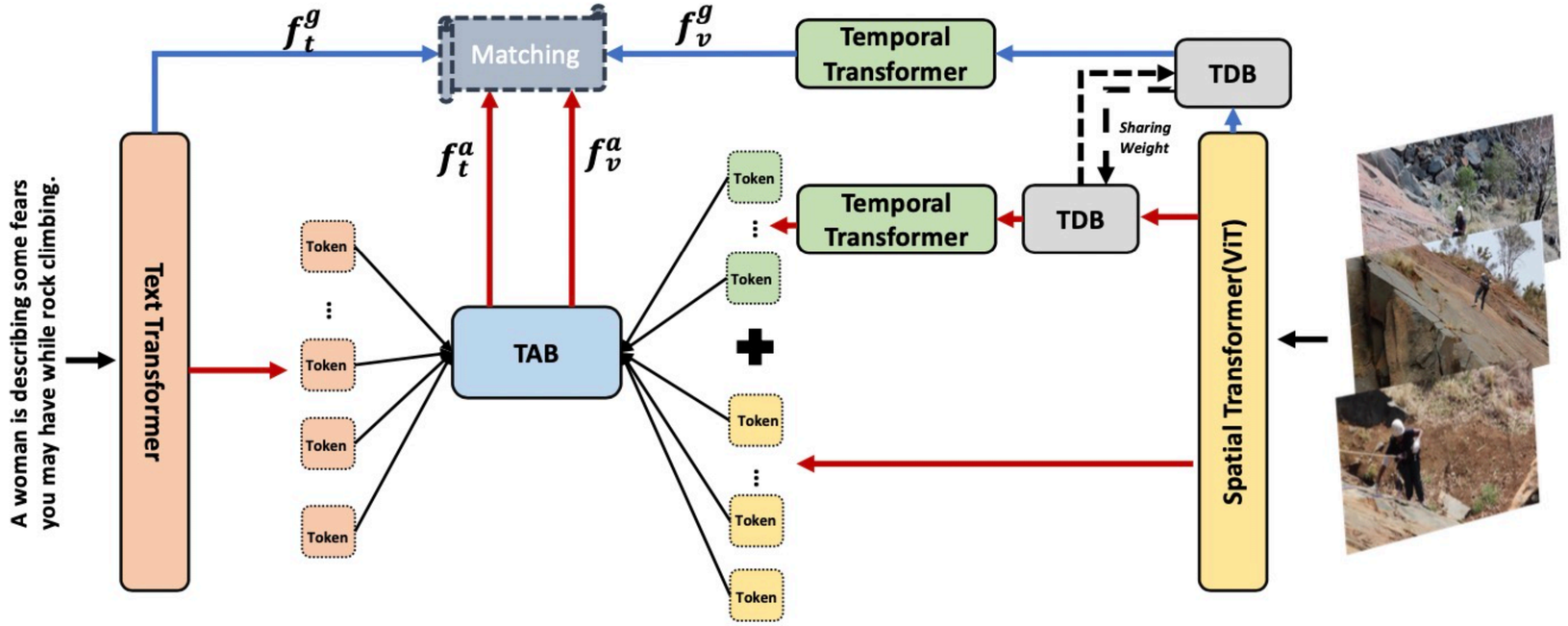
