Abstract

Text summarization aims to condense long documents and retain key information. Critical to the success of a summarization model is the faithful inference of latent representations of words or tokens in the source documents. Most recent models infer the latent representations with a transformer encoder, which is purely bottom-up and thus does not capture long-distance context well. Also, self-attention-based models face the challenge of quadratic complexity with respect to sequence length. We propose a method to improve summarization models on these two aspects. Our method assumes a hierarchical latent structure of a document where the top-level captures the long range dependency at a coarser time scale and the bottom token level preserves the details. Critically, our method enables token representations to be updated in both a bottom-up and top-down manner. In the bottom-up pass, token representations are inferred with local self-attention to leverage its efficiency. Top-down correction is then applied to allow tokens to capture global context. We demonstrate the effectiveness on a diverse set of summarization datasets, including narrative, conversational, scientific documents and news. Our model achieves state-of-the-art performance on a wide range of long document summarization benchmarks, compared to recent efficient transformers. We show that our model can summarize an entire book and achieve competitive performance using 0.27% parameters and much less training data, compared to a recent GPT-3-based model. These results indicate the general applicability and benefits of the framework.

1 Introduction

An abstractive summarization system aims to generate a semantically coherent and linguistically fluent summary by conditioning on the document. The dominant approach for abstractive summarization is to use a Seq2Seq model (Sutskever et al., 2014) with an encoder-decoder architecture instantiated with either RNNs (Hochreiter and Schmidhuber, 1997) or transformers (Vaswani et al., 2017). In such a model, an encoder computes or infers latent representations of observed tokens (words or subwords) in a document, conditioning on which a decoder generates a summary. This paper studies the problem of how to compute informative latent representations, which in turn would improve summarization.

We propose a method which synergizes bottom-up computation with top-down computation while assuming a multi-scale latent structure of a document. In a multi-scale structure, higher-level variables (like those representing sentences, segments) model the document at a coarser time-scale and abstract away details, and are suitable for capturing long range dependency of the document; in contrast, lower-level variables (like those representing tokens) preserve details, and prevent the summary from losing key details (such as the name of an entity). In our method, the summary is generated by conditioning on token representations (low-level variables), similar to recent abstractive summarization models (Zaheer et al., 2020; Beltagy et al., 2020). There is however a critical difference. In our method, token representations are first bottom-up inferred and then top-down updated with high level representations, hence rendering low-level representations aware of global context. See Figure 1 for an overview of our method.

Multi-level models have been widely studied in modeling for images (Sønderby et al., 2016), speech (Mehri et al., 2016), and language (Chung et al., 2016). It is also not new in the summarization literature. Prior summarization research has explored hierarchical models (Cheng and Lapata, 2016; Nallapati et al., 2016; Zhang et al., 2019; Xu et al., 2020; Cohan et al., 2018; Ruan et al., 2022). These works focus on the bottom-up computation

---

1In this paper, “compute” and “infer” (and “computation” and “inference”) are used interchangeably.
Figure 1: An overview of the top-down transformer. Suppose a document with 7 tokens is the inputs to the model, as shown on the bottom left. The bottom-up inference is achieved with local self-attention ($N_1$ layers) as shown in the left panel. To initialize the top-level representations, we pool bottom-up-inferred token representations with either equal weights or adaptive weights (see Section 2.3 for details). Top-level representations are then updated with full self-attention ($N_2$ layers) to capture global context. They are then used to update bottom-up-inferred token representations, as shown in the middle panel. The final token representations are attended by the decoder to generate a summary. Note that inference is used in the sense of statistical inference for latent variables and does not imply no training.

in a hierarchical model, computing higher-level representations (e.g., sentences, paragraphs) based on lower-level representations (e.g., words). In contrast, our method emphasizes the combination of bottom-up, as done in prior works, and top-down where lower-level representations are updated and enriched with higher-level representations (see the middle panel in Figure 1). This design is critical for summarization which requires global context. As shown in our ablations, removing the top-down update undermines the summarization performance.

The proposed method is agnostic to the model architecture. Due to the dominance of transformer models in NLP (Chen et al., 2018; Zhang et al., 2020; Sun et al., 2019; Martin et al., 2020), we instantiate our method with a transformer-based model. There is a bottleneck of applying transformers to long documents, because its computational and memory cost has a quadratic dependency on the sequence length. This issue is especially critical for summarization since we are more interested in summarizing long documents since short ones can be quickly read through by humans. To address this issue, a large amount of prior works have been devoted to develop efficient transformers with sub-quadratic complexity (Wang et al., 2020; Child et al., 2019; Beltagy et al., 2020; Zaheer et al., 2020; Kitaev et al., 2020; Roy et al., 2021).

Our method provides a natural way to diminish this quadratic complexity issue. In the bottom-up computation, we use local self-attention where each token only attends the tokens within a local fixed-length window, and thus the complexity does not grow as a function of the input sequence length. The top-down correction for (local) token representations enables them to capture more global context, reducing the limitation of local attention. In prior works like Longformer (Beltagy et al., 2020), Bigbird (Beltagy et al., 2020), local attention is also used. Our method is different from these models in terms of how to inject global information to locally computed representations. Longformer and BigBird utilize a few global tokens which attend and are attended by all local tokens, whereas we use top-down correction. Our approach can better capture global information compared to prior models, as demonstrated by clear performance improvements over these models in our experiments.

In summary, our methods have two key components: (1) local attention in bottom-up computation and (2) top-down correction for locally-computed-token-representations by high level representations. The first component alleviates the computational and memory cost and allows our model to process long documents, and the second component injects global information to local tokens and improves summarization performance. We call our model as top-down transformer, to emphasize the importance of the top-down update. We evaluate the model on a diverse set of summarization benchmarks. They cover documents from a variety of domains, including news articles and scientific, conversational, and narrative documents, and of various lengths ranging from hundreds of words (e.g., a news article), several thousands to over ten thousands of words (e.g., a scientific paper, a book chap-
ter), to even over hundred thousands of words (e.g., an entire book). Across all long document datasets, our models achieve competitive or state-of-the-art performance. We also show that our model is able to summarize a whole book. Compared to Wu et al. (2021) using GPT-3 and requiring humans to extensively label data, our model achieves competitive performance on book summary with only 0.27% parameters and a small amount of publicly available data. The diverse and strong empirical results support the effectiveness and wide applicability of the proposed model.

Our contributions are summarized as follows: (1) we propose a method which combines bottom-up computation and top-down update for long document summarization; (2) we conduct extensive evaluations and achieve strong performance on various long document benchmarks; and (3) we adapt our method to the challenging task of summarizing an entire book and achieve GPT-3-level performance with only 0.27% parameters.

2 Methods

Figure 1 gives a graphical overview of the top-down transformer. We introduce its details in this section. Suppose a document has \( N \) tokens, \( t = \{t_i\}_{i=1}^{N} \). In our method, token representations are computed by combining top-down and bottom-up processes. This leads to effective and efficient inference for token representations. They are then attended by a decoder to generate a summary, as in a regular encoder-decoder transformer.

