A Joint Model for Structure-based News Genre Classification with Application to Text Summarization

Zeyu Dai, Ruihong Huang
Department of Computer Science and Engineering
Texas A&M University
{jzdaizeyu, huangrh}@tamu.edu

Abstract

Journalists usually organize and present the contents of a news article following a well-defined structure. In this paper, we propose a novel joint model for structure-based news genre classification that simultaneously identifies one of four commonly used news structures (including *Inverted Pyramid* and three other structures) for a news article as well as recognizes a sequence of news elements within the article that define the corresponding news structure. Experiments show that the joint model consistently outperforms its variants that perform two tasks independently, which supports our motivation that preserving the two-way dependencies and constraints between a type of news structure and its sequence of news elements enables the model to better predict both of them. Although being not perfect, the system predicted news structure type and news elements have improved the performance of text summarization when incorporated into a recent neural network system.

1 Introduction

Journalists usually organize and report the contents of news following a well-defined structure. For example, when writing news briefs or breaking news, the *Inverted Pyramid* structure (Pottker, 2003) is often adopted to present the most newsworthy and key events first and then provide any additional details. However, while being commonly used, *Inverted Pyramid* is not the only news structure, there exist several other commonly used news structures as well, for example, a structure called *Kabob* is commonly used to present a narrative hook (Myers and Wukasch, 2003) first and then report the main story, where the narrative hook catches the reader's attention so that reader is willing to keep reading. Recognizing the overall structure of a news article can benefit many NLP tasks and applications, such as text summarization, text segmentation, discourse analysis, information extraction and text quality assessment, and many others.

Our recent research (Dai et al., 2018) first defines a small set of news elements, specifically five news elements, and then formally defines four commonly used news structures based on their different ways to select and organize news elements. News elements are defined based on their functions in a news story (introducing the main story or event, catching the reader's attention or providing details, etc.) as well as their writing styles (narrative or expository, also known as modes of discourse). Specifically, five news elements are defined, including two ledes, *Standard Lede* and *Image Lede*, with their functions as either introducing the main story or catching the reader's attention, as well as three other categories, *Synopsis*, *Narration* and a catch-all category *Body Section*. Each news element is realized as a set of one or more consecutive paragraphs in a news article. Using the well-defined news elements, four news structures, *Inverted Pyramid*, *Kabob*, *Martini Glass* and *Narrative* are introduced. The *Inverted Pyramid* structure can be represented as a *Standard Lede* followed by a *Body Section*, while the *Kabob* structure can be represented as an *Image Lede* followed by a *Synopsis* and a *Body Section*. Two more news structures, *Martini Glass* and *Narrative*, are defined and each of them has the *Narration* news element. We defer more details about news elements and news structures to the section 3.

Our previous work (Dai et al., 2018) created a dataset (the News Genre dataset) with both news structures and news elements annotated for structure-based news genre categorization, and has conducted news structure classification as a text classification task by building a machine learning classifier (SVM) using n-grams and several structure indicative features. However, we have not attempted to further recognize the annotated news elements within a news article yet. As each news

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3332–3342
August 1–6, 2021. ©2021 Association for Computational Linguistics

element carries a specific function in building a news story and features a writing style (narrative or expository), the recognized news elements are expected to be useful for many NLP applications.

In this work, we take one step further and propose to recognize both the news structure type of a news article as well as its corresponding news elements. We first implemented two pipeline approaches that first predict document-level news structure (or paragraph-level news element) tags using one single model, and then incorporate the predicted tags as features into another single model for predicting news element (or news structure) tags. Then, inspired by the idea that the overall news structure of a document determines the sequence of news elements within the document, and vice versa, we aim to recognize both the type of news structure and its news elements simultaneously in a joint model. Specifically, we build our joint model on top of a hierarchical BiLSTM neural networks that learn paragraph and document representations for predicting both a news structure type for a document and a sequence of news element tags for its paragraphs. The intrinsic evaluation on the News Genre dataset shows that the joint model consistently outperforms the pipeline models that accomplish two tasks independently, and achieves noticeable performance gains for predicting all four types of news structures and all five types of news elements, which supports our motivation that preserving the two-way dependencies and constraints between a news structure and its news elements enables the system to better predict both of them.

We believe that the identified news structures and news elements can be useful for many text-level NLP applications and tasks. In this paper, we further conduct experiments and use system predicted news structure and news element tags for improving text summarization. Informed by the predicted news structure genres, we expect to better locate the key event descriptions of a news story, and therefore improve the performance of extractive summarization models. Especially, we expect that recognizing news structures and news elements can boost the text summarization performance on news articles of a particular news structure, the *Kabob* structure, which is the second most frequent news genre and covers roughly 28% of news articles based on the annotated News Genre dataset.

For news documents with the *Kabob* structure, the beginning paragraphs (corresponding to a news element called *Image Lede*) do not directly present the key events of news, instead, the following paragraphs (corresponding to a news element called *Synopsis*) will summarize the main story. Therefore, this news genre brings additional difficulty to locate the correct paragraphs for extracting summary, and accordingly, recognizing this genre and its news elements is likely to noticeably improve text summarization performance on documents with the *Kabob* structure the most. Indeed, the extrinsic evaluation on the CNN/DailyMail dataset (Hermann et al., 2015) shows that a simple method for incorporating news genre tags as word features into a recent extractive summarization system (Liu and Lapata, 2019) improves the three ROUGE (Lin, 2004) scores, R-1, R-2 and R-L, consistently for all four types of news structure genres, with the *Kabob* structure receiving the largest improvements of 0.37, 0.14 and 0.34 points on R-1, R-2 and R-L respectively.

