Outline to Story: Fine-Grained Controllable Story Generation from Cascaded Events

Le Fang†, Tao Zeng§, Chaochun Liu†, Liefeng Bo§, Wen Dong†, Changyou Chen†
†University at Buffalo, §JD Finance America Corporation, AI Lab
{lefang, wendong, changyou}@buffalo.edu
{tao.zeng, chaochun.liu, liefeng.bo}@jd.com

Abstract

Large-scale pretrained language models have shown thrilling generation capabilities, especially when they generate consistent long text in thousands of words with ease. However, users of these models can only control the prefix of sentences or certain global aspects of generated text. It is challenging to simultaneously achieve fine-grained controllability and preserve the state-of-the-art unconditional text generation capability. In this paper, we first propose a new task named “Outline to Story” (O2S) as a test bed for fine-grained controllable generation of long text, which generates a multi-paragraph story from cascaded events, i.e., a sequence of outline events that guide subsequent paragraph generation. We then create dedicated datasets for future benchmarks, built by state-of-the-art keyword extraction techniques. Finally, we propose an extremely simple yet strong baseline method for the O2S task, which fine tunes pre-trained language models on augmented sequences of outline-story pairs with simple language modeling objective. Our method does not introduce any new parameters or perform any architecture modification, except several special tokens as delimiters to build augmented sequences. Extensive experiments on various datasets demonstrate state-of-the-art conditional story generation performance with our model, achieving better fine-grained controllability and user flexibility. Our paper is among the first ones by our knowledge to propose a model and to create datasets for the task of “outline to story”. Our work also instantiates research interest of fine-grained controllable generation of open-domain long text, where controlling inputs are represented by short text.

Introduction

Large-scale pretrained language models have shown thrilling generation capabilities to compose coherent and meaningful long text (Radford et al. 2019; Keskar et al. 2019; Zellers et al. 2019). However, in these models, users can only control the prefix or certain global aspects of generated text. Generation is also prone to deviate from the topic and wander to elsewhere freely. For more coherent generation, one may wish to pre-define the semantics flowing in each part, or even more explicitly control the words appearing in each paragraph. To this end, researchers face an open challenge to generate long text with fine-grained control, and simultaneously preserve the state-of-the-art unconditional text generation capabilities. Given the success of pre-training on a broad range of language processing tasks (Peters et al. 2018; Radford et al. 2018; Devlin et al. 2018), a dominant paradigm emerges to be pre-training a transformer (Vaswani et al. 2017) based language model on a super large unlabeled corpus, and fine-tuning the model on task specific supervised data. The paradigm sheds light on transfer learning to inherit capabilities of pre-trained models. Researchers (See et al. 2019; Ziegler et al. 2019) have accordingly studied the strength of massively pre-trained language models as long text generators and demonstrated their unparalleled advantages on context conditioning and generation quality. However, the need of fine-grained controllable generation is still not fulfilled, meaning that the control handle remains a single prefix or a certain global aspect such as sentiment.

In this paper, we propose a new task called “Outline to Story” (O2S) as a test bed of fine-grained controllable generation of long text. Here, story refers to open-domain multi-paragraph long text that is self-contained and semantically sound. Given a story outline as cascaded events, i.e., a sequence of events with one event per paragraph, where each event consists of a set of key words or phrases to appear in the corresponding paragraph, our task aims to generate a multi-paragraph story that is highly conditioned on and consistent with the outline (Table 1). Fine-grained controllable generation is therefore performed through designated outlines. The task is challenging comparing to trivial generation of short text considering the following facts.

- Firstly, we require the generated length to be much longer, i.e. from hundreds to one thousand words, to leverage the state-of-the-art text generation ability. Longer text leads to higher complexity and more flexibility in a broader space.
- Secondly, generated story should be highly conditioned on given cascaded events, while the events lack explicit con-

| Input: | Prompt (Optional) | Event_1 | Event_2 | Event_3 |
|-------|-------------------|---------|---------|---------|
| Output: | Paragraph_1 | Paragraph_2 | Paragraph_3 |

Table 1: Outline to Story. Prompt is optional but an useful addition to perform global control; each event is given to control the corresponding paragraph.
connections and details. The generation is more constrained given more control signals than a single prompt or prefix.

- Lastly, a successful model on the task should not only generate well to the purpose, but also be highly flexible and controllable in generation. For instance, we humans are flexible to write stories with partial or incomplete outlines. Accordingly, an ideal model on O2S should be easily adaptable to write with only a fixed beginning event and future events on demand.

In order to tackle the proposed task, we also create some dedicate datasets. When multi-paragraph narratives are generally available, large-scale human annotated outlines are expensive and formidable. We propose to use state-of-the-art keyword extraction techniques (Mihalcea and Tarau 2004; Rose et al. 2010) to extract a set of keywords from each paragraph and pair them with the paragraph accordingly. Note that how to get higher quality outlines is an independent and parallel research, which is out of the scope of this paper.

Furthermore, we propose an extremely simple yet strong baseline method for the O2S task. Inspired by the usage of control codes and artificial tokens in recent literatures (Keskar et al. 2019; Tsutsui and Crandall 2017) to specify desired features of generated text, we use special tokens as delimiters to connect outline events with paragraphs in an interleaving manner. This leads to an augmented sequence for each outline-story pair. We fine tune pre-trained language models with the original language modeling objective on augmented sequences to learn the functionality of special tokens and co-occurrence structures between events and stories. Our method of “fine-tuning with special tokens” (FIST) is featured with the following facts.

- Utilizing pre-trained language models enables cheap but powerful long text generation capability after transfer learning is performed.

- Fine-tuning on augmented sequences naturally grasps the conditioning relationship from outline to story and reserve the generation capabilities of a pre-trained model. The method does not introduce any new parameters except the latent representations of several special tokens. We also do not perform any architecture modification. Therefore, FIST has least architecture prerequisite, learning effort and model adaptation cost, while captures strong conditional dependencies.

