Enhancing Multi-Document Summarization with Cross-Document Graph-based Information Extraction

Zixuan Zhang[1], **Heba Elfardy**[2], **Markus Dreyer**[2], **Kevin Small**[2], **Heng Ji**[2], **Mohit Bansal**[2]
[1]University of Illinois at Urbana-Champaign
[2]Amazon Alexa
zixuan11@illinois.edu
{helfardy, mddreyer, smakevin, jihj, mobansal}@amazon.com

Abstract

Information extraction (IE) and summarization are closely related, both tasked with presenting a subset of the information contained in a natural language text. However, while IE extracts structural representations, summarization aims to abstract the most salient information into a generated text summary – thus potentially encountering the technical limitations of current text generation methods (e.g., hallucination). To mitigate this risk, this work uses structured IE graphs to enhance the abstractive summarization task. Specifically, we focus on improving Multi-Document Summarization (MDS) performance by using cross-document IE output, incorporating two novel components: (1) the use of auxiliary entity and event recognition systems to focus the summary generation model and; (2) incorporating an alignment loss between IE nodes and their text spans to reduce inconsistencies between the IE graphs and text representations. Operationally, both the IE nodes and corresponding text spans are projected into the same embedding space and pairwise distance is minimized. Experimental results on multiple MDS benchmarks show that summaries generated by our model are more factually consistent with the source documents than baseline models while maintaining the same level of abstractiveness.[1,2]

1 Introduction

Information extraction (IE) (Lin et al., 2020; Li et al., 2021) and summarization (Xiao et al., 2022; Pasunuru et al., 2021) are inherently similar tasks, sharing the objective of identifying and presenting a targeted subset of the information present in a natural language text. However, there are also conceptual and methodological distinctions. First of all, IE aims to extract specific structured information from natural language text while abstractive text summarization targets abstracting the most salient information of a given text into a natural language summary. Secondly, IE methods frequently have access to world knowledge via external schema and knowledge resources (e.g., Wikidata) whereas summarization methods often rely on the information encoded in large-scale pretrained embeddings to produce coherent summaries. The complementary aspects of these tasks imply an opportunity to transfer knowledge from one task to another.

Hence, in this paper, our primary motivation is to take advantage of the complementary nature of IE and summarization tasks, using the structured output of entity and event extraction systems to improve abstractive text summarization by focusing text generation toward explicitly observable grounded concepts. There are a few previous research studies exploring the mutual enhancement between IE and summarization. For example, Lu et al. (2022) use text summarization to improve relation extraction and Pasunuru et al. (2021) adopt open-domain IE to provide additional structural inputs for Multi-Document Summarization (MDS). However, these approaches have two notable limitations. First, IE is performed on single documents without analyzing cross-document interactions between the extracted knowledge elements. Such cross-document interactions could be essential to identifying salient parts of the source documents, which is especially useful for MDS. Moreover, previous studies use linearized graphs without actually constructing the graph as a whole, and hence failing to capture some global interactions between the extracted knowledge elements.

Based on these motivations, in this paper, we propose a text summarization model which is enhanced by IE. We focus on multi-document summarization (MDS) and improve the MDS model with cross-document IE graphs. Specifically, given a cluster of documents related to the same topic, we

[1]All our code will be publicly available at https://github.com/amazon-science/IESum.

[2]The work was done during the first author's internship at Amazon Alexa.

Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pages 1696–1707

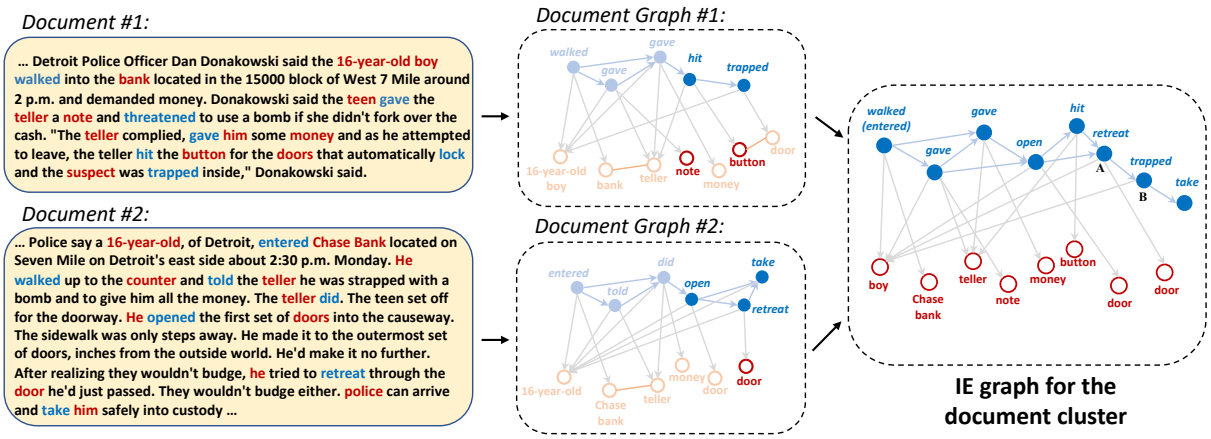

Figure 1: An example of an extracted cross-document IE graph. There are two documents in the cluster and both of them describe a story where a boy failed to rob a bank. However, each document lists different details about the event. For example, the first document mentions that the bank teller hit the button to lock the boy inside; while the second document mentions that boy was trapped between two different doors of the bank. We highlight the unique nodes and edges in the document graph. The merged graph has a more comprehensive description of the story.

