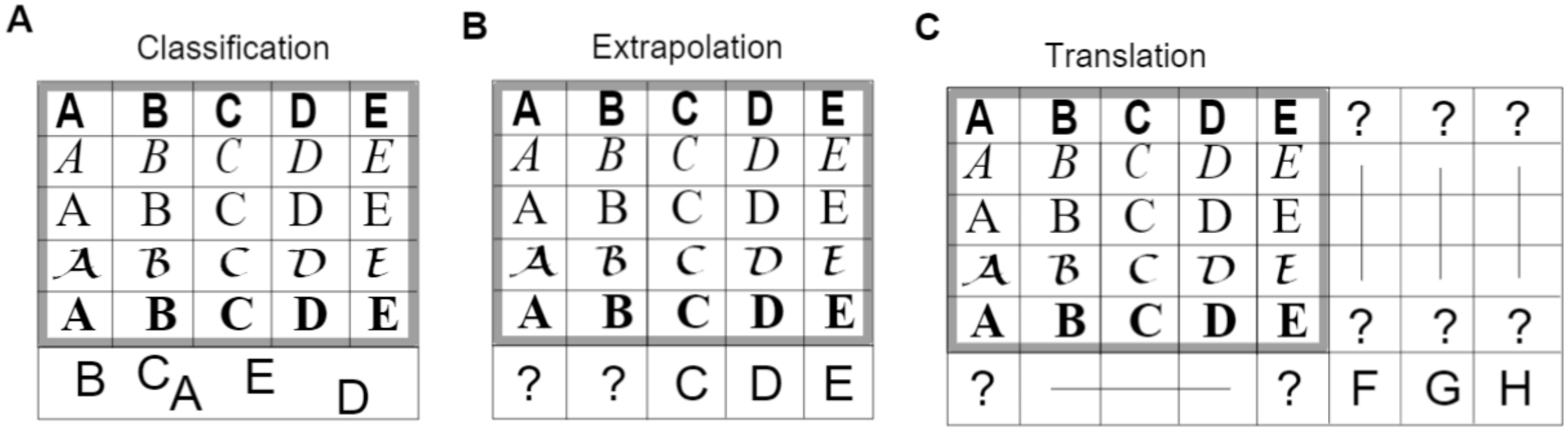
Bilinear Models

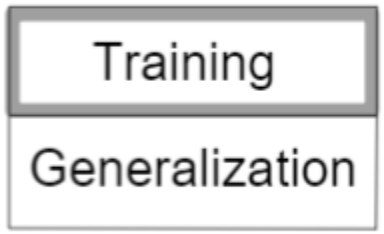


Given a labeled training set of observations in multiple styles (e.g., fonts) and content classes (e.g., letters), we want to

(A) classify content observed in a new style,

(B) extrapolate a new style to unobserved content classes, and

(C) translate from new content observed only in new styles into known styles or content classes.



双线性模型模型

- Linear

  假设V, W 为线性空间，$f : V \to W$ 两个线性空间的映射，如果满足：

$$f(v_1 + v_2) = f(v_1) + f(v_2)$$
$$f(\alpha v) = \alpha f(v)$$

  $f : V \to W$ 是线性的。

- Bilinear

  假设U, V, W 为线性空间，$f : V \times U \to W$ ，如果满足：

$$f(u_1 + u_2, v) = f(u_1, v) + f(u_2, v)$$
$$f(u, v_1 + v_2) = f(u, v_1) + f(u, v_2)$$
$$f(\alpha u, v) = \alpha f(u, v) = f(u, \alpha v)$$

  $f : V \times U \to W$ 是双线性的。

  当 v 固定，$f(u, v)$ 在 u 中是线性的：

$$f(u, v) = f_v(u) = f_v(u_1 + u_2) = f_v(u_1) + f_v(u_2)$$
$$f(\alpha u, v) = f_v(\alpha u) = \alpha f_v(u)$$