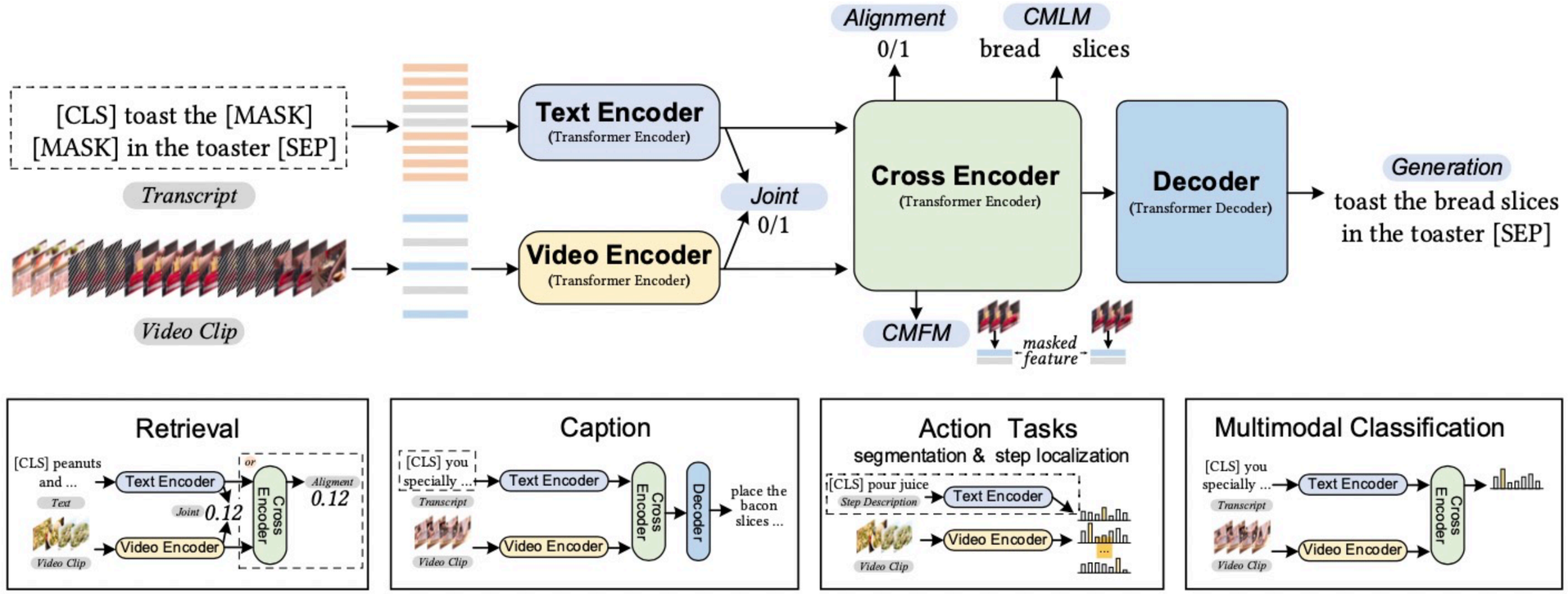