first use a cross-document fine-grained IE system to extract a cluster-level information graph, where each node could be an entity or an event trigger and each edge could be "*event-event*" temporal relations, "*event-argument*" links, or "*entity-entity*" relations. Each node in the graph is merged from separate documents according to entity and event coreference. After obtaining the cluster-level IE graph, we use an edge-conditioned graph attention network to encode the IE graph and to merge the graph information into the sequence-to-sequence summary generation pipeline. To better utilize the signals from IE, we further propose two novel training objectives. First, we propose an auxiliary task of entity and event recognition, where an additional classification module is incorporated to train a model to select the important entities and event triggers when performing summarization. The purpose of this auxiliary task is to help the model better recognize and remember the important events and entities which could be crucial for generating high-quality summaries. Second, we propose a graph and text alignment loss that minimizes the distance between IE graph nodes (e.g., nodes A and B in Figure 1) and their corresponding text segments (e.g., *retreat* and *trapped*) in a shared latent embedding space. Such an alignment loss can effectively incorporate IE graph information into the text representations and also mitigate the errors and inconsistencies caused by inevitable noise in the automatically extracted IE graphs. We conduct extensive experiments on multiple MDS benchmarks and show that our model outperforms several strong baselines both in terms of ROUGE scores as well as factual consistency metrics, all while maintaining the same level of abstractiveness. In summary, our main contributions are:

- We improve multi-document summarization (MDS) with cross-document IE graphs.
- We propose two novel training objectives to help the model better utilize the guidance from IE: (1) an entity and event recognition task loss and (2) a node-text alignment loss.
- Our proposed approach is proven effective by extensive experiments on multiple MDS benchmarks while achieving new state-of-the-art performance.

2 Problem Formulation

Our problem definition follows the typical formulation of abstractive multi-document summarization (MDS). Specifically, given a cluster of input documents $\mathcal{D} = \{D_1, D_2, \cdots, D_N\}$, we aim to build a model to generate a summary S of the document cluster. In this paper, we particularly focus on using IE to enhance summarization using the IE graph $\mathcal{G}$ merged from the individual graphs $\{G_1, G_2, \cdots, G_N\}$ extracted from N documents.

2.1 Cross-Document Information Extraction

We first perform cross-document information extraction on each document cluster using a state-of-the-art entity extraction and disambiguation system ReFinED (Ayoola et al., 2022) and event extraction and tracking system RESIN-11 (Du et al., 2022). Specifically, we first extract individual entity men-

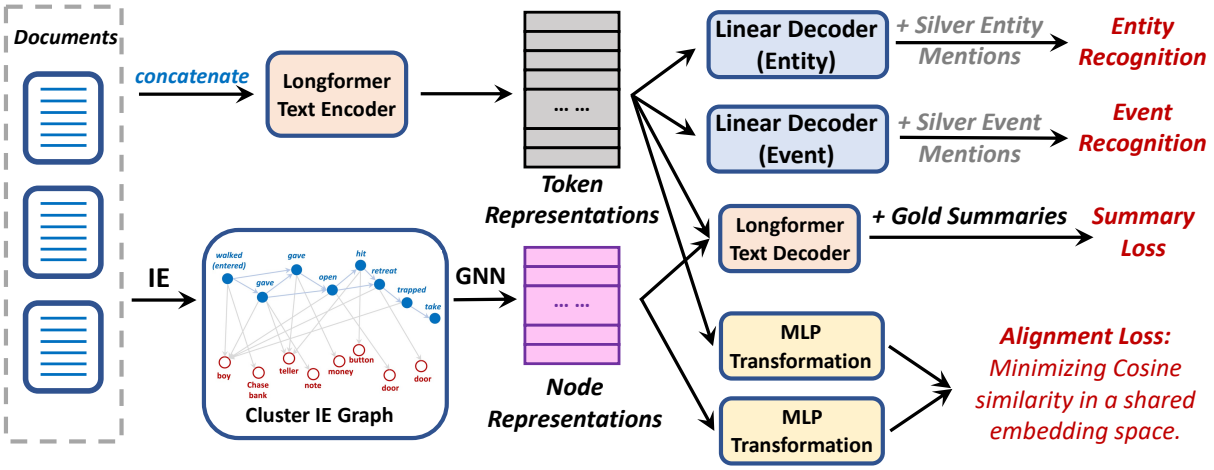

Figure 2: An overview of our IE-enhanced summarization pipeline. We first truncate and concatenate all documents in the cluster and feed them into the text encoder to obtain token representations. Meanwhile, we use a cross-document IE system to generate a cluster-level IE graph, and then use a GNN to get the node representations. During training, in addition to minimizing the distance between the generated and the reference summaries, we further use an entity and event recognition task and a node-text alignment loss to take advantage of the guidance from IE and improve MDS performance.

tions and event triggers as nodes from each document in the cluster. We then perform relation extraction, event argument role labeling, and event-event temporal relation extraction to add edges and to obtain a complete IE graph for each document. As shown in Figure 1, an example extracted event mention could be a "*Transport*" event triggered by "*walked*" with two event arguments "*boy*" and "*Chase bank*", where all such events are connected to form a unified document-level IE graph. To further connect document-level IE results into a cross-document IE graph, we then perform cross-document entity and event coreference resolution.[3] We merge all coreferenced entity and event nodes with their corresponding edges to form a cross-document IE graph. Specifically, if two nodes are labeled as the same entity or event, we merge these two nodes into a unified node and connect all related edges to it. It is worth noting that our framework does not rely on a specific IE systems and/or schema. Hence any form of structured IE outputs will work with our proposed method.

Notation Each node $v \in \mathcal{V}$ could be an entity or event trigger. We use $E = \{e_1, e_2, \cdots, e_{|E|}\}$ and $T = \{t_1, t_2, \cdots, t_{|T|}\}$ to denote the set of entities and event triggers respectively, where each e_i and t_i also act as a node in $\mathcal{V}$. Accordingly, there are three types of edges in $\mathcal{E}$ and we use p_{ij}, q_{ij}, and r_{ij} to represent the "*event-event*" temporal relations, "*event-entity*" argument roles, and "*entity-entity*" relations respectively. As shown in Figure 1, each blue node represents an event trigger (e.g., gave) while each brown node is an entity (e.g., bank), where the unique event triggers and entities are highlighted. The IE results include "*entity-entity*" relations connecting two different entities (e.g., <$button, door$>), "*event-argument*" links connecting an event trigger and an entity mention (e.g., <$gave, teller$>), and "*event-event*" temporal relations connecting two events (e.g., <$hit, retreat$>).