双线性模型是具有可分性的双因子模型：当任意一个因子保持不变时，模型的输出对于另一个因子是线 性的。

对称模型

对称模型中，用向量 $\mathbf{a}^s$ 和 $\mathbf{b}^c$ 来表示style和content，分别有 I 和 J 维，用 k 维的 y^{sc} 来表示样式s下c的观察向量。那么

$$y_k^{sc} = \sum_{i=1}^{I} \sum_{j=1}^{J} w_{ijk} a_i^s b_j^c$$

对称模型

$$y_k^{sc} = \sum_{i=1}^{I} \sum_{j=1}^{J} w_{ijk} a_i^s b_j^c$$

$$y_k^{sc} = \mathbf{a}^{s\,\mathrm{T}} \mathbf{W}_k \mathbf{b}^c$$

K个矩阵 $\mathbf{W}_k$ 描述了从style和content空间到K维observation空间的双线性映射。

$$\mathrm{y}^{sc} = \sum_{i,j} \mathbf{w}_{ij} a_i^s b_j^c$$

$\mathbf{w}_{ij}$ 是一个k维的向量，那么 y^{sc} 可以看作是由这些基向量混合$\mathbf{a}^s$和$\mathbf{b}^c$的张量积得到。

非对称模型

有时，在训练中学习的一些基本style的线性组合可能不能很好地描述新的style。可以通过让交互项 w_{ijk} 本身随style变化来获得更灵活的不对称模型。

$$y_k^{sc} = \sum_{i,j} w_{ijk}^s a_i^s b_j^c$$

非对称模型

$$y_k^{sc} = \sum_{i,j} w_{ijk}^s a_i^s b_j^c$$

讲上式的style相关项合并：

$$a_{jk}^s = \sum_i w_{ijk}^s a_i^s$$
$$y_k^{sc} = \sum_j a_{jk}^s b_j^c$$

用 $\mathbf{A}^s$ 来表示由分量 $\{a_{jk}^s\}$ 组成的 $I \times K$ 矩阵：

$$\mathbf{y}^{sc} = \mathbf{A}^s \mathbf{b}^c$$

这里 a_{jk}^s 可以看作从content space到observation space特定于style的映射。

非对称模型

$$y_k^{sc} = \sum_{i,j} w_{ijk}^s a_i^s b_j^c$$

让 $\mathbf{a}_j^s$ 表示含有分量 $\{a_{jk}^s\}$ 的k维向量，则式子可以写为：

$$\mathbf{y}^{\mathbf{sc}} = \sum_j \mathbf{a}_j^{\mathbf{s}} b_j^c$$

可以认为 a_{jk}^s 是一组特定于style的基向量，这些基向量和特定于content的系数 b_j^c 混合在一起后产生observation向量。

Second-order Pooling

文章提出了二阶池化的方法

$$\mathbf{G}_{avg}\left(R_j\right) = \frac{1}{\left|F_{R_j}\right|} \sum_{i:(\mathbf{f}_i \in R_j)} \mathbf{x}_i \cdot \mathbf{x}_i^\top$$

$$\mathbf{G}_{\max}\left(R_j\right) = \max_{i:(\mathbf{f}_i \in R_j)} \mathbf{x}_i \cdot \mathbf{x}_i^\top$$

文章发现用额外的局部描述信息可以获得很大的性能收益。

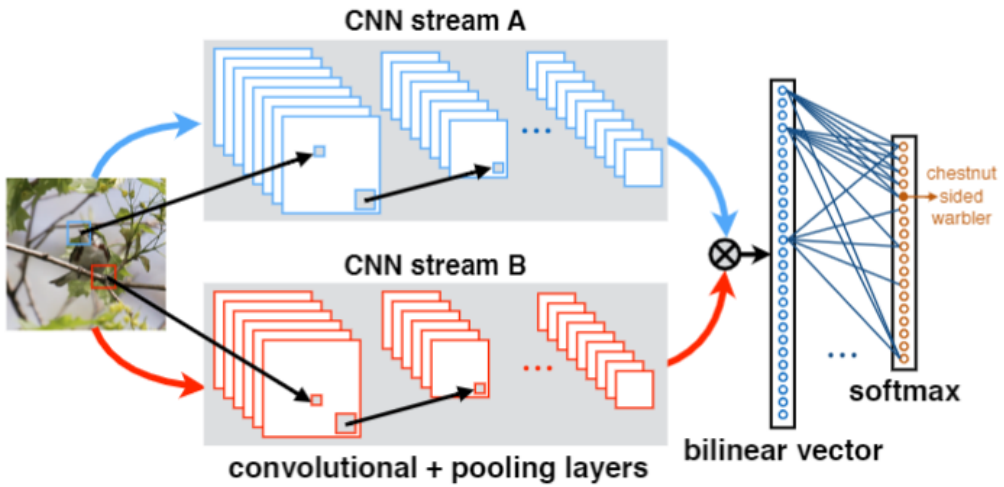
并且与2011年挑战赛的获胜者相比较，由于使用了线性模型而不是基于非线性内核的模型，因此在训练和测试方面速度明显更快

