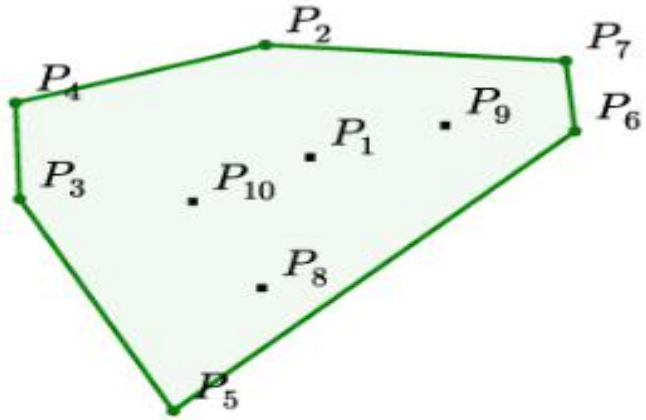
# Pointer Networks &CopyNet &Pointer-Generator networks

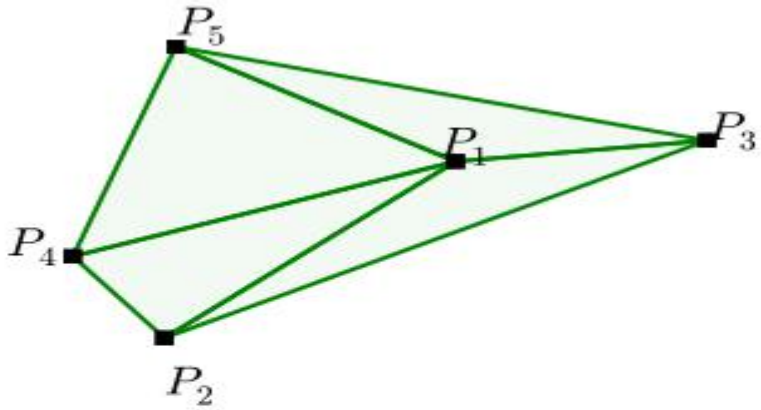
2022.4.28

# Pointer Networks-problems

- problem1： The number of target classes in each step of the output depends on the length of the input, which is variable.
- problem2： OOV （out-of-vocabulary）




(a) Input $\mathcal{P} = \{P_1, \ldots, P_{10}\}$, and the output sequence $\mathcal{C}^{\mathcal{P}} = \{\Rightarrow, 2, 4, 3, 5, 6, 7, 2, \Leftarrow\}$ representing its convex hull.