[3]The entity mentions extracted from ReFinED are merged according to the Wikidata IDs, while the event coreference resolution is done by a neural model (Lai et al., 2021).

3 Approach

In this paper, our main goal is to improve multi-document summarization (MDS) with the extracted cross-document IE graph. As illustrated in Figure 2, we first concatenate all documents in a cluster and feed this concatenated input into a Longformer encoder (Beltagy et al., 2020) that is capable of handling long text sequences. We also use the cross-document IE system to obtain a cluster-level IE graph, as shown in the example highlighted in Figure 1, and use a graph attention network to obtain the node representations. During training, in addition to the cross-entropy summary loss between the generated and the ground-truth summaries, we propose two additional novel training objectives: (1) an *entity and event recognition* task that makes the model aware of the locations of important events and entities; and (2) an *alignment loss* between the IE graph nodes and their corresponding text spans

to ensure that they are factually consistent in the latent space. We will go into details of our model design in the following sections.

3.1 Document Encoding

To handle the long input sequences, we use the encoder of the pre-trained *PRIMERA* (Xiao et al., 2022) model which is continually pre-trained from the Longformer-Encoder-Decoder (*LED*) model (Beltagy et al., 2020) to encode the documents and obtain the token representations $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots\}$. We truncate each document to the size of L_{max}/N (where N is the number of documents in the cluster), and concatenate all documents with a special token [doc-sep] to fit the maximum input length L_{max} of the *LED* model.[4]

$$\{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots\} = \text{Enc}\left(D_1, D_2, \cdots, D_N\right).$$

Similar to the work of Xiao et al. (2022), we assign the global attention on the [doc-sep] tokens to make sure that the model is aware of the document boundaries and that it analyzes the relationships between the documents.

In addition to directly encoding the documents, we also use the cross-document IE system described in Section 2.1 to extract a cross-document IE graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Similar to Zhang and Ji (2021), we use an edge-conditioned graph attention network to encode the entity nodes E and event nodes T respectively. The initial node representations of entities and events are computed by the average of the representations over all tokens in the entity mention or event trigger.

$$\boldsymbol{e}_i = \frac{1}{|e_T - e_S|}\sum_{j=e_S}^{e_T-1}\boldsymbol{w}_j, \;\; \boldsymbol{t}_i = \frac{1}{|t_T - t_S|}\sum_{j=t_S}^{t_T-1}\boldsymbol{w}_j,$$

where $[e_T, e_S]$ and $[t_T, t_S]$ denote the entity and event trigger spans respectively. After initializing the node embeddings, the updated entity embeddings are computed as follows:

$$\boldsymbol{e}_i^{L+1} = \boldsymbol{e}_i^{L} + \gamma \cdot \sum_{j\in\mathcal{N}_i}\alpha_{ij}\boldsymbol{f}_n\left(\boldsymbol{v}_j^{L}\right).$$

In this equation, $\boldsymbol{f}_n(\cdot)$ is a linear transformation layer and γ is a hyper-parameter controlling the level of neighborhood aggregation, where a larger γ means more information from the neighbors is incorporated when updating the node representations. The attention weights α_{ij} are determined by the node pair and the type of the edge connecting the pair of nodes.

$$\alpha_{ij} = \frac{\exp\left(\text{MLP}\left([\boldsymbol{v}_j, \boldsymbol{r}_{ij}, \boldsymbol{v}_i]\right)\right)}{\sum_{k=1}^{\mathcal{N}_i}\exp\left(\text{MLP}\left([\boldsymbol{v}_k, \boldsymbol{r}_{kj}, \boldsymbol{v}_i]\right)\right)},$$

where $\boldsymbol{r}_{ij}$ and $\boldsymbol{r}_{kj}$ are from a pre-initialized edge embedding matrix which could be optimized during training.[5] The event trigger embeddings are computed in the same way as the entities do. We use the node representations from the final layer as the output node representations.

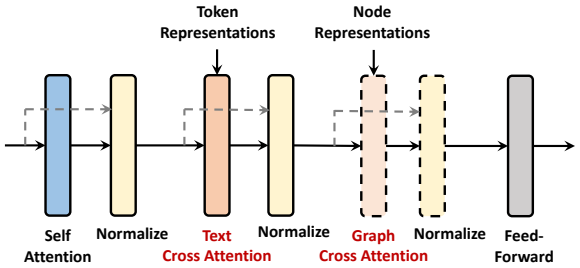

Figure 3: An example of the decoder layer components with graph cross-attention mechanism, where the dashed components are newly initialized weights while others are intialized from pre-trained weights.

3.2 Summary Generation

We use the pre-trained *LED* decoder to generate the summaries based on both token and node representations. In addition to the original pre-trained cross-attention mechanism $f_T(\cdot)$ for token representations, we include another similar cross-attention mechanism $f_G(\cdot)$ after f_T in all decoder layers for the system to model the relationships between each node in the graph and each token in the generated text. We use the pre-trained weights for text cross-attention mechanism $f_T(\cdot)$ and the graph cross-attention mechanism $f_G(\cdot)$ is randomly initialized, where both of them are continually optimized during the downstream training. An illustration of the pipeline in each decoder layer is shown in Figure 3. Therefore, each summary S_i is generated in an auto-regressive manner using the *LED* decoder $\text{Dec}(\cdot)$ with both text and graph cross-attention mechanism:

$$S_i = \text{Dec}\left([\text{BOS}], \{\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots\}, \{\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots\}\right),$$

where [BOS] is the *start* token in transformer decoders. Given a set of reference summaries

[4]The maximum length L_{max} is set as 4096 in pre-trained *PRIMERA* and *LED-large* models.

