Semantic Extractor-Paraphraser based Abstractive Summarization

Anubhav Jangra*  
IIT Patna, India  
anubhav0603@gmail.com

Raghav Jain*  
DTU, India  
araghavjain106@gmail.com

Vaibhav Mavi*  
IIT Delhi, India  
vaibhavgl52@gmail.com

Sriparna Saha  
IIT Patna, India  
sriparna.saha@gmail.com

Pushpak Bhattacharyya  
IIT Bombay, India  
pushpakbh@gmail.com

Abstract

The anthology of spoken languages today is inundated with textual information, necessitating the development of automatic summarization models. In this manuscript, we propose an extractor-paraphraser based abstractive summarization system that exploits semantic overlap as opposed to its predecessors that focus more on syntactic information overlap. Our model outperforms the state-of-the-art baselines in terms of ROUGE, METEOR and word mover similarity (WMS), establishing the superiority of the proposed system via extensive ablation experiments. We have also challenged the summarization capabilities of the state of the art Pointer Generator Network (PGN), and through thorough experimentation, shown that PGN is more of a paraphraser, contrary to the prevailing notion of a summarizer; illustrating it’s incapability to accumulate information across multiple sentences.

1 Introduction

Over the past few years, the Internet has become the most convenient and preferred form of information sharing worldwide. The evolution of technology has made it possible for anyone to convey their knowledge, opinions and ideals to the world, resulting in an increasing surge of information hindering users from accessing desired content. This increasing need to obtain key information makes the task of summarization paramount. Text is the most widely adopted form of communication, be it for personal messaging1 or for broadcasting, owing to its ability to convey almost any concept, its general flexibility to suit everyone’s needs, and its less storage requirement (opposed to other modes of communication like audio and video). Text summarization is a problem at the very core of natural language processing, and has various applications in the spoken languages, including summarization of conversations, and public speeches.

Some works have been done in the field of reinforcement learning based text summarization (Dong et al., 2018; Liu et al., 2018), the most prominent architecture being extractor-abstractor (EXT-ABS) model (Chen and Bansal, 2018). Inspired from this architecture, in this manuscript, we have proposed an extractor-paraphraser system that uses semantic information overlap as the underlying guidance strategy. The model is further enhanced to surpass its limits using reinforcement learning, for which we have proposed a novel semantic overlap based reward function. Word Mover Similarity (WMS) (Clark et al., 2019) is utilized to evaluate semantic similarity across generated sentences and the true ground truth summary sentences.

We assume that paraphrasing is a relatively simpler task than abstractive summarization, with the underlying intuition that paraphrasing is a sub-problem within abstractive summarization. To bolster our hypothesis, experiments are conducted on the extractor-abstractor (EXT-ABS) model (Chen and Bansal, 2018) and the Pointer Generator Network (PGN) (See et al., 2017), which is used as the basic abstraction unit in the former architecture. The results are rather staggering and reveal that the PGN model also paraphrases input document sentences, albeit implicitly. The major contributions of the paper are as follows:

• A novel semantic overlap based reward function is proposed for reinforcement of extractor-paraphraser model.
• To the best of our knowledge, we are the first ever to discover the fact that PGN networks are indeed doing an implicit extraction-paraphrasing operation, revealing the true nature of existing abstractive summarization models.

* means equal contribution.

1https://news.gallup.com/poll/179288/new-era-communication-americans.aspx
The rest of this paper is structured as follows: In Section 2 we have discussed related works of automatic text summarization. In Section 3 we have described the proposed model, and in Section 4 we have stated the experimental setup and the datasets used. A thorough discussion and state the results are provided in Section 5, followed by the conclusion and future work in Section 6.