- During the generation stage, the model itself is able to generate candidate outline events, and users reserve all the flexibility to modify, extend, and manipulate those events right before generating succeeding paragraphs. Users are free to start with incomplete outline. Actually it is all up to a user’s practical needs to decide the level of human supervision.

To summarize, our paper is among the first ones by our knowledge to study the “outline to story” task along with creating new datasets for the task. It fulfills the specific need of fine-grained controllable generation, where conditions are short text. Recently, we notice a concurrent work (Rashkin et al. 2020) that proposes a similar task and a dedicated architecture with memory mechanisms. Their method will be compared empirically with ours in this paper. Extensive experiments demonstrate that firstly, FIST achieves state-of-the-art conditional story generation performance; secondly, FIST has shown outstanding flexibility and controllability in the generation; lastly, FIST achieves comparable and even better metrics with its way much simpler design than (Rashkin et al. 2020). Our datasets and source code is publicly available.

### The Task and Data

#### New Task: Outline to Story

The “Outline to Story” (O2S) task aims to build strong conditional dependency between a given outline and a generated story. An outline consists of cascaded events, i.e., a sequence of events, each corresponding to a paragraph in the story. An event is formulated as a set of keywords or key phrases that summarize the main content of a paragraph. During training and generation, an outline is expected to be self-contained and follow a latent semantic flow. The challenge of O2S task is to connect the cascaded events and fill their gap with semantically and grammatically sound details. A high quality story output is expected to strongly condition on outlines and be fruitful in content. However, it may not necessarily use all the key phrases in the outline, since a novel, creative and coherent story following the given high level semantic flow is of interest, rather than an unique particular story.

#### Our Proposed Datasets

##### Raw stories

Our task requires datasets of paired human written outline and contentful long narratives. Unfortunately, existing public datasets rarely endow such a property. Although some tasks such as “data to text” (Chan et al. 2019) and “table to text” (Liu et al. 2019; Wang et al. 2020) consider similar conditional generation scenarios, the text in their datasets is generally too short and rigid. Those tasks emphasize faithful output from input data, rather than the generation capability of a model.

Neural story generation community has contributed several candidate datasets such as WritingPrompts (Fan, Lewis, and Dauphin 2018; Mao et al. 2019), WikiPlots and ROCStories (Mostafazadeh et al. 2016; Yao et al. 2019). However, ROCStories only consists of 5-lines stories that are too short for our task. For the other two datasets:

- WritingPrompts is a dedicated large-scale hierarchical story generation dataset collected from Reddit’s “WritingPrompts” forum. Based on a prompt as a rough guide or starting point, stories are multi-paragraph novels written by human users.

- WikiPlots contains plots about books, movies etc extracted from English language Wikipedia. Each plot is paired with a short title and given as one sentence per line without paragraphing. Therefore, the first pre-processing is to segment each plot into paragraphs. We map each sentence to a fixed-length vector representation using BERT (Devlin et al. 2018; Xiao 2018) and allocate adjacent sentences with less cosine proximity of representations into

#### Our Proposed Datasets

- **WritingPrompts**
- **WikiPlots**
- **ROCStories**

---

[1] https://github.com/fangleiai/Outline2Story
[2] https://github.com/markriedl/WikiPlots
different paragraphs. We also set a lower limit of 20 words per paragraph, since 20 words in average makes an English sentence according to linguist (Cutts 2020; Campaign 2004; Vincent 2014).

**Keyword Extraction** Without human annotated outlines, we use state-of-the-art keyword extraction techniques, such as TextRank (Mihalcea and Tarau 2004) and RAKE (Rose et al. 2010) to automatically extract outline from a multi-paragraph story. Specifically, we choose RAKE (Rapid Automatic Keyword Extraction), an unsupervised, domain-independent, and language-independent method, to extract key phrases from each paragraph. At least for English documents, RAKE is shown to be more computationally efficient than TextRank while achieving higher precision and comparable recall scores (Rose et al. 2010). We set minimum and maximum lengths of phrases as 1 and 4, respectively. We keep the number of extracted phrases linearly dependent on the paragraph length: it extracts one more key phrase for additional two sentences, which takes 40 words in average. These settings can be tuned as hyperparameters while the guideline is to achieve a balance between information completeness and generation creativity.

To summarize, an automatically extracted outline is expected to capture the key content and semantics flowing in a multi-paragraph story. Note that for WritingPrompts and WikiPlots, we keep the given prompt or title of a story as an optional global control signal and name it “the prompt” in the following paper. We summarize detailed dataset statistics in Table 2. The processed datasets will be released together with our source code for the community.

**Methodology**

At the core of our approach is the language modeling task, which is formulated as unsupervised distribution estimation of a sequence of tokens. Given a set of language examples, \((x_1, x_2, \cdots, x_k)\), each composed of variable length sequences of tokens \((s_1^k, s_2^k, \cdots, s_n^k)\), it is common to factorize the joint probabilities using the chain rule of conditional probabilities (Bengio et al. 2003):

\[
p(x) = \prod_{i=1}^{n} p(s_i | s_1, \cdots, s_{i-1})
\]

This decomposition allows both tractable sampling from and estimation of \(p(x)\) and any conditionals of the form \(p(s_{n-i} | s_1, \cdots, s_{n-i-1})\). Various models are trained as distribution approximators to minimize the negative log-likelihood over a dataset \(D = \{x_1, x_2, \cdots, x_{|D|}\}\):

\[
\mathcal{L}(D) = -\sum_{k=1}^{|D|} \sum_{i=1}^{n} \log p(s_i^k | s_{<i}^k)
\]

In recent years, large transformer-based models (Vaswani et al. 2017), especially the GPT-2 model (Radford et al. 2019), have shown superb capabilities to estimate the conditional probabilities owing to their self-attention architectures as powerful feature extractors. Considering the nature of languages in terms of sequential ordering and inherent co-occurrence, language modeling has been a fundamental task.

| Data set | Num. stories | Data split | Prompt avg. len. | Story avg. len. | Story avg. paragraphs | Event avg. phrases | Phrase avg. len. |
|----------|-------------|------------|-----------------|----------------|----------------------|------------------|-----------------|
| WP       | 303 K       | 90-5-5     | 25.4            | 674.5          | 6.3                  | 2.8              | 2.8             |
| WI       | 113 K       | 90-5-5     | 3.4             | 332.9          | 3.1                  | 3.3              | 3.1             |

Table 2: Statistics of two datasets. WP: WritingPrompts; WI: WikiPlots.
in the field.