2 Related Work

News structures have been extensively studied in the area of linguistics and journalism (Schokkenbroek, 1999; Van Dijk, 1985; Ytreberg, 2001). However, few computational studies tried to automatically categorize news articles according to news structures using data-driven methods. Our previous work (Dai et al., 2018) is the first work we are aware of that formulated four news structures using a small set of predefined news elements, created the first dataset for structure-based news genre categorization, and proposed a feature-based classifier to predict the news structure type of a document. With the motivation to better serve the needs of downstream applications, we developed a computational system to recognize news elements within a document as well as the overall news structure type. We built a joint model for these two tasks to preserve the two-way dependencies and constraints between them, and have empirically improved the performance of both tasks.

In the previous work, several well-studied genre-independent discourse structures have been explored for improving many NLP applications. For example, discourse structures including the RST-style tree structure (Mann and Thompson, 1988) and the PDTB-style discourse relations (Prasad et al., 2008) have been shown useful for a range of NLP applications, such as sentiment analysis (Bhatia et al., 2015; Märkle-Huß et al., 2017), text summarization (Marcu, 1997; Louis et al., 2010) and machine translation (Li et al., 2014; Guzmán et al., 2014). In addition, text segmentation (Hearst, 1994) that divides a text into a sequence of topically coherent segments by detecting topic transition boundaries have been shown useful for text summarization (Barzilay and Lee, 2004), sentiment analysis (Sauper et al., 2010) and dialogue systems (Shi et al., 2019). We believe that the genre-specific news structures can effectively complement the genre-independent discourse structures, and both of them are essential for achieving deep story-level text understanding.

In this work, we further apply our system predicted news structure and news element tags to help the task of extractive summarization, which aims to extract a summary by identifying the most important sentences in a news article. Nallapati et al. (2017) presents one of the earliest neural network systems for extractive summarization that adopt an RNN-based encoder for abstracting sentence representations. More recent work achieves higher performance for extractive summarization using more sophisticated neural network structures. SUMO (Liu et al., 2019) introduces structured attention to induce a dependency tree representation of a document while generating a summary. Liu and Lapata (2019) adapts BERT (Devlin et al., 2019) to text summarization which obtains contextualized representations of a document and its sentences using BERT's encoder by stacking several inter-sentence Transformer layers. Dong et al. (2019) fine-tunes a new Unified pre-trained Language Model (UniLM) for text summarization by employing a shared Transformer network and utilizing specific self-attention masks to control which context the predicting summary conditions on. The extrinsic evaluation on text summarization using (Liu and Lapata, 2019) as baseline demonstrates the usefulness of our system predicted genre-specific news structure tags in downstream NLP tasks.

In addition, our work is also related to text genre identification (Santini, 2007; Mehler et al., 2010; Rehm, 2002), but we focus on the genres of news structure which come from the area of journalism.

3 Structure-based News Genres

As shown in Figure 1, our previous work (Dai et al., 2018) formally defined four commonly used news structures based on the selection and organization of five predefined news elements.

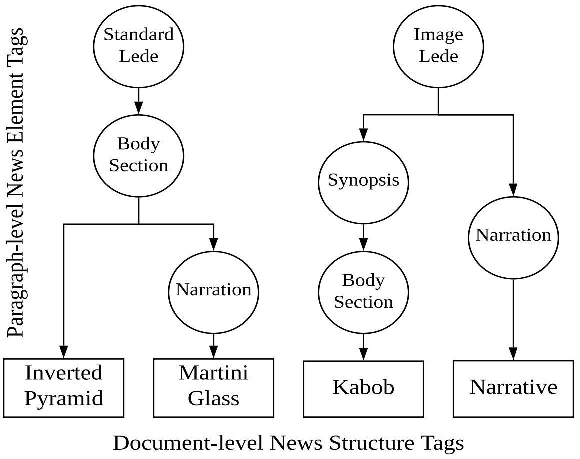

Figure 1: Four News Structures: Document-level News Structure Tags (in rectangle) and Paragraph-level News Element Tags (in circle). A News Element may include one or more consecutive paragraphs.

3.1 Five Paragraph-level News Elements

Standard Lede is used to introduce the key events and main story at the beginning of a news article; written in the expository style.

Image Lede [1] is used to catch the reader's attention by telling an anecdote, quoting a catchy slogan, or revealing an impressive fact or statistics (Jou, 2014); written in either narrative or expository style. Image Lede is located at the beginning of a news article as well, however, unlike Standard Lede, it does not directly discuss the key events of a news article, therefore, it may not represent a good summary of the news article.

Synopsis must follow an Image Lede and acts as a bridge that connects an Image Lede with the rest of a story. The function of Synopsis is to summarize the key events and main story of a news article; written in the expository style.

