Abstract

Generating a long, coherent text such as a paragraph requires a high-level control of different levels of relations between sentences (e.g., tense, coreference). We call such a logical connection between sentences as a paragraph flow. In order to produce a coherent flow of text, we explore two forms of inter-sentential relations in a paragraph: one is a human-created linguistical relation that forms a structure (e.g., discourse tree) and the other is a relation from latent representation learned from the sentences themselves. Our two proposed models incorporate each form of relations into document-level language models: the former is a supervised model that jointly learns a language model as well as discourse relation prediction, and the latter is an unsupervised model that is hierarchically conditioned by a recurrent neural network (RNN) over the latent information. Our proposed models with both forms of relations outperform the baselines in partially conditioned paragraph generation task. Our codes and data are publicly available.

1 Introduction

When composing multiple sentences into a paragraph, as in novels or academic papers, we often make design decisions in advance (Byrne, 1979) such as topic introduction and content ordering to ensure better coherence of the text. For instance, McKeown (1985); Swan (2002) proposed effective patterns for scientific writing: a hypothesis at first, followed by supporting sentences to validate the hypothesis, and lastly a concluding sentence. We call such a logical connection between sentences in a written paragraph as a flow. A coherent flow between sentences requires an understanding of various factors including tense, coreference, plans (Appelt, 1982; Hovy, 1991), scripts (Tomkins, 1978) and several others. We focus on the paragraph-level plan between sentences.

In text planning, underlying relations in text are broadly categorized into two forms: an explicit human-defined relation (e.g., a discourse tree) (Reiter and Dale, 2000) or an implicitly learned latent relation (Yang et al., 2016). While the former is defined and manually annotated based on linguistic theories, the latter is simply determinable from how people in fact put sentences together. In this work, we provide an empirical comparison between a linguistically-informed and a latent form of relations in context of a paragraph generation.

We compare the effectiveness of the two forms of relations using language modeling for paragraph generation. Due to the different characteristics of the two forms, we employ comparable but different components in addition to the base language model. For linguistic relations (e.g., discourse), we cast the problem into multi-task learning of supervised language modeling and discourse relation prediction. On the other hand, for latent relations, we learn an unsupervised hierarchical language model that is hierarchically conditioned by RNNs over linear operations between sentences.

We evaluate our models on partial paragraph generation task; producing the rest of text in a paragraph given some context of text. We observe that linguistically annotated discourse relations help produce more coherent text than the latent relations, followed by other baselines.

2 Related Work

There has been a variety of NLG systems that incorporate additional information between sentences (Appelt, 1982; Reiter and Dale, 2000; Gatt and Krahmer, 2018) which can be broadly categorized into two forms: linguistic and latent.
Linguistic relations are explicitly represented as external labels in the form of predefined rules or plans, formats, knowledge base, discourse parses, and more. Hovy (1985, 1990); Dalianis and Hovy (1996) integrated text planning in generation, where the plans are considered in knowledge, formatted rules and so forth. However, they are limited to small scale (i.e., few examples) and hand-written rules. Kang et al. (2017); Gardent et al. (2017); Kang et al. (2018b); Wang et al. (2018) used an external knowledge base to micro-planning for generating a corresponding text, while our work focuses on comparing two forms of relations from the text itself.

Moore and Paris (1993); Young and Moore (1994) utilized discourse structures such as rhetorical structure theory (RST) (Mann and Thompson, 1988) for parsing a document. A script (Tomkins, 1978) is another structured representation that describes a typical sequence of events in a particular context. Zhang et al. (2016); Ji and Eisenstein (2014) proposed better discourse parsers using neural networks. The prior works, however, used the discourse representations to describe the structure of the paragraph, while we focus on applicability of the discourse relations to language generation.

Latent relations use implicit information in a document such as hierarchical structure of the document: Lin et al. (2015); Chung et al. (2016) used hierarchical RNN for modeling a document. Similarly, the hierarchical model can be extended to other variants such as attention (Yang et al., 2016), encoder-decoder framework (Serban et al., 2017; Sordoni et al., 2015), auto-encoding (Li et al., 2015), and multiscale (Chung et al., 2016). However, the hierarchical recurrence of sentences, which is dependent on topics, are less likely modeling a flow of a document.

We further summarize the fundamental differences between the two forms of relations in Appendix.