In this paper, we highlight that language modeling provides a flexible way to cast various conditional distribution estimations of sequences into an unified estimation framework. A general supervised estimation task expressed as estimating \( P(\text{output}|\text{input}) \) is possibly solvable by language modeling on the augmented sequence of \((\text{input}, \text{output})\) connected by delimiters. Note that the key requirement is the sequential nature of both input and output, which is exactly the case for the “outline to story” task.

**Fine-tune with Special Tokens**

We adopt the common two-stage “pre-training + fine-tuning” paradigm. The pre-training exploits abundant unlabeled language corpus to model open-domain unconditional sequences. A continual language modeling task in fine-tuning leads to least modification on architecture and distribution space, which consequently ensures efficient model adaptation. Inspired by the usage of control codes and artificial tokens in recent literatures (Keskar et al. 2019; Tsutsui and Crandall 2017) to specify desired features of generated text, we use special tokens as delimiters to connect outline events with paragraphs. Consequently, each outline-story pair leads to an augmented sequence for fine-tuning.

Our method of “fine-tuning with special tokens” (FIST) naturally grasps the conditional relationship from outline to story and reserves the generation capabilities of the pre-trained model.

Take the GPT-2 (Radford et al. 2019) pre-trained model as an example. Like the token “<-endoftext>” to indicate the end of current article and the start of next article, special tokens will not be split during tokenization. Instead, they will be learned together with normal tokens and encode certain semantic meanings in a profound way. For each outline-story pair, FIST builds an augmented sequence as following:

1. For each outline event, *i.e.*, a set of key phrases, FIST builds an event sequence by separating key phrases with a “<-sepofcond>” token, prepending a “<-startofcond>” token to initiate the event, and appending a “<-endofcond>” token to halt the event. The optional global prompt will be initiated with a “<-startofcond>” token and put at the beginning of the first event sequence to interact with the whole content in the story.

2. For a list of event sequences and paragraphs, FIST joins them in an interleaving manner.

The architecture and example of augmented sequence are shown in Figure 1.

There are alternative ways to join a list of event sequences and paragraphs. As shown in Table 3, we advocate the best strategy in FIST to join them in an interleaving manner. Another possible way is to prepend all event sequences at the beginning and before all paragraphs. This has the advantage of exploiting all outline information starting from the very first paragraph. However, the disadvantages are also obvious. First, it weakens the paired and sequential correspondences between events and paragraphs; second, it implicitly requires the generation should have all outline events fixed before even generating the first word, hurting the generation flexibility and controllability. We will demonstrate in experiment section the benefits of FIST with an interleaving co-occurrence structure.

To summarize, special tokens are flexibly designed to accommodate various co-occurrence structures, whose representations are learned from scratch during the fine-tuning and play a significant role in the modeling of conditional dependence.

**Training and Generation**

With a pre-trained model, only a light fine-tuning on relatively small supervised data is needed. At the training stage, FIST performs language modeling on the outline-story augmented sequences; in generation, the story is sequentially decoded following the same format of augmented sequences.

Since event sequences are inherently modeled as part of the language, FIST model endows the ability to generate succeeding event sequences. During the generation stage, users reserve all flexibility to modify, extend, and manipulate those events right before generating succeeding paragraphs. For instance, users can replace a generated event sequence starting from “<-startofcond>” to “<-startofcond>” in place by another user written event sequence. FIST model may also wrap up a story indefinitely with less or more paragraphs than the provided length of outline, *i.e.*, the number of events. We will demonstrate through experiments the quality and characteristics of model generated outline events. It is all up to an user’s practical needs to decide the level of human supervision. Overall, FIST model keeps the highest level of generation controllability and flexibility to fulfill the promise of fine-grained controllable generation.

**Table 3: Two ways to build augmented sequence for each outline-story pair. Abbreviations follow that in Figure 1.**

| Input | Output |
|-------|--------|
| ST Prompt | SC E1 EC P1 |
| SC E2 EC P2 | SC E3 EC P3 |
| SC En EC Pn ET | SC E1 EC P1 |
| SC E2 EC P2 | SC E3 EC P3 |
| SC En EC Pn ET |

The interleaving manner:

The prepending manner:
Related Work

Controllable Text Generation

A number of previous studies of controllable text generation have focused on certain global aspects of the text. The most common aspects are sentiment and topic (Shen et al. 2017; Zhao et al. 2018; Hu et al. 2017; Fang et al. 2019; Dathathri et al. 2019; Keskar et al. 2019). Also, researchers attempt fine-grained control with plots, plans or the so-called storylines (Peng et al. 2018; Yao et al. 2019), leading to a wide usage and benchmark on 5-lines story dataset ROCStories (Mostafazadeh et al. 2016) and similar datasets with relatively short text. Later on, long story generation stands at the frontier (Fan, Lewis, and Dauphin 2018; 2019) of conditional and controllable generation. The task named as “neural story generation” is still under development and relatively under-explored so far.

So far, controllable generation with increasingly longer text and finer attributes has not been studied very well. This paper proposes the “outline to story” task to instantiate such research interest with controlled conditions as evolving short text. Recently, a concurrent work (Rashkin et al. 2020) proposes a similar task and a dedicated architecture with memory mechanisms integrated with pre-trained language models. This paper will benchmark both models and promote the application from outline to open-domain long stories.

