A Hierarchical Network for Abstractive Meeting Summarization with Cross-Domain Pretraining

Chenguang Zhu*, **Ruochen Xu***, **Michael Zeng, Xuedong Huang**
Microsoft Cognitive Services Research Group
{chezhu,ruox,nzeng,xdh}@microsoft.com

Abstract

With the abundance of automatic meeting transcripts, meeting summarization is of great interest to both participants and other parties. Traditional methods of summarizing meetings depend on complex multi-step pipelines that make joint optimization intractable. Meanwhile, there are a handful of deep neural models for text summarization and dialogue systems. However, the semantic structure and styles of meeting transcripts are quite different from articles and conversations. In this paper, we propose a novel abstractive summary network that adapts to the meeting scenario. We design a hierarchical structure to accommodate long meeting transcripts and a role vector to depict the difference among speakers. Furthermore, due to the inadequacy of meeting summary data, we pretrain the model on large-scale news summary data. Empirical results show that our model outperforms previous approaches in both automatic metrics and human evaluation. For example, on ICSI dataset, the ROUGE-1 score increases from 34.66% to 46.28%.

1 Introduction

Meetings are a very common forum where people exchange ideas, make plans, and share information. With the ubiquity of automatic speech recognition systems come vast amounts of meeting transcripts. Therefore, the need to succinctly summarize the content of a meeting naturally arises.

Several methods of generating summaries for meetings have been proposed (Mehdad et al., 2013; Murray et al., 2010; Wang and Cardie, 2013; Oya et al., 2014; Shang et al., 2018; Li et al., 2019). As Murray et al. (2010) points out, users prefer abstractive meeting summaries to extractive summaries. While these methods are mostly abstractive, they require complicated multi-stage machine learning pipelines, such as template generation, sentence clustering, multi-sentence compression, candidate sentence generation and ranking. As these approaches are not end-to-end optimisable, it is hard to jointly improve various parts in the pipeline to enhance the overall performance. Moreover, some components, e.g., template generation, require extensive human involvement, rendering the solution not scalable or transferrable.

Meanwhile, many end-to-end systems have been successfully employed to tackle document summarization, such as the pointer-generator network (See et al., 2017), reinforced summarization network (Paulus et al., 2018) and memory network

Meeting Transcript (163 turns)
... PM: ... another point is we have to skip the **teletext**, because in the world of upcoming internet we think **teletext** is going to be a thing of the past. ID: ... first about how it works. It's really simple. Everybody knows how a **remote** works. The user presses a button. The **remote** determines what button it is, PM: ... Few buttons, we talked about that. **Docking station**, **LCD**. general functions And default materials... ...
Summary from our model (23 sentences)
... The Project Manager announced that the project would not include a **teletext** feature. The Industrial Designer gave a presentation of the functions of the **remote**. The group decided on features to include in the remote, to include an **LCD** screen, and a **docking station** to change the layout of the interface. ...

Table 1: Example excerpt of a meeting transcript and the summary generated by our model in AMI dataset. Keywords are in bold. PM (program manager) and ID (industrial designer) are roles of the speakers. The meeting transcript contains word errors and grammatical glitches as it is the result from the automatic speech recognition system.

* Equal contribution

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 194–203

(Jiang and Bansal, 2018). These deep learning methods can effectively generate abstractive document summaries by directly optimizing pre-defined goals.

However, the meeting summarization task inherently bears a number of challenges that make it more difficult for end-to-end training than document summarization. We show an example of a meeting transcript from the AMI dataset and the summary generated by our model in Table 1.

First, the transcript and summary of a single meeting are usually much longer than those of a document. For instance, in CNN/Daily Mail dataset (Hermann et al., 2015), there are on average 781 tokens per article and 56 tokens per summary, while AMI meeting corpus contains meetings with 4,757 tokens per transcript and 322 tokens per summary on average. And the structure of a meeting transcript is very distinct from news articles. These challenges all prevent existing news summarization models to be successfully applied to meetings.

Second, a meeting is carried out between multiple participants. The different semantic styles, standpoints, and roles of each participant all contribute to the heterogeneous nature of the meeting transcript.

Third, compared with news, there is very limited labelled training data for meeting summary (137 meetings in AMI v.s. 312K articles in CNN/DM). This is due to the privacy of meetings and the relatively high cost of writing summaries for long transcripts.