[5]We only consider three edge types here: event-event temporal relations, event-entity argument relations, and entity-entity relations.

$\hat{S}_1, \cdots, \hat{S}_N$ and a set of generated summaries $S_1, \cdots, S_N$, the summary loss is defined to minimize the cross-entropy distance $f_{CE}(\cdot)$ between each pair of summary sequences.

$$\mathcal{L}_{\text{summ}} = \frac{1}{N} \sum_{i=1}^{N} f_{CE}\left(S_i, \hat{S}_i\right). \quad (1)$$

3.3 Entity and Event Recognition

The main goal of our model is to use IE results to enhance the performance of the summarization task. We first add an auxiliary entity and event recognition task to make the model more sensitive to the locations of important events and entities. This will ensure that the model will not miss these events and entities when summarizing the document. Specifically, we use a Multi-Layer Perceptron (MLP) based classifier to classify each token into three different types: [ENTITY], [EVENT] or [NONE], and we use the spans of entity mentions and event triggers extracted by our proposed IE system to provide silver-standard training signals. Each token $\boldsymbol{w}_i$ is transformed to logits $\boldsymbol{p}_i$ by an MLP classifier:

$$\boldsymbol{p}_i = \text{softmax}\left(\text{MLP}\left(\boldsymbol{w}_i\right)\right). \quad (2)$$

Given a set of input tokens $\boldsymbol{w}_1, \boldsymbol{w}_2, \cdots, \boldsymbol{w}_M$, the entity and event recognition loss is computed as:

$$\mathcal{L}_{\text{recognition}} = -\sum_{i=1}^{M} p_{ij},$$

where j is the index of the correct label for w_i.

3.4 Node and Text Alignment

Incorporating graph information for summarization could be challenging, since the IE graphs are extracted from automatic extraction systems which may introduce noise and errors. To this end, we propose a novel alignment loss to minimize the distance between node representations and their corresponding texts to ensure coordination between the graphs and summarization text. Specifically, we first use two MLPs to map the node and text representations into the same embedding space Z:

$$\boldsymbol{z}_i^w = \text{MLP}_w\left(\boldsymbol{w}_i\right), \quad \boldsymbol{z}_i^v = \text{MLP}_v\left(\boldsymbol{v}_i\right),$$

where $\boldsymbol{z}_i^w$ and $\boldsymbol{z}_i^v$ denote the representations for token w_i and node v_i in the shared embedding space. Given each node v_i and the set of its corresponding text tokens $\mathcal{W}_i$, we minimize the cosine similarity between the node embedding $\boldsymbol{v}_i$ and the average embedding of all tokens in $\mathcal{W}_i$:

$$\mathcal{L}_{\text{align}} = \sum_{v_i \in \mathcal{V}} d_{cos}\left(\boldsymbol{z}_i^v, \frac{1}{|\mathcal{W}_i|} \sum_{w_j \in \mathcal{W}_i} \boldsymbol{z}_j^w\right). \quad (3)$$

The intuition behind $\mathcal{L}_{align}$ is to ensure that the node embedding is centered around its corresponding text. This helps ensure that the graphs and input text are factually consistent with each other, thereby reducing the errors and noise propagated from the IE system. As an example in Figure 1, the latent distance between each pair of nodes and texts (e.g., the node representation of *boy* and the text representation of its corresponding tokens *16-year-old boy*) are minimized to reduce the noise of the extracted graph.

Multi-Task Training. We conduct multi-task training where the total loss is a weighted sum from Equation (1), (2), and (3). The weighting coefficients β_1,β_2, and β_3 are tunable hyper-parameters.

$$\mathcal{L} = \beta_1 \cdot \mathcal{L}_{\text{summ}} + \beta_2 \cdot \mathcal{L}_{\text{recognition}} + \beta_3 \cdot \mathcal{L}_{\text{align}}$$

4 Experiments

4.1 Data

Our experiments are conducted on three most widely-used MDS benchmarks, where the detailed dataset statistics are shown in Table 1.

Dataset	# Train / Val / Test	Docs per Cluster	Average Summary Length
Multi-News	44972 / 5622 / 5622	2.8	217
WCEP-10	8158 / 1020 / 1022	9.1	28
DUC-2004	0 / 0 / 50	10	115

Table 1: Statistics of the MDS Datasets

Multi-News. The Multi-News benchmark (Fabbri et al., 2019) is the most widely-used dataset for multi-document summarization. The summaries are long and informative news abstracts written by human editors, and the documents are extracted from multifarious news articles.

WCEP-10. The WCEP-10 (Gholipour Ghalandari et al., 2020) dataset is extracted from Wikipedia Current Event Portal, where each document cluster also describes a news event. Compared to Multi-News, the WCEP dataset has a much

larger number of documents in each cluster, and we manually reduce them to a maximum of 10 documents per cluster as previous research (Xiao et al., 2022; Parnell et al., 2022) did to obtain the WCEP-10 version of dataset. We include both Multi-News and WCEP-10 in our experiments to evaluate whether our model can stay effective in both long-summary and short-summary scenarios.

DUC-2004. There are only 50 test document clusters in DUC-2004 benchmark,[6] and we use this dataset to evaluate our model's zero-shot transfer ability. We train our model on Multi-News and directly test it on DUC-2004 since these two datasets have similar lengths of summaries.