Transfer Learning on Language Models

Language modeling (Bengio et al. 2003) have played an important role in natural language processing and understanding, especially when used as a pre-training technique. Word embeddings (Mikolov et al. 2013) and contextual word vectors (McCann et al. 2017; Howard and Ruder 2018; Peters et al. 2018) are some early but significant products. Recent works of large Transformer architectures (Vaswani et al. 2017; Radford et al. 2019; Devlin et al. 2019) have leveraged the power of both big models and big data to further improve language representation. A number of works also study different pre-training model bases (Song et al. 2019; Dong et al. 2019; Keskar et al. 2019; Lample and Conneau 2019; Zellers et al. 2019). In terms of generation tasks, the GPT-2 models attract huge attention due to its dedicated design for unconditional language generation. Researchers (See et al. 2019; Mao et al. 2019; Ziegler et al. 2019) have studied transfer learning on GPT-2 models to solve conditional generation tasks. For instance, (Ziegler et al. 2019) impressively introduces modification to self-attention architecture for adapting a pre-trained model to arbitrary conditional input that goes beyond text. However, we note that how to leverage pre-trained language models for fine-grained controllable generation of long text is still an open problem.

Experiments and Discussions

Experimental Settings

We first evaluate various models’ story generation capability; then evaluate and emphasize controllability and flexibility in generation, a.k.a. production or inference stage.

We conduct experiments on WritingPrompts and WikiPlots as introduced in “the Task and Data” section, which meet our target of open-domain long text corpora with paired outlines.

We notice that our model generating story from outline doesn’t necessarily need prompt as part of input. However, since conventional story generation does use prompt as input and the concurrent work (Rashkin et al. 2020) uses prompt as input to form outlines, we always conduct experiments using prompt as an useful addition to perform global control for a fair comparison between models.

We use the smallest public version GPT-2 as a pre-trained model base, which is a large auto-regressive transformer based language model trained on 40 GB of non-Wikipedia text (Radford et al. 2019). It is, by our knowledge, the most widely used model base in relevant literature. Note that there are many other pre-trained language models that may be larger and more powerful than our used one (Song et al. 2019; Dong et al. 2019; Keskar et al. 2019; Lample and Conneau 2019; Zellers et al. 2019). Our argument is that, the purpose of our experiment is not to compare powers of different pre-trained model bases. It’s orthogonal to this work to investigate different pre-trained model bases.

Benchmark Models

We compare models with designated purposes. By comparing with a specialized-architecture task-specific story generation model (Fan, Lewis, and Dauphin 2018), we evaluate the model’s in-domain generation performances. The fusion model in (Fan, Lewis, and Dauphin 2018) takes a convolutional seq2seq model structure with a fusion training mechanism.

By comparing with a concurrent work (Rashkin et al. 2020), we compare FIST with PlotMachines, a dedicated architecture using memory mechanisms for outline-conditioned story generation. Note that the benchmarked PlotMachines use the same GPT2 model base to construct its architecture.

By comparing with state-of-the-art transfer learning method based on GPT-2 models (Ziegler et al. 2019), we evaluate different ways to absorb input for conditional generation. The encoder-agnostic model adaptation in (Ziegler et al. 2019) has certain advantage of absorbing arbitrary conditional input other than text. Its key architecture is the pseudo self attention (PSA), which introduces new projection matrices to absorb input embeddings to the self-attention framework.

By comparing different inputs, we perform ablation study on input side to evaluate effect of outlines in conditional generation. One input source is only the prompt, whereas the other is “prompt + outline” connected with necessary delimiters.

By comparing with another way to build augmented sequence for fine-tuning, we perform ablation study on the FIST model side. The alternative way is the prepending manner shown in Table 3, while our FIST advocates to join cascaded events and paragraphs in an interleaving manner.

Implementation Details

We implement FIST using the “Huggingface Transformers” library in Pytorch (Wolf et al. 2019). Special tokens are conveniently added to the original 50K BPE GPT-2 vocabulary. All GPT-2 model settings remain the same. Other models are re-trained and re-evaluated with the same datasets using their own implementation repos-
Automatic and Qualitative Evaluation

Automatic Metrics We basically evaluate the following automatic metrics towards target stories:

- **Perplexity (PPL)** is used to evaluate language models and often regarded as a proxy for generation quality. All models based on GPT-2 use BPE tokenization scheme, where PPL values are not directly comparable with some previous models such as Keskar et al. [2019], where PPLs are computed at natural word level. Similarly to Keskar et al. [2019], we additionally compute word-level perplexity of GPT-2 models to enable the comparison with previous models. That is, we normalize the total negative log probability of the target text by the number of word level tokens. To ensure the fairness, when we evaluate perplexity on augmented sequences using FIST, we only count over story tokens, but not tokens of the event sequences, since other models only count over story tokens.

- **BLEU and ROUGE scores** are computed as n-gram overlap of generated stories versus target stories. We compute 4-gram BLEU using the “NLTK” library and ROUGE scores (ROUGE-1, ROUGE-2, and ROUGE-L) using Lin and Hovy [2002]. The ROUGE score includes both precision (P), recall (R), and F1, with ROUGE precision having similar interpretation as BLEU.

Automatic Evaluation Results The automatic evaluation results are presented in Table 4. Overall, the proposed FIST with full input source (prompt and outline) generally achieves better results, with lower PPL, higher BLEU and higher ROUGE scores, demonstrating state-of-the-art conditional story generation performance.

Methods based on pre-trained models, i.e. PSA / PlotMachines / FIST, show better overall performance with relatively less task specific efforts than Fusion models. This demonstrates the power of large-scale pre-training. Fusion models still show relatively high precision scores in setting models, presenting a much simpler approach based on solely language modeling rather than dedicated architecture. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?

