1 Summarization

1.1 Baseline

First X Sentences Baseline Works Well, but not always.

Example 1: New York Times Documents

This figure shows There is a long tail of useful sentences later in the document, as seen by the fact that the oracle sentence counts drop off relatively slowly.[Durrett et al. 2016]

Example 2: Multi-Document Case
Given many documents about Cyrus the Great, produce a summary.
1.2 Evaluation

Evaluation can be more subjective than other tasks. One typical choice is ROUGE [Lin, 2004]

1.2.1 ROUGE-1

Unigram overlap between generated text and reference. (Recall)
Example:

"Cyrus the Great defeated the Greek empire."
"Cyrus the Great founded the Persian empire."
Recall = 5/7

1.2.2 ROUGE-1 vs BLEU

ROUGE-1 is a recall focused metric, where the length of the generated text is constrained.
BLEU is a precision focused metric, where there is a brevity penalty to encourage the input to be longer.

1.2.3 ROUGE-L

ROUGE-L -ROUGE based on LCS (longest common subsequence statistics), does not require specifying a predetermined n-gram length.

Example

\[
S \quad \text{"Cyrus the Great defeated the Greek empire"}
\]
\[
G \quad \text{"Cyrus the Great founded the Persian empire."
}\]
\[
Recall \quad R_{lsc} = \frac{LCS(S, G)}{|G|}
\]
\[
Precision \quad P_{lsc} = \frac{LCS(S, G)}{|S|}
\]
\[
Rouge - L \quad F_{lsc} = \frac{(1 + \beta^2)R_{lsc}P_{lsc}}{R_{lsc} + \beta^2P_{lsc}}
\]

1.2.4 Shortcomings

ROUGE contains practically no measure of coherence.
ROUGE doesn't really capture precision of facts.
1.3 Approaching as Sentence Selection

Generally involves trading off between two criteria (subject to a length constraint):

**Relevance** Selected sentences should be important.

**Diversity** Selected sentences should be different from each other.

### 1.3.1 Maximum Marginal Relevance

Greedily pick sentences based on the following MMR objective:

\[
\text{MMR} = \arg\max_{s_i \in D \setminus A} (\lambda \times \text{Sim}_1(s_i, q) - (1 - \lambda) \times \max_{s_j \in A} \text{Sim}_2(s_i, s_j))
\]

### 1.3.2 TextRank

Construct a graph where vertices are sentences and edges are similarities. Compute a score for each vertex using PageRank:

\[
\text{score}(s_i) = (1 - d) + \sum_{j \in \text{In}(s_i)} \frac{1}{|\text{Out}(s_j)|} \text{score}(s_j)
\]

### 1.3.3 Using Word Frequency Only

Repeat the below steps:

1. Value\(w_i\) = \(P_D(w_i)\)
2. Value\(s_i\) = sum of its word values
3. Choose \(s_i\) with largest value that also contains highest scoring word
4. Adjust \(P_D(w)\)
5. Repeat until length constraint

Here, step (2) is for relevance, and step (4) is for diversity.
1.3.4 Global Inference

Solve global (NP-hard) objective:

\[ D = \{ t_1, \ldots, t_n \} \]
\[ \hat{A} = \arg \max_{A \subseteq D} \sum_{t_i \in A} \text{Rel}(t_i) - \sum_{t_i, t_j \in A, i < j} \text{Red}(t_i, t_j) \]
\[ \text{such that } \sum_{t_i \in A} \text{len}(t_i) \leq K \]

Here, \( D \) is textual units, \( \hat{A} \) is the summary, Rel is relevance, Red is redundancy, K is length constraint.