To tackle these challenges, we propose an end-to-end deep learning framework, **H**ierarchical **M**eeting summarization **Net**work (HMNet). HMNet leverages the encoder-decoder transformer architecture (Vaswani et al., 2017) to produce abstractive summaries based on meeting transcripts. To adapt the structure to meeting summarization, we propose two major design improvements.

First, as meeting transcripts are usually lengthy, a direct application of the canonical transformer structure may not be feasible. For instance, conducting the multi-head self-attention mechanism on a transcript with thousands of tokens is very time consuming and may cause memory overflow problem. Therefore, we leverage a hierarchical structure to reduce the burden of computing. As a meeting consists of utterances from different participants, it forms a natural multi-turn hierarchy. Thus, the hierarchical structure carries out both token-level understanding within each turn and turn-level understanding across the whole meeting. During summary generation, HMNet applies attention to both levels of understanding to ensure that each part of the summary stems from different portions of the transcript with varying granularities.

Second, to accommodate multi-speaker scenario, HMNet incorporates the role of each speaker[1] to encode different semantic styles and standpoints among participants. For example, a program manager usually emphasizes the progress of the project while a user interface designer tends to focus on user experience. In HMNet, we train a role vector for each meeting participant to represent the speaker's information during encoding. This role vector is appended to the turn-level representation for later decoding.

To tackle the problem of insufficient training data for meeting summarization, we leverage the idea of pretraining (Devlin et al., 2018). We collect summarization data from the news domain and convert them into the meeting format: a group of several news articles forms a multi-person meeting and each sentence becomes a turn. The turns are reshuffled to simulate a mixed order of speakers. We pretrain the HMNet model on the news task before finetuning it on meeting summarization. Empirical results show that this cross-domain pretraining can effectively enhance the model quality.

To evaluate our model, we employ the widely used AMI and ICSI meeting corpus (McCowan et al., 2005; Janin et al., 2003). Results show that HMNet significantly outperforms previous meeting summarization methods. For example, on ICSI dataset, HMNet achieves 11.62 higher ROUGE-1 points, 2.60 higher ROUGE-2 points, and 6.66 higher ROUGE-SU4 points compared with the previous best result. Human evaluations further show that HMNet generates much better summaries than baseline methods. We then conduct ablation studies to verify the effectiveness of different components in our model.

2 Problem Formulation

We formalize the problem of meeting summarization as follows. The input consists of meeting transcripts $\mathcal{X}$ and meeting participants $\mathcal{P}$. Suppose there are s meetings in total. The tran-

[1]Both datasets in experiments only provide role information for each participant. In real applications, we can use a vector to represent each participant when a personal identifier is available.

scripts are $\mathcal{X} = \{X_1, ..., X_s\}$. Each meeting transcript consists of multiple turns, where each turn is the utterance of a participant. Thus, $X_i = \{(p_1, u_1), (p_2, u_2), ..., (p_{L_i}, u_{L_i})\}$, where $p_j \in \mathcal{P}, 1 \leq j \leq L_i$, is a participant and $u_j = (w_1, ..., w_{l_j})$ is the tokenized utterance from p_j. The human-labelled summary for meeting X_i, denoted by Y_i, is also a sequence of tokens. For simplicity, we will drop the meeting index subscript. So the goal of the system is to generate meeting summary $Y = (y_1, ..., y_n)$ given the transcripts $X = \{(p_1, u_1), (p_2, u_2), ..., (p_m, u_m)\}$.

3 Model

Our hierarchical meeting summarization network (HMNet) is based on the encoder-decoder transformer structure (Vaswani et al., 2017), and its goal is to maximize the conditional probability of meeting summary Y given transcript X and network parameters θ: $P(Y|X;\theta)$.

3.1 Encoder

3.1.1 Role Vector

Meeting transcripts are recorded from various participants, who may have different semantic styles and viewpoints. Therefore, the model has to take the speaker's information into account while generating summaries.

To incorporate the participants' information, we integrate the *speaker role* component. In the experiments, each meeting participant has a distinct role, e.g., program manager, industrial designer. For each role, we train a vector to represent it as a fixed-length vector $r_p, 1 \leq p \leq P$, where P is the number of roles. Such distributed representation for a role/person has been proved to be useful for sentiment analysis (Chen et al., 2016). This vector is appended to the embedding of the speaker's turn (Section 3.1.2). According to the results in Section 4.5, the vectorized representation of speaker roles plays an important part in boosting the performance of summarization.