4.2 Baselines and Implementation Details

For baselines, we mainly compare our model with state-of-the-art multi-document summarization models *PRIMERA* (Xiao et al., 2022) and *REFLECT* (Song et al., 2022). *REFLECT* only reports ROUGE scores on the Multi-News dataset and we directly use the reported scores for comparison. Besides, we also include a previous model *BART-Graph* (Pasunuru et al., 2021), which uses a linearized IE graph to improve summarization. We compare our model with it to see whether encoding the graph structurally improves the summarization performance. We also experiment with three ablation variants of our proposed model: (1) *Recognition-Only:* for the model with only the entity and event recognition loss; (2) *Alignment-Only:* for the model with only the graph encoder and the node-text alignment loss. (3) *Separate-Graphs:* for encoding the IE graphs for each document separately and using a collated matrix as the node representations. For Multi-News and WCEP-10, we train all of these models on the training set, choose the best model checkpoint based on the performance on the validation set, and test the models on the test set. For DUC-2004, we use the trained checkpoint on Multi-News dataset for evaluation, since the summary length on Multi-News is more similar to DUC-2004 compared with WCEP-10.[7]

4.3 Evaluation Metrics

Co-occurrence. Similar to previous research studies, we first include the most widely-used ***ROUGE-F1*** score which measures the overlap between the generated summaries and the reference summaries in terms of overlapping n-grams and longest common subsequence.

Factual Consistency. Intuitively, our proposed IE-enhanced summarization should improve factual consistency of the generated summary with the source documents, since the entities and events in the original documents are mined and memorized by the model through the two proposed IE enhancement loss. Therefore, we include several factuality metrics to measure the improvements in terms of factuality of the generated summaries. Specifically we use ***FactCC*** (Kryscinski et al., 2020), ***FactGraph*** (Ribeiro et al., 2022), ***EntityPrecision*** (Nan et al., 2021), **SUMMAC** (Laban et al., 2022), and **BERTSCORE** (Pagnoni et al., 2021).

Abstractiveness. To measure abstractiveness of our generated summaries, we use the MINT score (Dreyer et al., 2023), which is based on contiguous and non-contiguous extractive overlaps between summaries and their source documents. Our goal is to measure whether the novelty of the generated summary is sacrificed due to the improvements of factual consistency, e.g., by generating a more extractive summary.

4.4 Results

Table 2 shows the results of our proposed model, as well as the baselines on the three datasets. In general, the full version of our proposed model outperforms the baselines in terms of both ROUGE scores and factuality metrics while maintaining the same level of MINT scores. This shows that our model can generate high-quality summaries that are factually consistent without sacrificing any novelty. Specifically, entity and event recognition mainly improve factual consistency, while node-text alignment improves the similarity with the referenced summaries. This follows our intuition since the recognition task is mainly designed to help the model better notice the important event triggers and entity mentions, which prevents the model from hallucination and thereby improves factual consistency. On the other hand, the alignment loss can reduce the noise and errors in those extracted IE graphs, which makes the model better optimized on the ground-truth summaries.

4.5 Human Evaluation

We conduct a human evaluation on Amazon Mechanical Turk to evaluate the effect of adding our

[6] https://duc.nist.gov/duc2004/

[7] More detailed hyper-parameter settings can be found in Appendix A.

Evaluation	*Co-occurrence*			*Factual Consistency*					*Abstractiveness*
Metrics	R-1	R-2	R-L	FactCC	FactGraph	SUMMAC	Bert-P	EntityPrec	MINT
Multi-News									
REFLECT	49.3	20.0	24.8	-	-	-	-	-	-
BART-Graph	49.2	19.0	24.0	74.2	74.1	86.0	**87.3**	89.9	81.8
PRIMERA	49.9	20.9	25.8	73.1	75.0	86.2	87.0	89.3	82.1
Separate-Graphs	49.8	20.4	25.8	74.7	75.2	86.2	87.0	89.5	82.1
Recognition-Only	50.0	20.8	26.0	77.4	76.1	86.5	87.1	**91.1**	82.0
Alignment-Only	50.3	20.9	26.3	75.6	74.9	87.7	87.1	90.8	82.1
Full Model	**50.3**	**21.1**	**26.4**	**77.8**	**76.5**	**87.9**	87.1	**91.1**	82.1
WCEP-10									
PRIMERA	46.1	24.9	37.8	68.0	71.3	56.9	94.1	88.0	86.6
Separate-Graphs	46.1	24.8	37.8	69.1	71.6	57.0	94.0	89.1	86.6
Recognition-Only	46.1	24.8	**37.9**	71.2	**71.7**	57.1	94.0	91.0	86.5
Alignment-Only	**47.3**	**25.0**	**37.9**	68.5	71.4	57.6	**94.4**	90.5	86.5
Full Model	**47.3**	24.9	37.8	**71.5**	**71.7**	**57.7**	**94.4**	**91.3**	86.8
DUC-2004									
PRIMERA	32.6	6.7	16.8	53.0	48.8	77.9	84.2	79.6	70.1
Separate-Graphs	32.6	6.6	16.8	54.2	49.9	77.6	**85.1**	80.4	70.1
Recognition-Only	32.5	6.8	16.8	54.2	**51.2**	76.8	84.7	82.3	70.4
Alignment-Only	32.8	**7.2**	17.1	53.2	49.1	78.9	84.3	80.0	70.2
Full Model	**32.9**	**7.2**	**17.3**	**54.8**	**51.2**	**79.1**	85.0	**84.0**	70.1

Table 2: Evaluation results with various metrics on the three MDS datasets. We primarily compare our results with three most recent transformer-based baselines *BART-Graph*, *REFLECT*, and *PRIMERA*. We also include two variants of our own model for ablation study, where we remove the recognition loss and the alignment loss respectively and test the model on these MDS datasets.

two proposed training objectives. We randomly select 300 document clusters for each of Multi-News and WCEP and use all the 50 document clusters in DUC-2004, asking three annotators per summary to score the factual consistency: 1 for major factual errors, 2 for minor factual errors, and 3 for no factual errors. Figure 5 in the Appendix shows the annotation guidelines. We aggregate the three judgements per summary using majority voting. We follow the qualification procedure for annotators described in Dreyer et al. (2023). Table 3 shows the percentages for each factuality score, where *baseline* denotes the *PRIMERA* (Xiao et al., 2022) model and *ours* denotes our proposed model. We find that our model can substantially reduce major factual errors on all three datasets, and is able to obtain higher average factuality scores on Multi-News and DUC-2004. Particularly, on DUC-2004 where the model is directly transferred from another dataset, our model can especially outperform the baseline in terms of factual consistency.