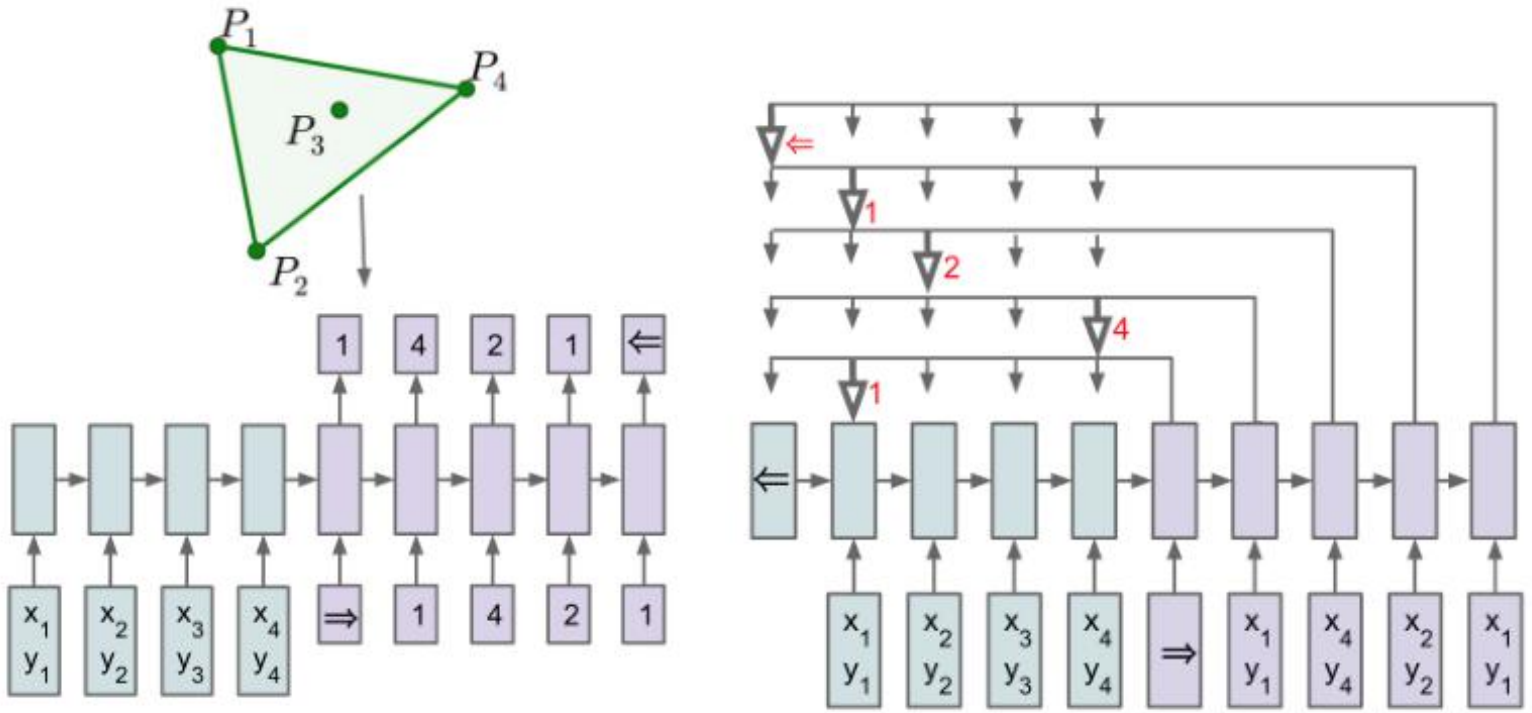



(b) Input $\mathcal{P} = \{P_1, \ldots, P_5\}$, and the output $\mathcal{C}^{\mathcal{P}} = \{\Rightarrow, (1, 2, 4), (1, 4, 5), (1, 3, 5), (1, 2, 3), \Leftarrow\}$ representing its Delaunay Triangulation.

# Pointer Networks-model

$$p(\mathcal{C}^{\mathcal{P}}|\mathcal{P};\theta) = \prod_{i=1}^{m(\mathcal{P})} p_\theta(C_i|C_1,\ldots,C_{i-1},\mathcal{P};\theta).$$

$$\theta^* = \arg\max_\theta \sum_{\mathcal{P},\mathcal{C}^{\mathcal{P}}} \log p(\mathcal{C}^{\mathcal{P}}|\mathcal{P};\theta),$$




(a) Sequence-to-Sequence

(b) Ptr-Net

$$\begin{aligned} u_j^i &= v^T \tanh(W_1 e_j + W_2 d_i) \quad j \in (1,\ldots,n) \\ a_j^i &= \text{softmax}(u_j^i) \quad j \in (1,\ldots,n) \\ d_i' &= \sum_{j=1}^{n} a_j^i e_j \end{aligned}$$

$$\begin{aligned} u_j^i &= v^T \tanh(W_1 e_j + W_2 d_i) \quad j \in (1,\ldots,n) \\ p(C_i|C_1,\ldots,C_{i-1},\mathcal{P}) &= \text{softmax}(u^i) \end{aligned}$$

# Pointer Networks-code

```python
def attention(self,ref, query, with_softmax, scope="attention"):
    """

    :param ref: encoder的输出
    :param query: decoder的输出
    :param with_softmax:
    :param scope:
    :return:
    """
    with tf.variable_scope(scope):
        W_1 = tf.get_variable("W_e", [self.hidden_dim, self.attention_dim], initializer=self.initializer)  # L x A
        W_2 = tf.get_variable("W_d", [self.hidden_dim, self.attention_dim], initializer=self.initializer) # L * A

        dec_portion = tf.matmul(query, W_2)

        scores = [] # S * B
        v_blend = tf.get_variable("v_blend", [self.attention_dim, 1], initializer=self.initializer)  # A x 1
        bais_blend = tf.get_variable("bais_v_blend", [1], initializer=self.initializer)  # 1 x 1
        for i in range(self.max_enc_length + 1):
            refi = tf.matmul(tf.squeeze(ref[:,i,:]),W_1)
            ui = tf.add(tf.matmul(tf.nn.tanh(dec_portion+refi),v_blend),bais_blend) # B * 1
            scores.append(tf.squeeze(ui))
        scores = tf.transpose(scores,[1,0]) # B * S
        if with_softmax:
            return tf.nn.softmax(scores,dim=1)
        else:
            return scores
```

# Pointer Networks-Discussion

Sequence-to-Sequence Model

Attention Mechanism

variable size output dictionaries

# Incorporating Copying Mechanism in Sequence-to-Sequence Learning

CopyNet

# CopyNet-problems

I: Hello Jack, my name is Chandralekha.

R: Nice to meet you, Chandralekha.

---

I: This new guy doesn't perform exactly as we expected.

R: What do you mean by "doesn't perform exactly as we expected"?

problem1: the rote memorization; copying mechanism

problem2: out-of-vocabulary (OOV)