2.1 Bottom-Up Computation

In the bottom-up path, contextual embeddings of the tokens, \( \{e_i \mid e_i \in \mathbb{R}^{d_i}\}_{i=1}^{N} \), are computed with \( N_1 \) layers of local self-attention. In particular, each token \( t_i \) only attends to nearby tokens within a window of size of \( w \). The complexity is hence \( O(Nw) \), in contrast to \( O(N^2) \) for full self-attention models.

2.2 Top-Down Computation

The efficiency with local self-attention in the bottom-up path nevertheless comes with a limitation, that is, each \( e_i \) only captures the context within a local window instead of that of the whole document. To mitigate this issue, we propose a top-down update for token representations.

Consider a two-level multi-scale latent structure for a document. The lower level consists of token representations, \( \{e_i\}_{i=1}^{N} \), computed by the bottom-up computation. The top level consists of units at a coarser level. It is affordable to apply full self-attention at the top level due to its coarser granularity, allowing these top-level units to capture global document context. The self-attention mechanism for the top-level representations is the original multi-head self-attention proposed in Vaswani et al. (2017).

Denote the top level representations after self-attention update as \( \{s_j \mid s_j \in \mathbb{R}^{d_j}\}_{j=1}^{M} \) (see Section 2.3 for details on top-level representation initialization methods). We can then update the bottom-up-inferred token representations with the top-level representations. This is achieved with \( N_3 \) top-down computation layers, as illustrated by the middle panel in Figure 1. Each layer contains three transformations on \( \{e_i\} \): (1) token self-attention, (2) token-segment cross-attention, (3) feed-forward. (1) and (3) are the same as those in the bottom-up layers or regular self-attention layer with local attention. (2) implementing the cross-attention between the top and bottom levels is the critical operation. In particular, each \( e_i \) is updated with cross-attention,

\[
\tilde{e}_i = e_i + \text{LayerNorm}( \sum_{j=1}^{M} \alpha_{ij} f_v(s_j) ), \tag{1}
\]

\[
\alpha_{ij} = \frac{\exp (f_q(e_i)^T f_k(s_j) )}{\sqrt{d} \sum_{l=1}^{M} \exp (f_q(e_i)^T f_k(s_l) )} \tag{2}
\]

where \( f_q, f_k, \) and \( f_v \) indicate query, key, and value linear mappings, respectively. For notational clarity, Equation 1 only illustrates the case with a single attention head. In practice, we use multi-heads. The cross-attention operation injects global contextual information into bottom-up-inferred token representations, \( e_i \), and yields global-context-aware token representations, \( \tilde{e}_i \), conditioning on which a summary can be generated by a decoder.

To instantiate the top-down computation, we need to make two choices: (1) the number of top-levels above the token level and (2) the unit representation for each top-level. We choose to use one top level since it is sufficiently coarser to apply full self-attention for a wide range of long document benchmarks we experimented on. A natural choice for top level units is sentence, paragraph, and chapter, depending on the number top level considered. Such a choice however leads to complicated implementations and reduced scalability due to the varying length of these units. We hence choose a
simpler approach, where the top level consists of fixed-length segments of the documents. While we use a single top level, multiple top levels can be simply achieved with segments with increasingly coarser granularity.

In the top-down computation, segment-level self-attention has a complexity of $O(M^2)$, and token-segment cross-attention has a complexity of $O(NM)$. Thus, together with bottom-up inference, the complexity is $O(Nw + M^2 + NM)$. In practice, we use relatively small $w$ (window size) and $M$ (number of segments).

### 2.3 Pooling Methods

As aforementioned, we use a single top level, consisting of fixed-length segments. The segment representations are initialized by pooling token representations. Following the notation above, suppose a document is divided into $M$ segments, and the embedding of the $j$th segment is initialized as,

$$
s_{j}^{(0)} = \sum_{n=1}^{k} p_{n} e_{j \times d+n} \tag{3}
$$

where $k$ is the kernel size and $d$ is the stride. $p_{n}$ is the weight for the $n$th token. We introduce two approaches to compute the weights. The first method is average pooling (AvgPool) and hence $p_{n} = \frac{1}{k}$, which is simple and convenient. In the second approach, we leverage the reference summary to define the importance of each token to assign adaptive weights (AdaPool). Particularly, we learn an importance tagger with labels constructed with the reference summaries, which involves three steps:

1. Construct training labels for the importance tagger: (1) word lemmatization for document and reference words; (2) label a document word as important if it appears in the reference word list and is a non-stopword
2. Train a top-down transformer encoder with constructed labels as the importance tagger
3. Train the summarization model with oracle weights (i.e., constructed labels from Step 1) and test it with the adaptive importance weight assigned by the learned tagger

In our experiments, we also used OracleAdaPool where the weights are obtained from Step 1 with the reference summaries. Note that if $\{p_{n}\}_{n=1}^{k}$ does not form a valid probability distribution, $s_{j}$ can be computed with a normalized weight distribution within each pooling window as follows,

$$
\sum_{n=1}^{k} \exp(p_{n}) e_{j \times d+n} \tag{4}
$$

$\{s_{j}^{(0)}\}_{j=1}^{M}$ are updated with self-attention, yielding $\{s_{j}\}_{j=1}^{M}$, which are then used in top-down inference for token representations, as discussed in Section 2.2.

### 3 Experiments

#### 3.1 Overview

We thoroughly evaluate the proposed method on various summarization datasets. See Table 7 in the appendix for a summary of datasets used in the current work. Our model is first evaluated on two standard long document summarization benchmarks, PubMed and arXiv (Cohan et al., 2018). It outperforms various efficient transformers and other approaches and achieves state-of-the-art performance. Although we focus on long document summarization, models under our framework is also applicable to shorter documents. We test our model on CNN-Dailymail (See et al., 2017), the most widely used short summarization dataset. Compared to a full self-attention model, our model achieves competitive or better performance. Recently, a more challenging benchmark, SummScreen (Chen et al., 2021), is proposed, where summarization systems need to summarize TV show scripts. These documents convey plot events often indirectly and implicitly in dialogues, in contrast to news and scientific articles where statements follow a logical order and facts are offered explicitly. Moreover, a typical episode contains multiple subplots that proceed in parallel. Solving this benchmark thus requires a system to draw information from utterances spreading out through the entirety of the input and integrate them to a concise description. Our model outperforms strong baselines on this challenging benchmark by a significant margin. Another challenging dataset, BookSum (Kryściński et al., 2021), is also recently released. It covers books from the literature domain, including stories, plays, and novels. Similar to ScreenSum, it requires integrating plot events from indirectly expressed descriptions. A further challenge is to process long-form texts up to hundreds of pages or over 100,000 words. Our method does well on this challenge, achieving competitive or superior performance compared to
Table 1: Results on Scientific Articles. Best performance (no oracle) is in bold, and the second best is underlined.