With full input sources, the FIST model generally outperforms PSA, showing that simple language modeling on naturally co-occurred text conditions could grasp conditional dependence to some considerable extent. In other words, architecture modification is not always necessary with text conditions. The comparison so far on automatic metrics poses an interesting and open-ended question that, does dedicated architecture really benefit as expected?
The performance gap caused by two different input sources vary between different models: both Fusion models and PSA have limited performance improvement when provided with full inputs, while the FIST model is observed with a relatively larger performance gap. It is reasonable since the FIST model works directly and universally on augmented sequences, which is more sensitive and easier to reflect the enhancement of input information; while others “separate” input and story, and are less sensitive to the input.

The FIST model with interleaving joining mechanism shows a little performance drop comparing with the prependinger manner, due to the fact that the latter exposes more information along the decoding. We will see that the performance drop is a quite acceptable sacrifice, as FIST model wid interleaving joining mechanism achieves great decoding flexibility and controllability.

**Qualitative Evaluation** We also present some generation examples on the test datasets in Tables 5 and 9 in the Appendix. Stories are seen as semantically and grammatically sound, moreover, highly conditioned on and consistent with given outlines. Due to the pairing nature between extracted outlines and stories, the model presents obvious grasp on at least a lexical level. Tables 7, 8, 10, 11 are also good examples. A large scale human evaluation is underway, which is quite expensive due to text length and scale.

**Analysis on Automatically Generated Events** Since event sequences are inherently modeled as a part of “the language”, FIST model has its own ability to generate succeeding event sequences, which future story paragraphs and events are conditioned on. We perform event analysis under various decoding scenarios that involve different levels of event supervision and utilization.

- **FIST-α**: exactly as the test setting in previous evaluation section, the prompt and all the automatically extracted outline events are given following the chronological order.
- **FIST-ρ**: with the prompt and the first event sequence given, the following test events are also given, but randomly shuffled in order.
- **FIST-β**: with only the prompt and the first event sequence given, the model will depend on its own generated event sequence starting from the second paragraph till the end.

We compare automatic metrics of generated stories in different decoding scenarios versus target stories in Table 5.

| Dataset: WritingPrompts | Story | Event | Phrase | ROUGE-1↑ | ROUGE-2↑ | ROUGE-L↑ |
|-------------------------|-------|-------|--------|----------|----------|----------|
| FIST-α                  | 2.9   | 2.4   | 3.0    | 0.377    | 0.13     | 0.22     |
| FIST-ρ                  | 2.9   | 2.4   | 3.0    | 0.371    | 0.12     | 0.21     |
| FIST-β                  | 3.0   | 2.4   | 2.9    | 0.358    | 0.04     | 0.21     |

| Dataset: WikiPlots      | Story | Event | Phrase | ROUGE-1↑ | ROUGE-2↑ | ROUGE-L↑ |
|-------------------------|-------|-------|--------|----------|----------|----------|
| FIST-α                  | 2.9   | 2.6   | 3.3    | 0.333    | 0.21     | 0.239    |
| FIST-ρ                  | 2.9   | 2.6   | 3.3    | 0.325    | 0.20     | 0.229    |
| FIST-β                  | 2.9   | 2.5   | 3.2    | 0.321    | 0.06     | 0.229    |

Table 5: Automatic metrics for event analysis.

In this paper, we propose and create dataset for the task of “outline to story” to instantiate research interest of fine-grained controllable generation of open-domain long text with short-text controlled conditions. By leveraging pre-trained language models, our method FIST represents an improvement of “outline to story” to instantiate research interest of fine-grained controllable generation of open-domain long text with short-text controlled conditions. By leveraging pre-trained language models, our method FIST represents an improvement of "fine-grained controllable generation of open-domain long text with short-text controlled conditions."
References

Bengio, Y.; Ducharme, R.; Vincent, P.; and Jauvin, C. 2003. A neural probabilistic language model. Journal of machine learning research 3(Feb): 1137–1155.

Campaign, P. E. 2004. How to write in plain English. Plain English Campaign.

Chan, Z.; Chen, X.; Wang, Y.; Li, J.; Zhang, Z.; Gai, K.; Zhao, D.; and Yan, R. 2019. Stick to the Facts: Learning towards a Fidelity-oriented E-Commerce Product Description Generation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 4960–4969.

Cutts, M. 2020. Oxford guide to plain English. Oxford University Press, USA.

Dathathri, S.; Madotto, A.; Lan, J.; Hung, J.; Frank, E.; Molino, P.; Yosinski, J.; and Liu, R. 2019. Plug and Play Language Models: A Simple Approach to Controlled Text Generation. In International Conference on Learning Representations.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Dong, L.; Yang, N.; Wang, W.; Wei, F.; Liu, X.; Wang, Y.; Gao, J.; Zhou, M.; and Hon, H.-W. 2019. Unified language model pre-training for natural language understanding and generation. In Advances in Neural Information Processing Systems, 13042–13054.

Fan, A.; Lewis, M.; and Dauphin, Y. 2018. Hierarchical Neural Story Generation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 889–898.

Fan, A.; Lewis, M.; and Dauphin, Y. 2019. Strategies for structuring story generation. arXiv preprint arXiv:1902.01109.

Fang, L.; Li, C.; Gao, J.; Dong, W.; and Chen, C. 2019. Implicit Deep Latent Variable Models for Text Generation. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 3937–3947.

Holtzman, A.; Buys, J.; Du, L.; Forbes, M.; and Choi, Y. 2019. The Curious Case of Neural Text Degeneration. In International Conference on Learning Representations.

Howard, J.; and Ruder, S. 2018. Universal Language Model Fine-tuning for Text Classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 328–339.

Hu, Z.; Yang, Z.; Liang, X.; Salakhutdinov, R.; and Xing, E. P. 2017. Toward controlled generation of text. In Proceedings of the 34th International Conference on Machine Learning-Vol. 70, 1587–1596. JMLR. org.