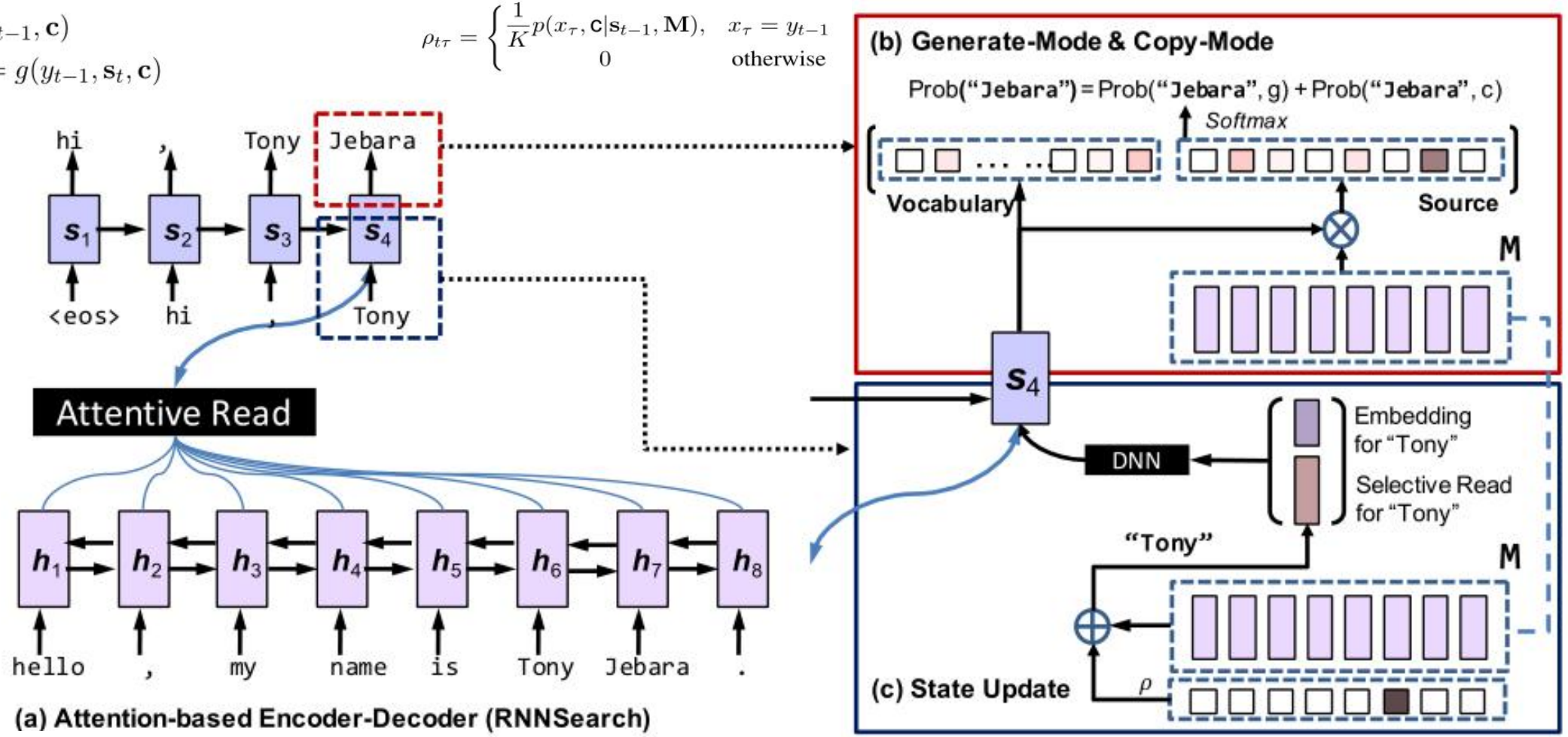
# CopyNet-Model

State Update

$y_{t-1}$ will be represented as $[\mathbf{e}(y_{t-1}); \zeta(y_{t-1})]^\top$

$$\mathbf{h}_t = f(x_t, \mathbf{h}_{t-1}); \quad \mathbf{c} = \phi(\{\mathbf{h}_1, ..., \mathbf{h}_{T_S}\})$$
$$\mathbf{s}_t = f(y_{t-1}, \mathbf{s}_{t-1}, \mathbf{c})$$
$$p(y_t|y_{<t}, X) = g(y_{t-1}, \mathbf{s}_t, \mathbf{c})$$

$$\zeta(y_{t-1}) = \sum_{\tau=1}^{T_S} \rho_{t\tau} \mathbf{h}_\tau$$
$$\rho_{t\tau} = \begin{cases} \frac{1}{K} p(x_\tau, \mathbf{c}|\mathbf{s}_{t-1}, \mathbf{M}), & x_\tau = y_{t-1} \\ 0 & \text{otherwise} \end{cases}$$