3 FlowNet: Language Modeling with Inter-sentential Relations

We propose language models that incorporate each relation to capture a high-level flow of text.

3.1 Discourse-driven FlowNet

As a linguistic relation, we employ RST (Mann and Thompson, 1988) trees to represent discourse connections in the text. For simplicity, we limit usage of the discourse trees by only considering relations between adjacent phrases; relations are inserted between adjacent phrases and represented as a flattened sequence of phrases and relations. If two consecutive RST relations are given, the deeper level of relation is chosen. If the central elementary discourse unit (EDU) or phrase is after its dependent, the relation is excluded. We consider each sequence of the flattened discourse relations as a writing flow. For example, people often write a text by elaborating basic information (Elaboration) and then describing a following statement attributed to the information (Attribution).

We view discourse relations as additional labels to predict at the same time we predict next words in language modeling. Specifically, we propose to jointly train a model that predicts a sequence of words and a sequence of RST labels by taking advantage of shared representations, following previous sequence labeling problems such as named entity recognition (Collobert et al., 2011) and part-of-speech tagging (Huang et al., 2015). Note that the RST relations are only used during training to obtain better representation for the two tasks, but not at test time.

Figure 1(a) shows our FlowNet using discourse relations. Let a paragraph be a sequence of sentences $D = \{s_1, s_2, ..., s_M\}$. This model treats adjacent sentences as pairs for learning the standard seq2seq model. The first objective is to maximize the likelihood of the current sentence given the previous sentence. Hence, we maximize the following:

$$
\log P(s_t|s_{<t}, s_{t-1})
$$

where $s_i = \{w_{i1}, w_{i2}, \ldots, w_{iT_i}\}$, and $T_i$ is the number of tokens of $s_i$.

To better guide the model with discourse context, we use the shared representations to predict RST relations at the same time. For each paragraph, we run the pre-trained RST parser (Ji and Eisenstein, 2014) and flatten the parse tree to obtain RST relations for each sentence $Y_i = (y_1, \ldots, y_{K_i})$, where $K_i$ is the number of discourse relations in $s_i$. We then make a label sequence over tokens in the sentence with

\[ 2 \] The full discourse tree can be incorporated using other types of language model such as Tai et al. (2015).
was dead because she

3.2 Delta-driven FlowNet

In this model, we aim to utilize latent representations to characterize the flow between sentences. Specifically, we define delta, subtractions of hidden representations of adjacent sentences as such latent information. Figure 1(b) shows how we hierarchically model different levels of information: words, sentences, and deltas.

Each word is encoded using a RNN encoder \( g_{\text{word}} \). We take the last hidden representation of word as sentence embeddings \( s_1, ..., s_M \). Similar to hierarchical RNN (Lin et al., 2015), each sentence representation is encoded using another RNN encoder \( g_{\text{sent}} \). While discourse flow provides an explicit relation symbol, delta flow calculates a latent relation by subtracting previous representation \( s_i \) from current representation \( s_i \):

\[
d(s_{i-1}, s_i) = d_{i-1} = s_i - s_{i-1}
\]

Given a sequence of \( M \)-1 delta relations \( d_1, ..., d_{M-1} \) for a paragraph of \( M \) sentences, we again encode them using another RNN encoder \( g_{\text{delta}} \). The model takes the word, sentence, and delta information altogether to predict the next \( (t) \) word in the \( m \)-th sentence:

\[
h_t = f(h_{t-1}, x_t, s_{m-1}, d_{m-2})
\]

where \( x_t \) is a word representation, \( s_{m-1} \) is a sentence representation and \( d_{m-2} \) is a delta information. Note that sentence representation is from the previous sentence, and delta information is calculated by two previous sentences. If there is no previous information given, the parameters are randomly initialized.

4 Experiment

Due to the absence of goal-oriented language generation task, we collect paragraph data and define a new task of generating partial text of a paragraph given some context.

4.1 Data

We collect paragraphs from three different domains: Papers are paragraphs extracted from...
Table 1: Number of paragraphs in our dataset.