This equation can reformulate as integer linear program:

\[
\text{maximize} \sum_i \alpha_i \text{Rel}(t_i) - \sum_{i < j} \alpha_{ij} \text{Red}(t_i, t_j) \\
\text{such that} \\
1. \alpha_i, \alpha_j \in \{0, 1\} \\
2. \sum_i \alpha_i \text{len}(t_i) \leq K \\
3. \alpha_{ij} - \alpha_i \leq 0 \\
4. \alpha_{ij} - \alpha_j \leq 0 \\
5. \alpha_i + \alpha_j - \alpha_{ij} \leq 1
\]

1.3.5 A Scalable Global Model

Aims to solve two problems:
1. Reduce the number of variables in the ILP so that it can be solved efficiently.
2. Model redundancy more globally than via pairwise constraints

In a scalable global model, define:

**Concepts** could be n-grams, named entities, syntactic subtrees etc.

**Weights** are fixed (not learned). Gillick and Favre choose frequency count.

Then let

\[ c_i = \text{concept is in summary} \]
\[ s_i = \text{sentence is in summary} \]
The objective is to

\[
\text{maximize}_{c_i, s_i} \sum_i w_i c_i
\]

such that

1. \( c_i \in \{0, 1\} \forall i \)
2. \( s_j \in \{0, 1\} \forall j \)
3. \( s_j \text{Occ}_{ij} \leq c_i \forall i, j \)
4. \( \sum_j s_j \text{Occ}_{ij} \geq c_i \forall i \)

where \( \text{Occ}_{ij} \) indicat the occurrence of concept \( i \) in sentence \( j \).

### 1.3.6 Submodularity

Define an objective that is monotone submodular.

It is NP-hard to find the globally optimal solution, but a greedy approximation is guaranteed to give a constant factor approximation

\[
\mathcal{F}(\hat{A}) \geq (1 - 1/e) \mathcal{F}(A_{\text{opt}}) \approx 0.632 \mathcal{F}(A_{\text{opt}})
\]

Definition:

**Submodularity** Let \( \Omega \) be a set and \( \mathcal{F} \) be a function that maps subsets \( X \subseteq \Omega \) to real values in \( \mathbb{R} \). Then \( \mathcal{F} \) is submodular if for every set \( X, Y \subseteq \Omega \) with \( X \subseteq Y \subseteq \Omega \setminus v \)

\[
\mathcal{F}(X + v) - \mathcal{F}(X) \geq \mathcal{F}(Y + v) - \mathcal{F}(Y)
\]

which means that \( v \) relatively adds more to \( X \) than it does to \( Y \).

**Monotone Submodularity** Furthermore, \( \mathcal{F} \) is monotone submodular if for all \( X \subseteq Y \subseteq \Omega \), \( \mathcal{F}(Y) \geq \mathcal{F}(X) \),

which means that adding more sentences to the summary shouldnt decrease the objective.

Submodularity reward diversity instead of penalizing redundancy:

\[
\mathcal{F} := L(A) + \lambda R(A)
\]

where \( L \) is relevence and \( R \) is diversity.

If sentences are clustered into \( K \) clusters: \( C_1, \ldots, C_k \):

\[
R(A) = \sum_{k=1}^{K} \sqrt{\sum_{j \in C_k \cap A} w_j}
\]

where \( w_j \) is the relevance score for sentence \( j \).

This square root function lowers reward for picking too many sentences from the same cluster.

### 2 Case Study: Challenge in Data Generation

#### 2.1 The Challenge

For text summarization, nowadays neural models as so called deep learning technique have made breaking-through results on such tasks. However, in this work, they would like to propose that for large context tasks,
which is document generation, experiments show that current models (e.g seq2seq with attention) produce fluent text, but fail to convincingly approximate human-generated documents.

2.2 The New Dataset

There are several benchmark datasets used for such text generation task. Such as WEATHERGOV(Liang, 2009), ROBOCUP(Chen and Moony, 2008), and WIKIBIO(Lebret, 2016). However, these datasets, for instance WEATHERGOV is at least partially machine-generated (Reiter, 2017). And generations contained within WIKIBIO are mostly a one sentence short description.

To fulfill the purpose to get a long-enough and also sampled from real world data. They collected ROTOWIRE (RW) and SBNATION (SBN), which came from two sources of articles summarizing NBA basketball games, paired with their corresponding game scoring tables. There are 15 metrics for a line-score tables, and 24 metrics for a box-score tables containing at most 25 different players. The summarization is in different style as RW is a more well-structured and formal, on the other hand, style for SBN is more fans-targeted.