Keskar, N. S.; McCann, B.; Varshney, L. R.; Xiong, C.; and Socher, R. 2019. Ctrl: A conditional transformer language model for controllable generation. arXiv preprint arXiv:1909.05858.

Lample, G.; and Conneau, A. 2019. Cross-lingual language model pretraining. arXiv preprint arXiv:1901.07291.

Lin, C.-Y.; and Hovy, E. 2002. Manual and automatic evaluation of summaries. In Proceedings of the ACL-02 Workshop on Automatic Summarization-Volume 4, 45–51. Association for Computational Linguistics.

Liu, T.; Luo, F.; Yang, P.; Wu, W.; Chang, B.; and Sui, Z. 2019. Towards Comprehensive Description Generation from Factual Attribute-value Tables. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 5985–5996.

Mao, H. H.; Majumder, B. P.; McAuley, J.; and Cottrell, G. 2019. Improving Neural Story Generation by Targeted Common Sense Grounding. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 5990–5995.

McCann, B.; Bradbury, J.; Xiong, C.; and Socher, R. 2017. Learned in translation: Contextualized word vectors. In Advances in Neural Information Processing Systems, 6294–6305.

Mihalcea, R.; and Tarau, P. 2004. Textrank: Bringing order into text. In Proceedings of the 2004 conference on empirical methods in natural language processing, 404–411.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems, 3111–3119.

Mostafazadeh, N.; Chambers, N.; He, X.; Parikh, D.; Batra, D.; Vanderwende, L.; Kohli, P.; and Allen, J. 2016. A corpus and evaluation framework for deeper understanding of commonsense stories. arXiv preprint arXiv:1604.01696.

Peng, N.; Ghazvininejad, M.; May, J.; and Knight, K. 2018. Towards controllable story generation. In Proceedings of the First Workshop on Storytelling, 43–49.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; and Zettlemoyer, L. 2018. Deep contextualized word representations. In Proceedings of NAACL-HLT, 2227–2237.

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018. Improving language understanding by generative pre-training. URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/languageunsupervised. pdf.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. OpenAI Blog 1(8): 9.

Rashkin, H.; Celikyilmaz, A.; Choi, Y.; and Gao, J. 2020. PlotMachines: Outline-Conditioned Generation with Dynamic Plot State Tracking. arXiv preprint arXiv:2004.14967.
Rose, S.; Engel, D.; Cramer, N.; and Cowley, W. 2010. Automatic keyword extraction from individual documents. *Text mining: applications and theory* 1: 1–20.

See, A.; Pappu, A.; Saxena, R.; Yerukola, A.; and Manning, C. D. 2019. Do Massively Pretrained Language Models Make Better Storytellers? In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 843–861.

Shen, T.; Lei, T.; Barzilay, R.; and Jaakkola, T. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in neural information processing systems*, 6830–6841.

Song, K.; Tan, X.; Qin, T.; Lu, J.; and Liu, T.-Y. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. In *International Conference on Machine Learning*, 5926–5936.

Tsutsui, S.; and Crandall, D. 2017. Using artificial tokens to control languages for multilingual image caption generation. *arXiv preprint arXiv:1706.06275*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Vincent, S. 2014. Sentence length: why 25 words is our limit. *Blog*, *InsideGOVUK*. Retrieved from: https://insidegovuk.blog.gov.uk/2014/08/04/sentence-length-why-25-words-is-our-limit*.

Wang, Z.; Wang, X.; An, B.; Yu, D.; and Chen, C. 2020. Towards Faithful Neural Table-to-Text Generation with Content-Matching Constraints.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; and Brew, J. 2019. HuggingFace’s Transformers: State-of-the-art Natural Language Processing. *ArXiv* abs/1910.03771.

Xiao, H. 2018. bert-as-service. https://github.com/hanxiao/bert-as-service.

Yao, L.; Peng, N.; Weischedel, R.; Knight, K.; Zhao, D.; and Yan, R. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 7378–7385.

Zellers, R.; Holtzman, A.; Rashkin, H.; Bisk, Y.; Farhadi, A.; Roesner, F.; and Choi, Y. 2019. Defending against neural fake news. In *Advances in Neural Information Processing Systems*, 9051–9062.

Zhao, J.; Kim, Y.; Zhang, K.; Rush, A.; and LeCun, Y. 2018. Adversarially regularized autoencoders. In *International Conference on Machine Learning*, 5902–5911. PMLR.

Ziegler, Z. M.; Melas-Kyriazi, L.; Gehrmann, S.; and Rush, A. M. 2019. Encoder-Agnostic Adaptation for Conditional Language Generation. *arXiv preprint arXiv:1908.06938*.
Appendix
You discover a grand hall filled with legendary weapons like Mjölnir and Excalibur. Each generation or so, warriors come to the hall to inherit a weapon that they are worthy enough to wield. Across the hall you see a forgotten weapon that’s been collecting dust. You hear it call to you.

A searing pain manifested in the back of Renard’s skull, jarring him awake. He found himself lying in a dusty forgotten hall of stone. "How did I get here?" He asked himself. But no memories were forthcoming. "Greetings child." The voice sent a chill down his spine and somehow relieved the ache in his head. "Who am I?" He asked loudly into the dimness. "Come find me and all shall be clear."

Far and away from the other weapons was a black horn. Simple and eerie. Without any confirmation he knew what must be done. He brought it to his lips and blew, one solid blast. Its mournful baritone echoed and somehow agitated the other weapons. Renard collapsed to the ground shrieking as he felt his body covered in invisible flames. His clothes burned away but no matter how much pain he felt the flames did not damage his body.

Hours, days, weeks, time had fallen to the wayside in the wake of this endless torment. Until suddenly it stopped. He rose, naked in the hall of legend, a new existence brought forth in pain. His name forgotten only the horn remained. "The age of man has reached its pinnacle." The horn whispered. "Go forth and bring cleansing. My King of Salt."