Figure 1. Overview of CLIP2Video. It consists of two key components: Temporal Difference Block (TDB), which is used to enhance temporal interaction between frames; Temporal Alignment Block (TAB), which is adopted to align video clips and contextual words in the same space, capturing the motion change by cross-modal understanding.

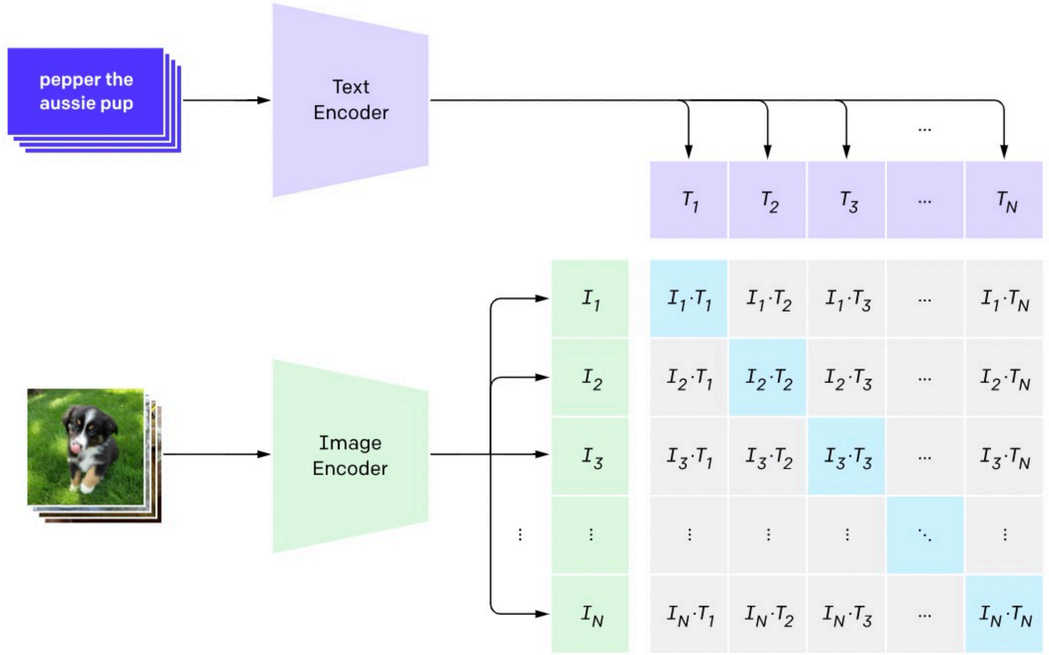
1	**CAMoE**	3	32.9	58.3	68.4	42.6	3.8	1	59.8	92.8	86.2	✓	Video-Text Retrieval by Multi-Stream Corpus Alignment and Dual Softmax Loss		2021
2	**CLIP2Video**	4	29.8	55.5	66.2	45.4	5.3	1	54.6	82.1	90/8	✓	CLIP2Video: Mastering Video-Text Retrieval via Image CLIP		2021
3	**TACo**	5	24.8		64.0	52.1						✓	TACo: Token-aware Cascade Contrastive Learning for Video-Text Alignment		2021
4	**MDMMT**	6	23.1	52.8	61.8	49.8						✓	MDMMT: Multidomain Multimodal Transformer for Video Retrieval		2021
5	**UniVL**	6	21.2		63.1	49.6						✓	UniVL: A Unified Video and Language Pre-Training Model for Multimodal Understanding and Generation		2020

https://paperswithcode.com/sota/video-retrieval-on-msr-vtt

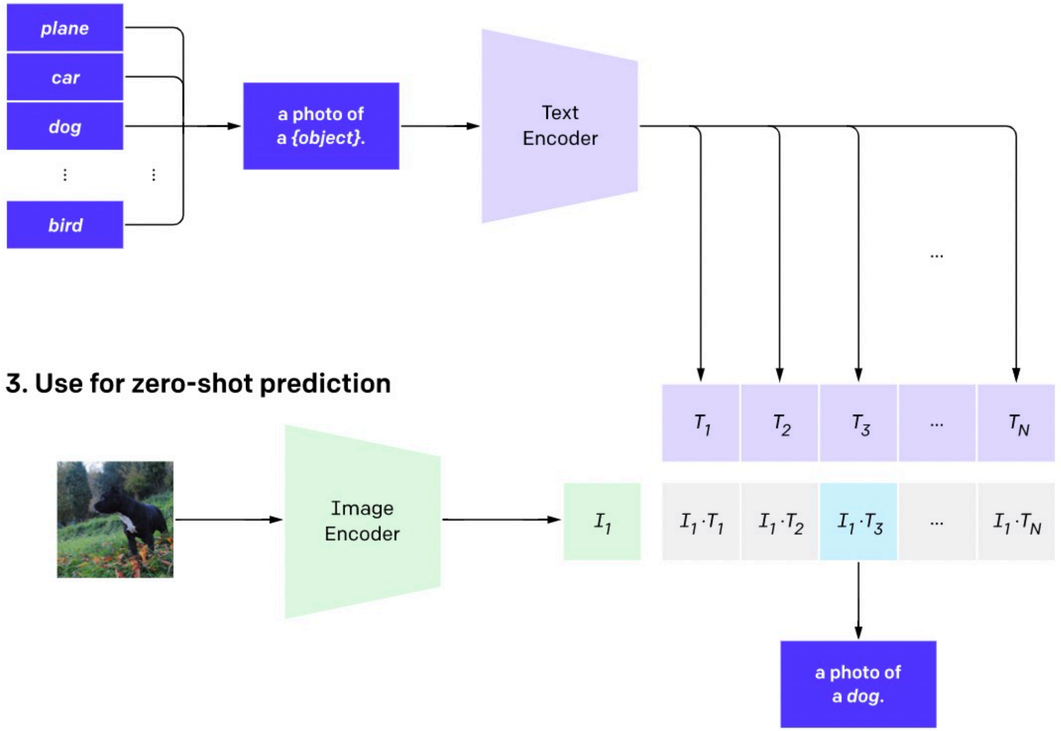
UniVL: A Unified Video and Language Pre-Training Model for Multimodal Understanding and Generation