Narration gives great details about key events and often contains a sequence of events (or subevents) in chronological order (Mani, 2012); written in the narrative style (Lavelle, 1997).

Body Section presents additional details and supplementary information about key events; written in the expository style. Paragraphs that do not belong to any of the four above categories were annotated as a Body Section (Dai et al., 2018).

3.2 Four Document-level News Structures

Inverted Pyramid, known as the most popular news article structure (Pottker, 2003), presents the

[1] In some news articles, an image is presented first to catch the readers' eyes and the Image Lede acts as the description of the image.

content in the descending order of importance and relevance (Scanlan, 2003). For this structure, key events and main story will be introduced first, then additional information will be provided later. This structure is represented as a Standard Lede followed by a Body Section, shown in Figure 1.

Martini Glass (Jou, 2014) begins by presenting a summary of a story following the Inverted Pyramid structure, and then transitions into a chronological elaboration of the story in detail. Therefore, the Martini Glass structure contains a Standard Lede, an optional Body Section and a Narration.

Kabob (Jou, 2014) first tries to catch the reader's eyes using an anecdote (or a catchy slogan, etc), then introduces the key events, and discusses the main story with more details at last. Therefore, the Kabob structure is defined to start with an Image Lede, then uses a Synopsis as the transition, and finally ends with a Body Section.

Narrative structure presents a chronologically ordered sequence of events with a greater amount of details than normal news articles. Dai et al. (2018) annotated this news structure when the majority of paragraphs form a single Narration with an optional preceding Image Lede.

3.3 The News Genre Dataset

Dai et al. (2018) created the first structure-based news genre dataset [2]. This dataset contains 853 English news articles across four news domains, including politics, crime, business and disaster. In this dataset, each article was annotated with a news structure label and a sequence of news element tags for its paragraphs. The same news element tag will be assigned to all paragraphs in a consecutive sequence that a news element spans over.

The four common news structures applied to most of the annotated news articles, with only 21 documents were not annotated with any of the four news structures and did not receive paragraph-level news element tags either, so we removed these 21 documents in our experiments. Table 1 shows the statistics of news structure and news element tags, from which we can see that the distribution of news structures is highly imbalanced, with *Inverted Pyramid* and *Kabob* as two major structure types.

News Structure	#	News Element	#
Inverted Pyramid	482	Standard Lede	519
Martini Glass	37	Image Lede	244
Kabob	237	Synopsis	237
Narrative	76	Narration	113
Total	832	Body Section	746

Table 1: Data Statistics of the News Genre Dataset.

4 Model

4.1 The Joint Model for Predicting both News Structures and News Elements

Figure 2 illustrates the overall architecture of our joint model, which can simultaneously predict both document-level news structure label and paragraph-level news element tags. The model processes a whole news article containing a sequence of paragraphs each time, and predicts a document-level label as well as a sequence of paragraph-level tags with one tag for each paragraph using the standard BIO tagging schema (Ratinov and Roth, 2009) for sequence labeling. Specifically, we treat the news element *Body Section* as the "other" (or 'O') tag since this tag can't help determine document-level news structure type (shown in Figure 1) and was used as a catch-all "other" label during the data annotation as well. For other paragraph-level news element tags except for the *Body Section*, we assign a "B-" prefix to the first paragraph that starts the news element and assign "I-" prefix to other paragraphs inside the same news element.

The model employs the two-level hierarchical BiLSTM layers (Schuster and Paliwal, 1997) with max-pooling (Collobert and Weston, 2008) operation in between to learn both word and paragraph representations, followed by a max-pooling operation to calculate the document representation and a softmax classification layer for predicting the document-level label. Added on top of the paragraph-level representations, a linear-chain Conditional Random Field (CRF) layer (Lafferty et al., 2001) is utilized to jointly decode a sequence of paragraph-level tags considering their inter-dependencies. As shown in Figure 2, the model consists of the following components:

Feature-rich Word Vector: Given a sequence of words $(w_1, w_2, ..., w_L)$ as the input document, for each word w_i, we construct a feature-rich word vector by concatenating its word embedding $\boldsymbol{w}_i^{word}$ with its character-level representation [3], and extra

[2] Available at https://github.com/ZeyuDai/Fine-grained_Structure-based_News_Genre_Categorization