Dataset	Major (1.0)	Minor (2.0)	No (3.0)	Avg Scores
baseline	6.0%	11.3%	82.7%	2.767
Multi-News (ours)	**4.7%**	12.7%	82.7%	**2.780**
baseline	10.7%	9.0%	80.3%	**2.697**
WCEP-10 (ours)	**9.0%**	19.3%	71.7%	2.627
baseline	22.0%	22.0%	56.0%	2.340
DUC-2004 (ours)	**18.0%**	12.0%	70.0%	**2.520**

Table 3: Human evaluation results.

4.6 Qualitative Analysis

To better understand the effects made by our proposed training objectives, we look into the prediction results and show a typical example in Figure 4, explaining how our proposed method works to improve the summaries. In this example, the document cluster is mainly talking about a shut-down incident of the Nasdaq trading market. Compared to the summary from the baseline model, our model is better at memorizing the important facts and showing them in the output summary, e.g., the exact Nasdaq Index (3631.17) when the trading was suddenly suspended. Some other facts such as *"three hours"* are also memorized by our model but ignored by the baseline model. Moreover, our model is able to generate more informative mentions of those key entities (e.g., *NYSE*), where the baseline model fails to generate a named mention and only writes *"the exchange"*.

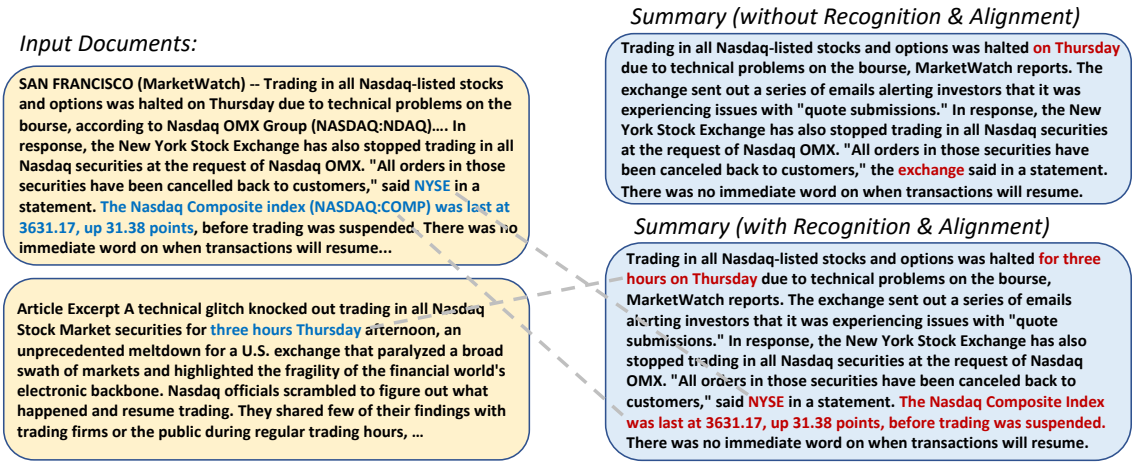

Figure 4: A qualitative example from our full model compared to the baseline *PRIMERA* model. Our model is better at preserving important facts and utilizing more informative mentions of the key entities.

5 Related Work

Multi-Document Summarization. Abstractive multi-document summarization (MDS) aims to build models to generate summaries given a set of similar documents related to the same topic. With the tremendous success of sequence-to-sequence pre-trained language models such as BART (Lewis et al., 2020) and T5 (Guo et al., 2022), finetuning on pre-trained models, like DeYoung et al. (2021); Parnell et al. (2022); Zhao et al. (2022); Moro et al. (2022); Song et al. (2022); Ernst et al. (2022), has become the primary style of methods for summarization tasks. There are also research studies on how to handle cross-document information overlap and redundancy. For example, Pasunuru et al. (2021) propose to use graph structures generated by OpenIE systems to make the model more sensitive about the main message of the document cluster. More recently, Xiao et al. (2022) propose to integrate entity overlap into the pre-training scheme, where the overlapping entities are used to select out salient sentences for pre-training.

Cross-Document Information Extraction. Information Extraction (IE) aims to extract structured representations from unstructured text, which includes various subtasks from Named Entity Recognition (Reich et al., 2022; Ayoola et al., 2022; Ding et al., 2021), to Relation Extraction (Yu et al., 2022; Tian et al., 2022), and Event Extraction (Xu et al., 2021; Yu et al., 2021) on news documents. There are also a number of research studies (Yao et al., 2021; Wu et al., 2022; Du et al., 2022) focusing on corpus-level cross-document extraction models. However, all these models still rely on cross-document entity and event coreference systems, which could bottleneck the efficiency and effectiveness of corpus-level IE models.

Joint IE and Summarization. IE and summarization share inherent similarities; both of them are designed to find the main information from an input natural language text. Therefore, it is promising to design a joint learning framework so that the two tasks could provide each other with mutual enhancement. There are some preliminary explorations of previous studies to train a model to learn IE and natural language generation (NLG) tasks jointly. For example, Li et al. (2021) train a template-based generative model for event argument extraction, and Du and Cardie (2020) propose to generate natural questions to ask the model for event extraction. However, although generation-based methods are proposed, these models are still doing a single task (IE) without multi-task settings for both IE and NLG. Recently, Lu et al. (2022) use summarization to provide indirect training signal for relation extraction tasks, however, their method is only suitable for relation extraction tasks and cannot cover general-concept IE tasks.