Clothing appeared from nonexistence to cover his form. With a mere flick of his wrist the wall broke and he found himself in London. "First comes destruction, I create nothingness." stunned Londoners stared at him. Woth a broad wave of his hand they disintegrated into piles of salt. He rose to the sky on blackened wings. "Hello child. You have reached the Hall of Greetings."

He saw several others storm out of the hall, holding the feeble weapons built in a bygone era to destroy him.

His voice boomed over the city like the crashing of thunder. "Hail the arisen king for the end of days is not as god prophesized. And cower for the land will be cleansed of the human disease."

---

Table 6: Generation examples on test dataset of WritingPrompts with FIST-α decoding.
You discover a grand hall filled with legendary weapons like Mjölnir and Excalibur. Each generation or so, warriors come to the hall to inherit a weapon that they are worthy enough to wield. Across the hall you see a forgotten weapon that’s been collecting dust. You hear it call to you.

A searing pain manifested from the mauve handle of my mighty sword and as it pierced the air, I knew I would be rewarded for my courage. I had come here at the wrong time. I looked around the dusty forgotten hall for any signs of the many who had come before me. I would have no fear. I would have a rest. A rest for my soul and the rest for my friends and family.

"Hello child. You have reached the Hall of Greetings."

I stared at my mother, looking over her shoulder at the men and women I had come to serve. They had come from many places. They had brought with them knowledge, from each great nation to its depths and it was all worth it. They were here to bring me freedom, and the next time, they would make it into the Hall of Salt... This time, however, my new existence brought forth a sadness. I wasn’t ready. I was ready to take a break. I didn’t want to go forth...

Suddenly, I saw several others storm from my little house, with weapons of all types and from all different stories. The first comes destruction, the second comes joy. I stepped out of my chair to find the first to be wounded and fallen, and the last to be killed by the first. As the horrified and stunned Londoners stared in shock and fear at the blade of the first, I heard the familiar voice of my mother speak to me. "I’m sorry. I didn’t mean to... But this is an honor and our journey isn’t over yet."

"This is your one chance. Please help me, I have lost the last of my friends. Let me rest, I want to protect the human disease."

| Table 7: Generation examples on test dataset of WritingPrompts with FIST-r decoding. |
Table 8: Generation examples on test dataset of WritingPrompts with FIST-f decoding.
To protect his friend, he is revealed to have been one of the original members of the terrorist organisation, and forced to work together to defend themselves, killing their attackers. Meanwhile, Coulson and Lincoln search one of Blake's old safe houses and arrest him. The agents capture him, but not before his brothers work together to try to stop him. They are able to stop him and he is able to make it out of the building. As the agents prepare to take his place, he escapes into a field where his usual desires. Ruben was recently made redundant and is struggling financially, blaming the government for his problems and sympathising with the Watchdogs. Trying to track down Andrew, May enlists Simmons, who suggests that as Andrew is becoming more animalistic, he is running on base instincts and not his usual desires. She also advises against May’s intended plan to kill Andrew, believing they can still devise a cure for Terrigenesis before he becomes Lash permanently. Tracing the Watchdogs’ online activity, Daisy and Fitz track down one of their non-radical members and coerce him into revealing the location of the Watchdogs’ local headquarters, at a farm. Daisy, Fitz and Mack spy on the farm, but Ruben arrives, intending to apologise to Mack, and blows their cover, forcing the agents to defend him from Watchdogs. In the ensuing fight Fitz is shot with a nitramene gel projectile, but Daisy stops the implosion by freezing the gel with liquid nitrogen. Following Ruben home, Mack admits he works for SHIELD. They are attacked by five Watchdogs, who saw Daisy’s powers in use at the farm but believed Mack to be the Inhuman. The brothers work together to defend themselves, killing their attackers. Meanwhile, Coulson and Lincoln search one of Blake’s old safe houses and find him there. Blake espouses his hatred of Inhumans and his belief that SHIELD will always use dangerous alien elements to spread terror, and Lincoln tries to kill him with Coulson’s permission, only to find ‘Blake’ is a hologram.

Mack gets into an argument with Ruben, who is unaware that Mack is a SHIELD agent and resents him for barely being present in his life and for helping care for their elderly parents. Ruben is recently made redundant and is struggling financially, blaming the government for his problems and sympathising with the Watchdogs. Trying to track down Andrew, May enlists Simmons, who suggests that as Andrew is becoming more animalistic, he is running on base instincts and not his usual desires. She also advises against May’s intended plan to kill Andrew, believing they can still devise a cure for Terrigenesis before he becomes Lash permanently. Tracing the Watchdogs’ online activity, Daisy and Fitz track down one of their non-radical members and coerce him into revealing the location of the Watchdogs’ local headquarters, at a farm. Daisy, Fitz and Mack spy on the farm, but Ruben arrives, intending to apologise to Mack, and blows their cover, forcing the agents to defend him from Watchdogs. In the ensuing fight Fitz is shot with a nitramene gel projectile, but Daisy stops the implosion by freezing the gel with liquid nitrogen. Following Ruben home, Mack admits he works for SHIELD. They are attacked by five Watchdogs, who saw Daisy’s powers in use at the farm but believed Mack to be the Inhuman. The brothers work together to defend themselves, killing their attackers. Meanwhile, Coulson and Lincoln search one of Blake’s old safe houses and find him there. Blake espouses his hatred of Inhumans and his belief that SHIELD will always use dangerous alien elements to spread terror, and Lincoln tries to kill him with Coulson’s permission, only to find ‘Blake’ is a hologram.

Table 9: Generation examples on test dataset of WikiPlots with FIST-α decoding.
### Target

**Prompt:** Watchdogs (Agents of S.H.I.E.L.D.)

**Event:** 
- militant terrorist organisation dedicated
- improve nitramene weaponry
- reluctantly leaves

**Paragraph:** The Watchdogs, a militant terrorist organisation dedicated to eradicating the Inhumans, destroy an Indiana ATCU facility using nitramene in gel projectiles. Mack, who is in the area visiting his brother Ruben, reluctantly leaves and joins Daisy and Fitz to investigate, and they inform Coulson of the use of nitramene, leading him to suspect the Watchdogs are being led by former SHIELD agent Felix Blake, who tried to improve nitramene weaponry in the past.