[3] For character-level representation, we adopted one layer

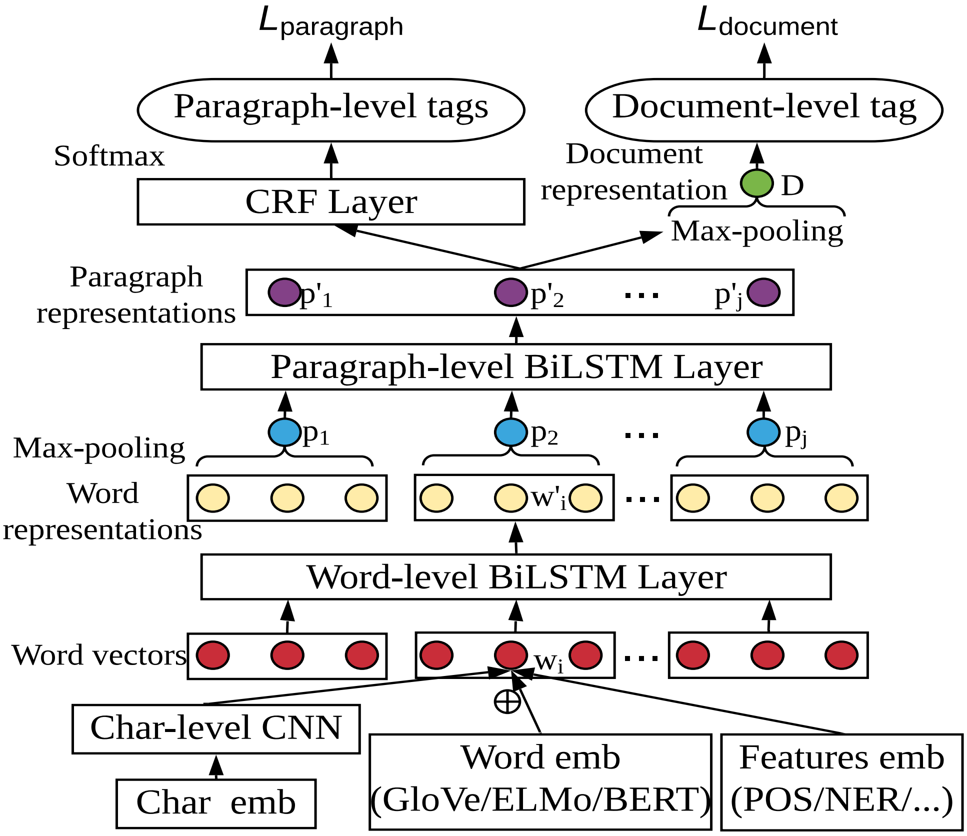

Figure 2: The Joint Model Architecture for both Document-level and Paragraph-level News Genre Tags Prediction.

word-level features embedding [4] as:

$$\boldsymbol{w}_i = [\boldsymbol{w}_i^{word}; \boldsymbol{w}_i^{char}; \boldsymbol{w}_i^{features}]$$

To take advantage of the recent progress about contextualized word representation from pre-trained language models, our framework supports three options including 300 dimensional GloVe (Pennington et al., 2014), 1024 dimensional ELMo (Peters et al., 2018) and the "bert-base-cased" version of BERT (Devlin et al., 2019) to initialize [5] the $\boldsymbol{w}_i^{word}$.

Word-level BiLSTM Layer: Given a sequence of feature-rich word vectors $(\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_L)$ as the input, the word-level BiLSTM layer will refine the word w_i's hidden representation $(\boldsymbol{w}_i')$ by modeling the word-level inter-dependencies:

$$\boldsymbol{w}_i' = BiLSTM(\boldsymbol{w}_1, ..., \boldsymbol{w}_i, ..., \boldsymbol{w}_L)$$

Paragraph-level BiLSTM Layer: Given a sequence of word representations $(\boldsymbol{w}_1', \boldsymbol{w}_2', ..., \boldsymbol{w}_L')$, we build the paragraph representation $(\boldsymbol{p}_j)$ for the j-th paragraph in the document, by applying max-pooling operation over the sequence of word representations for all words within the j-th paragraph:

$$\boldsymbol{p}_j = \max_{w_i \in p_j} \boldsymbol{w}_i'$$

Then, the paragraph-level BiLSTM layer will update the j-th paragraph's hidden representation $(\boldsymbol{p}_j')$ by modeling the paragraph-level inter-dependencies:

$$\boldsymbol{p}_j' = BiLSTM(\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_j, ...)$$

Softmax Classification Layer for Document-level News Structure Type Prediction: We compute the document representation $(\boldsymbol{D})$ by applying max-pooling operation over all paragraph representations $(\boldsymbol{p}_1', \boldsymbol{p}_2', ..., \boldsymbol{p}_j', ...)$.

Then, for the i-th training instance with $\boldsymbol{y}_{doc\_gold}^{(i)}$ as the gold annotation of document-level tag, our model predict the document-level tag $\boldsymbol{y}_{doc\_pred}^{(i)}$ using the softmax classification layer:

$$\boldsymbol{y}_{doc\_pred}^{(i)} = softmax(\boldsymbol{W}_{doc}\boldsymbol{D}^{(i)} + \boldsymbol{b}_{doc})$$

of CNN with 50 hidden units followed by a max-pooling layer.

[4] For word-level features, we collected the corresponding paragraph's position (PARA) index, capitalization (CAP) flag, Part-of-speech (POS) tag and named entity (NER) tag of each word. The embedding sizes for PARA/CAP/POS/NER were 20/5/35/20 respectively. We used Standford CoreNLP toolkit (Manning et al., 2014) to generate POS and NER tags.

[5] GloVe embeddings were fixed during training. For ELMo and BERT, we also froze its parameters during model training.

And we want to minimize the following cross-entropy loss during model training:

$$L_{document} = -\sum_{i} \boldsymbol{y}^{(i)}_{doc\_gold} * \log \boldsymbol{y}^{(i)}_{doc\_pred}$$

CRF Layer for Paragraph-level News Elements Sequence Labeling: For the task of sequence labeling, it is important to model the label dependencies (e.g., "I-*" must follow "B-*" in BIO tagging schema.) and capture the label continuity and transition patterns. Therefore, a CRF layer is added on top of the paragraph-level BiLSTM layer to jointly decode the news element tags sequence.