1. Contrastive pre-training

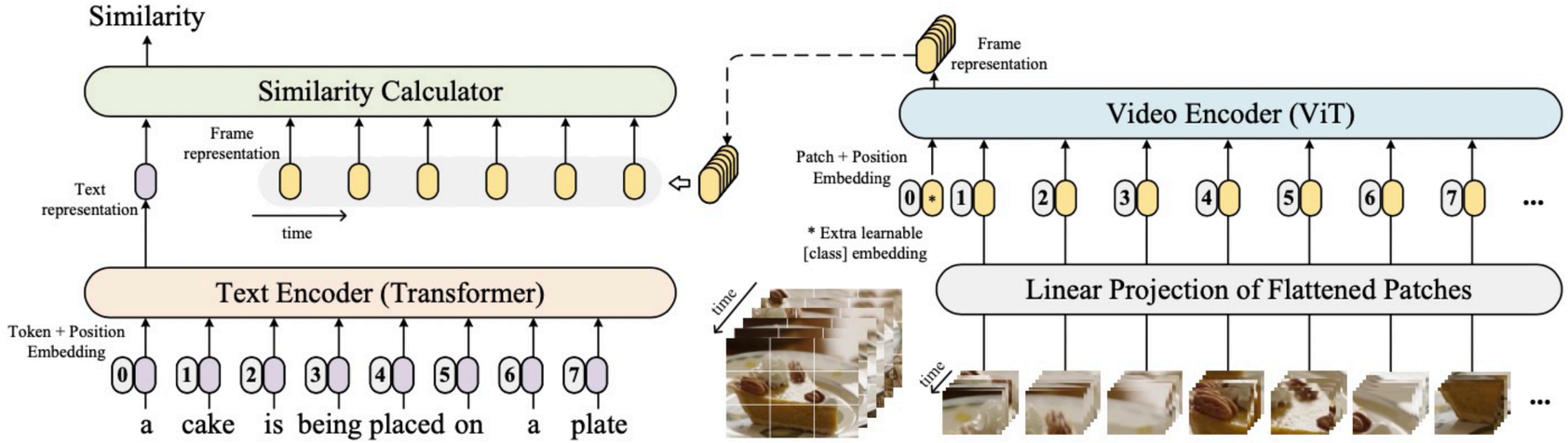


2. Create dataset classifier from label text

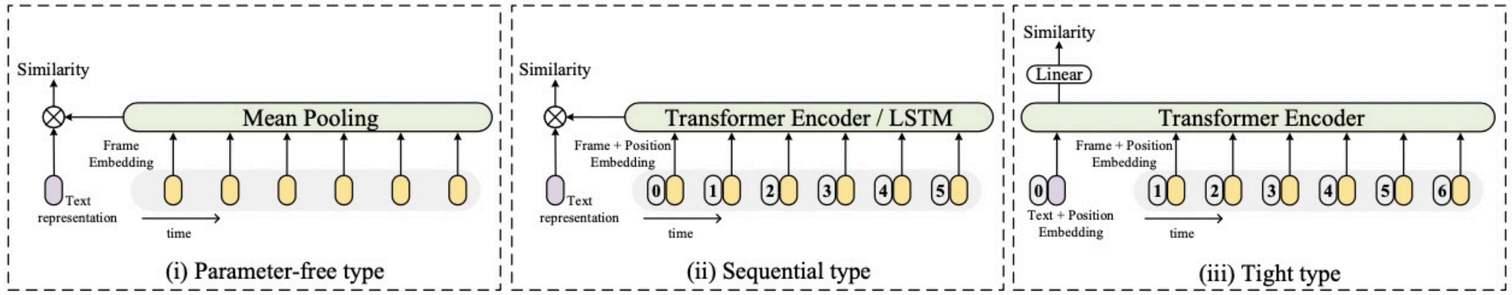