	O_2P	BERKELEY	BONN-FGT	BONN-SVR	BROOKES	NUS-C	NUS-S
background	**85.4**	83.4	83.4	84.9	79.4	77.2	79.8
aeroplane	**69.7**	46.8	51.7	54.3	36.6	40.5	41.5
bicycle	22.3	18.9	23.7	**23.9**	18.6	19.0	20.2
bird	45.2	36.6	**46.0**	39.5	9.2	28.4	30.4
boat	**44.4**	31.2	33.9	35.3	11.0	27.8	29.1
bottle	46.9	42.7	**49.4**	42.6	29.8	40.7	47.4
bus	**66.7**	57.3	66.2	65.4	59.0	56.4	61.2
car	**57.8**	47.4	56.2	53.5	50.3	45.0	47.7
cat	**56.2**	44.1	41.7	46.1	25.5	33.1	35.0
chair	13.5	8.1	10.4	**15.0**	11.8	7.2	8.5
cow	46.1	39.4	41.9	**47.4**	29.0	37.4	38.3
diningtable	32.3	**36.1**	29.6	30.1	24.8	17.4	14.5
dog	**41.2**	36.3	24.4	33.9	16.0	26.8	28.6
horse	**59.1**	49.5	49.1	48.8	29.1	33.7	36.5
motorbike	**55.3**	48.3	50.5	54.4	47.9	46.6	47.8
person	**51.0**	50.7	39.6	46.4	41.9	40.6	42.5
pottedplant	**36.2**	26.3	19.9	28.8	16.1	23.3	28.5
sheep	50.4	47.2	44.9	**51.3**	34.0	33.4	37.8
sofa	**27.8**	22.1	26.1	26.2	11.6	23.9	26.4
train	**46.9**	42.0	40.0	44.9	43.3	41.2	43.5
tv/monitor	44.6	43.2	41.6	37.2	31.7	38.6	**45.8**
Mean	**47.6**	40.8	41.4	43.3	31.3	35.1	37.7

Carreira, Joao, et al. "Semantic segmentation with second-order pooling." European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2012.

BIlinear CNN

双线性映射 $f: V \times U \rightarrow W$ 中，V 和 U 可以用来表示不同的信息，文章想通过two-stream结构来分别学习到位置（location）信息和图像（image）信息。



$$\text{bilinear}\ (l, I, f_A, f_B) = f_A(l, I)^T f_B(l, I)$$

$$f_A(l, I) \in \mathbb{R}^{K \times D_A}$$

$$f_B(l, I) \in \mathbb{R}^{K \times D_B}$$

$$f_A(l, I)^T f_B(l, I) \in \mathbb{R}^{D_A \times D_B}$$

$$\Phi(I) = \sum_{l \in \mathcal{L}} \text{bilinear}\ (l, I, f_A, f_B) = \sum_{l \in \mathcal{L}} f_A(l, I)^T f_B(l, I)$$