Figure 1 shows one of an example from ROTOWIRE, taken from the origin paper of the dataset (Wiseman, 2017).

```
<table>
<thead>
<tr>
<th>WIN</th>
<th>LOSS</th>
<th>PTS</th>
<th>FG_PCT</th>
<th>RB</th>
<th>AS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heat</td>
<td>11</td>
<td>12</td>
<td>103</td>
<td>49</td>
<td>47</td>
</tr>
<tr>
<td>Hawks</td>
<td>7</td>
<td>15</td>
<td>95</td>
<td>43</td>
<td>33</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TEAM</th>
<th>WIN</th>
<th>LOSS</th>
<th>PTS</th>
<th>FG_PCT</th>
<th>RB</th>
<th>AS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heat</td>
<td>11</td>
<td>12</td>
<td>103</td>
<td>49</td>
<td>47</td>
<td>27</td>
</tr>
<tr>
<td>Hawks</td>
<td>7</td>
<td>15</td>
<td>95</td>
<td>43</td>
<td>33</td>
<td>20</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PLAYER</th>
<th>AS</th>
<th>RB</th>
<th>PT</th>
<th>FG</th>
<th>FGA</th>
<th>CITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tyler Johnson</td>
<td>5</td>
<td>2</td>
<td>27</td>
<td>8</td>
<td>16</td>
<td>Miami</td>
</tr>
<tr>
<td>Dwight Howard</td>
<td>4</td>
<td>17</td>
<td>23</td>
<td>9</td>
<td>11</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Paul Millsap</td>
<td>2</td>
<td>9</td>
<td>21</td>
<td>8</td>
<td>12</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Goran Dragic</td>
<td>4</td>
<td>2</td>
<td>21</td>
<td>8</td>
<td>17</td>
<td>Miami</td>
</tr>
<tr>
<td>Wayne Ellington</td>
<td>2</td>
<td>3</td>
<td>19</td>
<td>7</td>
<td>15</td>
<td>Miami</td>
</tr>
<tr>
<td>Dennis Schroder</td>
<td>7</td>
<td>4</td>
<td>17</td>
<td>8</td>
<td>15</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Rodney McGruder</td>
<td>5</td>
<td>5</td>
<td>11</td>
<td>3</td>
<td>8</td>
<td>Miami</td>
</tr>
<tr>
<td>Thabo Sefolosha</td>
<td>5</td>
<td>5</td>
<td>10</td>
<td>5</td>
<td>11</td>
<td>Atlanta</td>
</tr>
<tr>
<td>Kyle Korver</td>
<td>5</td>
<td>3</td>
<td>9</td>
<td>3</td>
<td>9</td>
<td>Atlanta</td>
</tr>
</tbody>
</table>
```

Figure 1: A sample from the dataset.

2.3 Neural data-to-document models

The task they proposed, however, could be generalized as a text generation problem, however with longer texts and given data tables.
2.3.1 baseline

The paper suggested that for the base model, we could map each record \( r \in s \) into a vector \( r' \) by first embedding r.t (e.g., POINTS), r.e (e.g., RUSSELL WESTBROOK), and r.m (e.g., 50), and then applying a 1-layer MLP (similar to Yang et al. (2016)). Our source data-records are then represented as \( s' = \{r_j\}^J_{j=1} \). Given this we could apply nowadays standard seq2seq model as the baseline, where the model takes the data-to-document as the sequence-to-sequence for end to end training.

2.3.2 Source Copying Variation

There are several variants of this model proposed on other tasks, where copying for text summarization (Gu, 2016) and reconstruction losses for language translation (Tu, 2017) are introduced. The variation can be then re-parameterized as shown in Figure 2, Figure 3, and Figure 4.

\[
p(\hat{y}_t | \hat{y}_{1:t-1}, s) = \sum_{z \in \{0,1\}} p(\hat{y}_t, z = 1, \hat{y}_{1:t-1}, s).
\]

Figure 2: Reparameterized for copying, where \( y^t \) is copied from the source or generated.

\[
p(\hat{y}_t, z_t | \hat{y}_{1:t-1}, s) \propto \\
p_{\text{copy}}(\hat{y}_t, \hat{y}_{1:t-1}, s), & \quad z_t = 1, \hat{y}_t \in s \\
0, & \quad z_t = 1, \hat{y}_t \notin s \\
p_{\text{gen}}(\hat{y}_t, \hat{y}_{1:t-1}, s), & \quad z_t = 0,
\]