For the i-th training instance, given the annotated paragraph-level news element tags sequence $\boldsymbol{y}^{(i)}_{para\_gold} = (y^{(i)}_1, y^{(i)}_2, ..., y^{(i)}_j, ...)$ and hidden paragraph representations $\boldsymbol{P'}^{(i)} = (\boldsymbol{p'}^{(i)}_1, \boldsymbol{p'}^{(i)}_2, ..., \boldsymbol{p'}^{(i)}_j, ...)$, we minimize the following CRF loss during model training:

$$L_{paragraph} = -\sum_{i} \log p(\boldsymbol{y}^{(i)}_{para\_gold} | \boldsymbol{P'}^{(i)})$$

For model testing, we use the Viterbi algorithm to search for the optimal label sequence.

Joint Model vs. Single Model Training: The overall loss function for training our joint model is:

$$L = L_{document} + L_{paragraph}$$

Clearly, we can easily make it a single-task model for either document-level news structure type prediction or paragraph-level news element sequence labeling, by removing unrelated loss term from the overall loss function. We will compare the performance of our joint model with single models in the following intrinsic evaluation section 5.

4.2 Parameter Settings and Implementation Details

We manually tuned all hyperparameters of our model based on the development set using the macro-average F1-score as the selection criterion. After the hyperparameter search, we used the hidden size of 512 (tuned from the list [100, 300, 512, 1024]) for each BiLSTM layer and all hidden representations ($\boldsymbol{w'}_i, \boldsymbol{p}_j, \boldsymbol{p'}_j, \boldsymbol{D}$). For regularization, we applied 50% (tuned from [10%, 20%, 30%, 50%]) dropout to both input and output vectors of each BiLSTM layer. To alleviate the problem of gradient exploding for BiLSTM training, we clipped the gradient L2 norm at threshold 5.0 (tuned from [5.0, 10.0]) and utilized L2 regularization with coefficient 10^{-6}. Parameters were optimized using SGD optimizer with momentum 0.9 (tuned from [0.9, 0.95] and no momentum) and initial learning rate 0.015 (tuned from [0.0001, 0.001, 0.01, 0.015, 0.05, 0.1]), decreasing by 5% after each epoch. The batch size was 32 (tuned from [8, 16, 32, 64]) in the normal case, but it will be much smaller (1 or 2 depending on the model size) when using BERT because of the GPU CUDA memory limitation.

News Structure	#	News Element	#
Inverted Pyramid	434	Standard Lede	467
Martini Glass	33	Image Lede	219
Kabob	214	Synopsis	214
Narrative	69	Narration	102
Total	750	Body Section	673

Table 2: Data Statistics of the Cross-validation Set.

We implemented our model using Pytorch, with ELMo from AllenNLP [6] and BERT-base from HuggingFace [7]. Since BERT used the subword tokenizer, we used the first token's representation as word embedding if one word was split into several subword tokens. We trained our model for 50/20/3 epochs when using GloVe/ELMo/BERT word embeddings respectively, considering that different word representation techniques require a different number of fine-tuning epochs. To diminish the effects of randomness in neural network training, we ran our proposed model, its variants as well as our own baselines using 5 different random seeds and the reported performance is the average score across 5 runs. The full model training took around 8-12 hours on one NVIDIA GTX 1080Ti GPU.

5 Intrinsic Evaluation

5.1 Experimental Settings

Considering that the News Genre corpus is relatively small and cross-validation is more robust for a small dataset, we followed our previous work (Dai et al., 2018) and evaluated our models using 5-fold cross-validation. Specifically, we created our own cross-validation/development set splits containing 750/82 news articles respectively, and randomly split the cross-validation set into five folds with even domain distribution. Table 2 reports the distribution of news structure and element tags on

[6] https://github.com/allenai/allennlp

[7] https://github.com/huggingface/transformers

	Document-level News Structure Types						Paragraph-level News Element Tags						
Model	Acc	Mac	IP	MG	Kab	Nar	Acc	Mac	SL	IL	Sy	Na	BS
Feature-based (2018)	71.8	50.3	81.2	17.8	54.3	48.0	-	-	-	-	-	-	-
Our Models													
Single Model (GloVe)	75.6	51.8	81.7	19.5	56.0	50.2	73.4	48.6	67.0	28.8	28.8	36.0	82.6
Single Model (ELMo)	78.0	54.2	84.0	22.0	58.6	52.0	76.0	50.9	68.3	30.4	30.4	38.0	87.2
Single Model (BERT)	77.6	53.6	83.5	21.5	58.0	51.5	75.6	50.3	68.0	30.2	30.2	37.5	85.5
Joint Model (GloVe)	77.8	53.8	83.7	21.7	58.3	51.6	75.8	50.6	68.2	30.0	30.0	38.0	86.6
Joint Model (ELMo)	**80.0**	**56.0**	**86.0**	**24.6**	**60.5**	**52.8**	**78.3**	**53.2**	**70.5**	**32.4**	**32.4**	**40.5**	**90.4**
Joint Model (BERT)	79.2	55.5	85.5	24.2	60.0	52.2	77.6	52.4	70.0	32.0	32.0	38.2	90.0
Pipeline Models (ELMo)													
Pipeline (doc → *para*)	78.0	54.2	84.0	22.0	58.6	52.0	77.4	52.2	69.7	31.6	31.6	39.2	89.0
Pipeline (para → *doc*)	78.8	55.0	84.8	23.4	59.5	52.4	76.0	50.9	68.3	30.4	30.4	38.0	87.2