| Model                  | PubMed R-1 | PubMed R-2 | PubMed R-L | arXiv R-1 | arXiv R-2 | arXiv R-L |
|------------------------|------------|------------|------------|-----------|-----------|-----------|
| Pegasus (568M)         | 44.21      | 16.95      | 38.83      | 44.21     | 16.95     | 38.83     |
| Dancer                 | 46.34      | 19.97      | 42.42      | 45.01     | 17.60     | 40.56     |
| TLM-I+E (460M)         | 42.13      | 16.27      | 39.21      | 41.62     | 14.69     | 38.03     |
| SSN-DM                 | 46.73      | 21.00      | 42.42      | 45.01     | 19.06     | 32.77     |
| BigBird (578M)         | 46.52      | 20.65      | 42.88      | 46.63     | 19.62     | 41.83     |
| Longformer (460M)      | 46.97      | 20.23      | 42.88      | 46.63     | 19.62     | 41.83     |
| LSH                    | 48.12      | 21.06      | 42.72      |           |           |           |
| TopDownFormer (AvgPool) (464M) | 48.34 | 21.40 | 44.22 | 48.67 | 20.70 | 43.91 |
| TopDownFormer (AdaPool) (464M) | 51.05 | 23.26 | 46.47 | 50.99 | 21.93 | 45.61 |
| TopDownFormer (OracleAdaPool) | 55.15 | 26.55 | 50.25 | 64.16 | 33.39 | 56.88 |

3.3 Scientific Documents

We first test the effectiveness of our framework on two widely used datasets based on scientific documents, PubMed and arXiv. They consist of long documents of length ranging from several thousands of words to over ten thousands words. Three variants of our model with various pooling weights are presented. AvgPool, AdaPool, and OracleAdaPool in Table 1 indicate average pooling, pooling with adaptive weights, pooling with adaptive weights determined by references, respectively (see Section 2.3 for more details).

The experiment results are displayed in Table 1. Pegasus (Zhang et al., 2020) is pretrained on a large-scale of dataset with a pretraining objective specifically designed for summarization. It uses a full self-attention encoder and thus has to truncate the source document due to the quadratic memory complexity. The summarization-oriented large-scale pre-training makes it a strong baseline. Dancer (Gidiotis and Tsoumakas, 2020) takes a divide-and-conquer approach in which the summary is divided into sections and each section is paired to the appropriate section of the document and the model is trained on short sequences and has a low memory requirement. This is a straightforward approach achieving strong performance.

TLM-I+E (Pilault et al., 2020) first extracts salient sentences and then uses a GPT-style model to generate a summary by conditioning on the introduction section and extracted sentences (instead of the whole document), thus reducing memory requirement. SSN-DM (Cui and Hu, 2021) is an extractive model and uses a sliding encoder to process segments of a document and a memory module to capture autoregressive dependency between segments. These two models bear similarities to our model in that they use a multi-scale structure. The extracted salient sentences in TLM-I+E can be considered a representation of the document at a coarser granularity since salient information
is retained. Instead of keeping the coarser representations in the latent space, TLM-I+E reads out them to the observed word space. In SSN-DM, the fixed-size memory module pooling information from each segments can also be considered a high level representation of the document. Despite these similarities, our model, synergizing bottom-up and top-down inference, clearly outperforms these prior models.

BigBird (Zaheer et al., 2020), Longformer (Beltagy et al., 2020), and LSH (Kitaev et al., 2020; Huang et al., 2021) are efficient transformers. BigBird based on Pegasus pre-training combines local attention, random attention tokens, and global attention tokens. LSH uses content-dependent sparse attention based on local sensitivity hashing. Longformer is closely related to our models. It uses the same local attention as in our bottom-up computation except it has an extra [CLS] token which is a global attention token. Longformer is also initialized from BART. The only difference is that our models compute token representations with both top-down and bottom-up processes, in contrary to pure bottom-up in Longformer. The clear performance improvement over Longformer and other efficient transformers indicates the effectiveness of the synergy of bottom-up and top-down computation.

3.4 Short Documents

| CNN-DailyMail | R-1 | R-2 | R-L |
|---------------|-----|-----|-----|
| BART (Reported) | 44.15 | 21.28 | 40.90 |
| BART (Re-eval) | 43.93 | 20.81 | 40.79 |
| TopDownFormer (AvgPool) | 44.32 | 21.03 | 41.40 |
| TopDownFormer (AdaPool) | 44.85 | 21.31 | 41.15 |
| TopDownFormer (OracleAdaPool) | 63.87 | 38.42 | 59.10 |

Table 2: Results on CNN-DailyMail. Best performance (no oracle) is in bold, and the second best is underlined.

To demonstrate the general applicability of the proposed framework, we show its effectiveness on short document summarization and compare it to full self-attention model. We hypothesize that although the bottom-up computation uses local self-attention, our method with the top-down correction would lead to competitive or better summarization performance.

Our model parameters are initialized from BART. Hence, BART with full self-attention forms a natural baseline, allowing for direct comparison. In the bottom-up inference, the local attention window size of our models is 256. As shown in Table 2, our models achieve slightly better performance, especially in terms of R-1 and R-L, than BART. It confirms our hypothesis that a synergy of bottom-up with local attention and top-down inference with global attention is effective and achieves on-par or better performance as full self-attention.

3.5 SummScreen

Scientific and news articles often require that facts are offered explicitly and statements follow a logical order, which might allow summarization models to exploit layout and stylistic biases. We next test the proposed method on a more challenging dataset, SummScreen, which requires a model to draw and integrate information from indirect expressions across a wide range of the document. SummScreen (Chen et al., 2021) provides two datasets, TVMegaSite and ForeverDreaming, collecting from two different TV show transcript websites. Each document is the transcript of a TV show episode and the summary is an associated recap.

Table 3 summarizes the results. Extractive oracle is an extractive method by extracting nearest neighbors based on Rouge scores. Longformer is an abstractive method and takes the whole document as input. Hybrid models first select salient sentences and then input them to BART. Our models outperform these strong baselines and even achieves comparable or superior performance than prior models having access to oracle information.

3.6 BookSum

BookSum (Kryściński et al., 2021) is another challenging dataset, consisting of books from the literature domain including stories, plays and novels. It includes examples on three levels of granularity with increasing difficulty: (1) paragraph-level with inputs with hundreds of words, (2) chapter-level, with inputs with several thousands or over ten thousands of words, (3) book-level, with inputs spanning up to hundreds of pages and over hundred thousands of words. The chapter-level examples have comparable lengths to other popular long-form summarization datasets such as PubMed, arXiv. We first test our models on the chapter level. The book-level summarization is extremely challenging. First, the number of examples (313 books) is limited. Second, a book is too long to fit in current models. We train our model in a curriculum and recursive way to address the two issues.
Table 3: Results on SummScreen. Best performance (no oracle) is in bold, and second best is underlined.

![Table 3](image)

Table 4: Results on BookSum Chapter Level. Best performance (no oracle) is in bold, and second best is underlined.