Figure 3: The models of Gu et al. (2016) and Yang et al. (2016) parameterize the joint distribution table over \( y_t \) and \( z_t \) directly.

\[
p(\hat{y}_t, z_t | \hat{y}_{1:t-1}, s) = \\
p_{\text{copy}}(\hat{y}_t | z_t, \hat{y}_{1:t-1}, s) p(z_t | \hat{y}_{1:t-1}, s), & \quad z_t=1 \\
p_{\text{gen}}(\hat{y}_t | z_t, \hat{y}_{1:t-1}, s) p(z_t | \hat{y}_{1:t-1}, s), & \quad z_t=0,
\]

Figure 4: Conditional Copy Model Gulcehre et al.(2016), on the other hand, decompose the joint probability as this way.

2.3.3 Source Reconstruction

Reconstruction-based techniques can also be applied at the document or sentence-level during training. One simple approach to this problem is to utilize the hidden states of the decoder to try to reconstruct the database.
$\mathcal{L}(\theta) = -\sum_{k=1}^{K} \min_{r \in s} \log p_k(r | b_i; \theta)$

$= -\sum_{k=1}^{K} \min_{r \in s} \sum_{x \in \{e, m, t\}} \log p_k(r, x | b_i; \theta)$.

Figure 5: We can train with the reconstruction loss for a particular block $b_i$, where $p_k$ is the $k$th predicted distribution over records, and where we have modeled each component of $r$ independently.

2.4 Advantage of the dataset

One main purpose of this new dataset is to improve some current metrics for natural language generation (NLG). The evaluation of NLG has been a difficult problem for a long time and researchers have developed different type of automatic evaluation metrics, for example, BLEU (Habash, 2004; Belz, 2005) which is widely used in machine translation field and NIST (Doddington, 2002) evaluation metrics which emphasize on the less frequent terms and could be more sensitive than BLEU. And also there are ROUGE metric and different variants of it (Lin, 2004). These metrics are mostly based on the n-gram and reward the systems that generate short term fluent content but unable to evaluate the correctness and completeness of the information conveyed in the generated text. So often human evaluation of the NLG system are required while its very inconvenient.

2.4.1 Quantitative

Within the dataset, we already have all the information that we want to express in natural language and we feed those information to the NLG model to get text. So we can evaluate the NLG model by inspecting the information in the generated text and compare them with the information in the human written text. The information should be contained in the generated text should be relations about the entities (basketball team and players etc.) and scores.

2.4.2 Novel Evaluation Metrics

This work introduced in particular three induced metrics:

- Content Selection (CS): precision and recall of unique relations $r$ extracted from $y_{1:T}$ that are also extracted from $y_{1:T}$. This measures how well the generated document matches the gold document in terms of selecting which records to generate.
- Relation Generation (RG): precision and number of unique relations $r$ extracted from $y_{1:T}$ that also appear in $s$. This measures how well the system is able to generate text containing factual (i.e., correct) records.
- Content Ordering (CO): normalized Damerau-Levenshtein Distance (Brill and Moore, 2000) between the sequences of records extracted from $y_{1:T}$ and that extracted from $y_{1:T}$. This measures how well the system orders the records it chooses to discuss.

Note that CS primarily targets the what to say aspect of evaluation, CO targets the how to say it aspect, and RG targets both.
2.5 Some Experiment Results

2.5.1 The information is wrong

Figure 6 shows a document generated by the Conditional Copy model, using a beam of size 5. This particular generation evidently has several nice properties: it nicely learns the colloquial style of the text, correctly using idioms such as 19 percent from deep. It is also partially accurate in its use of the records.

This generation also contains major logical errors. First, there are basic copying mistakes, such as flipping the teams' win/loss records. The system also makes obvious semantic errors; for instance, it generates the phrase the Rockets were able to out-rebound the Rockets. Finally, we see the model hallucinates factual statements, such as in front of their home crowd, which is presumably likely according to the language model, but ultimately incorrect (and not supported by anything in the box- or linescores).

Figure 6: Highlights in blue when it generates text that is licensed by a record in the associated box- and line-scores. While the erroneous text is highlighted in red.