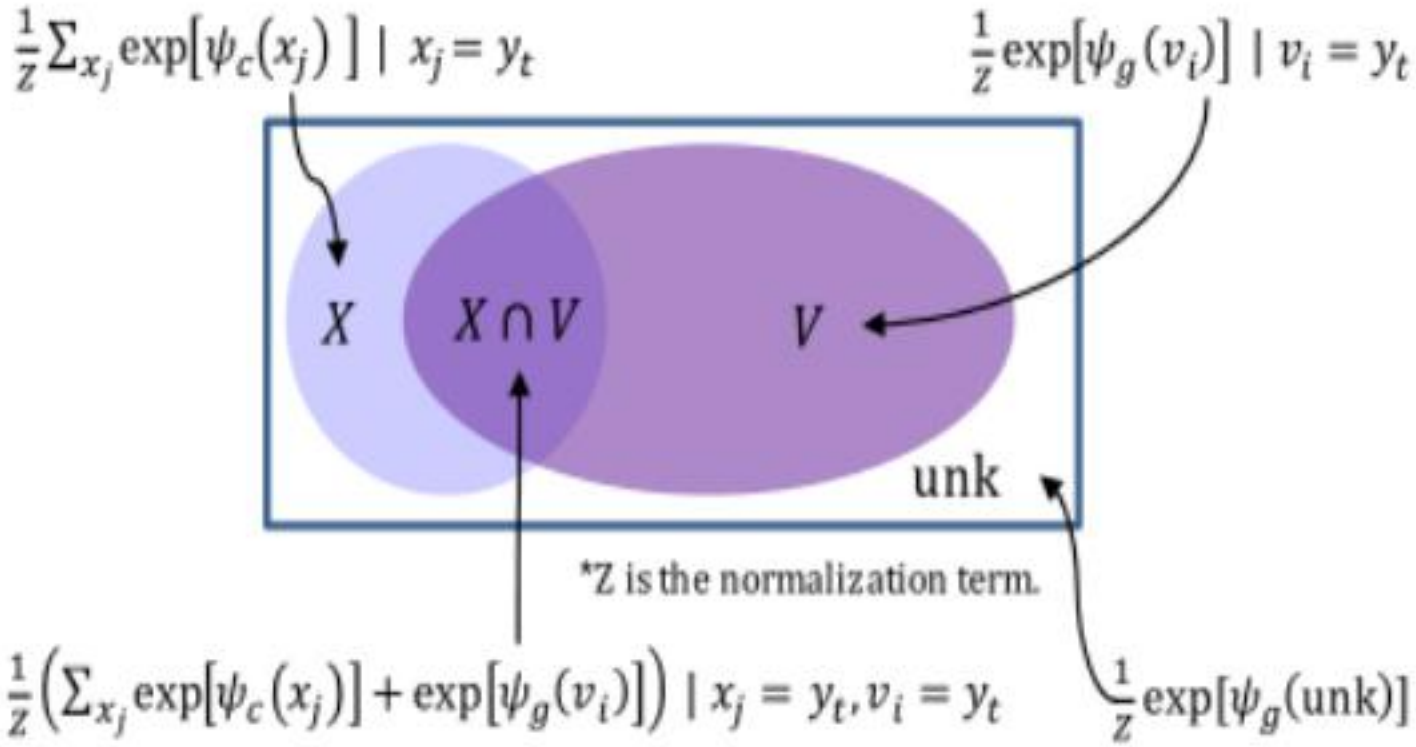



$$\mathbf{c}_t = \sum_{\tau=1}^{T_S} \alpha_{t\tau} \mathbf{h}_\tau; \quad \alpha_{t\tau} = \frac{e^{\eta(\mathbf{s}_{t-1}, \mathbf{h}_\tau)}}{\sum_{\tau'} e^{\eta(s_{t-1}, \mathbf{h}_{\tau'})}}$$

M

$$\zeta(y_{t-1}) \xrightarrow{\text{update}} \mathbf{s}_t \xrightarrow{\text{predict}} y_t \xrightarrow{\text{sel. read}} \zeta(y_t)$$

# CopyNet-Model




$$p(y_t, \mathsf{g}|\cdot) = \begin{cases} \frac{1}{Z}e^{\psi_g(y_t)}, & y_t \in \mathcal{V} \\ 0, & y_t \in \mathcal{X} \cap \bar{V} \\ \frac{1}{Z}e^{\psi_g(\text{UNK})} & y_t \notin \mathcal{V} \cup \mathcal{X} \end{cases}$$

$$p(y_t, \mathsf{c}|\cdot) = \begin{cases} \frac{1}{Z}\sum_{j:x_j=y_t} e^{\psi_c(x_j)}, & y_t \in \mathcal{X} \\ 0 & \text{otherwise} \end{cases}$$

$$\psi_g(y_t = v_i) = \mathbf{v}_i^\top \mathbf{W}_o \mathbf{s}_t, \quad v_i \in \mathcal{V} \cup \text{UNK}$$

$$\psi_c(y_t = x_j) = \sigma\left(\mathbf{h}_j^\top \mathbf{W}_c\right)\mathbf{s}_t, \quad x_j \in \mathcal{X}$$

$$\mathcal{L} = -\frac{1}{N}\sum_{k=1}^{N}\sum_{t=1}^{T}\log\left[p(y_t^{(k)}|y_{<t}^{(k)}, X^{(k)})\right]$$

# CopyNet-Code

```
# Copy mechanism
transformed_hidden2 = self.copy_W(output).view(batch_size, self.hidden_size, 1)
copy_score_seq = torch.bmm(encoder_outputs, transformed_hidden2)  # this is linear. add activation function before multiplying.
copy_scores = torch.bmm(torch.transpose(copy_score_seq, 1, 2), one_hot_input_seq).squeeze(1)  # [b, vocab_size + seq_length]
missing_token_mask = (one_hot_input_seq.sum(dim=1) == 0)  # tokens not present in the input sequence
missing_token_mask[:, 0] = 1  # <MSK> tokens are not part of any sequence
copy_scores = copy_scores.masked_fill(missing_token_mask, -1000000.0)

# Generate mechanism
gen_scores = self.out(output.squeeze(1))  # [b, vocab_size]
gen_scores[:, 0] = -1000000.0  # penalize <MSK> tokens in generate mode too
```