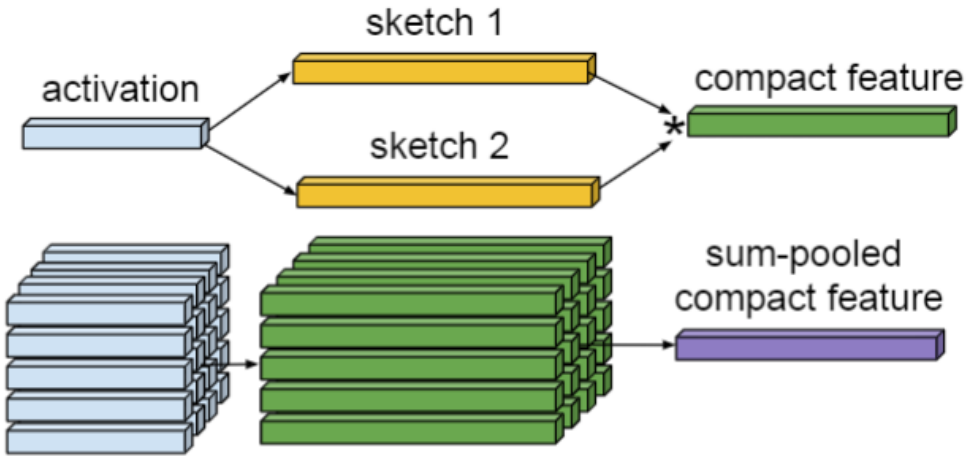
Lin, Tsung-Yu, Aruni RoyChowdhury, and Subhransu Maji. "Bilinear cnn models for fine-grained visual recognition." Proceedings of the IEEE international conference on computer vision. 2015.

Compact Bilinear Pooling

$$B(\mathcal{X}) = \sum_{s \in \mathcal{S}} x_s x_s^T$$

Bilinear的运算使得特征的维度变成了原来的平方，这增加后续的计算量，这篇文章提出一种紧凑的双线性池化。

原始特征经过双线性池化后用于后的分类预测，如果从核方法的角度来，那么双线性可以写成如下形 式：



$$\begin{aligned}\langle B(\mathcal{X}), B(\mathcal{Y}) \rangle &= \left\langle \sum_{s \in \mathcal{S}} x_s x_s^T, \sum_{u \in \mathcal{U}} y_u y_u^T \right\rangle \\ &= \sum_{s \in \mathcal{S}} \sum_{u \in \mathcal{U}} \left\langle x_s x_s^T, y_u y_u^T \right\rangle \\ &= \sum_{s \in \mathcal{S}} \sum_{u \in \mathcal{U}} \langle x_s, y_u \rangle^2\end{aligned}$$

如果能够找到一个低维映射函数 $\phi(x) \in R^d$， 其中 $d << c^2$，且满足 $\langle \phi(x), \phi(y) \rangle \approx k(x, y) = \langle x_s, y_u \rangle^2$，那么上式就可以写成：

$$\begin{aligned}\langle B(\mathcal{X}), B(\mathcal{Y}) \rangle &= \sum_{s \in \mathcal{S}} \sum_{u \in \mathcal{U}} \langle x_s, y_u \rangle^2 \\ &\approx \sum_{s \in \mathcal{S}} \sum_{u \in \mathcal{U}} \langle \phi(x), \phi(y) \rangle \\ &\equiv \langle C(\mathcal{X}), C(\mathcal{Y}) \rangle\end{aligned}$$

Gao, Yang, et al. "Compact bilinear pooling." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

Compact Bilinear Pooling

Algorithm 1 Random Maclaurin Projection

Input: $x \in \mathbb{R}^c$

Output: feature map $\phi_{RM}(x) \in \mathbb{R}^d$, such that $\langle \phi_{RM}(x), \phi_{RM}(y) \rangle \approx \langle x, y \rangle^2$

1. Generate random but fixed $W_1, W_2 \in \mathbb{R}^{d \times c}$, where each entry is either $+1$ or -1 with equal probability.
2. Let $\phi_{RM}(x) \equiv \frac{1}{\sqrt{d}}(W_1 x) \circ (W_2 x)$, where $\circ$ denotes element-wise multiplication.

Algorithm 2 Tensor Sketch Projection

Input: $x \in \mathbb{R}^c$

Output: feature map $\phi_{TS}(x) \in \mathbb{R}^d$, such that $\langle \phi_{TS}(x), \phi_{TS}(y) \rangle \approx \langle x, y \rangle^2$

1. Generate random but fixed $h_k \in \mathbb{N}^c$ and $s_k \in \{+1, -1\}^c$ where $h_k(i)$ is uniformly drawn from $\{1, 2, \ldots, d\}$, $s_k(i)$ is uniformly drawn from $\{+1, -1\}$, and $k = 1, 2$.
2. Next, define sketch function $\Psi(x, h, s) = \{(Qx)_1, \ldots, (Qx)_d\}$, where $(Qx)_j = \sum_{t:h(t)=j} s(t) x_t$