2 Related Work

Automatic text summarization has been extensively researched over more than three decades, and has shown a lot of progress and promise over the course of time. Various approaches have been explored to tackle both extractive and abstractive summarization. Initial research (Paice, 1990; Kupiec et al., 1995) focused on extractive summarization due to its easier setup. Various techniques ranging from integer linear programming (Galanis et al., 2012), graph based approaches (Mihalcea and Tarau, 2004; Mihalcea, 2004), genetic algorithms (Saini et al., 2019a,b), and neural networks (Nallapati et al., 2017; Zhang et al., 2016) have been adopted to solve the extractive summarization task. The majority of the research in abstractive summarization revolves around deep learning (See et al., 2017; Chopra et al., 2016; Nallapati et al., 2016). Liu et al. (2018) proposed a generative adversarial network based model to generate document abstracts. A handful of works however also use ILP (Banerjee et al., 2015) and graph-based (Ganesan et al., 2010) techniques to attempt to solve the problem. A lot of domain specific summarization techniques have also been explored, like radiology findings summarization (Zhang et al., 2018), across-time summarization (Duan and Jatowt, 2019), movie review summarization (Zhuang et al., 2006), book summarization (Mihalcea and Ceylan, 2007), and customer review based opinion summarization (Pecar, 2018). Lately, multi-modal summarization (Jangra et al., 2020a,b; Zhu et al., 2020; Saini et al., 2020) has also gained popularity.

Recently, people have also explored reinforcement learning to tackle the problem of automatic text summarization in both extractive (Dong et al., 2018; Gao et al., 2019) and abstractive domains (Xiao et al., 2020; Chen and Bansal, 2018). Chen and Bansal (2018) have proposed an extractor-abstractor architecture, separating the relevant data searching part and the paraphrasing part to individual modules. In this work, we have proposed a system inspired from Chen and Bansal (2018), stressing on the significance of semantic information over the traditional syntactic overlap. The literature on text summarization is rich, and has an abundance of survey papers (Yao et al., 2017; Gambhir and Gupta, 2017) to get an in depth overview of the domain.

3 Proposed Method

Problem Definition: Given the training data \{X, Y\} where \(X = \{d_1, d_2, ..., d_N\}\) is the set of input documents and \(Y = \{y_1, y_2, ..., y_N\}\) is the set of corresponding output summaries, the task of automatic summarization is defined as the problem of discovering a function \(f: X \mapsto Y\), such that \(f(d_i) = y_i; \forall i \in \{1, 2, ..., N\}\).

We have proposed an extractor-paraphraser framework, which is inspired from the extractor-abstractor (EXT-ABS) framework introduced by
Chen and Bansal (2018). The summarization function \( f(.) \) is approximated as the composition, \( f(d_i) = h(g(d_i)) \), where the functions \( g(.) \) and \( h(.) \) are modeled as the extractor and the paraphraser components of the model, respectively. Given an input document \( d_i = \{s_1^d, s_2^d, \ldots, s_{|d_i|}^d\} \), the extractor \( g(.) \) extracts relevant sentences, acting as the primary noise filter. These extracted set of sentences are fed to the paraphraser \( h(.) \), which accumulates the information into a concise gist of the extracted sentences, simulating the natural language generation module in the proposed system (Fig. 1). The paraphraser in our system is capable of summarizing multiple extracted sentences to generate one sentence, in contrast with its predecessor ‘abstractor’ from EXT-ABS model (Chen and Bansal, 2018) which rephrases one extracted sentence at a time. The proposed framework consists of an ‘n-to-one paraphraser’, that compiles \( n \) sentences into one sentence, generating a richer summary. Formally, given a document \( d_i = \{s_1^d, s_2^d, \ldots, s_{|d_i|}^d\} \), the extractor is defined as:

\[
g : X \mapsto \mathbb{Z}^{[|y_i| \times n]},
\]

\[
s.t.\ g(d_i) = \{k_{j,l} \mid 1 \leq k_{j,l} \leq |d_i|, y_i = 1, \ldots, y_i = n\}
\]

where \( k_{p,q} \) represents the index of the \( q^{th} \) extracted sentence corresponding to the \( p^{th} \) sentence in the final gold summary. For modelling the same, we have used an encoder-decoder model, where the encoder consists of a temporal convolitional model (Kim, 2014) cascaded with a bidirectional LSTM network (Schuster and Paliwal, 1997) and the decoder is a uni-directional LSTM model (Hochreiter and Schmidhuber, 1997) (Fig. 1).