# CopyNet-Experiments-Synthetic Dataset

| Rule-type | Examples (e.g. **x** = i h k, **y** = j c) |
|---|---|
| **x** → ∅ | a b c d **x** e f → c d g |
| **x** → **x** | a b c d **x** e f → c d **x** g |
| **x** → **x** **x** | a b c d **x** e f → **x** d **x** g |
| **x** **y** → **x** | a b **y** d **x** e f → **x** d i g |
| **x** **y** → **x** **y** | a b **y** d **x** e f → **x** d **y** g |

| Rule-type | **x** → ∅ | **x** → **x** | **x** → **x** **x** | **x** **y** → **x** | **x** **y** → **x** **y** |
|---|---|---|---|---|---|
| Enc-Dec | **100** | 3.3 | 1.5 | 2.9 | 0.0 |
| RNNSearch | 99.0 | 69.4 | 22.3 | 40.7 | 2.6 |
| COPYNET | 97.3 | **93.7** | **98.3** | **68.2** | **77.5** |

# CopyNet-Experiments-Text Summarization

**Input(1):** 今天上午 9 点半， 复旦 投毒案 将 在 上海 二中院 公开审理 。 被害 学生 黄洋 的 亲属 已 从 四川 抵达 上海 ， 其父 称待 刑事 部分结束 后 ， 再 提 民事 赔偿 ， 黄洋 92 岁 的 奶奶 依然 不知情 。 今年 4 月 ， 在 复旦 上海医学院 读 研究生 的 黄洋 疑遭 室友 林森 浩 投毒 ， 不幸身亡 。 新民 网

Today 9:30, the Fudan poisoning case will be will on public trial at the Shanghai Second Intermediate Court. The relatives of the murdered student Huang Yang has arrived at Shanghai from Sichuan. His father said that they will start the lawsuit for civil compensation after the criminal section. HuangYang 92-year-old grandmother is still unaware of his death. In April, a graduate student at Fudan University Shanghai Medical College, Huang Yang is allegedly poisoned and killed by his roommate Lin Senhao. Reported by Xinmin

**Golden:** 林森 浩 投毒案 今日 开审 92 岁 奶奶 尚不知 情 ( the case of Lin Senhao poisoning is on trial today, his 92-year-old grandmother is still unaware of this)

**RNN context:** 复旦投毒案： 黄洋疑遭室友投毒凶手已从四川飞往上海，父亲命案另有4人被通知家属不治？

**CopyNet:** 复旦 投毒案 今 在 沪 上 公开审理 ( the Fudan poisoning case is on public trial today in Shanghai )

**Input(2):** 华谊 兄弟 （ 300027 ） 在 昨日 收盘 后 发布公告 称， 公司 拟 以 自有 资金 3.978亿元 收购 浙江 永乐 影视 股份 有限公司 若干股东 持有 的 永乐 影视 51 % 的 股权 。 对于 此项 收购 ， 华谊 兄弟 董秘 胡明 昨日 表示 ： “ 和 永乐 影视 的 合并 是 对 华谊 兄弟 电视剧 业务 的 一个 加强 。

Huayi Brothers (300027) announced that the company intends to buy with its own fund 397.8 million 51% of Zhejiang Yongle Film LTD's stake owned by a number of shareholders of Yongle Film LTD. For this acquisition, the secretary of the board, Hu Ming, said yesterday: "the merging with Yongle Film is to strengthen Huayi Brothers on TV business".

**Golden:** 华谊 兄弟 拟 收购 永乐 影视 5 1 % 股权 ( Huayi Brothers intends to acquire 51% stake of Zhejiang Yongle Film )

**RNN context:** 华谊兄弟收购永乐影视51%股权：与永乐影视合并为“和唐”影视合并的“UNK”和“UNK”的区别？

**CopyNet:** 华谊 兄弟 拟 3.978 亿 收购 永乐 影视 董秘 称 加强 电视剧 业务 ( Huayi Brothers is intended to 397.8 million acquisition of Yongle Film secretaries called to strengthen the TV business )

**Input(3):** 工厂 ， 大门 紧锁 ， 约 20 名 工人 散 坐在 树荫下 。 “ 我们 就是 普通工人 ， 在 这里 等 工资 。 ” 其中 一人 说道 。 7 月 4 日 上午 ， 记者 抵达 深圳 龙华区 清 湖 路上 的 深圳 愿景 光电子 有限公司 。 正如 传言 一般 ， 愿景 光电子 倒闭 了 ， 大 股东 邢毅 不知 所踪 。

The door of factory is locked. About 20 workers are scattered to sit under the shade. “We are ordinary workers, waiting for our salary” one of them said. In the morning of July 4th, reporters arrived at Yuanjing Photoelectron Corporation located at Qinghu Road, Longhua District, Shenzhen. Just as the rumor, Yuanjing Photoelectron Corporation is closed down and the big shareholder Xing Yi is missing.