3. Finally, define $\phi_{TS}(x) \equiv \text{FFT}^{-1}(\text{FFT}(\Psi(x, h_1, s_1)) \circ \text{FFT}(\Psi(x, h_2, s_2)))$, where the $\circ$ denotes element-wise multiplication.

$$E[\langle \Psi(x, h, s), \Psi(y, h, s) \rangle] = \langle x, y \rangle, \text{ 除此之外 } \Psi(x \otimes y, h, s) = \Psi(x, h, s) * \Psi(y, h, s)$$

Gao, Yang, et al. "Compact bilinear pooling." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

Low-rank Bilinear Pooling

前面的双线性融合方法，$\mathbf{X} \in \mathbb{R}^{c \times hw}$ 经过双线性池化得到双线性特征后，展开成向量 $\mathbf{z} = \mathrm{vec}\left(\mathbf{X}\mathbf{X}^T\right) \in \mathbb{R}^{c^2}$，使用线性分类器做最后的预测。假设是一个用 $\mathbf{w} \in \mathbb{R}^{c^2}$ 和 b 参数化的线性分类器，标准的soft margin SVM目标函数为：

$$\min_{\mathbf{w},b} \frac{1}{N} \sum_{i=1}^{N} \max\left(0, 1 - y_i \mathbf{w}^T \mathbf{z}_i + b\right) + \frac{\lambda}{2} \|\mathbf{w}\|_2^2$$

上式写成矩阵的形式可以表示为：

$$\min_{\mathbf{W},b} \frac{1}{N} \sum_{i=1}^{N} \max\left(0, 1 - y_i \,\mathrm{tr}\left(\mathbf{W}^T \mathbf{X}_i \mathbf{X}_i^T\right) + b\right) + \frac{\lambda}{2} \|\mathbf{W}\|_F^2$$

利用拉格朗日函数对参数求偏导，可以得到上述两个式子的最优解为：

$$\mathbf{w}^* = \sum_{y_i=1} \alpha_i \mathbf{z}_i - \sum_{y_i=-1} \alpha_i \mathbf{z}_i$$

$$\mathbf{W}^* = \sum_{y_i=1} \alpha_i \mathbf{X}_i \mathbf{X}_i^T - \sum_{y_i=-1} \alpha_i \mathbf{X}_i \mathbf{X}_i^T$$

$$\text{where } \alpha_i \geq 0, \forall i = 1, \ldots, N$$

Kong, Shu, and Charless Fowlkes. "Low-rank bilinear pooling for fine-grained classification." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Low-rank Bilinear Pooling

因为 $\mathbf{z} = \mathrm{vec}\left(\mathbf{X}\mathbf{X}^T\right) \in \mathbb{R}^{c^2}$，所以$\mathbf{w}^* = \mathrm{vec}(\mathbf{W}^*)$。 $\mathbf{W}^*$ 是对称矩阵的求和，因此 $\mathbf{W}^*$ 也是对称矩阵。从上式看出 $\mathbf{W}^*$ 是正样本和负样本分别对应的矩阵的差值，所以可以做以下特征值分解：

$$\begin{aligned}\mathbf{W}^* &= \mathbf{\Psi}\mathbf{\Sigma}\mathbf{\Psi}^T = \mathbf{\Psi}_+\mathbf{\Sigma}_+\mathbf{\Psi}_+^T + \mathbf{\Psi}_-\mathbf{\Sigma}_-\mathbf{\Psi}_-^T \\ &= \mathbf{\Psi}_+\mathbf{\Sigma}_+\mathbf{\Psi}_+^T - \mathbf{\Psi}_-\left|\mathbf{\Sigma}_-\right|\mathbf{\Psi}_-^T \\ &= \mathbf{U}_+\mathbf{U}_+^T - \mathbf{U}_-\mathbf{U}_-^T\end{aligned}$$

