1. Introduction

- **Problem:** Traditional attention mechanisms in image captioning models usually assume one-to-one mapping from image regions to caption words, which is impossible.
  - visual information forcibly given to non-visual words;
  - hard to understand interactions between objects;
  - earlier decoding steps have less knowledge of the image.

- **Solution:** We propose Adaptive Attention Time (AAT), which allows adaptive alignment for image captioning and can address all these issues.
  - an image region can be mapped to an arbitrary number of caption words (from zero to multiple); and vice versa.

With AAT,
  - an image region can be mapped to an arbitrary number of caption words (from zero to multiple); and vice versa.

2. Method

- **Confidence network:** measures how confident the decoder currently is to output a word
  \[ p_{n} = \begin{cases} \sigma(\max(h_{n}^i, W_{1} + b_{1} + b_{1})) & n = 0 \\ \sigma(\max(h_{n}^i, W_{1} + b_{1} + b_{2} + b_{2})) & n > 0 \end{cases} \]

- **Attention time (steps):** determined by the confidence
  \[ N(t) = \min(M_{max}, \min\{n': \prod_{i=0}^{n-1}(1-p_{i}^{n}) < c\}) \]

- **Final output:** average of all attention steps
  \[ \begin{align*}
  h_{n}^i &= \beta_{n}h_{n}^i + \sum_{j=1}^{N(N(t))} \beta_{j}h_{j}^i, \\
  m_{n}^i &= \beta_{n}m_{n}^i + \sum_{j=1}^{N(N(t))} \beta_{j}m_{j}^i, \\
  \beta_{n} &= \begin{cases} 1 & n = 0 \\ \prod_{i=0}^{n-1}(1-p_{i}^{n}) & n > 0 \end{cases}
  \end{align*} \]

- **Time cost penalty:** encourages to take fewer attention steps
  \[ L_{t} = \ell(N(t)) \sum_{n=1}^{N(N(t))} (1-p_{i}^{n}) \]

3. Results

**Table 1: Attention Model (Time)**

<table>
<thead>
<tr>
<th>Att Model</th>
<th>Time</th>
<th>Cider-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base</td>
<td>1.23</td>
<td>123.76</td>
</tr>
<tr>
<td>Recurrent</td>
<td>2.12</td>
<td>124.22</td>
</tr>
<tr>
<td>Adaptative (0-4)</td>
<td>2.55</td>
<td>126.48</td>
</tr>
</tbody>
</table>

**Table 2: Time Cost Penalty**

<table>
<thead>
<tr>
<th>Att Type</th>
<th>Head(s)</th>
<th>Time</th>
<th>Cider-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addictive</td>
<td>1</td>
<td>2.54</td>
<td>126.76</td>
</tr>
<tr>
<td>Recurrent</td>
<td>1-3</td>
<td>1.03</td>
<td>125.76</td>
</tr>
<tr>
<td>Dot-Product</td>
<td>1-3</td>
<td>2.54</td>
<td>126.76</td>
</tr>
<tr>
<td>Adaptive (0-4)</td>
<td>2.55</td>
<td>126.48</td>
<td></td>
</tr>
</tbody>
</table>

**Table 3: Attention Heads**

<table>
<thead>
<tr>
<th>Att Type</th>
<th>Head(s)</th>
<th>Time</th>
<th>Cider-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Addictive</td>
<td>1</td>
<td>2.54</td>
<td>126.76</td>
</tr>
<tr>
<td>Recurrent</td>
<td>1-3</td>
<td>1.03</td>
<td>125.76</td>
</tr>
<tr>
<td>Dot-Product</td>
<td>1-3</td>
<td>2.54</td>
<td>126.76</td>
</tr>
<tr>
<td>Adaptive (0-4)</td>
<td>2.55</td>
<td>126.48</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 2: Attention time at different decoding steps**

AAT demonstrates the best performance while requires relatively little attention time.

**Figure 3: Comparisons with state-of-the-arts**

- **Additive**
  - BLEU: 20.09
  - ROUGE: 23.52
  - METEOR: 22.56
  - CIDEr: 22.24
- **GRU**
  - BLEU: 17.84
  - ROUGE: 22.52
  - METEOR: 21.76
  - CIDEr: 21.88
- **Attention Based (Anderson et al. 2018)**
  - BLEU: 17.59
  - ROUGE: 22.34
  - METEOR: 21.26
  - CIDEr: 21.22
- **Additive**
  - BLEU: 21.50
  - ROUGE: 24.32
  - METEOR: 23.19
  - CIDEr: 23.13

Code available at: https://github.com/husthuaan/AAT