![Table 4](image)

### 3.6.1 Chapter Level

Table 4 displays the results. Kryściński et al. (2021) takes a divide-and-conquer approach to summarize chapters. They finetune BART, T5, and Pegasus on the paragraph level data and the chapter summary is obtained by concatenating the paragraph summary. This might miss the intra-paragraph context. Our models directly summarize the whole chapters and outperform these divide-and-conquer models. Efficient transformers, Longformer and BigBird, are also able to take in the whole chapters as inputs. But these bottom-up approaches clearly underperform our models.

### 3.6.2 Book Level

We first train a top-down transformer on chapter-level and then fine-tune it on book-level data. The inputs to the book-level model are (1) the concatenated chapter reference summaries in training or (2) the concatenated chapter summaries generated by the chapter-level model in testing. The chapter-to-book curriculum training is to mitigate the scarcity of book-level data. The recursive summarization of chapters and then books can be considered abstractive content selection applied to book data.

Table 5 summarizes the book-level results. The middle section shows the performance for the models with the divide-and-conquer approach (Kryściński et al., 2021), same as those for the chapter-level data. Wu et al. (2021) also attempts to summarize books using GPT-3 with reinforcement learning (RL) finetuning. The results are shown in third section in Table 5. Their method shares similarity with ours in that they decompose books into shorter sequences and train the model and summarize the text segments recursively. There are three differences between our approach and theirs. First, we train our model with the limited data from BookSum, while (Wu et al., 2021) requires human labelers to write summaries, which is highly costly. Second, our model has lower complexity, allowing it to takes in longer input. Thus, we only need to decompose the book one time (into chapters), in contrast to multiple recursive decomposition steps. Multiple recursive summarization steps is prone to accumulating errors. Third, GPT-3 uses bottom-up inference to infer token representations, in contrast to the synergy of bottom-up and top-down inference in our approach. The last two differences might account for our competitive performance using a much smaller model (0.46B vs. 175B) and less data.

### 3.7 Ablation Studies

Our method has two key components: (1) local attention in bottom-up computation, and (2) top-down update to inject global context. We conduct ablation studies on these two factors. All ablation experiments are performed with PubMed.

We first ablate top-down update (TDU). The results are summarized in Table 6. The first row shows the performance of the top-down transformer with top-down update via cross-attention and window size 1024, which is our final model. The second row shows the performance for a vari-
of top-down update. In this variant, to update the bottom-up inferred token representations, we concatenate the token representations with the corresponding top-level segment representations, in contrast to the cross-attention approach used in the final model. We can see a clear performance degradation, indicating the importance of the cross-attention-based top-down update. The third row displays the results without top-down update, and the decoder attends the bottom-up-inferred token representations to generate summaries. Compared to our final model, the performance is also degraded, suggesting the effectiveness of the top-down update.

The lower panel of Table 6 presents ablations on window size (WS) of local attention. As the window size increases, the performance on all metrics enhances. The effect is quite large when the window size is increased from 32 to 256. The effect becomes smaller after 256, but the model performance can still benefit from larger window size.

| Method                      | WS | R-1  | R-2  | R-L  |
|-----------------------------|----|------|------|------|
| TDU via cross-attention     | 32 | 46.30| 19.55| 42.21|
| TDU via concat              | 64 | 47.25| 20.37| 42.12|
| TDU via cross-attention     | 128| 47.44| 20.56| 43.35|
| TDU via cross-attention     | 256| 47.89| 21.06| 43.77|
| TDU via cross-attention     | 512| 48.08| 21.16| 44.05|

Table 6: Ablation studies of Top-Down Transformer. TDU: top-down update. WS: window size.

4 Related Work

Summarization Models Prior works have proposed extractive models (Nallapati et al., 2017; Cui and Hu, 2021), abstractive models (Nallapati et al., 2016; Zhang et al., 2020), and hybrid models combining extractive and abstractive methods (Gehrmann et al., 2018; Pilault et al., 2020), for text summarization. Although our model mostly follows the abstractive approach, it also has connections to the hybrid models. These models usually first extract salient sentences from the source document and then summarize the extracted sentences with an abstractive model. Extracted sentences can be viewed as a high level representation of the document, although it is the observed space but not in the latent space as in our framework. A continuous representations in the latent space facilitates end-to-end learning. Moreover, assigning importance weight with the importance tagger in our method resembles an extractive step in a hybrid model, and thus top down transformer with learned importance tagger can be considered a hybrid model.

Efficient Transformers Despite the effectiveness of transformers on a variety of tasks, its quadratic complexity with respect to the sequence length has limited its application to problems with long sequences. A large amount of works have attempted to address this limitation. A major line of work focuses on designing various sparse attention mechanisms. These works can be roughly categorized into two groups, depending on whether the sparsity pattern is content-dependent (Kitaev et al., 2020; Roy et al., 2021; Wang et al., 2021; Liu et al., 2021) or content-independent (Child et al., 2019; Beltagy et al., 2020; Ainslie et al., 2020; Zaheer et al., 2020). Our work is mostly related to content-independent sparse attention. A main assumption of content-independent sparse attention is that the context temporally and/or spatially proximate to the query token is more important, which is intuitively sensible and supported by empirical attention analysis (Child et al., 2019). Thus, a common sparse attention pattern is local attention, where each query token only attends to a neighborhood within a fixed temporal and/or spatial window. While this reduces the complexity to be linear, a model with only local attention cannot model long-range dependency. Prior works combine local attention with other attention patterns with wider or global receptive field such as dilated attention, random attention tokens, and global attention tokens (Beltagy et al., 2020; Zaheer et al., 2020). Our models also use local attention for its efficiency and leverage top-down inference to enable global-context awareness.

5 Conclusion

In this work, we propose a summarization method which combines bottom-up computation with top-down computation to improve token representation inference. In the bottom-up pass, token representations are inferred with local self-attention to capture global context. Our model achieves (1) state-of-the-art performance on a wide range of long document summarization benchmarks, and (2) competitive performance on summarizing whole books using 0.27% parameters and much less training data, compared to a recent GPT-3-based model. These results indicate the general applicability and benefits of the proposed
framework.
Limitations

In the current work, we only explore a model with a single top-level layer. It would be a fruitful research direction to study models with multiple layers, with growing level of abstraction. This might improve both the efficiency and performance of the current model, since long range dependency is mostly captured by higher-level layers and the window size at the low-level can be small.

References

Joshua Ainslie, Santiago Ontanon, Chris Alberti, Vaclav Cvicek, Zachary Fisher, Philip Pham, Anirudh Ravula, Sumit Sanghai, Qifan Wang, and Li Yang. 2020. ETC: Encoding long and structured inputs in transformers. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 268–284, Online. Association for Computational Linguistics.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. arXiv preprint arXiv:2004.05150.

Mia Xu Chen, Orhan Firat, Ankur Bapna, Melvin Johnson, Wolfgang Macherey, George Foster, Llion Jones, Mike Schuster, Noam Shazeer, Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Ziheng Chen, Yonghui Wu, and Macduff Hughes. 2018. The best of both worlds: Combining recent advances in neural machine translation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 76–86, Melbourne, Australia. Association for Computational Linguistics.