This idea can be extended if richer data is available in practice:

- If an organization chart of participants is available, we can add in representations of the relationship between participants, e.g., manager and developers, into the network.
- If there is a pool of registered participants, each participant can have a personal vector which acts as a user portrait and evolves as more data about this user is collected.

3.1.2 Hierarchical Transformer

Transformer. Recall that a transformer block consists of a multi-head attention layer and a feed-forward layer, both followed by layer-norm with residuals: LayerNorm$(x + Layer(x))$, where $Layer$ can be the attention or feed-forward layer (Vaswani et al., 2017).

As the attention mechanism is position agnostic, we append positional encoding to input vectors:

$$\text{PE}_{(i,2j)} = \sin(i/10000^{\frac{2j}{d}}) \quad (1)$$

$$\text{PE}_{(i,2j+1)} = \cos(i/10000^{\frac{2j}{d}}), \quad (2)$$

where $\text{PE}_{(i,j)}$ stands for the j-th dimension of positional encoding for the i-th word in input sequence. We choose sinusoidal functions as they can extend to arbitrary input length during inference.

In summary, a transformer block on a sequence of n input embeddings can generate n output embeddings of the same dimension as input. Thus, multiple transformer blocks can be sequentially stacked to form a transformer network:

$$\text{Transformer}(\{x_1, ..., x_n\}) = \{y_1, ..., y_n\} \quad (3)$$

Long transcript problem. As the canonical transformer has the attention mechanism, its computational complexity is quadratic in the input length. Thus, it struggles to handle very long sequences, e.g. 5,000 tokens. However, meeting transcripts are usually fairly long, consisting of thousands of tokens.

We note that meetings come with a natural multi-turn structure with a reasonable number of turns, e.g. 289 turns per meeting on average in AMI dataset. And the number of tokens in a turn is much less than that in the whole meeting. Therefore, we employ a two-level transformer structure to encode the meeting transcript.

Word-level Transformer. The word-level transformer processes the token sequence of one turn in the meeting. We encode each token in one turn using a trainable embedding matrix $\mathcal{D}$. Thus, the j-th token in the i-th turn, $w_{i,j}$, is associated with a uniform length vector $\mathcal{D}(w_{i,j}) = g_{i,j}$. To incorporate syntactic and semantic information, we also train two embedding matrices to represent the part-of-speech (POS) and entity (ENT) tags. Therefore, the token $w_{i,j}$ is represented by the vector