The paraphraser function \( h(.) \) is defined as follows:

\[
h\{\{k_{j,l}\}_{l=1}^{n} \mid y_i \mid, d_i \} = p \left( \oplus_{l=1}^{n} s_{k_{j,l}}^{d_l} \right)_{j=1}^{\mid y_i \mid}
\]

where \( \oplus \) represents the concatenation of sentences, \( k_{j,l} \) represents the extractor output and \( p \) is written as:

\[
p(s_j^{ext}) = s_j^g
\]

where \( s_j^{ext} = \oplus_{l=1}^{n} s_{k_{j,l}}^{d_l} \) and \( s_j^g \) is the \( j^{th} \) gold summary sentence. As shown in Fig. 1, a pointer-generator framework (See et al., 2017) is used to model the paraphraser.

### 3.1 Training the Submodules

In order to train the paraphraser beforehand, we need to mimic the output of an extractor\(^2\). To fulfil this requirement, exemplary-extracted sentences are generated using gold summaries. Word-mover distance (WMD) (Kusner et al., 2015) is adopted to create these exemplary-extracted sentences, with the motivation that it captures the semantic overlap between sentences better than its predecessors. For a training pair \( \{d, y\} \), the labels are generated as:

\[
\forall l \in \{1, 2, \ldots, n\}:
\]

\[
\forall s_j \in y:
\]

\[
k_{j,l} = \text{argmin}_{i} \{\text{WMD}(s_i^d, s_j^y)\};
\]

\[
\forall s_i^d \in d - \{s_k^d\}_{k, l}^{j-1}
\]

(4)

where \( k_{j,l} \) represents the \( l^{th} \) exemplary-extracted sentence corresponding to the \( j^{th} \) gold summary sentence, \( s_j^y \).

The \( n \) exemplary-extracted sentences corresponding to each summary sentence are concatenated and fed into the paraphraser as the input and the summary sentence is fed as the output. We argue that this should allow the paraphraser to generate information rich summaries. To facilitate the expected behaviour during the testing phase, we require the extractor to feed the paraphraser with input similar to the exemplary-extracted sentences. Hence, we first pre-train the extractor on these exemplary-extracted sentences as opposed to random initialization. Cross-entropy loss is used for training the paraphraser and pre-training the extractor.

### 3.2 Extractor agent

**Enhancement using reinforcement learning:**

Here, the extractor is trained as part of an actor-critic model (Mnih et al., 2016) which takes an action based on the current state and current value of parameters to maximize a given reward at each time step, where the action is to extract \( n \) sentences, \( \{k_{l,i}\}_{l=1}^{n} \), and the state refers to the set of document sentences, \( d_i \), and already extracted sentences, \( \{d_{k_{l,i}}, d_{k_{l,i}}^{2}, \ldots, d_{k_{l,i}-1}\} \); \( l \in \{1, 2, \ldots, n\} \). The predicted sentences are concatenated and passed to

\(^2\)Note that the extractor’s output can also be used to train the paraphraser at this step, however, since the extractor’s weights get fine-tuned using reinforcement learning at a later stage, training paraphraser on extractor’s non-ideal outputs might lead to unsatisfactory performance of the paraphraser.
the paraphraser to get an output sentence. A novel semantic-based reward function using word mover distance (WMD) is used as the reward function. Since the reward is to be maximised, WMD needs to be converted to a similarity function, for which, a generalised version of word mover similarity (WMS), proposed in (Clark et al., 2019), is used. Formally, at a time step $t$, given an action $j_{t,l}$, and a summary sentence, $s_t$, the short term reward, $r$ is calculated as:

$$r = \frac{a + 1}{a + e^{b \times WMD(s_t^\text{ext}, p(s_t'))}}$$ (5)

where $s_t^\text{ext} = \bigoplus_{l=1}^{n} s_{k_{t,l}}$, and $\bigoplus$ represents the concatenation of sentences, $\{s^d, s^n\}$ are the sentences belonging to a training pair and $a$ and $b$ are the hyper-parameters introduced. To avoid redundant phrases and words, tri-gram avoidance through beam search (Paulus et al., 2017) is applied at a sentence level. For a fair comparison, details regarding the beam search reranking and other nuances of implementation are kept the same as (Chen and Bansal, 2018).