3. Use for zero-shot prediction

CLIP4Clip: An Empirical Study of CLIP for End to End Video Clip Retrieval

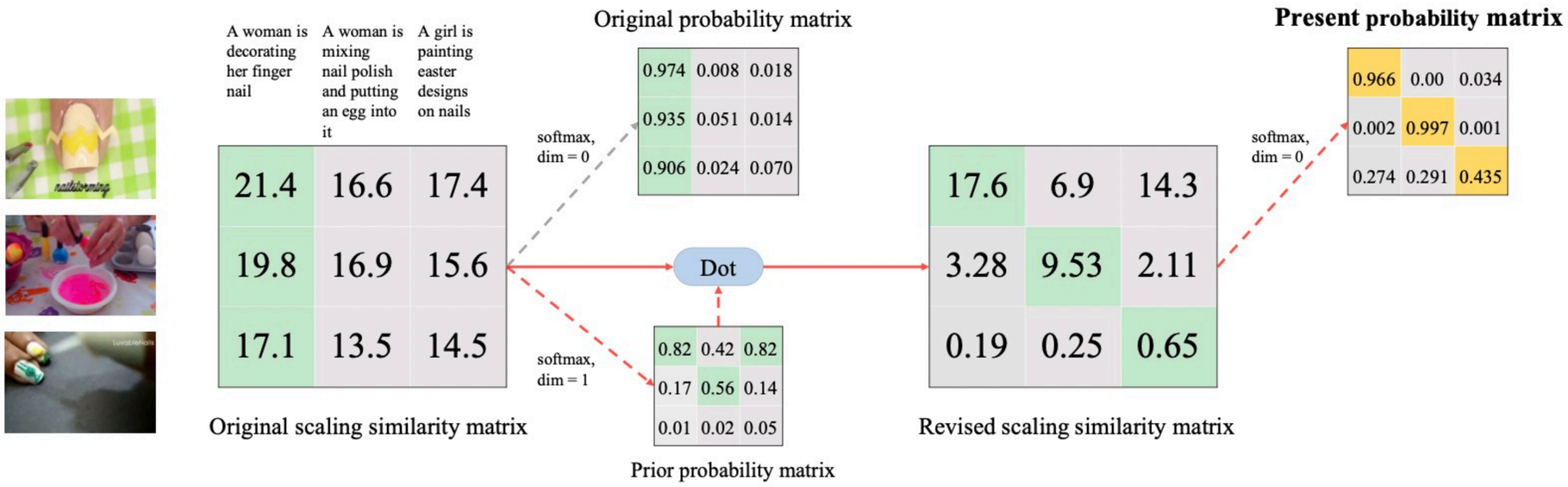


(a) Main structure

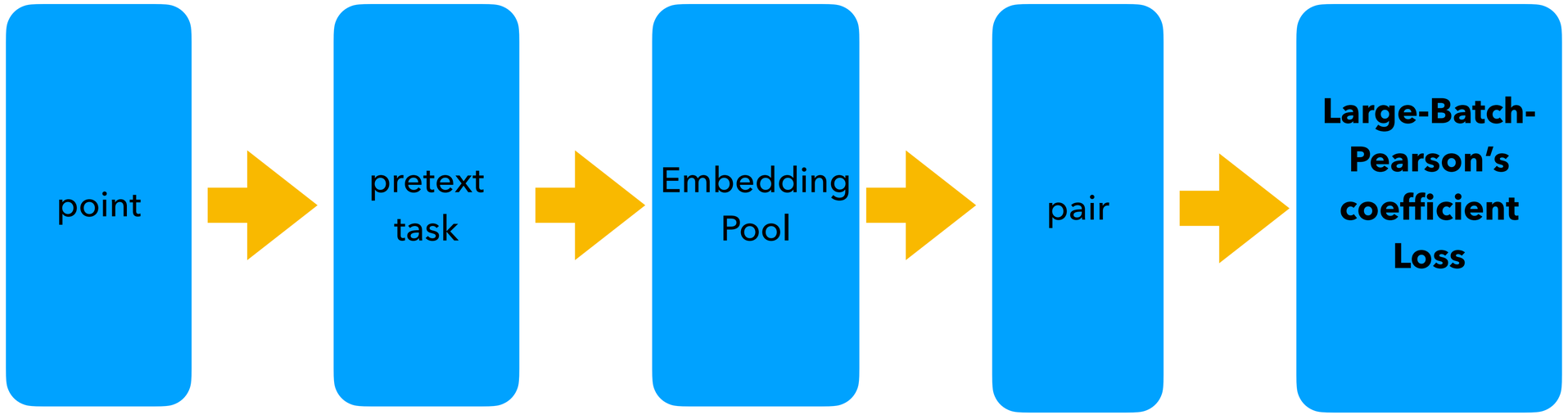