Mingda Chen, Zewei Chu, Sam Wiseman, and Kevin Gimpel. 2021. Summrscreen: A dataset for abstractive screenplay summarization. arXiv preprint arXiv:2104.07091.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 484–494, Berlin, Germany. Association for Computational Linguistics.

R. Child, Scott Gray, Alec Radford, and Ilya Sutskever. 2019. Generating long sequences with sparse transformers. ArXiv, abs/1904.10509.

Junyoung Chung, Sungjin Ahn, and Yoshua Bengio. 2016. Hierarchical multiscale recurrent neural networks. arXiv preprint arXiv:1609.01704.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

Peng Cui and Le Hu. 2021. Sliding selector network with dynamic memory for extractive summarization of long documents. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 5881–5891, Online. Association for Computational Linguistics.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4098–4109, Brussels, Belgium. Association for Computational Linguistics.

Alexios Gidiotis and Grigorios Tsoumakas. 2020. A divide-and-conquer approach to the summarization of long documents. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 28:3029–3040.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. Neural computation, 9(8):1735–1780.

Luyang Huang, Shuyang Cao, Nikolaus Parlialian, Heng Ji, and Lu Wang. 2021. Efficient attentions for long document summarization. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1419–1436, Online. Association for Computational Linguistics.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. In International Conference on Learning Representations.

Wojciech Kryściński, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir Radev. 2021. Booksum: A collection of datasets for long-form narrative summarization. arXiv preprint arXiv:2105.08209.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In Text Summarization Branches Out, pages 74–81, Barcelona, Spain, Association for Computational Linguistics.

Ye Liu, Jianguo Zhang, Yao Wan, Congying Xia, Lifang He, and Philip Yu. 2021. HETFORMER: Heterogeneous transformer with sparse attention for long-text
**extractive summarization.** In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 146–154, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7203–7219, Online. Association for Computational Linguistics.

Soroush Mehri, Kundan Kumar, Ishaan Gulrajani, Rithesh Kumar, Shubham Jain, Jose Sotelo, Aaron Courville, and Yoshua Bengio. 2016. Samplernn: An unconditional end-to-end neural audio generation model. arXiv preprint arXiv:1612.07837.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In Thirty-First AAAI Conference on Artificial Intelligence.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. arXiv preprint arXiv:1602.06023.

Jonathan Pilault, Raymond Li, Sandeep Subramanian, and Chris Pal. 2020. On extractive and abstractive neural document summarization with transformer language models. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 9308–9319, Online. Association for Computational Linguistics.

Aurko Roy, Mohammad Saffari, Ashish Vaswani, and David Grangier. 2021. Efficient content-based sparse attention with routing transformers. Transactions of the Association for Computational Linguistics, 9:53–68.

Qian Ruan, Malte Ostendorff, and Georg Rehm. 2022. Histruct+: Improving extractive text summarization with hierarchical structure information. arXiv preprint arXiv:2203.09629.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: summarization with pointer-generator networks. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Casper Kaæ Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaæ Sønderby, and Ole Winther. 2016. Ladder variational autoencoders. Advances in neural information processing systems, 29:3738–3746.
A Data Statistics
| Dataset                | # Docs. | # Input Words | # Summary Words | Domain       |
|------------------------|---------|---------------|-----------------|--------------|
| PubMed                 | 133K    | 3,224         | 214             | Scientific   |
| arXiv                  | 215K    | 6,913         | 292             | Scientific   |
| TVMegaSite             | 22.5K   | 6,420         | 380             | Conversational |
| ForeverDreaming        | 4.3K    | 7,605         | 113             | Conversational |
| BookSum-Chapter-Level  | 12K     | 5,102         | 505             | Narrative    |
| BookSum-Book-Level     | 436     | 112,885       | 1,167           | Narrative    |
| CNN-DM                 | 311K    | 906           | 63              | News         |

Table 7: Summarization Datasets. It shows the total number of documents, the average number of input words, the average number of summary words, and the domain for each dataset.
B Qualitative Examples
 PubMed Example #1: Reference

A new class of water-soluble C60 transfecting agents has been prepared using Hirschbingel chemistry and assessed for their ability to act as gene-delivery vectors in vitro. In an effort to elucidate the relationship between the hydrophobicity of the fullerene core, the hydrophilicity of the water-solubilizing groups, and the overall charge state of the C60 vectors in gene delivery and expression, several different C60 derivatives were synthesized to yield either positively charged, negatively charged, or neutral chemical functionalities under physiological conditions. These fullerene derivatives were then tested for their ability to transfect cells grown in culture with DNA carrying the green fluorescent protein (GFP) reporter gene. Statistically significant expression of GFP was observed for all forms of the C60 derivatives when used as DNA vectors and compared to the ability of naked DNA alone to transfect cells. However, efficient in vitro transfection was only achieved with the two positively charged C60 derivatives, namely, an octa-amino derivatized C60 and a dodeca-amino derivatized C60 vector. All C60 vectors showed an increase in toxicity in a dose-dependent manner. Increased levels of cellular toxicity were observed for positively charged C60 vectors, relative to the negatively charged and neutral vectors. Structural analyses using dynamic light scattering and size exclusion chromatography reveal interesting intermolecular correlations between the various derivatized C60 compounds, the C60 vector/dna complexes, their physical characteristics (aggregation, charge), and their transfection efficiencies. Recently, similar G80/C60-based compounds have demonstrated potential as advanced contrast agents for magnetic resonance imaging (MRI). Thus, the successful demonstration of intracellular DNA uptake and gene expression from DNA using C60 vectors suggests the possibility of developing analogous G80/C60-based vectors to serve simultaneously as both therapeutic and diagnostic agents.

PubMed Example #1: Model Output

| Table 8: Summary Samples for PubMed |

ArXiv Example #1: Reference

We study experimentally the thermal fluctuations of energy input and dissipation in a harmonic oscillator driven out of equilibrium, and search for fluctuation relations. We study transient evolution from the equilibrium state, together with non-equilibrium steady states. Fluctuations relations are obtained experimentally for both the work and the heat, for the stationary and transient eigenvalues. A stationary state fluctuation theorem is verified for the two prescriptions of the work, but a transient fluctuation theorem is satisfied for the work given to the system but not for the heat dissipated by the system in the case of linear forcing. Experimental observations on the statistical and dynamical properties of the fluctuations of the angle, we derive analytical expressions for the probability density functions of the work and the dissipated heat, and for their finite time corrections, both for a forcing linear in time and a forcing sinusoidal in time. From these expressions, we present some analytical derivations of fluctuation theorems based on experimental observations. The probability density function of the heat, agreement between experiments and our modeling is excellent.