**Golden:** 深圳 亿元 级 LED 企业倒闭 烈日 下 工人 苦 等 老板 ( Hundred-million CNY worth LED enterprise is closed down and workers wait for the boss under the scorching sun)

**RNN context:** 深圳 “<UNK>”：深圳<UNK><UNK>，<UNK>，<UNK>，<UNK>

**CopyNet:** 愿景 光电子 倒闭 20 名 工人 散 坐在 树荫下 (Yuanjing Photoelectron Corporation is closed down, 20 workers are scattered to sit under the shade )

# CopyNet-Experiments-Single-turn Dialogue

# Copynet--Discussion

Hybrid Addressing of M( Location-based Addressing)

**Copy mechanism**

Handling Out-of-Vocabulary Words

Copying and Generation

State Update

# Get To The Point: Summarization with Pointer-Generator Networks

PGN

# PGN-problems

**Original Text (truncated):** lagos, nigeria (cnn) a day after winning nigeria's presidency, *muhammadu buhari* told cnn's christiane amanpour that **he plans to aggressively fight corruption that has long plagued nigeria** and go after the root of the nation's unrest. *buhari* said he'll "rapidly give attention" to curbing violence in the northeast part of nigeria, where the terrorist group boko haram operates. by cooperating with neighboring nations chad, cameroon and niger, **he said his administration is confident it will be able to thwart criminals** and others contributing to nigeria's instability. for the first time in nigeria's history, the opposition defeated the ruling party in democratic elections. *buhari* defeated incumbent goodluck jonathan by about 2 million votes, according to nigeria's independent national electoral commission. **the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.**

---

**Baseline Seq2Seq + Attention:** **UNK UNK** says his administration is confident it will be able to **destabilize nigeria's economy**. **UNK** says his administration is confident it will be able to thwart criminals and other **nigerians**. **he says the country has long nigeria and nigeria's economy.**

---

**Pointer-Gen:** *muhammadu buhari* says he plans to aggressively fight corruption **in the northeast part of nigeria**. he says he'll "rapidly give attention" to curbing violence **in the northeast part of nigeria**. he says his administration is confident it will be able to thwart criminals.

---

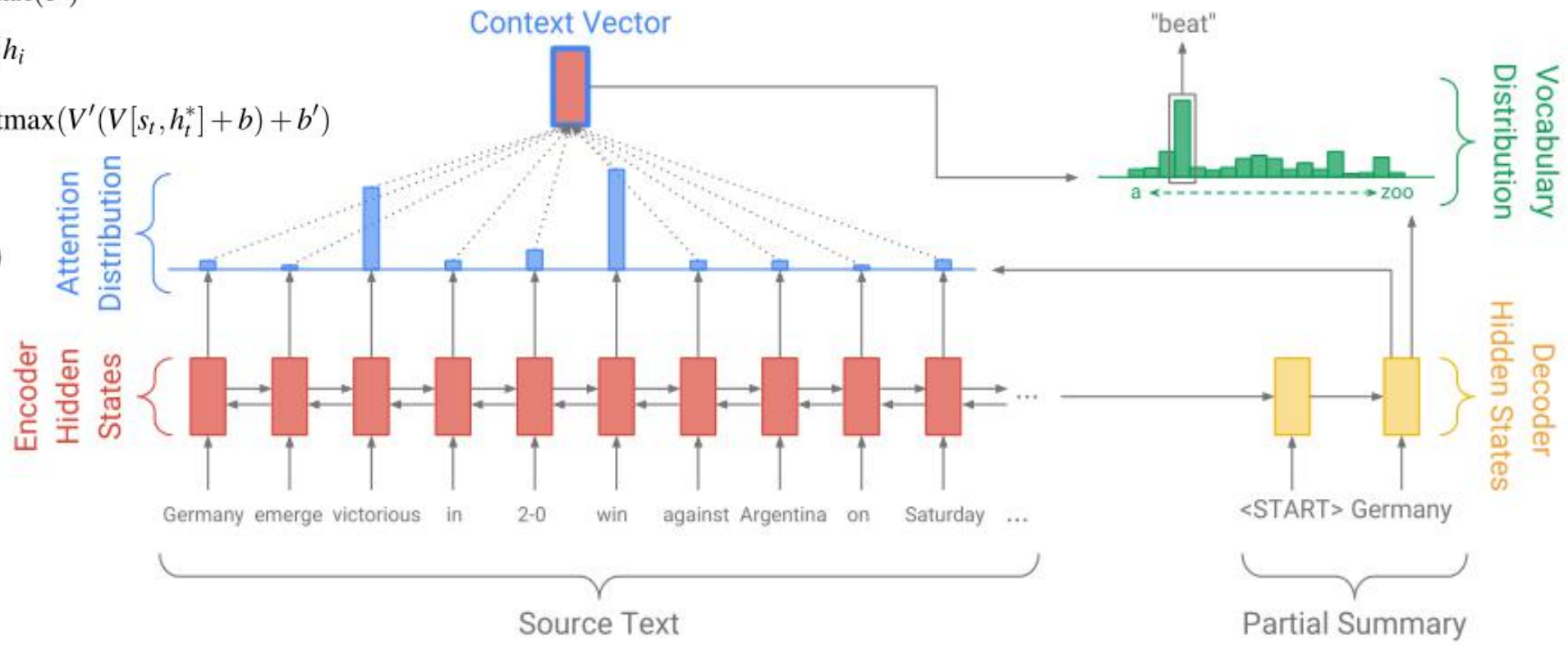
**Pointer-Gen + Coverage:** *muhammadu buhari* says he plans to aggressively fight corruption that has long plagued nigeria. he says his administration is confident it will be able to thwart criminals. the win comes after a long history of military rule, coups and botched attempts at democracy in africa's most populous nation.