4 Experiments

4.1 Dataset

For all the following experiments we have used the CNN / DailyMail dataset (Nallapati et al., 2016), which contains online news articles, with the bullet highlights treated as the gold standard summaries. The experiments in this work are conducted on the non-anonymized version of this dataset. The dataset consists of 277,226 training, 13,368 validation and 11,490 test article-summary pairs. An article contains $\sim$780 tokens per document, whereas the summary consists of $\sim$56 tokens with the average number of sentences per summary being $\sim$3.75. An article sentence, on average contains $\sim$30 tokens.

4.2 Comparative methods

To highlight the superiority of the proposed semantic-overlap based methodology over existing syntactic measures, we set the value of $n = 1$ for initial experiments. To compare the complexity of the summarization task with paraphrasing objectively, the experiments are further extended for $n = 2$ as well. We evaluate our model with sufficient baselines, including:

**Pointer-Generator network [PGN]:** An encoder-decoder attention based framework for abstractive summarization (See et al., 2017).

**Pre-trained extractor - Extractor-Abstractor framework [EXT – ABS (Ext only)]:** The pre-trained extractor of the extractor-abstractor framework proposed by Chen and Bansal (2018) on ROUGE-L based exemplary-extracted sentences.

**Extractor-Abstractor framework without reinforcement [EXT – ABS (w/o RL)]:** The extractor-abstractor framework proposed by Chen and Bansal (2018) at a stage before reinforcing the extractor.

**Extractor-Abstractor framework [EXT – ABS + RLX]:** The complete extractor-abstractor framework proposed by Chen and Bansal (2018) including X as the reward function for reinforcement learning (X is either ROUGE-L or the reward function defined in Eq. 5) along with beam search.

**Pre-trained extractor - One-to-one extractor-paraphraser model [O2O (Ext only)]:** The pre-trained extractor of the proposed extractor-paraphraser framework (with $n = 1$) on WMD based (Eq. 4) exemplary-extracted sentences.

**One-to-one extractor-paraphraser model without reinforcement [O2O (w/o RL)]:** A particular setting of the proposed methodology where $n = 1$, without any reinforcement or beam search.

**One-to-one extractor-paraphraser model [O2O + RLX]:** A particular setting of the proposed methodology where $n = 1$, including extractor, paraphraser, and X as the reward function for reinforcement learning (X is either ROUGE-L or the reward function defined in Eq. 5) and beam search.

**Pre-trained extractor - Two-to-one extractor-paraphraser model [M2O, $n = 2$ (Ext only)]:** The pre-trained extractor of the proposed extractor-paraphraser framework (with $n = 2$) on WMS based (Eq. 4) exemplary-extracted sentences.

---

Note that we obtain the word mover similarity proposed in (Clark et al., 2019) for the case $a = 0$, and $b = 1$.

The hyperparameters are set to $a = 1$ and $b = 0.5$ after extensive experimentation, and are used throughout this paper.

---

6Statistical analysis on all the variations of the proposed model has been done.

7For fair comparison of extraction capabilities across all models, we limit the model to output 4 sentences during evaluation on test dataset.
Table 1: Evaluation scores for the generated text summary using ROUGE, METEOR and Word Mover Similarity (WMS). ‘Id-ext’ refers to the ideal extractor experiments. The ‘-’ denotes unavailability of a score.