其中 $\mathbf{\Sigma}_+$ 和 $\mathbf{\Sigma}_-$ 分别是正值和负值的特征值， $\mathbf{\Psi}_+$ 和 $\mathbf{\Psi}_-$ 为对应的特征向量。 第三行的 $\mathbf{U}_+ = \mathbf{\Psi}_+\mathbf{\Sigma}_+^{\frac{1}{2}}$ 以及 $\mathbf{U}_- = \mathbf{\Psi}_-\left|\mathbf{\Sigma}_-\right|^{\frac{1}{2}}$。因此 $\mathbf{W}^*$ 可能会有好的低秩的分解，即 $\mathbf{U}_-$ 和 $\mathbf{U}_+$ 是低秩的。文章直接施加了一个强硬的低秩约束 $\mathrm{rank}(\mathbf{W}) = r \ll c$，具体的方法是使用 $\mathbf{U}_+, \mathbf{U}_- \in \mathbb{R}^{c\times r/2}$ 来近似表示 $\mathbf{W}$：

$$\begin{aligned}y &= \mathrm{tr}\left(\boldsymbol{W}\boldsymbol{X}\boldsymbol{X}^T\right) + b \\ &= \mathrm{tr}\left(\boldsymbol{U}_+\boldsymbol{U}_+^T - \boldsymbol{U}_-\boldsymbol{U}_-^T\right)\boldsymbol{X}\boldsymbol{X}^T + b \\ &= \mathrm{tr}\left(\boldsymbol{U}_+\boldsymbol{U}_+^T\boldsymbol{X}\boldsymbol{X}^T\right) - \mathrm{tr}\left(\boldsymbol{U}_-\boldsymbol{U}_-^T\boldsymbol{X}\boldsymbol{X}^T\right) + b \\ &= \mathrm{tr}\left(\left(\boldsymbol{U}_+^T\boldsymbol{X}\right)^T\left(\boldsymbol{U}_+^T\boldsymbol{X}\right)\right) - \mathrm{tr}\left(\left(\boldsymbol{U}_-^T\boldsymbol{X}\right)^T\left(\boldsymbol{U}_-^T\boldsymbol{X}\right)\right) + b \\ &= \left\|\boldsymbol{U}_+^T\boldsymbol{X}\right\|_{\mathrm{F}}^2 - \left\|\boldsymbol{U}_-^T\boldsymbol{X}\right\|_{\mathrm{F}}^2 + b\end{aligned}$$

因此不需要计算$\mathbf{X}\mathbf{X}^T$ ，而降低了计算量。

Kong, Shu, and Charless Fowlkes. "Low-rank bilinear pooling for fine-grained classification." Proceedings of the IEEE conference on computer vision and pattern recognition. 2017.

Factorized Bilinear Model

根据前面介绍的双线性池化，双线性池化结合全连接层的模型可以写为：

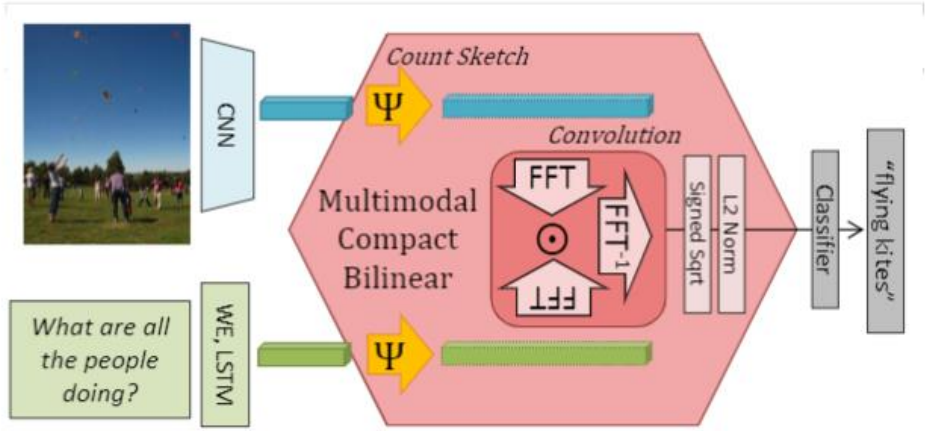
$$
\begin{aligned}
y_j &= b_j + \mathbf{W}_{j\cdot}^T \operatorname{vec}\left(\sum_{i\in\mathbb{S}} \mathbf{x}_i \mathbf{x}_i^T\right) \\
&= b_j + \sum_{i\in\mathbb{S}} \mathbf{x}_i^T \mathbf{W}_{j\cdot}^R \mathbf{x}_i
\end{aligned}
$$