Problem 1: The summaries sometimes reproduce factual details inaccurately (e.g. Germany beat Argentina 3-2). This is especially common for rare or out-of-vocabulary words such as 2-0.

Problem 2: The summaries sometimes repeat themselves (e.g. Germany beat Germany beat Germany beat...)

# PGN-baselinemodel

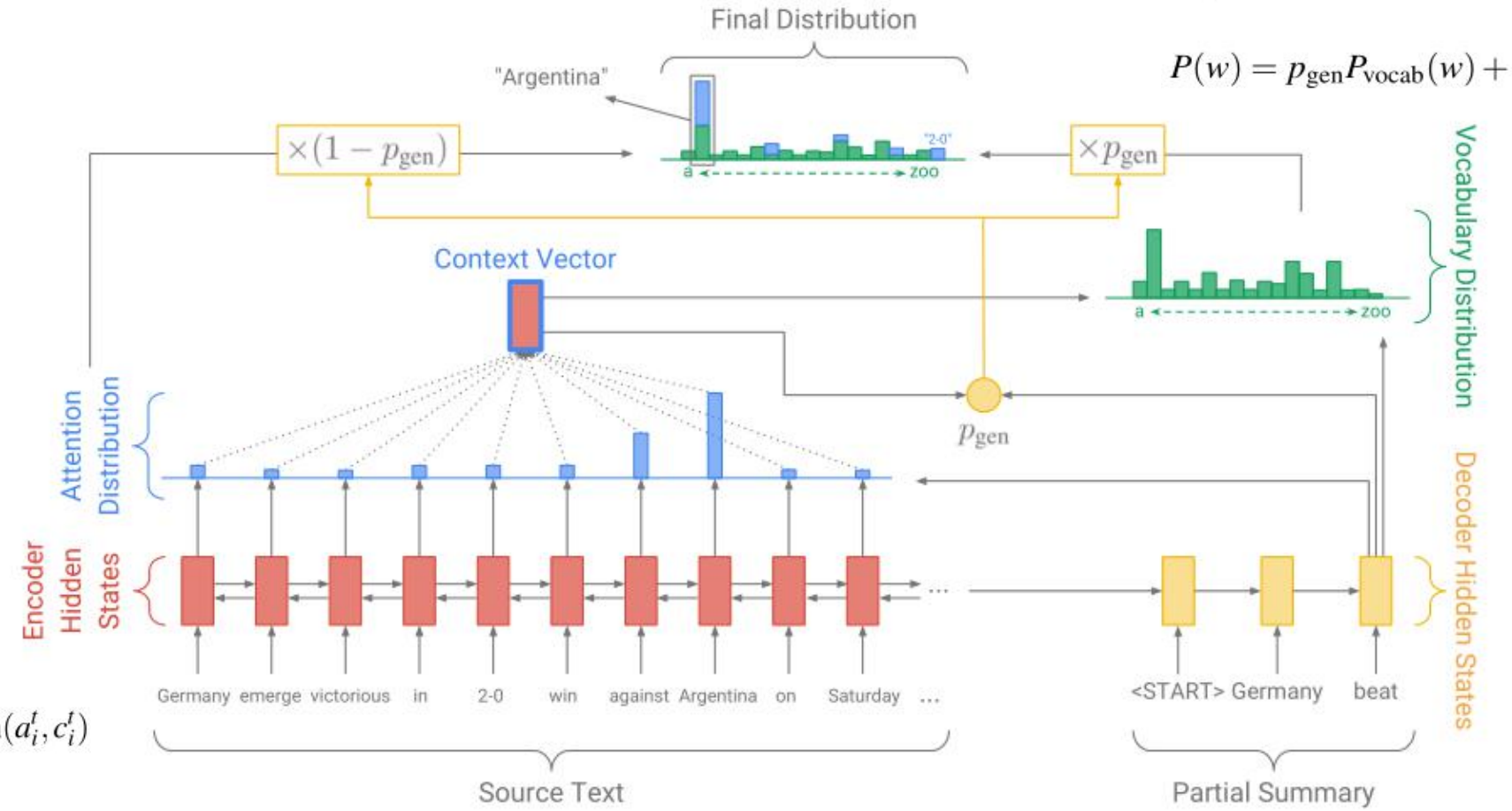
$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + b_{\text{attn}})$$
$$a^t = \text{softmax}(e^t)$$
$$h_t^* = \sum_i a_i^t h_i$$
$$P_{\text{vocab}} = \text{softmax}(V'(V[s_t, h_t^*] + b) + b')$$
$$P(w) = P_{\text{vocab}}(w)$$
$$\text{loss}_t = -\log P(w_t^*)$$
$$\text{loss} = \frac{1}{T}\sum_{t=0}^{T} \text{loss}_t$$