6 Conclusions

In this paper, we focus on improving multi-document summarization (MDS) model with cross-document Information Extraction (IE). We propose two novel training objectives – an entity and event recognition loss and a node-text alignment loss – that can help the model better utilize the signals from IE. Experimental results show that our model can generate summaries that are more factual, while not losing any abstractiveness.

7 Limitations

One limitation of our proposed method is the IE graphs are pre-extracted separately, where the IE model is not optimized during the model training and the IE results are only used as side inputs for summarization. It would be more exciting if we can really build a joint IE and Summarization model which are trained simultaneously in the pipeline, although it is very difficult since passing the gradients through a cross-document system is nearly intractable. We intend to address this limitation in our future work.

Acknowledgement

We thank the anonymous reviewers for their valuable feedback.

References

Tom Ayoola, Joseph Fisher, and Andrea Pierleoni. 2022. Improving entity disambiguation by reasoning over a knowledge base. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2899–2912, Seattle, United States. Association for Computational Linguistics.

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

Jay DeYoung, Iz Beltagy, Madeleine van Zuylen, Bailey Kuehl, and Lucy Wang. 2021. MSˆ2: Multi-document summarization of medical studies. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7494–7513, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ning Ding, Guangwei Xu, Yulin Chen, Xiaobin Wang, Xu Han, Pengjun Xie, Haitao Zheng, and Zhiyuan Liu. 2021. Few-NERD: A few-shot named entity recognition dataset. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3198–3213, Online. Association for Computational Linguistics.

Markus Dreyer, Mengwen Liu, Feng Nan, Sandeep Atluri, and Sujith Ravi. 2023. Evaluating the tradeoff between abstractiveness and factuality in abstractive summarization. In *Proceedings of the 2023 Conference of the European Chapter of the Association for Computational Linguistics*.

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 671–683, Online. Association for Computational Linguistics.

Xinya Du, Zixuan Zhang, Sha Li, Pengfei Yu, Hongwei Wang, Tuan Lai, Xudong Lin, Ziqi Wang, Iris Liu, Ben Zhou, Haoyang Wen, Manling Li, Darryl Hannan, Jie Lei, Hyounghun Kim, Rotem Dror, Haoyu Wang, Michael Regan, Qi Zeng, Qing Lyu, Charles Yu, Carl Edwards, Xiaomeng Jin, Yizhu Jiao, Ghazaleh Kazeminejad, Zhenhailong Wang, Chris Callison-Burch, Mohit Bansal, Carl Vondrick, Jiawei Han, Dan Roth, Shih-Fu Chang, Martha Palmer, and Heng Ji. 2022. RESIN-11: Schema-guided event prediction for 11 newsworthy scenarios. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: System Demonstrations*, pages 54–63, Hybrid: Seattle, Washington + Online. Association for Computational Linguistics.

Ori Ernst, Avi Caciularu, Ori Shapira, Ramakanth Pasunuru, Mohit Bansal, Jacob Goldberger, and Ido Dagan. 2022. Proposition-level clustering for multi-document summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1765–1779, Seattle, United States. Association for Computational Linguistics.

Alexander Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy. Association for Computational Linguistics.

Demian Gholipour Ghalandari, Chris Hokamp, Nghia The Pham, John Glover, and Georgiana Ifrim. 2020. A large-scale multi-document summarization dataset from the Wikipedia current events portal. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1302–1308, Online. Association for Computational Linguistics.

Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. LongT5: Efficient text-to-text transformer for long sequences. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 724–736, Seattle, United States. Association for Computational Linguistics.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. 2020. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346, Online. Association for Computational Linguistics.

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177.

Tuan Lai, Heng Ji, Trung Bui, Quan Hung Tran, Franck Dernoncourt, and Walter Chang. 2021. A context-dependent gated module for incorporating symbolic semantics into event coreference resolution. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3491–3499, Online. Association for Computational Linguistics.

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

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 894–908, Online. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Keming Lu, I Hsu, Wenxuan Zhou, Mingyu Derek Ma, Muhao Chen, et al. 2022. Summarization as indirect supervision for relation extraction. *arXiv preprint arXiv:2205.09837*.

Gianluca Moro, Luca Ragazzi, Lorenzo Valgimigli, and Davide Freddi. 2022. Discriminative marginalized probabilistic neural method for multi-document summarization of medical literature. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 180–189, Dublin, Ireland. Association for Computational Linguistics.

Feng Nan, Ramesh Nallapati, Zhiguo Wang, Cicero Nogueira dos Santos, Henghui Zhu, Dejiao Zhang, Kathleen McKeown, and Bing Xiang. 2021. Entity-level factual consistency of abstractive text summarization. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2727–2733, Online. Association for Computational Linguistics.

Artidoro Pagnoni, Vidhisha Balachandran, and Yulia Tsvetkov. 2021. Understanding factuality in abstractive summarization with FRANK: A benchmark for factuality metrics. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4812–4829, Online. Association for Computational Linguistics.

Jacob Parnell, Inigo Jauregi Unanue, and Massimo Piccardi. 2022. A multi-document coverage reward for RELAXed multi-document summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5112–5128, Dublin, Ireland. Association for Computational Linguistics.

Ramakanth Pasunuru, Mengwen Liu, Mohit Bansal, Sujith Ravi, and Markus Dreyer. 2021. Efficiently summarizing text and graph encodings of multi-document clusters. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4768–4779, Online. Association for Computational Linguistics.

Aaron Reich, Jiaao Chen, Aastha Agrawal, Yanzhe Zhang, and Diyi Yang. 2022. Leveraging expert guided adversarial augmentation for improving generalization in named entity recognition. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1947–1955, Dublin, Ireland. Association for Computational Linguistics.

Leonardo F. R. Ribeiro, Mengwen Liu, Iryna Gurevych, Markus Dreyer, and Mohit Bansal. 2022. FactGraph: Evaluating factuality in summarization with semantic graph representations. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3238–3253, Seattle, United States. Association for Computational Linguistics.