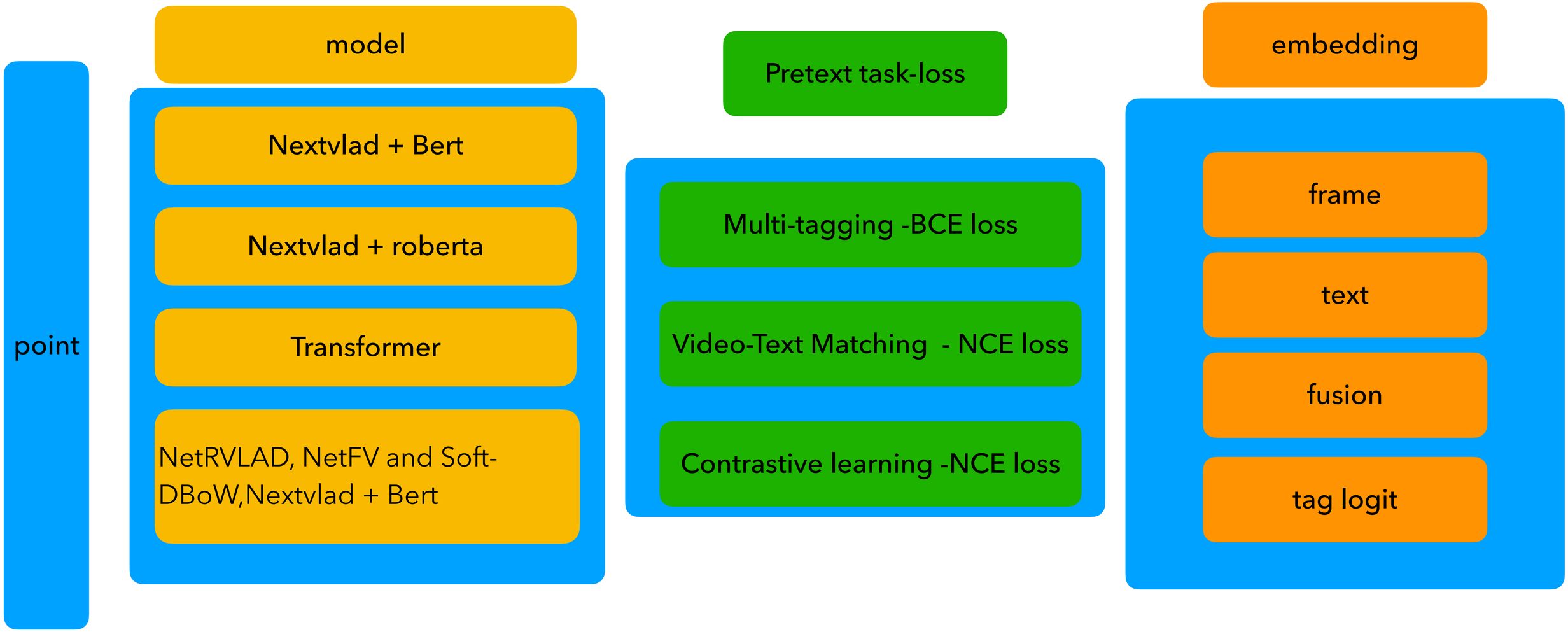
(b) Similarity calculator

Improving Video-Text Retrieval by Multi-Stream Corpus Alignment and Dual Softmax Loss