**Event:** 
- recently made redundant
- old safe houses
- brothers work together

**Paragraph:** Mack gets into an argument with Ruben, who is unaware that Mack is a SHIELD agent and resents him for barely being present in his life and for helping care for their elderly parents. Ruben was recently made redundant and is struggling financially, blaming the government for his problems and sympathising with the Watchdogs. Trying to track down Andrew, May enlists Simmons, who suggests that as Andrew is becoming more animalistic, he is running on base instincts and not his usual desires. She also advises against May’s intended plan to kill Andrew, believing they can still devise a cure for Terrigenesis before he becomes Lash permanently. Tracing the Watchdogs’ online activity, Daisy and Fitz track down one of their non-radical members and coerce him into revealing the location of the Watchdogs’ local headquarters, at a farm. Daisy, Fitz and Mack spy on the farm, but Ruben arrives, intending to apologise to Mack, and blows their cover, forcing the agents to defend him from Watchdogs. In the ensuing fight Fitz is shot with a nitramene gel projectile, but Daisy stops the implosion by freezing the gel with liquid nitrogen. Following Ruben home, Mack admits he works for SHIELD. They are attacked by five Watchdogs, who saw Daisy’s powers in use at the farm but believed Mack to be the Inhuman. The brothers work together to defend themselves, killing their attackers. Meanwhile, Coulson and Lincoln search one of Blake’s old safe houses and find him there. Blake espouses his hatred of Inhumans and his belief that SHIELD will always use dangerous alien elements to spread terror, and Lincoln tries to kill him with Coulson’s permission, only to find ‘Blake’ is a hologram.

**Event:** 
- team later learn

**Paragraph:** The team later learn that the Watchdogs were directed to the ATCU facility by Malick, and destroyed the building to cover up the theft of a nuclear weapon, which Blake (a paraplegic after being attacked by Mike Peterson) delivers to Giyera in exchange for more powerful weapons.

### Generation

| Prompt | Watchdogs (Agents of S.H.I.E.L.D.) |
| --- | --- |
| Event | militant terrorist organisation dedicated improve nitramene weaponry reluctantly leaves |
| Paragraph | In Los Angeles, S.Inspector Howerd and dr George Bailey are called to investigate a terrorist organisation dedicated to gaining control over the world. While trying to improve nitramene weaponry, the agents are helped by S.Wade who reluctantly leaves her due to the death of her daughter. While on the rooftop, Bailey gets a call from Amanda who claims the deaths are her fault but S.Wade informs her he has become too attached to her daughter and has decided to destroy the organization. |
| Event | team later learn powerful weapons |
| Paragraph | The team later learn of Amanda’s betrayal, and after an altercation they decide to kill both her and the leader of the group. After discovering the group is still active, they decide to destroy the group while keeping their powers a secret. |

Table 10: Generation examples on test dataset of WikiPlots with FIST-$r$ decoding.
Paragraph:
The Watchdogs, a militant terrorist organisation dedicated to eradicating the Inhumans, destroy an Indiana ATCU facility using nitramene in gel projectiles. Mack, who is in the area visiting his brother Ruben, reluctantly leaves and joins Daisy and Fitz to investigate, and they inform Coulson of the use of nitramene, leading him to suspect the Watchdogs are being led by former SHIELD agent Felix Blake, who tried to improve nitramene weaponry in the past.

Paragraph:
Mack gets into an argument with Ruben, who is unaware that Mack is a SHIELD agent and resents him for barely being present in his life and for helping care for their elderly parents. Ruben was recently made redundant and is struggling financially, blaming the government for his problems and sympathising with the Watchdogs. Trying to track down Andrew, May enlists Simmons, who suggests that as Andrew is becoming more animalistic, he is running on base instincts and not his usual desires. She also advises against May’s intended plan to kill Andrew, believing they can still devise a cure for Terrigenesis before he becomes Lash permanently. Tracing the Watchdogs’ online activity, Daisy and Fitz track down one of their non-radical members and coerce him into revealing the location of the Watchdogs’ local headquarters, at a farm. Daisy, Fitz and Mack spy on the farm, but Ruben arrives, intending to apologise to Mack, and blows their cover, forcing the agents to defend him from Watchdogs. In the ensuing fight Fitz is shot with a nitramene gel projectile, but Daisy stops the implosion by freezing the gel with liquid nitrogen. Following Ruben home, Mack admits he works for SHIELD. They are attacked by five Watchdogs, who saw Daisy’s powers in use at the farm but believed Mack to be the Inhuman. The brothers work together to defend themselves, killing their attackers. Meanwhile, Coulson and Lincoln search one of Blake’s old safe houses and find him there. Blake espouses his hatred of Inhumans and his belief that SHIELD will always use dangerous alien elements to spread terror, and Lincoln tries to kill him with Coulson’s permission, only to find ‘Blake’ is a hologram.

Paragraph:
In Los Angeles, S.Inspector Howerd and dr George Bailey are called to investigate a terrorist organisation dedicated to gaining control over the world. While trying to improve nitramene weaponry to enslave Simmon’s, the agents are helped by S.Wade who reluctantly leaves her due to the death of her daughter. While on the rooftop, Bailey gets a call from Amanda who claims the deaths are her fault but S.Wade informs her he has become too attached to her daughter and has decided to destroy the organization.

Paragraph:
The team later learn that the Watchdogs were directed to the ATCU facility by Malick, and destroyed the building to cover up the theft of a nuclear weapon, which Blake (a paraplegic after being attacked by Mike Peterson) delivers to Giyera in exchange for more powerful weapons.