<table>
<thead>
<tr>
<th>Domain</th>
<th>Train</th>
<th>Valid</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Papers</td>
<td>16,173</td>
<td>899</td>
<td>899</td>
</tr>
<tr>
<td>SciFi</td>
<td>157,031</td>
<td>8,724</td>
<td>8,724</td>
</tr>
<tr>
<td>Fantasy</td>
<td>317,654</td>
<td>17,649</td>
<td>17,649</td>
</tr>
</tbody>
</table>

academic manuscripts in computer science domain from the PeerRead (Kang et al., 2018a), and Fantasy and SciFi are paragraphs of two frequent categories extracted from the BookCorpus (Zhu et al., 2015), where paragraphs are extracted using the line breaker in the dataset.

We only use paragraphs whose lengths are from 4 to 7, in order to measure the performance change according to paragraph length. The dataset is randomly split by 0.9/0.05/0.05 for train, valid, and test set, respectively. Table 1 shows the numbers of paragraphs for each domain. All paragraphs are parsed into RST trees using the state-of-the-art discourse parser by Ji and Eisenstein (2014).

4.2 Bridging: Partial Paragraph Generation

We evaluate our models on partial text generation task; given a partial information (e.g., some sentences), producing the rest of text.

Figure 2: Bridging task: given [1] and [4] sentences, guessing [2,3] sentences (red, underlined).

Figure 2 shows our bridging task. It requires a generation of masked sentences in the middle of a paragraph given the first and the last sentences. If only the first sentence is given, the generation can be too divergent. The existence of the last sentence makes the generation more coherent and converged to some point.

We evaluate it with one hard and one soft automatic metrics: METEOR (M) (Banerjee and Lavie, 2005) and VectorExtrema (VE) (Liu et al., 2016) by calculating cosine similarity of averaged word embeddings (Pennington et al., 2014), and human performance.

4.3 Models and Setup

We compare various baseline seq2seq models which encode the context; a concatenated first and last sentences, and decode the intermediate words: S2S is attentional seq2seq model (Bahdanau et al., 2014), and HS2S: is a hierarchical version of the S2S by combining two baselines: HRNN (Lin et al., 2015) hierarchically models sequence of words and sentences, and HRED (Serban et al., 2017; Sordoni et al., 2015) encodes the given context and decodes the words. FlowNet (delta/disc.) is our proposed language model with delta and discourse relations, respectively.

We find the best hyper-parameters on validation set using grid search. Here are the final parameters used: 32 for batch size, 25 for maximum sentence length, 300 for word embedding size initialized by GloVe (Pennington et al., 2014), 1 LSTM layer (Hochreiter and Schmidhuber, 1997) with 512 size, clipping by 0.25, 0.2 learning rate and 0.5 decay rate with Adagrad (Duchi et al., 2011) optimizer, and 50,000 for the vocabulary size. The total number of distinct discourse relations is 44.

4.4 Results

Table 2: Performance on bridging task. METEOR and VectorExtrema are used. The higher the better.

Table 3: Comparison of different delta functions.
MLP which is a multi-layer perceptron network. All scores are macro-averaged across datasets. While ADD shows good performance on METEOR, SUBTRACT does on the soft metric (i.e., VecExt), indicating that subtraction can help the model capture the better semantics than the other functions. Figure 3 shows how performance changes on Fantasy as the paragraph lengths increase. Both of FlowNet achieve more improvements when generating longer paragraphs. Especially, discourse relations achieve the best performance at length 6 and 7.

Table 4 shows an example paragraph with text produced by the models as well as reference and human annotation. Given FIRST and LAST sentences, the models generate middle sentences (e.g., [M1] → [M2] → [M3]). REF and HUMAN are reference middle sentences and sentences written by human annotator, respectively. Please find more examples in the appendix.

**5 Conclusion and Discussion**

We explore two forms of inter-sentential relations: linguistic relation such as discourse relations and a latent representation learned from the text. The proposed models for both relations achieve significant improvements over the baselines on partial paragraph generation task.

Despite the empirical effectiveness and difference between the linguistic and latent relations, they are not directly aligned for comparison. A potential direction for future study is to directly couple them together and see whether one form contains the other, or vice versa. Another direction is to check their effectiveness on top of the recent pre-trained language models.

**Acknowledgements**

We also thank Jason Weston, Dan Jurafsky, and anonymous reviewers for their helpful comments.
References


Jiwei Li, Minh-Thang Luong, and Daniel Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. In ACL.


Biao Zhang, Deyi Xiong, Jinsong Su, Qun Liu, Rongrong Ji, Hong Duan, and Min Zhang. 2016. Variational neural discourse relation recognizer. In EMNLP.