# PGN-model

$$p_{\text{gen}} = \sigma(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{\text{ptr}})$$

$$P(w) = p_{\text{gen}} P_{\text{vocab}}(w) + (1 - p_{\text{gen}}) \sum_{i:w_i=w} a_i^t$$




Coverage mechanism

$$\text{covloss}_t = \sum_i \min(a_i^t, c_i^t)$$

$$c^t = \sum_{t'=0}^{t-1} a^{t'}$$

$$\text{loss}_t = -\log P(w_t^*) + \lambda \sum_i \min(a_i^t, c_i^t)$$

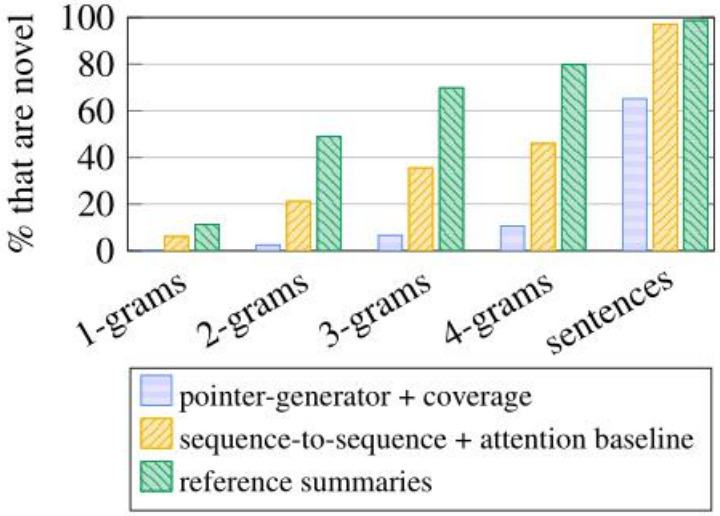
$$e_i^t = v^T \tanh(W_h h_i + W_s s_t + w_c c_i^t + b_{\text{attn}})$$

# PGN-Code

```
if use_coverage and coverage is not None: # non-first step of coverage
  # Multiply coverage vector by w_c to get coverage_features.
  coverage_features = nn_ops.conv2d(coverage, w_c, [1, 1, 1, 1], "SAME") # c has shape (batch_size, attn_length, 1, attention_vec_size)

  # Calculate v^T tanh(W_h h_i + W_s s_t + w_c c_i^t + b_attn)
  e = math_ops.reduce_sum(v * math_ops.tanh(encoder_features + decoder_features + coverage_features), [2, 3])  # shape (batch_size,attn_length)

  # Calculate attention distribution
  attn_dist = masked_attention(e)

  # Update coverage vector
  coverage += array_ops.reshape(attn_dist, [batch_size, -1, 1, 1])
else:
  # Calculate v^T tanh(W_h h_i + W_s s_t + b_attn)
  e = math_ops.reduce_sum(v * math_ops.tanh(encoder_features + decoder_features), [2, 3]) # calculate e

  # Calculate attention distribution
  attn_dist = masked_attention(e)

  if use_coverage: # first step of training
    coverage = tf.expand_dims(tf.expand_dims(attn_dist,2),2) # initialize coverage
```

# PGN-Experiments

| | ROUGE | | | METEOR | |
|---|---|---|---|---|---|
| | 1 | 2 | L | exact match | + stem/syn/para |
| abstractive model (Nallapati et al., 2016)* | 35.46 | 13.30 | 32.65 | - | - |
| seq-to-seq + attn baseline (150k vocab) | 30.49 | 11.17 | 28.08 | 11.65 | 12.86 |
| seq-to-seq + attn baseline (50k vocab) | 31.33 | 11.81 | 28.83 | 12.03 | 13.20 |
| pointer-generator | 36.44 | 15.66 | 33.42 | 15.35 | 16.65 |
| pointer-generator + coverage | **39.53** | **17.28** | **36.38** | 17.32 | 18.72 |
| lead-3 baseline (ours) | 40.34 | 17.70 | 36.57 | 20.48 | 22.21 |
| lead-3 baseline (Nallapati et al., 2017)* | 39.2 | 15.7 | 35.5 | - | - |
| extractive model (Nallapati et al., 2017)* | 39.6 | 16.2 | 35.3 | - | - |



# PGN--Discussion

summarization: extractive and abstractive

Coverage mechanism

# Thank You~