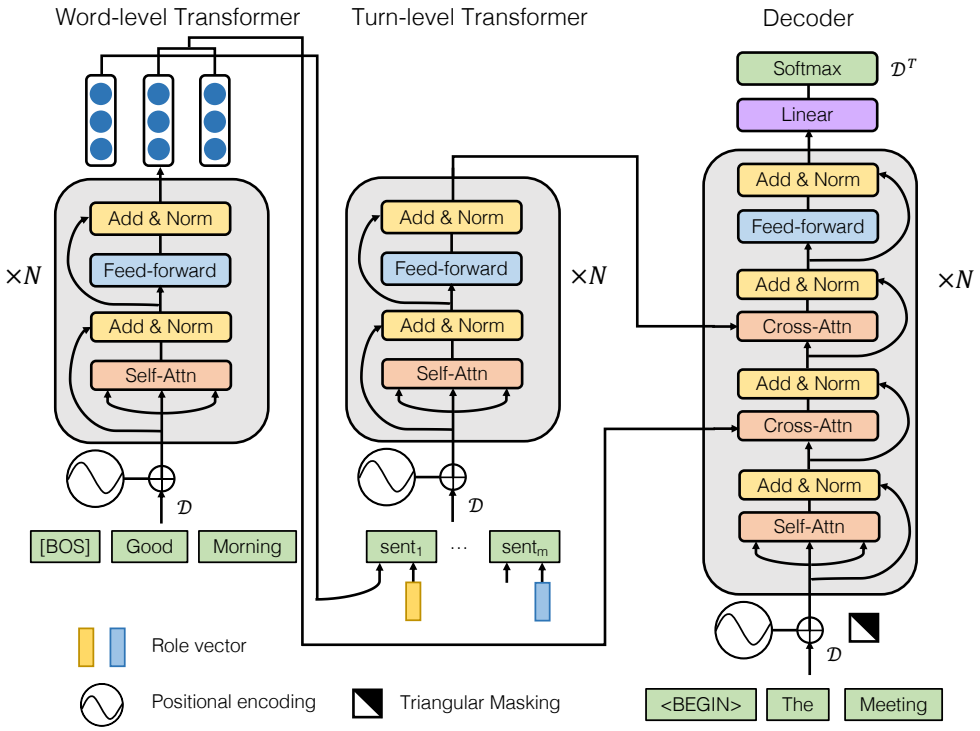

Figure 1: Hierarchical Meeting Summary Network (HMNet) model structure. [BOS] is the special start token inserted before each turn, and its encoding is used in turn-level transformer encoder. Other tokens' encodings enter the cross-attention module in decoder.

$x_{i,j} = [g_{i,j}; POS_{i,j}; ENT_{i,j}]$. Note that we add a special token $w_{i,0}$=[BOS] before the sequence to represent the beginning of a turn. Then, we denote the output of the word-level transformer as follows: Word-Transformer$(\{x_{i,0}, ..., x_{i,L_i}\}) = \{x^{\mathcal{W}}_{i,0}, ..., x^{\mathcal{W}}_{i,L_i}\}$.

Turn-level Transformer. The turn-level transformer processes the information of all m turns in a meeting. To represent the i-th turn, we employ the output embedding of the special token [BOS] from the word-level transformer, i.e. $x^{\mathcal{W}}_{i,0}$. Furthermore, we concatenate it with the role vector of the speaker for this turn, p_i. It follows that the output of the turn-level transformer is: Turn-Transformer$(\{[x^{\mathcal{W}}_{1,0}; p_1], ..., [x^{\mathcal{W}}_{m,0}; p_m]\}) = \{x^{\mathcal{T}}_1, ..., x^{\mathcal{T}}_m\}$.

3.2 Decoder

The decoder is a transformer to generate the summary tokens. The input to the decoder transformer contains the $k-1$ previously generated summary tokens $\hat{y}_1, ..., \hat{y}_{k-1}$. Each token is represented by a vector using the same embedding matrix $\mathcal{D}$ as the encoder, $\mathcal{D}(\hat{y}_i) = g_i$.

The decoder transformer uses a lower triangular mask to prevent the model to look at future tokens. Moreover, the transformer block includes two cross-attention layers. After self-attention, the embeddings first attend with token-level outputs $\{x^{\mathcal{W}}_{i,j}\}^{m,L_i}_{i=1,j=1}$, and then with turn-level outputs $\{x^{\mathcal{T}}_i\}^m_{i=1}$, each followed by layer-norm. This makes the model attend to different parts of the inputs with varying scales at each inference step.

The output of the decoder transformer is denoted as: Decoder-Transformer$(\{g_1, ..., g_{k-1}\}) = \{v_1, ..., v_{k-1}\}$.

To predict the next token $\hat{y}_k$, we reuse the weight of embedding matrix $\mathcal{D}$ to decode v_{k-1} into a probability distribution over the vocabulary:

$$P(\hat{y}_k|\hat{y}_{<k}, X) = \text{softmax}(v_{k-1}\mathcal{D}^T) \quad (4)$$

We illustrate the Hierarchical Meeting summary Network (HMNet) in Fig. 1.

Training. During training, we seek to minimize the cross entropy:

$$L(\theta) = -\frac{1}{n}\sum_{k=1}^{n} logP(y_k|y_{<k}, X) \quad (5)$$

We use teacher-forcing in decoder training, i.e. the decoder takes ground-truth summary tokens as input.

Inference. During inference, we use beam search to select the best candidate. The search starts with the special token ⟨BEGIN⟩. We employ the commonly used trigram blocking (Paulus et al., 2018): during beam search, if a candidate word would create a trigram that already exists in the previously generated sequence of the beam, we forcibly set the word's probability to 0. Finally, we select the summary with the highest average log-likelihood per token.