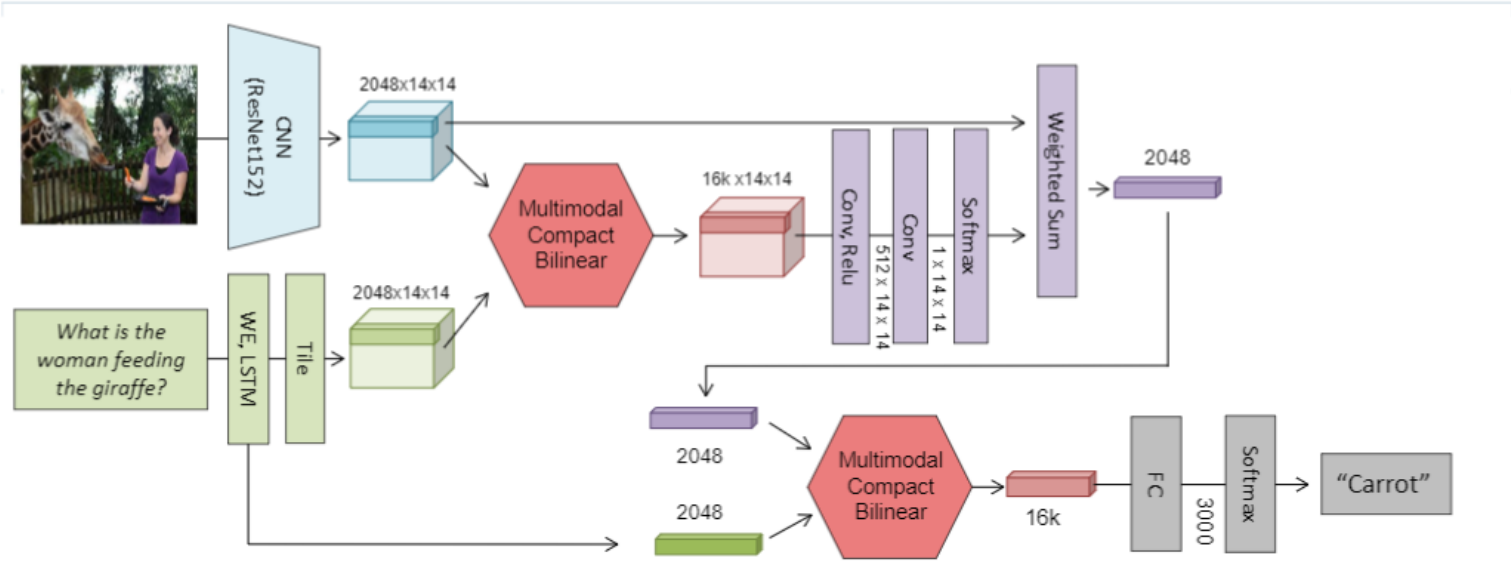
其中 $\mathbf{W}_{j\cdot}$ 是 $\mathbf{W}$ 的第 j 行，$\mathbf{W}_{j\cdot}^R \in \mathbb{R}^{n\times n}$ 是 $\mathbf{W}_{j\cdot}$ reshape 之后的矩阵，y_j 和 b_j 分别是 $\mathbf{y}$ 和 $\mathbf{b}$ 第 j 个值。因此根据这个形式提出了结合一阶和二阶特征，并且使用低秩矩阵 $\mathbf{F}$ 来代替 $\mathbf{W}$ 简化计算的模型：

$$
y = b + \mathbf{w}^T\mathbf{x} + \mathbf{x}^T\mathbf{F}^T\mathbf{F}\mathbf{x}
$$

Li, Yanghao, et al. "Factorized bilinear models for image recognition." Proceedings of the IEEE International Conference on Computer Vision. 2017.

Multimodal Compact Bilinear Pooling





Fukui, Akira, et al. "Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding." Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. 2016.