<table>
<thead>
<tr>
<th>Models</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-L</th>
<th>METEOR</th>
<th>WMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extractive baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXT - ABS (Ext only)</td>
<td>40.17</td>
<td>18.11</td>
<td>36.41</td>
<td>22.81</td>
<td>-</td>
</tr>
<tr>
<td>O2O (Ext only)</td>
<td><strong>40.97</strong></td>
<td><strong>18.42</strong></td>
<td><strong>37.35</strong></td>
<td>21.86</td>
<td><strong>14.28</strong></td>
</tr>
<tr>
<td>M2On=2 (Ext only)</td>
<td>40.19</td>
<td>17.98</td>
<td>36.60</td>
<td>21.62</td>
<td>14.24</td>
</tr>
<tr>
<td>Abstractive baselines</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PGN</td>
<td>39.53</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXT - ABS (w/o RL)</td>
<td>38.38</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>O2O (w/o RL)</td>
<td><strong>39.82</strong></td>
<td></td>
<td><strong>37.21</strong></td>
<td>19.24</td>
<td>13.81</td>
</tr>
<tr>
<td>M2On=2 (w/o RL)</td>
<td>32.82</td>
<td>11.29</td>
<td>31.08</td>
<td>16.13</td>
<td>12.92</td>
</tr>
<tr>
<td>Reinforced models</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EXT - ABS + RLROUGE</td>
<td>40.88</td>
<td>17.8</td>
<td>38.53</td>
<td>20.38</td>
<td>13.7</td>
</tr>
<tr>
<td>O2O + RLROUGE</td>
<td><strong>41.32</strong></td>
<td><strong>18.16</strong></td>
<td><strong>38.89</strong></td>
<td>20.52</td>
<td>14.56</td>
</tr>
<tr>
<td>EXT - ABS + RLWMS</td>
<td>40.82</td>
<td>17.82</td>
<td>38.45</td>
<td><strong>21.47</strong></td>
<td>14.42</td>
</tr>
<tr>
<td>O2O + RLWMS</td>
<td>41.2</td>
<td>18.12</td>
<td>38.81</td>
<td>21.34</td>
<td><strong>14.6</strong></td>
</tr>
<tr>
<td>M2On=2 + RLROUGE</td>
<td>39.71</td>
<td>16.7</td>
<td>37.32</td>
<td>18.25</td>
<td>13.52</td>
</tr>
<tr>
<td>Ablation experiments (Ideal Extractor)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Id - ext EXT - ABS</td>
<td>49.73</td>
<td>26.53</td>
<td>47.2</td>
<td>24.36</td>
<td>19.34</td>
</tr>
<tr>
<td>Id - ext O2O</td>
<td><strong>50.04</strong></td>
<td>26.31</td>
<td><strong>47.34</strong></td>
<td><strong>24.61</strong></td>
<td><strong>20.14</strong></td>
</tr>
<tr>
<td>Id - ext M2On=2</td>
<td>47.00</td>
<td>23.37</td>
<td>44.23</td>
<td>22.22</td>
<td>17.99</td>
</tr>
</tbody>
</table>

Two-to-one extractor-paraphraser model without reinforcement [M2On=2 (w/o RL)]: A specific case of the proposed many-to-one paraphrasing where \( n = 2 \). This specific model comprises of only the extractor and paraphraser modules.

Two-to-one extractor-paraphraser model [M2On=2 + RLX]: A specific case of the proposed many-to-one paraphrasing where \( n = 2 \); including X as the reward function for reinforced extractor (X is either ROUGE-L or the reward function defined in Eq. 5) and beam search.

Ideal extractor-abstractor model [Id – ext EXT – ABS]: To determine the performance of the abstractor component, the EXT – ABS baseline is evaluated with the assumption that the extractor is ideal, subsequently feeding the exemplary-extracted sentences generated for the test data using ROUGE-L score to the paraphraser directly.

Ideal extractor-paraphraser model [Id – ext O2O]: Similar to the Id – ext EXT – ABS setup, the ‘n-to-1’ paraphraser is evaluated with \( n = 1 \), given that the extractor performs ideally (generating exemplary-extracted sentences as proposed in Eq. 4).