Yun-Zhu Song, Yi-Syuan Chen, and Hong-Han Shuai. 2022. Improving multi-document summarization through referenced flexible extraction with credit-awareness. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1667–1681, Seattle, United States. Association for Computational Linguistics.

Yuanhe Tian, Yan Song, and Fei Xia. 2022. Improving relation extraction through syntax-induced pre-training with dependency masking. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1875–1886, Dublin, Ireland. Association for Computational Linguistics.

Xueqing Wu, Kung-Hsiang Huang, Yi Fung, and Heng Ji. 2022. Cross-document misinformation detection based on event graph reasoning. In *Proceedings of the 2022 Conference of the North American Chapter*

of the Association for Computational Linguistics: Human Language Technologies, pages 543–558, Seattle, United States. Association for Computational Linguistics.

Wen Xiao, Iz Beltagy, Giuseppe Carenini, and Arman Cohan. 2022. PRIMERA: Pyramid-based masked sentence pre-training for multi-document summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5245–5263, Dublin, Ireland. Association for Computational Linguistics.

Runxin Xu, Tianyu Liu, Lei Li, and Baobao Chang. 2021. Document-level event extraction via heterogeneous graph-based interaction model with a tracker. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3533–3546, Online. Association for Computational Linguistics.

Yuan Yao, Jiaju Du, Yankai Lin, Peng Li, Zhiyuan Liu, Jie Zhou, and Maosong Sun. 2021. CodRED: A cross-document relation extraction dataset for acquiring knowledge in the wild. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4452–4472, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiaxin Yu, Deqing Yang, and Shuyu Tian. 2022. Relation-specific attentions over entity mentions for enhanced document-level relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1523–1529, Seattle, United States. Association for Computational Linguistics.

Pengfei Yu, Heng Ji, and Prem Natarajan. 2021. Lifelong event detection with knowledge transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5278–5290, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zixuan Zhang and Heng Ji. 2021. Abstract Meaning Representation guided graph encoding and decoding for joint information extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 39–49, Online. Association for Computational Linguistics.

Chao Zhao, Tenghao Huang, Somnath Basu Roy Chowdhury, Muthu Kumar Chandrasekaran, Kathleen McKeown, and Snigdha Chaturvedi. 2022. Read top news first: A document reordering approach for multi-document news summarization. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 613–621, Dublin, Ireland. Association for Computational Linguistics.

A Experiment Details

We list our detailed hyper-parameter settings for training our model on each of the datasets in Table 4 and Table 5, where each hyper-parameter is determined based on grid search among 5 candidate values. We train our model on 8 NVIDIA V100 GPUs with 32GB memory, and the total training time is about 7 hours for Multi-News and 3 hours for WCEP-10.

Hyper-parameters	Values
Num of features for each node	1,024
Num of GNN layers	1
Message Passing Level γ	0.01
Weights of the losses β_1, β_2, β_3	1.0, 0.2, 0.2
Learning Rate	3e-5
Batch Size	16
Maximum Length of Generated Summaries	256
Maximum Training Steps	25,000
Warm-up Steps	2,500
Beam Size for Generation	5

Table 4: Detailed hyper-parameter settings for model training on Multi-News.

Hyper-parameters	Values
Num of features for each node	1,024
Num of GNN layers	1
Message Passing Level γ	0.005
Weights of the losses β_1, β_2, β_3	1.0, 0.1, 0.1
Learning Rate	3e-5
Batch Size	16
Maximum Length of Generated Summaries	50
Maximum Training Steps	5,000
Warm-up Steps	500
Beam Size for Generation	5

Table 5: Detailed hyper-parameter settings for model training on WCEP-10.

B Annotation Guidelines

We use Amazon MTurk to do human evaluation, where the detailed annotation guidelines for human evaluators are shown in Figure 5.

Please evaluate how consistent the blue sentence from the summary is with respect to the information in the articles.

- **1 star**: **Major error.** The blue sentence contains a **major** factual error or multiple minor errors.
- **2 stars**: **Minor error.** The blue sentence contains one **minor** factual error.
- **3 stars**: **No errors.** The blue sentence contains no factual errors.

Major errors:

- **Definition:** Readers knowledgeable in the space would likely recognize the error in the blue sentence. If printed in a newspaper, the newspaper would have to print a correction or retraction to maintain its reputation.
- **Example 1:** The blue sentence might say "A fire broke out in Seattle", but an article says it broke out in Portland.
- **Example 2:**The blue sentence might say "the Republicans won the election", but the articles indicate that the Democrats won instead.
- **Example 3:** The blue sentence might say that "A fire broke out at 2am", but the articles don't mention the time when the fire broke out, or they mention it was during the day.

Minor errors:

- **Definition:** Most readers would not notice the error or find it less important. If printed in a newspaper, the newspaper may not need to print a correction.
- **Example 1:** The blue sentence might say that a celebrity couple shared a video of their daughter, but the articles says that the *mom* shared the video.
- **Example 2:** The blue sentence might misspell a name.
- **Example 3:** The blue sentence might contain a repetition that is not literally correct, e.g., "the soccer team won the game 1-2 and 1-2".
- **Example 4:** The blue sentence might say "Lady Celia Vestey was one of Prince Harry's six godmothers", but it should be *godparents*.
- **Example 5:** The blue sentence might say "The Game Awards will take place in Los Angeles and London", but the articles say they take place "virtually from Los Angeles and London".

Meaning of the colors:

- **Summary**: The gray sentences in the summary are displayed to give context only. Please evaluate the blue sentence only.
- **Articles**: The sentences in the articles have green background color to help you find information more quickly. Article sentences with darker green background color are more related to the blue sentence. The least related sentence have been removed, indicated by three dots (...).

Figure 5: Annotation instructions to annotate factual consistency on Mechanical Turk.