ArXiv Example #1: Model Output

We study the fluctuations of energy and dissipation in a thermostated harmonic oscillator driven out of equilibrium, and search for fluctuation relations. We study transient evolution from the equilibrium state, together with non-equilibrium steady states. Fluctuations relations are obtained experimentally for both the work and the heat, for the stationary and transient eigenvalues. A stationary state fluctuation theorem is verified for the two prescriptions of the work, but a transient fluctuation theorem is satisfied for the work given to the system but not for the heat dissipated by the system in the case of linear forcing. Experimental observations on the statistical and dynamical properties of the fluctuations of the angle, we derive analytical expressions for the probability density functions of the work and the dissipated heat, and for their finite time corrections, both for a forcing linear in time and a forcing sinusoidal in time. From these expressions, we present some analytical derivations of fluctuation theorems based on experimental observations.

ArXiv Example #2: Reference

The pairing Hamiltonian constitutes an important approximation in many-body systems, it is exactly soluble and quantum integrable. On the other hand, the continuum single-particle density (cpd) contains information about the continuum energy spectrum. The question whether one can use the Hamiltonian with constant pairing strength for correlations in the continuum is still unanswered. In this paper we generalize the richardson exact solution for the pairing Hamiltonian including correlations in the continuum. The resonant and non-resonant continuum are included through the continuum single-particle level density (cspld) for the nucleus @sm@. The resonant correlations are made explicit by using the cauchy theorem. Low lying states with seniority zero and two are described in detail.

ArXiv Example #2: Model Output

The exact solution of the richardson pairing hamiltonian is obtained by including the resonant and non resonant continuum through the continuum single-particle level density (cspd). The ground states, which appear in the complex energy representation, provide the main contribution from the continuum; the inclusion of the continuum has been used to study the bound and unbound states. It is found that the continuum pairs (pair energies with positive real component) converge to the real part of the uncorrelated pair energy and they do not appear in complex conjugate partners: as a result the total energy of the system may be complex. From the exact solutions of the pairing and pairing-like hamiltonians one can then be used to investigate many-body correlations.

Table 9: Summary Samples for ArXiv

| Table 9: Summary Samples for ArXiv |

1281
Michelle MacLaren is no longer set to direct the first “Wonder Woman” theatrical movie. MacLaren left the project over “creative differences.” Movie is currently set for 2017.

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program when the incident happened in January. He was flown back to Chicago via air on March 20 but died on Sunday. Initial police reports indicated the fall was an accident but authorities are investigating the possibility that Mogni was robbed. His cousin claims he was attacked and thrown off a 40 ft bridge.

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program in Italy when the incident happened in January. He was flown back to Chicago via air ambulance on March 20, but he died on Sunday after falling off a 40 ft bridge in Rome in a suspected robbery attack in Rome. Police reports indicated the fall was an accident but authorities are investigating the possibility he was robbed.

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program in Italy when the incident happened in January. He was flown back to Chicago via air ambulance on March 20, but he died on Sunday after falling off a 40 ft bridge in Rome in a suspected robbery attack in Rome. Police reports indicated the fall was an accident but authorities are investigating the possibility he was robbed.

Table 1: Summary Samples for CNN-DailyMail

| CNN-DailyMail Example #1: Reference | Michelle MacLaren is no longer set to direct the first “Wonder Woman” theatrical movie. MacLaren left the project over “creative differences.” Movie is currently set for 2017. |
| CNN-DailyMail Example #2: Reference | Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program when the incident happened in January. He was flown back to Chicago via air on March 20 but he died on Sunday. Initial police reports indicated the fall was an accident but authorities are investigating the possibility that Mogni was robbed. His cousin claims he was attacked and thrown off a 40 ft bridge. |
| CNN-DailyMail Example #2: Model Output | CNN confirms that director Michelle MacLaren is leaving the “Wonder Woman” movie. The movie, starring Gal Gadot, is still set for release on June 23, 2017. |
| CNN-DailyMail Example #2: Model Output | Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program in Italy when the incident happened in January. He was flown back to Chicago via air ambulance on March 20, but he died on Sunday after falling off a 40 ft bridge in Rome in a suspected robbery attack in Rome. Police reports indicated the fall was an accident but authorities are investigating the possibility he was robbed. |

Table 10: Summary Samples for TVMegaSite

| TVMegaSite Example #1: Reference | TVMegaSite Example #2: Reference |
| **TVMegaSite Example #1: Model Output** | **TVMegaSite Example #2: Model Output** |
| **TVMegaSite Example #1: Model Output** | **TVMegaSite Example #2: Reference** |
| **TVMegaSite Example #2: Reference** | **TVMegaSite Example #2: Model Output** |

Table 1: Summary Samples for TVMegaSite

| TVMegaSite Example #1: Reference | TVMegaSite Example #2: Reference |
| TVMegaSite Example #1: Model Output | TVMegaSite Example #2: Model Output |
| TVMegaSite Example #1: Model Output | TVMegaSite Example #2: Reference |
| TVMegaSite Example #2: Reference | TVMegaSite Example #2: Model Output |

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program when the incident happened in January. He was flown back to Chicago via air on March 20 but he died on Sunday. Initial police reports indicated the fall was an accident but authorities are investigating the possibility that Mogni was robbed. His cousin claims he was attacked and thrown off a 40 ft bridge.

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program in Italy when the incident happened in January. He was flown back to Chicago via air ambulance on March 20, but he died on Sunday after falling off a 40 ft bridge in Rome in a suspected robbery attack in Rome. Police reports indicated the fall was an accident but authorities are investigating the possibility he was robbed.

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program in Italy when the incident happened in January. He was flown back to Chicago via air ambulance on March 20, but he died on Sunday after falling off a 40 ft bridge in Rome in a suspected robbery attack in Rome. Police reports indicated the fall was an accident but authorities are investigating the possibility he was robbed.

Table 10: Summary Samples for CNN-DailyMail

| TVMegaSite Example #1: Reference | TVMegaSite Example #2: Reference |
| TVMegaSite Example #1: Model Output | TVMegaSite Example #2: Model Output |
| TVMegaSite Example #1: Model Output | TVMegaSite Example #2: Reference |
| TVMegaSite Example #2: Reference | TVMegaSite Example #2: Model Output |

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program when the incident happened in January. He was flown back to Chicago via air on March 20 but he died on Sunday. Initial police reports indicated the fall was an accident but authorities are investigating the possibility that Mogni was robbed. His cousin claims he was attacked and thrown off a 40 ft bridge.

Andrew Mogni, 20, from Glen Ellyn, Illinois, had only just arrived for a semester program in Italy when the incident happened in January. He was flown back to Chicago via air ambulance on March 20, but he died on Sunday after falling off a 40 ft bridge in Rome in a suspected robbery attack in Rome. Police reports indicated the fall was an accident but authorities are investigating the possibility he was robbed.
Sydney finds a way to save Vaughn’s life while trying to track down Sark’s base of operations in order to obtain the antidote to a deadly virus that Sloane has been infected with. Meanwhile, Vaughn is in critical condition after being exposed to Rambaldi’s toxin, and Sydney is forced to make a dangerous deal with Sark in exchange for his life and getting rid of him. Will meets with a professor to find out if any of the CIA's standardized tests were ever used to recruit American children in the 1980’s, and discovers that in one tenth thousand first graders could pass the test.