Ideal extractor-paraphraser model [Id – ext M2On=2]: The ‘n-to-1’ paraphraser with \( n = 2 \) and the exemplary-extracted sentences assumed as the extractor (generating exemplary-extracted sentences as proposed in Eq. 4).

5 Results

Results of different baselines and the proposed approach are discussed in this section. Table 1 illustrates that our proposed techniques perform better than the rest of the systems. We have used ROUGE-1, ROUGE-2, ROUGE-L (Lin, 2004), METEOR (Banerjee and Lavie, 2005) and word mover similarity (WMS) (Clark et al., 2019) as evaluation metrics. We believe that ROUGE as an evaluation metric is incapable of judging the quality of an abstractive summary due to its emphasis on syntactic overlap over semantic overlap (Liu et al., 2016; Clark et al., 2019; Novikova et al., 2017). To overcome this, we have also used WMS as an evaluation metric.

5.1 Semantic information overlap

It is noticed that the proposed paraphraser in model O2O (w/o RL) outperforms the abstractor from EXT – ABS (w/o RL) in terms of ROUGE scores, while scoring marginally less in terms of METEOR. The extractor counterparts EXT – ABS (Ext only) and O2O (Ext only)
We stress that semantic information overlap is more significant than the syntactic overlap, we believe that WMS is better suited for the evaluation task as well. Examples of generated summaries for the $EXT - ABS$ and $O2O RL_{WMS}$ models are illustrated in Fig. 3. An important observation in the generated summaries is that the former model produces the name “shao li” which is not present in the input document or the gold standard summary, whereas this mistake is avoided by the $O2O RL_{WMS}$ model.

5.2 Summarization vs paraphrasing

Theoretically the $M2O_{n=2}$ setting should surpass the $O2O$ setting, since the former has extra input information at the paraphraser stage that the latter lacks. However, it is observed that this does not happen; in actuality, the $M2O_{n=2}$ model is outperformed by the $O2O$ model in all aspects (Table 1). After manual scrutiny of exemplary-extracted sentence pairs fed to the paraphraser and the generated sentences, it is observed that the expected accumulation of information does not take place. To quantify this observation, WMD based overlap of information is computed between the exemplary-extracted sentence and the generated sentence. It is discovered that on average, the more similar sentence has a WMD of 1.775 while the other one obtains a value of 2.894, illustrating the inability of the paraphraser to combine information across the two sentences into one.

Hence, we hypothesize that the PGN model intrinsically paraphrases one input sentence to generate the corresponding sentence in the generated summary, innately mimicking the extractor-paraphraser behaviour. An experiment is formulated to evince the truth of this proposed hypothesis. For this experiment, three different collections of documents are used: 1) ground truth summary, 2) summaries generated by PGN (See et al., 2017), and 3) summaries generated by the $EXT - ABS$ model (Chen and Bansal, 2018), all corresponding to the data in the test set. For every summary sentence, its semantic overlap (using WMD) is computed between the generated sentences, it is observed that the expected accumulation of information does not take place. To quantify this observation, WMD based overlap of information is computed between the exemplary-extracted sentence and the generated sentence. It is discovered that on average, the more similar sentence has a WMD of 1.775 while the other one obtains a value of 2.894, illustrating the inability of the paraphraser to combine information across the two sentences into one.

One critical observation is that the reward function introduces a bias in the evaluation process. It can be clearly observed from the fact that the model $EXT - ABS + RL_{ROUGE}$ obtains better ROUGE scores while it pales in comparison to $EXT - ABS + RL_{WMS}$ in terms of WMS. Since we stress that semantic information overlap is more significant than the syntactic overlap, we believe that WMS is better suited for the evaluation task as well as a better choice for the reward function.