Table 3: Intrinsic Evaluation Results on the Cross-validation Set of News Genre Dataset using 5-fold Cross-validation. We report accuracy (Acc), macro-average F1-score (Mac), and class-wise F1-scores for document-level structure and paragraph-level element tags, including Inverted Pyramid (IP), Martini Glass (MG), Kabob (Kab), Narrative (Nar), Standard Lede (SL), Image Lede (IL), Synopsis (Sy), Narration (Na) and Body Section (BS).

the cross-validation set. The hyperparameter tuning was conducted on the development set using the cross-validation set for model training.

5.2 Baselines

Feature-based (Dai et al., 2018): To compare with previous work, we replicated the feature-based model of (Dai et al., 2018) that performs document-level news structure type classification only.

Pipeline (doc → para) && Pipeline (para → doc): We implemented two pipeline approaches that first predict document-level news structure (or paragraph-level news element) tags using our single model, and then incorporate the predicted tags as word-level features (with embedding size 10) into another single model for predicting paragraph-level (or document-level) tags. The pipeline approach that first predicts document-level news structure tags is marked as Pipeline (doc → para); the reverse one is marked as Pipeline (para → doc).

5.3 Experimental Results

Table 3 summarizes the evaluation results on the cross-validation set using 5-fold cross-validation. The first row shows the performance of our replicated feature-based baseline (Dai et al., 2018) which achieves similar performance as in the original paper. The second section reports the performance of our models for predicting both document-level news structure types and paragraph-level news element tags, which compares the results of our models trained with different loss functions (joint model vs. single model) when using different word embeddings (GloVe vs. ELMo vs. BERT).

We can see that the joint model consistently outperforms (statistical significant t-test with $p < 0.05$) the corresponding single model independent from the word embeddings, which supports our motivation that document-level news structure type identification can not be separated from learning paragraph-level news element representations and features, and vice versa. Among the three word representation techniques, the ELMo word embeddings consistently give the best performance, followed by BERT and GloVe. One possible reason why BERT performs worse in our experiments is that we have to use a very small batch size and large learning rate when using BERT due to the limitation of GPU CUDA memory. The best joint model using the ELMo embeddings achieves 80.0% accuracy and 56.0% macro F1-score for predicting document-level news structure types, which outperforms the previous feature-based baseline by a large margin, and simultaneously achieves 78.3% accuracy and 53.2% macro F1-score for identifying paragraph-level news element tags.

The third section shows the performance of the two pipeline models. Note that, for fair comparisons, both pipeline models use the ELMo word embeddings that perform the best for our tasks (in both single and joint models). We can see that our joint model consistently outperforms both pipeline approaches. This is reasonable because pipeline models suffer from error propagation which poses an even bigger challenge in our task when the predicted news element sequence can not be compatible with any of the four news structure types.

In addition, Table 4 reports the experimental results on the development set, where we used the whole cross-validation set for training the models. On the development set, we observe similar comparisons among models and consistent performance gains achieved by the joint model.

	Document-level News Structure Types						Paragraph-level News Element Tags						
Model	Acc	Mac	IP	MG	Kab	Nar	Acc	Mac	SL	IL	Sy	Na	BS
Feature-based (2018)	72.8	51.2	81.7	18.5	56.1	48.5	-	-	-	-	-	-	-
Our Models													
Single Model (GloVe)	76.2	52.7	82.4	19.6	56.9	51.7	74.6	49.1	68.0	29.5	29.5	35.9	83.0
Single Model (ELMo)	78.8	54.6	84.9	22.2	59.0	52.3	76.8	51.3	68.6	30.8	30.8	38.5	87.8
Single Model (BERT)	78.4	54.3	84.5	22.7	58.8	51.2	76.2	50.5	68.4	30.4	30.4	37.1	86.3
Joint Model (GloVe)	78.5	54.5	84.4	23.1	58.9	51.4	76.2	50.7	68.1	30.0	30.0	38.5	86.9
Joint Model (ELMo)	**81.1**	**56.5**	**86.6**	**25.2**	**60.9**	**53.1**	**79.4**	**53.8**	**71.0**	**33.2**	**33.2**	**40.8**	**90.6**
Joint Model (BERT)	79.5	55.6	85.7	23.8	60.4	52.3	77.9	52.9	70.4	32.5	32.5	39.0	90.1
Pipeline Models (ELMo)													
Pipeline (doc → *para*)	78.8	54.6	84.9	22.2	59.0	52.3	77.6	52.6	70.0	32.0	32.0	39.2	89.9
Pipeline (para → *doc*)	79.8	55.6	86.1	23.2	60.2	52.8	76.8	51.3	68.6	30.8	30.8	38.5	87.8

Table 4: Intrinsic Evaluation Results on the Development Set using the whole Cross-validation Set for Training.