2.6 Drawback: Long term reference

It turns out that current vanilla sequence to sequence style model is incapable to handle this kind of long-term reference while preserving a good language model. Some evaluations are shown in Figure 7. Obviously, they can produce some good results on BLEU score and perplexity, but fail far behind even a 0-cost template system. This task shows difficulties in generation problems from different aspects. Also, they proposed the work to motivate researchers to focus further on generation problems that are relevant both to content selection and surface realization, but may not be reflected clearly in the models perplexity.

3 Conclusion

So the lecture closed with some important key points for open research problem (Not only limited to NLP):

- Always starts with setup Good data + Good evaluation.
- Always starts from simple models as strong baseline => A more complex approach
- Always starts with a certain strong knowledge base $\Rightarrow$ Connects the dots.

<table>
<thead>
<tr>
<th>Development</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Beam</td>
<td>Model</td>
<td>RG</td>
<td>CS</td>
<td>CO</td>
<td>PPL</td>
<td>BLEU</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>P%</td>
<td>#</td>
<td>P%</td>
<td>R%</td>
<td>DLD%</td>
<td></td>
</tr>
<tr>
<td>Gold</td>
<td>91.77</td>
<td>12.84</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>Template</td>
<td>99.35</td>
<td>49.7</td>
<td>18.28</td>
<td>65.52</td>
<td>12.2</td>
<td>N/A</td>
<td>6.87</td>
</tr>
<tr>
<td>B=1</td>
<td>Joint Copy</td>
<td>47.55</td>
<td>7.53</td>
<td>20.53</td>
<td>22.49</td>
<td>8.28</td>
<td>7.46</td>
</tr>
<tr>
<td>Joint Copy + Rec</td>
<td>57.81</td>
<td>8.31</td>
<td>23.65</td>
<td>23.30</td>
<td>9.02</td>
<td>7.25</td>
<td>10.00</td>
</tr>
<tr>
<td>Joint Copy + Rec + TVD</td>
<td>60.69</td>
<td>8.95</td>
<td>23.63</td>
<td>24.10</td>
<td>8.84</td>
<td>7.22</td>
<td>12.78</td>
</tr>
<tr>
<td>Conditional Copy</td>
<td>68.94</td>
<td>9.09</td>
<td>25.15</td>
<td>22.94</td>
<td>9.00</td>
<td>7.44</td>
<td>13.31</td>
</tr>
<tr>
<td>B=5</td>
<td>Joint Copy</td>
<td>47.00</td>
<td>10.67</td>
<td>16.52</td>
<td>26.08</td>
<td>7.28</td>
<td>7.46</td>
</tr>
<tr>
<td>Joint Copy + Rec + TVD</td>
<td>57.51</td>
<td>11.41</td>
<td>18.28</td>
<td>25.27</td>
<td>8.05</td>
<td>7.22</td>
<td>12.04</td>
</tr>
<tr>
<td>Conditional Copy</td>
<td>71.07</td>
<td>12.61</td>
<td>21.90</td>
<td>27.27</td>
<td>8.70</td>
<td>7.44</td>
<td>14.46</td>
</tr>
<tr>
<td>Test</td>
<td>Template</td>
<td>99.30</td>
<td>49.61</td>
<td>18.50</td>
<td>64.70</td>
<td>8.04</td>
<td>N/A</td>
</tr>
<tr>
<td>Joint Copy + Rec (B=5)</td>
<td>61.23</td>
<td>11.02</td>
<td>21.56</td>
<td>26.45</td>
<td>9.06</td>
<td>7.47</td>
<td>10.88</td>
</tr>
<tr>
<td>Joint Copy + Rec + TVD (B=1)</td>
<td>60.27</td>
<td>9.18</td>
<td>23.11</td>
<td>23.69</td>
<td>8.48</td>
<td>7.42</td>
<td>12.96</td>
</tr>
<tr>
<td>Conditional Copy (B=5)</td>
<td>71.82</td>
<td>12.82</td>
<td>22.17</td>
<td>27.16</td>
<td>8.68</td>
<td>7.67</td>
<td>14.49</td>
</tr>
</tbody>
</table>

Figure 7: Performance of introduced metrics on gold and system outputs of RotoWire development and test data.