3.3 Pretraining

As there is limited availability of meeting summarization data, we propose to utilize summary data from the news domain to pretrain HMNet. This can warm up model parameters on summarization tasks. However, the structure of news articles is very different from meeting transcripts. Therefore, we transform news articles into the meeting format.

We concatenate every M news articles into an M-people meeting, and treat each sentence as a single turn. The sentences from article i is considered to be utterances from the i-th speaker, named as [Dataset-i]. For instance, for each XSum meeting, the speakers' names are [XSum-1] to [XSum-M]. To simulate the real meeting scenario, we randomly shuffle all the turns in these pseudo meetings. The target summary is the concatenation of the M summaries.

We pretrain HMNet model with a large collection of news summary data (details in Section 4.1), and then finetune it on real meeting summary task.

4 Experiment

4.1 Datasets

We employ the widely used AMI (McCowan et al., 2005) and ICSI (Janin et al., 2003) meeting corpora. The two datasets contain meeting transcripts from automatic speech recognition (ASR), respectively. We follow Shang et al. (2018) to use the same train/development/test split: 100/17/20 for AMI and 43/10/6 for ICSI. Each meeting has an abstractive summary written by human annotators. Furthermore, each participant has an associated role, e.g. project manager, marketing expert[2]. Since there is only one speaker per role in each meeting and no other speaker identification information, we use a single role vector to model both speaker and role information simultaneously.

In AMI, there are on average 4,757 words with 289 turns in the meeting transcript and 322 words in the summary. In ICSI, there are on average 10,189 words with 464 turns in the meeting transcript and 534 words in the summary. As the transcript is produced by the ASR system, there is a word error rate of 36% for AMI and 37% for ICSI (Shang et al., 2018).

The pretraining is conduct on the news summarization datasets CNN/DailyMail (Hermann et al., 2015), NYT (Sandhaus, 2008) and XSum (Narayan et al., 2018), containing 312K, 104K and 227K article-summary pairs. We take the union of three datasets for the pretraining. We choose groups of $M = 4$ news articles to match the 4-speaker setting in AMI dataset. These converted meetings contain on average 2,812 words with 128 turns and 176 words in the summary.

4.2 Baseline models

For comparison, we select a variety of baseline systems from previous literatures: the basic baselines **Random** (Riedhammer et al., 2008) and **Copy from Train**, which randomly copies a summary from the training set as the prediction[3]; the template-based method **Template** (Oya et al., 2014); the ranking systems **TextRank** (Mihalcea and Tarau, 2004) and **ClusterRank** (Garg et al., 2009); the unsupervised method **UNS**; the document summarization model **PGNet**[4] (See et al., 2017); and the multi-modal model **MM** (Li et al., 2019).

In addition, we implement the baseline model **Extractive Oracle**, which concatenates top sentences with the highest ROUGE-1 scores with the golden summary. The number of sentences is determined by the average length of golden summary: 18 for AMI and 23 for ICSI.

4.3 Metrics

Following Shang et al. (2018), we employ ROUGE-1, ROUGE-2 and ROUGE-SU4 metrics (Lin, 2004) to evaluate all meeting summarization models. These three metrics respectively evaluate the accuracy of unigrams, bigrams, and unigrams plus

[2]We select the Scenario Meetings of AMI as in Shang et al. (2018)

[3]To reduce variance, for each article, we randomly sample 50 times and report the averaged metrics.

[4]PGNet treats the whole meeting transcript as an article and generates the summary.

	AMI			ICSI		
Model	ROUGE-1	R-2	R-SU4	ROUGE-1	R-2	R-SU4
Random	35.13	6.26	13.17	29.28	3.78	10.29
Template	31.50	6.80	11.40	/	/	/
TextRank	35.25	6.9	13.62	29.7	4.09	10.64
ClusterRank	35.14	6.46	13.35	27.64	3.68	9.77
UNS	37.86	7.84	14.71	31.60	4.83	11.35
Extractive Oracle	39.49	9.65	13.20	34.66	8.00	10.49
PGNet	40.77	14.87	18.68	32.00	7.70	12.46
Copy from Train	43.24	12.15	14.01	34.65	5.55	10.65
MM (TopicSeg+VFOA)*	**53.29**	13.51	/	/	/	/
MM (TopicSeg)*	51.53	12.23	/	/	/	/
HMNet	53.02	**18.57****	**24.85****	**46.28****	**10.60****	**19.12****

Table 2: ROUGE-1, ROUGE-2, ROUGE-SU4 scores of generated summary in AMI and ICSI datasets. Numbers in bold are the overall best result. * The two baseline MM models require additional human annotations of topic segmentation and visual signals from cameras. ** Results are statistically significant at level 0.05.

skip-bigrams with a maximum skip distance of 4. These metrics have been shown to highly correlate with the human judgment (Lin, 2004).