5.4 Qualitative Analysis

To better understand the strengths and weaknesses of the joint model, we analyze the news structure and news element tags prediction made by our single model and joint model (both using ELMo embeddings) on the development set. Among the 82 documents, we find that the joint model clearly made less inconsistent predictions than the single model (18 vs. 27) where the predicted news element sequence can not be compatible with the predicted news structure type, e.g., *Inverted Pyramid* structure with *Image Lede* news element. This result proves the effectiveness of our joint model that preserves the two-way dependencies between the predicted news structure type and news elements.

We further examine the wrong predictions generated by our best joint model. About 70% errors happen because the model failed to distinguish the first news element between *Standard Lede* and *Image Lede*, which can be improved if the model is aware of the key events (Choubey et al., 2018) in a news article. The remaining errors come from identifying the *Narration* paragraphs written in narrative style, which by itself is a challenging task.

6 Extrinsic Evaluation on Text Summarization

We expect the news genre tags predicted by our joint model to be useful for extracting news summaries because our tags (e.g., *Standard Lede* in *Inverted Pyramid*; and *Synopsis* in *Kabob*) can help locate the key event descriptions of a news story which should be the right section to select sentences for extractive summarization.

To verify our expectations, we choose a recent BERT-based framework for text summarization proposed by Liu and Lapata (2019), which used to achieve the state-of-the-art performance on the CNN/DailyMail dataset (Hermann et al., 2015). We use exactly the same experiment settings as in (Liu and Lapata, 2019) and implement our text summarization models based on their source code [8]. We leave all components of the summarization model unchanged, but add an embedding layer to the input of BERT, which encodes the paragraph-level news elements and document-level news structure tags generated by our system trained on the whole cross-validation set. Specifically, the embedding layer will encode each tag or the combination of a news structure type and a news element tag (e.g., Kabob-Image Lede) into a vector with 10 dimensions, which will be concatenated with the original BERT's word embeddings. For each input token, the added embedding layer will incorporate its news structure information (e.g., the paragraph-level tag for the paragraph where the token locates in) into the hidden token representation, and therefore influence the model.

Model	R-1	R-2	R-L
LEAD-3	40.42	17.62	36.67
SUMO (Liu et al., 2019)	41.00	18.40	37.20
UniLM (Dong et al., 2019)	43.33	20.21	40.51
Baseline (Liu and Lapata, 2019)	43.25	20.24	39.63
+ News Element tags (ours)	43.42	20.28	39.74
+ News Structure types (ours)	**43.48**	**20.30**	**39.78**

Table 5: Text Summarization Results on the CNN/DailyMail Dataset. R-1 and R-2 stand for ROUGE score using unigram and bigram overlap; R-L is the ROUGE score using longest common subsequence. LEAD-3 is a simple baseline which selects the first three sentences in a news article.

6.1 Experimental Results

Table 5 shows the text summarization results on the CNN/DailyMail dataset using the automatic evaluation package ROUGE (Lin, 2004). Incorporating

[8] Available at `https://github.com/nlpyang/PreSumm`

News Structure	Inverted Pyramid	Martini Glass	Kabob	Narrative
LEAD-3	40.58/17.86/36.83	40.48/17.80/36.75	40.13/17.18/36.33	40.25/17.52/36.54
Baseline (Liu and Lapata, 2019)	43.38/20.30/39.76	43.33/20.28/39.72	43.05/20.12/39.38	43.17/20.20/39.58
+ News Element tags (ours)	43.43/20.33/39.80	43.38/20.30/39.75	43.32/20.22/39.61	43.35/20.26/39.70
+ News Structure types (ours)	**43.49/20.35/39.82**	**43.47/20.33/39.80**	**43.42/20.26/39.72**	**43.44/20.28/39.74**

Table 6: Text Summarization Results divided by News Structure Genres. Each cell reports R-1/R-2/R-L scores.

the system predicted paragraph-level news element tags into the baseline (Liu and Lapata, 2019) improves the R-1, R-2 and R-L by 0.17, 0.04 and 0.11 points respectively, which is non-trivial considering the difficulties of text summarization. Adding our document-level news structure types into the summarization model further improves the performance slightly, which outperforms the baseline by 0.23 R-1, 0.06 R-2 and 0.15 R-L.

6.2 Effects on Different News Genres

To understand which type of news structure is the bottleneck for news summarization, we evaluate the ROUGE scores on each subset of the CNN/DailyMail test set divided by our predicted news structure types, and report the text summarization results in Table 6. We can see that *Kabob* structure is the most difficult genre for news summarization, which is not surprising because news documents with the *Kabob* structure will not present the key events at the beginning of the story, and therefore brings additional difficulty to locate the correct paragraphs for extracting summary. By incorporating our news structure types and news element tags into the model, all genres of news documents receive better performance for extractive summarization. Especially for the news articles with the *Kabob* structure, our news genre tags improve the ROUGE scores by 0.37, 0.14 and 0.34 points on R-1, R-2 and R-L respectively, which is the largest improvement among four types of news structures.