\footnote{Here $X \in \{ROUGE, WMS\}$, which remains same when comparing the two models}

It is established by the fact that the models using WMS as the reward function attain comparable ROUGE scores as well (while the reverse is not true), indicating that incorporating semantic information can assist in capturing syntactic information as well. Examples of generated summaries for the $EXT - ABS$ and $O2O RL_{WMS}$ models are illustrated in Fig. 3. An important observation in the generated summaries is that the former model produces the name “shao li” which is not present in the input document or the gold standard summary, whereas this mistake is avoided by the $O2O RL_{WMS}$ model.
and second most similar sentences is the highest in the case of the PGN model (approx. 1.57), while the value is around 0.41 for the gold standard summaries. The curve for gold standard has a steady slope, where as the slope for PGN rises quickly from the first point to the second point, and then follows a constant growth afterwards, illustrating a very high similarity with the most similar sentence, thus endorsing the hypothesis that the PGN intrinsically paraphrases one input sentence to generate one sentence in final summary.

Gold Standard:

```plaintext
tong shao, 20, was an international student from china attending iowa state university. her body was found in the trunk of her car in iowa city on september 26. police believe it had been for three weeks. she died of blunt force trauma and asphyxiation. her boyfriend, xiangnan li, 23, was the last to see her, but flew to china on september 8, before shao was officially missing. according to tong’s father, an arrest warrant has now been issued. however li has disappeared.
```

EXT-ABS + RLROUGE:

```plaintext
police are issuing a warrant for the girl’s boyfriend. shao li, 23, was listed as a person of interest in the case. tong shao was a chemical engineering student at iowa state university. xiangnan li, 23, found murdered in september after going missing. the girl’s body was discovered in iowa last year. tong shao, 20, was found dead in her car for three weeks.
```

O2O + RLwms:

```plaintext
tong shao went missing in september 2014. police found her body stuffed in the trunk of her toyota camry. shao’s boyfriend, xiangnan li, 23, was listed as a person of interest. police are issuing a warrant for the girl’s boyfriend. a 20-year-old daughter was found murdered in iowa last year.
```

**Figure 3:** Example of generated summaries for $EXT−ABS + RL_{ROUGE}$ model, $O2O + RL_{WMS}$. The text in **red** denotes the novel information that is not present in the gold summary, the text in **green** denotes the information overlap that is present exclusively in the generated summary and the text in **blue** denotes the information covered in all three scenarios.

### 5.3 Error Analysis

As the results illustrate in Table 1, overall the one-to-one setting of the proposed n-to-one paraphraser performs better than the two-to-one setting, revealing a major blockade of the existing sequential frameworks - the incapability of combining multiple sentences into one, deviating from the ideal notion of ‘abstraction’. We also observe that the extractors outperform their corresponding ($w/o RL$) counterparts, and are competing with the complete models as well to some extent, portraying the difficulty of abstractive summarization over sentence extraction, along with the inability of current evaluation metrics like ROUGE and METEOR to focus on semantic information over syntactic subtleties. As elucidated by the ideal extractor experiments, the paraphraser has great potential for even achieving state-of-the-art results in the summarization task, provided the extractor works ideally. Our future work would focus on mollifying the loss introduced at the extraction step, and obtaining a better harmony in the extractor-paraphraser network as a whole.

### 6 Conclusion

In this paper, we propose the extractor-paraphraser model, that comprises of the n-to-one paraphraser and a novel word mover similarity based reward function. We show that the semantic overlap based techniques surpass the strong baselines that rely on syntactic information overlap. We also develop experiments to unveil that the state-of-the-art pointer-generator network (PGN) indeed paraphrases input sentences intrinsically, and is unable to merge two sentences into one agglomerate sentence when explicitly conditioned to do so, changing our perception of sequence-to-sequence abstractive summarization models. We show that the existing works are still nowhere close to mimicking a true human summary, portraying the difficulty of true abstraction over its simplified formulation of extracting and then paraphrasing.

**Acknowledgement:** Dr. Sriparna Saha would like to acknowledge the support of Early Career Research Award of Science and Engineering Research Board (SERB) of Department of Science and Technology, India to carry out this research.

**References**