Multimodal Low-rank Bilinear Pooling

$$f_i = \sum_{j=1}^{N}\sum_{k=1}^{M} w_{ijk} x_j y_k + b_i = \mathbf{x}^T \mathbf{W}_i \mathbf{y} + b$$

还是一样的双线性模型，$\mathbf{W}_i \in \mathbb{R}^{N\times M}$ 是 f_i 对应的参数，共有 L 个，那么参数一共有 $L \times (N \times M + 1)$ 个，引入前面提到的低秩分解，降低计算量：

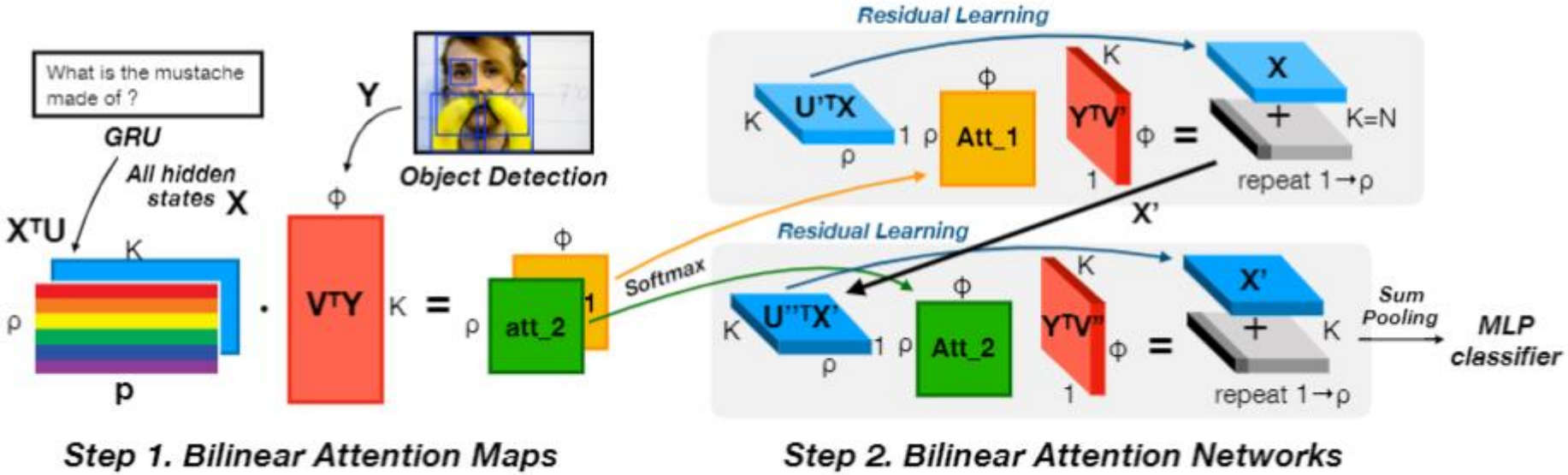
$$f_i = \mathbf{x}^T \mathbf{W}_i \mathbf{y} + b_i = \mathbf{x}^T \mathbf{U}_i \mathbf{V}_i^T \mathbf{y} + b_i = 1^T \left(\mathbf{U}_i^T \mathbf{x} \circ \mathbf{V}_i^T \mathbf{y}\right) + b_i$$

$1 \in \mathbb{R}^d$ 表示为1的列向量，$\circ$ 表示哈达玛积，但仍然需要两个三阶张量 $\mathbf{U}$ 和 $\mathbf{V}$，为了减少参数张量的阶数，用 $\mathbb{P} \in \mathbb{R}^{d\times c}$ 来代替 1：

$$\mathbf{f} = \mathbf{P}^T \left(\mathbf{U}^T \mathbf{x} \circ \mathbf{V}^T \mathbf{y}\right) + \mathbf{b}$$

Kim, Jin-Hwa, et al. "Hadamard product for low-rank bilinear pooling." arXiv preprint arXiv:1610.04325 (2016).

Multimodal Compact Bilinear Pooling



Step 1. Bilinear Attention Maps

Step 2. Bilinear Attention Networks

$$A_{i,j} = \mathbf{p}^T \left(\left(\mathbf{U}^T \mathbf{X}_i \right) \circ \left(\mathbf{V}^T \mathbf{Y}_j \right) \right)$$

$$\mathbf{f}'_k = \left(\mathbf{X}^T \mathbf{U}' \right)_k^T \mathcal{A} \left(\mathbf{Y}^T \mathbf{V}' \right)_k$$

Kim, Jin-Hwa, Jaehyun Jun, and Byoung-Tak Zhang. "Bilinear attention networks." Advances in Neural Information Processing Systems. 2018.