7 Conclusion

We have presented a joint neural network model for structure-based news genre identification that predicts both the news structure type for a document and a sequence of news element tags for its paragraphs. The joint model preserves the two-way dependencies and constraints between a type of news structure and its sequence of news elements, and consistently outperforms its variants that perform two tasks independently or in a pipeline. While being imperfect, the system predicted news structure types and news element tags have been shown effective for improving text summarization models.

For the future work, we will further improve the performance on identifying minority classes of news structures and news elements (e.g., *Narration*), by conducting semi-supervised learning. Meanwhile, we are keen to explore uses of our news genres in other applications as well, such as text quality assessment and information extraction.

Acknowledgments

We thank our anonymous reviewers for providing insightful review comments. We gratefully acknowledge support from National Science Foundation via the awards IIS-1942918. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. Disclaimer: the views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF or the U.S. Government.

References

2014. Journalism story structure.

Regina Barzilay and Lillian Lee. 2004. Catching the Drift: Probabilistic Content Models, with Applications to Generation and Summarization. In *NAACL*.

Parminder Bhatia, Yangfeng Ji, and Jacob Eisenstein. 2015. Better document-level sentiment analysis from RST discourse parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2212–2218, Lisbon, Portugal. Association for Computational Linguistics.

Prafulla Kumar Choubey, Kaushik Raju, and Ruihong Huang. 2018. Identifying the most dominant event in a news article by mining event coreference relations. In *NAACL-HLT*, pages 340–345.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *ICML*, pages 160–167.

Zeyu Dai, Himanshu Taneja, and Ruihong Huang. 2018. Fine-grained structure-based news genre categorization. In *Proceedings of the Workshop Events* *and Stories in the News 2018*, pages 61–67, Santa Fe, New Mexico, U.S.A. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, volume 1, pages 4171–4186.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, pages 13063–13075.

Francisco Guzmán, Shafiq Joty, Lluís Màrquez, and Preslav Nakov. 2014. Using discourse structure improves machine translation evaluation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 687–698.

Marti A Hearst. 1994. Multi-paragraph segmentation of expository text. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 9–16. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

John Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, 2001, pages 282–289.

Ellen Lavelle. 1997. Writing style and the narrative essay. *British Journal of Educational Psychology*, 67(4):475–482.

Junyi Jessy Li, Marine Carpuat, and Ani Nenkova. 2014. Assessing the discourse factors that influence the quality of machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 283–288.

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.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *EMNLP-IJCNLP*, pages 3721–3731.

Yang Liu, Ivan Titov, and Mirella Lapata. 2019. Single document summarization as tree induction. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1745–1755, Minneapolis, Minnesota. Association for Computational Linguistics.

Annie Louis, Aravind Joshi, and Ani Nenkova. 2010. Discourse indicators for content selection in summarization. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 147–156. Association for Computational Linguistics.

Inderjeet Mani. 2012. Computational modeling of narrative. *Synthesis Lectures on Human Language Technologies*, 5(3):1–142.

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *ACL System Demonstrations*, pages 55–60.

Daniel Marcu. 1997. From discourse structures to text summaries. In *Intelligent Scalable Text Summarization*.

Joscha Märkle-Huß, Stefan Feuerriegel, and Helmut Prendinger. 2017. Improving sentiment analysis with document-level semantic relationships from rhetoric discourse structures. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Alexander Mehler, Serge Sharoff, and Marina Santini. 2010. *Genres on the web: Computational models and empirical studies*, volume 42. Springer Science & Business Media.

Jack Myers and Don C Wukasch. 2003. *Dictionary of poetic terms*. University of North Texas Press.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI*, volume 31.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*, volume 1, pages 2227–2237.

Horst Pottker. 2003. News and its communicative quality: The inverted pyramid—when and why did it appear? *Journalism Studies*, 4(4):501–511.

R. Prasad, N. Dinesh, Lee A., E. Miltsakaki, L. Robaldo, Joshi A., and B. Webber. 2008. The Penn Discourse Treebank 2.0. In *lrec2008*.

Lev Ratinov and Dan Roth. 2009. Design challenges and misconceptions in named entity recognition. In *Proceedings of the thirteenth conference on computational natural language learning*, pages 147–155. Association for Computational Linguistics.

Georg Rehm. 2002. Towards automatic web genre identification: a corpus-based approach in the domain of academia by example of the academic's personal homepage. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pages 1143–1152. IEEE.

Marina Santini. 2007. *Automatic identification of genre in web pages*. Ph.D. thesis, University of Brighton.

Christina Sauper, Aria Haghighi, and Regina Barzilay. 2010. Incorporating Content Structure into Text Analysis Applications.

Chip Scanlan. 2003. Writing from the top down: Pros and cons of the inverted pyramid.

Christina Schokkenbroek. 1999. News stories: structure, time and evaluation. *Time & Society*, 8(1):59–98.

Mike Schuster and Kuldip K Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Weiyan Shi, Tiancheng Zhao, and Zhou Yu. 2019. Unsupervised dialog structure learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1797–1807, Minneapolis, Minnesota. Association for Computational Linguistics.

Teun A Van Dijk. 1985. Structures of news in the press. *Discourse and communication: New approaches to the analysis of mass media discourse and communication*, 10:69.

Espen Ytreberg. 2001. Moving out of the inverted pyramid: narratives and descriptions in television news. *Journalism Studies*, 2(3):357–371.