The Doctor and Martha are trapped on board a space station in the Torgi system, where the crew are trying to prevent the ship from colliding with the sun. The Doctor uses the sonic screwdriver on Martha’s mobile phone to activate Universal Roaming Activation, which allows him to travel anywhere in space and time without interference from the ship’s computers. The Doctor begins being transported to the planet Earth, but he refuses to board the ship. He explains that he does not want to have any tea, but instead says that he likes drink it anyway. He asks him how long they have known each other, and he says that it has been ten years since he first met her. He has not had a day of rest since he has known her, and has toiled without repose or a day’s freedom ever since. He had a typhoid epidemic in the third week of Lent, and when he returned home he was almost fifty and is weary and irritable. He complains about his brother-in-law, Serebryakov, Serebryakov’s young second wife, Helen, and about how their visit has been a disaster. He discoursed volubly on the patterns of deforestation until he sees that Helen is uninterested. Helen insists she is interested, and offers him tea. She tells him that she is simply bored and life is too much for her to bear. In Act II, Serebryakov complains to Helen of how he is old and no one respects him. His querulous behavior only annoys Helen, who begs him to stop it. Serebryakov ignores her and bemoans how his life of scholarship somehow seems to be nothing to her. Helen joins him and tells him Serebryakov must see Astrov now; she wants her father to stop behaving like a child. The elder Marina confides to Helen that she loves Astrov, and Helen suggests that she say something to see if the doctor loves Sonya too. Sonya offers her permission for Helen to do this. Astrov and Helen meet to discuss the estate's future. He discusses modestly on the patterns of deforestation until he sees that Helen is interested. Helen insists she is interested but says they should talk about something else. She points blankly at Serebryakov, and asks him not to. She then moves in to seduce Helen, but she wants none of it. As he tries to kiss her, Helen enters the room with flowers. Helen is heralded by the situation of Vanya to tell her husband that they must stay alive. A moment later, Serebryakov and the others entertain and Serebryakov announces his idea to sell the estate because he and Helen need to afford a certain amount of money to buy a cottage in Finland. They also have to figure out what to do with the rest of the property. The estate is worth way too much money to make sure they can afford a certain amount of money to buy a cottage in Finland. They also have to figure out what to do with the rest of the property. The estate is worth way too much money.

The scene opens in a country house in the Russian countryside. Ivan, a young man, is sitting with his mother, who is knitting him a Christmas stocking. He tells her that he wants to show them to Helena and Sonia, and Helena asks him if he finds it interesting. Helena is sitting next to him, and tells him to look for Batushka's plantations every year. He wants to show them to Helena and Sonia, and Helena asks him if he finds it interesting. Helena is sitting next to him, and tells him to look for Batushka's plantations every year. He wants to show them to Helena and Sonia, and Helena asks him if he finds it interesting. Helena is sitting next to him, and tells him to look for Batushka's plantations every year. She tells him that she is simply bored and life is too much for her to bear. Astrov and Helen meet to discuss the estate's future. He discusses modestly on the patterns of deforestation until he sees that Helen is interested. Helen insists she is interested but says they should talk about something else. She points blankly at Serebryakov, and asks him not to. She then moves in to seduce Helen, but she wants none of it. As he tries to kiss her, Helen enters the room with flowers. Helen is heralded by the situation of Vanya to tell her husband that they must stay alive. A moment later, Serebryakov and the others entertain and Serebryakov announces his idea to sell the estate because he and Helen need to afford a certain amount of money to buy a cottage in Finland. They also have to figure out what to do with the rest of the property. The estate is worth way too much money to make sure they can afford a certain amount of money to buy a cottage in Finland. They also have to figure out what to do with the rest of the property. The estate is worth way too much money.
In his London studio, artist Basil Hallward puts the finishing touches on his latest portrait, that of a young man. Although Lord Henry, who is visiting with Basil, asks about the young man’s identity, Basil declines to answer, noting his preference for secrecy. Basil never intends to exhibit the painting, because if he did, it would bore the deepest feelings in his soul. However, Basil lets slip that the subject of the portrait is Dorian Gray, who shortly thereafter pays the two men a house call. Lord Henry immediately begins to influence Dorian, suggesting that he should treasure and guard his youth and beauty while he has them, because they will soon fade. Terrified of aging, Dorian wishes he could trade his soul to stay as young as he looks in the portrait; a short while later, he again wishes that he could stay young while the image in the painting aged. The portrait thus begins to take on a life-like existence; in fact, Basil’s threat to burn the portrait is likened to “sunder” and Basil confesses that he would prefer the company of the portrait to the real Dorian. Dorian falls in love with a young actress, Sibyl Vane, a woman he barely knows. She plays a different woman at each night’s performance, running the label of “genius” from Dorian, who is as smitten with her acting more than with her personality. They become engaged, much to the surprise of Lord Henry and Basil. The sweet, wholesome Sibyl discusses her engagement with her family. Because her mother is indebted to the theatre manager, Mr. Isaacs, for fifty pounds, she is against the marriage unless Dorian is wealthy; they do not know that he is. Sibyl’s angry brother, James, is leaving for Australia, but he vows to kill Dorian if he wrongs his sister in any way. James also confronts his mother about gossip he has heard—that his mother and deceased father never married, which Mrs. Vane admits is true. Dorian attends a performance of Sibyl’s with Lord Henry and Basil, but the performance is terrible. Sibyl tells Dorian she can no longer act, because she has shown her a beautiful reality. Dorian is disgusted by her poor acting, because her performances were what drew him to her; he dismisses her and returns home. To his surprise, the portrait shows marks of cruelty around the mouth, lines that do not show on Dorian’s face. He begins to suspect that his wish is coming true, so he vows to be good so that both he and the portrait can remain young. He, therefore, intends to apologize to Sibyl the next day and makes to marry her after all. However, he is too late: Sibyl commits suicide at the theatre that night. Dorian first feels responsibility for her death, but then views it both as wonderful entertainment and a selfish act on her part. Lord Henry tries to keep Dorian’s name out of the scandal. Dorian and Lord Henry spend the evening at the opera. The next morning, Basil arrives and expresses concern for Dorian, given the events of the previous day. Dorian, however, is completely unconcerned about Sibyl or her family; he wants to talk only of happy subjects. The next day, he covers his portrait and moves it to the attic, to which Dorian has the only key. Then he settles into to read a yellow book sent by Lord Henry: the book becomes Dorian’s Blackpintick for life. Several years pass, and Dorian lives a hedonistic life according to the guidelines established by Lord Henry and the yellow book. While the face in the portrait has turned ugly, Dorian remains young, beautiful, and innocent. People talk about Dorian’s “madness of pleasure” and his dreadful influence on the people around him, but that is of no consequence to him. Finally, when he is thirty-eight years old, Dorian shows the portrait to Basil, who begs Dorian to repent of his sin and asks that the wish be revoked. Instead, Dorian kills Basil and hides his body. Blackmailing his old friend Alan Campbell, Dorian is able to dispose of Basil’s body. An hour later, Dorian attends a party, but is bored and distracted. He then heads for an opium den and, out on the street, meets Sibyl’s younger brother, who has been waiting for an opportunity to harm Dorian for nearly twenty years. Dorian makes a case for mistaken identity when he claims to have the face of a twenty-year-old and cannot be the man James is looking for. A woman in the street reveals that Dorian “sold himself to the devil for a pretty face.” so James again pursues Dorian. At his country estate one week later, Dorian entreats guests but believes James in hunting him. Dorian soon learns, however, that a man accidentally killed in a hunting accident is James, and so he feels safe. The novel concludes six months later: Dorian and Lord Henry dine, and talk turns serious—Dorian talks of Basil, and Lord Henry reflects on a sermon he heard the previous Sunday while walking in the park. Lord Henry also inquires about the secret of Dorian’s youth, which Dorian dismisses. Then Dorian asks Lord Henry never to give the yellow book to anyone else. That evening, while Dorian examines the portrait, he decides to destroy it with the knife used to murder Basil. Soon after, Dorian’s servant and a police officer find an old, ugly man lying dead on the ground in front of a portrait of a young and innocent Dorian.