Solution

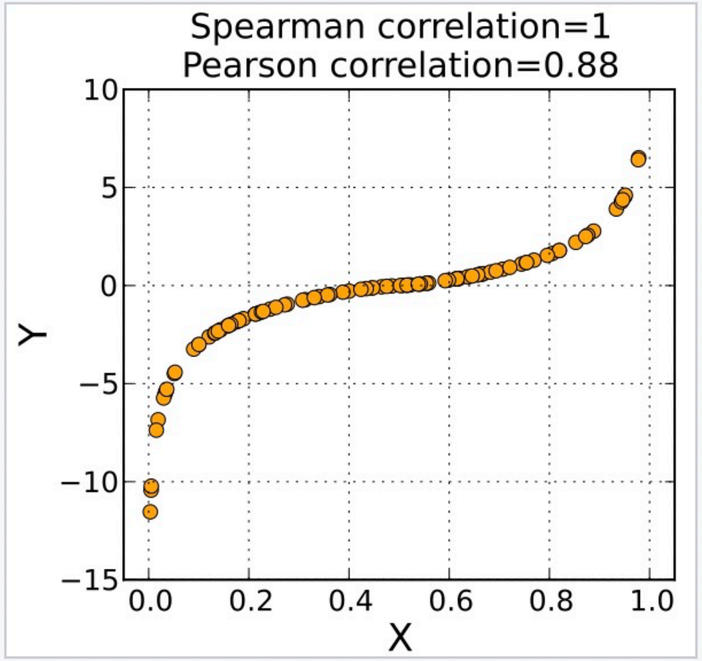

point
pretext task
Embedding Pool
pair
Large-Batch-Pearson's coefficient Loss


point
model
Nextvlad + Bert
Nextvlad + roberta
Transformer
NetRVLAD, NetFV and Soft-DBoW,Nextvlad + Bert
Pretext task-loss
Multi-tagging -BCE loss
Video-Text Matching - NCE loss
Contrastive learning -NCE loss
embedding
frame
text
fusion
tag logit

Large-Batch-Pearson's coefficient Loss

斯皮尔曼相关系数被定义成等级变量之间的皮尔逊相关系数。[1]对于样本容量为n的样本，n个 原始数据X_i, Y_i被转换成等级数据x_i, y_i，相关系数ρ为

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}.$$



斯皮尔曼等级相关系数为1表明两个被比较的变量是相关的，即使它们之间的关系并非线性的。相较而言，它并未给出完整的皮尔逊相关系数。

两个变量之间的皮尔逊相关系数定义为两个变量的协方差除以它们标准差的乘积：

$$\rho_{X,Y} = \frac{\operatorname{cov}(X,Y)}{\sigma_X \sigma_Y} = \frac{E[(X - \mu_X)(Y - \mu_Y)]}{\sigma_X \sigma_Y}$$

上式定义了**总体**相关系数，常用希腊小写字母 ρ (rho) 作为代表符号。估算样本的协方差和标准差，可得到**样本相关系数**(样本皮尔逊系数)表示：

$$r = \frac{\sum_{i=1}^{n}(X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i - \overline{Y})^2}}$$

```python
class SpearmanCorrelationLoss(nn.Module):
    def __init__(self, temp=0.2):
        super(SpearmanCorrelationLoss, self).__init__()
        self.temp = temp
    def forward(self, input1, input2, label, return_sim=False):
        similarity = cosine(input1, input2)
        similarity_sm = F.softmax(similarity / self.temp, dim=0)
        similarity_sm = similarity_sm -  torch.mean(similarity_sm)
        label = label - torch.mean(label)
        t_m1 = torch.sqrt(torch.sum(similarity_sm ** 2))
        t_m2 = torch.sqrt(torch.sum(label ** 2))
        correlation = torch.sum(similarity_sm*label) / (t_m1 * t_m2 + 0.00001)
        if return_sim:
            return -correlation, similarity
        else:
            return -correlation
```