In his London studio, artist Basil Hallward puts the finishing touches on his latest portrait, that of a young man. Although Lord Henry, who is visiting with Basil, asks about the young man’s identity, Basil declines to answer, noting his preference for secrecy. Basil never intends to exhibit the painting, because if he did, it would bore the deepest feelings in his soul. However, Basil lets slip that the subject of the portrait is Dorian Gray, who shortly thereafter pays the two men a house call. Lord Henry immediately begins to influence Dorian, suggesting that he should treasure and guard his youth and beauty while he has them, because they will soon fade. Terrified of aging, Dorian wishes he could trade his soul to stay as young as he looks in the portrait; a short while later, he again wishes that he could stay young while the image in the painting aged. The portrait thus begins to take on a life-like existence; in fact, Basil’s threat to burn the portrait is likened to “sunder” and Basil confesses that he would prefer the company of the portrait to the real Dorian. Dorian falls in love with a young actress, Sibyl Vane, a woman he barely knows. She plays a different woman at each night’s performance, running the label of “genius” from Dorian, who is as smitten with her acting more than with her personality. They become engaged, much to the surprise of Lord Henry and Basil. The sweet, wholesome Sibyl discusses her engagement with her family. Because her mother is indebted to the theatre manager, Mr. Isaacs, for fifty pounds, she is against the marriage unless Dorian is wealthy; they do not know that he is. Sibyl’s angry brother, James, is leaving for Australia, but he vows to kill Dorian if he wrongs his sister in any way. James also confronts his mother about gossip he has heard—that his mother and deceased father never married, which Mrs. Vane admits is true. Dorian attends a performance of Sibyl’s with Lord Henry and Basil, but the performance is terrible. Sibyl tells Dorian she can no longer act, because she has shown her a beautiful reality. Dorian is disgusted by her poor acting, because her performances were what drew him to her; he dismisses her and returns home. To his surprise, the portrait shows marks of cruelty around the mouth, lines that do not show on Dorian’s face. He begins to suspect that his wish is coming true, so he vows to be good so that both he and the portrait can remain young. He, therefore, intends to apologize to Sibyl the next day and makes to marry her after all. However, he is too late: Sibyl commits suicide at the theatre that night. Dorian first feels responsibility for her death, but then views it both as wonderful entertainment and a selfish act on her part. Lord Henry tries to keep Dorian’s name out of the scandal. Dorian and Lord Henry spend the evening at the opera. The next morning, Basil arrives and expresses concern for Dorian, given the events of the previous day. Dorian, however, is completely unconcerned about Sibyl or her family; he wants to talk only of happy subjects. The next day, he covers his portrait and moves it to the attic, to which Dorian has the only key. Then he settles into to read a yellow book sent by Lord Henry: the book becomes Dorian’s Blackpintick for life. Several years pass, and Dorian lives a hedonistic life according to the guidelines established by Lord Henry and the yellow book. While the face in the portrait has turned ugly, Dorian remains young, beautiful, and innocent. People talk about Dorian’s “madness of pleasure” and his dreadful influence on the people around him, but that is of no consequence to him. Finally, when he is thirty-eight years old, Dorian shows the portrait to Basil, who begs Dorian to repent of his sin and asks that the wish be revoked. Instead, Dorian kills Basil and hides his body. Blackmailing his old friend Alan Campbell, Dorian is able to dispose of Basil’s body. An hour later, Dorian attends a party, but is bored and distracted. He then heads for an opium den and, out on the street, meets Sibyl’s younger brother, who has been waiting for an opportunity to harm Dorian for nearly twenty years. Dorian makes a case for mistaken identity when he claims to have the face of a twenty-year-old and cannot be the man James is looking for. A woman in the street reveals that Dorian “sold himself to the devil for a pretty face.” so James again pursues Dorian. At his country estate one week later, Dorian entreats guests but believes James in hunting him. Dorian soon learns, however, that a man accidentally killed in a hunting accident is James, and so he feels safe. The novel concludes six months later: Dorian and Lord Henry dine, and talk turns serious—Dorian talks of Basil, and Lord Henry reflects on a sermon he heard the previous Sunday while walking in the park. Lord Henry also inquires about the secret of Dorian’s youth, which Dorian dismisses. Then Dorian asks Lord Henry never to give the yellow book to anyone else. That evening, while Dorian examines the portrait, he decides to destroy it with the knife used to murder Basil. Soon after, Dorian’s servant and a police officer find an old, ugly man lying dead on the ground in front of a portrait of a young and innocent Dorian.

Dorian Gray is sitting in the garden of his uncle’s house in London, playing the piano. In the center of the room is a portrait of a young man of extraordinary beauty, and in front of it is a box containing a black book. Dorian goes to his bedroom and finds a small box of lacquer, which he takes out and puts inside. He puts the box back, gets into a horse-drawn carriage, and gives the driver an address. The driver takes him to the address, and as he is leaving the house, he sees the dead body of a man on the table. When Campbell returns, he tells Alan not to disturb the body, but to come back at seven o’clock in the evening. When the man arrives, he throws the picture over the table, but Dorian does not believe that it has been disturbed. He returns home and finds that Campbell has brought back the chemicals and the icons, and the other things that he needs to do the job. He opens the cabinet where he had hidden Basil’s coat and bag, and finds the green paste. At midnight, he gets a hansom and leaves the house with the instructions to meet him at 7 o’clock the next day. He sits in the back of the carriage as the driver drives him through the streets. He wonders if it is possible to cure the soul by means of the senses and the body by way of the soul. He wakes up in the middle of the night to find that the portrait has not changed.