4.4 Implementation Details

We employ spaCy (Honnibal and Johnson, 2015) as the word tokenizer and embed POS and NER tags into 16-dim vectors. The dimension of the role vector is 32.

All transformers have 6 layers and 8 heads in attention. The dimension for each word is 512 and thus the input and output dimensions of transformers d_{model} are 512 for the decoder, $512 + 16 + 16 = 544$ for the word-level transformer, and $512 + 16 + 16 + 32 = 576$ for the turn level transformer. For all transformers, the inner-layer always has dimensionality $d_{ff} = 4 \times d_{model}$. HMNet has 204M parameters in total. We use a dropout probability of 0.1 on all layers.

We pretrain HMNet on news summarization data using the RAdam optimizer (Liu et al., 2020) with $\beta_1 = 0.9$, $\beta_2 = 0.999$. The initial learning rate is set to $1e-9$ and linearly increased to 0.001 with 16000 warmup steps. For finetuning on the meeting data, the optimization setup is the same except the initial learning rate is set to 0.0001. We use gradient clipping with a maximum norm of 2 and gradient accumulation steps as 16.

4.5 Results

Table 2 shows the ROUGE scores of generated summaries in AMI and ICSI datasets. As shown, except for ROUGE-1 in AMI, HMNet outperforms all baseline models in all metrics, and the result is statistically significant at level 0.05, under paired t-test with the best baseline results. On ICSI dataset, HMNet achieves 11.62, 2.60 and 6.66 higher ROUGE points than previously best results.

Note that MM is a multi-modal model which requires human annotation of topic segmentation (TopicSeg) and visual focus on attention (VFOA) collected from cameras, which is rarely available in practice. In comparison, our model HMNet is entirely based on transcripts from ASR pipelines. Still, on AMI dataset, HMNet outperforms MM(TopicSeg) by 1.49 points in ROUGE-1 and 6.34 points in ROUGE-2, and is higher than MM(TopicSeg+VFOA) by 5.06 points in ROUGE-2.

Moreover, HMNet significantly outperforms the document summarization model PGNet, indicating that traditional summarization models must be carefully adapted to meeting scenarios. HMNet also compares favorably to the extractive oracle, showing that human summaries are more abstractive rather than extractive for meetings.

It's worth noting that Copy from Train obtains a surprisingly good result in both AMI and ICSI, higher than most baselines including PGNet. The reason is that the meetings in AMI and ICSI are not isolated events. Instead, they form a series of related discussions on the same project. Thus, many project keywords appear in multiple meetings and their summaries. It also explains the relatively high ROUGE scores in the evaluation. However, HMNet can focus on salient information and as a

Model	ROUGE-1	R-2	R-SU4
AMI			
HMNet	**53.0**	**18.6**	**24.9**
−pretrain	48.7	18.4	23.5
−role vector	47.8	17.2	21.7
−hierarchy	45.1	15.9	20.5
ICSI			
HMNet	**46.3**	**10.6**	**19.1**
−pretrain	42.3	**10.6**	17.8
−role vector	44.0	9.6	18.2
−hierarchy	41.0	9.3	16.8

Table 3: Ablation study of HMNet.

result, achieves a considerably higher score than Copy from Train baseline.

Ablation Study. Table 3 shows the ablation study of HMNet on the test set of AMI and ICSI. As shown, the pretraining on news summarization data can help increase the ROUGE-1 on AMI by 4.3 points and on ICSI by 4.0 points. When the role vector is removed, the ROUGE-1 score drops 5.2 points on AMI and 2.3 points on ICSI. When HMNet is without the hierarchy structure, i.e. the turn-level transformer is removed and role vectors are appended to word-level embeddings, the ROUGE-1 score drops as much as 7.9 points on AMI and 5.3 points on ICSI. Thus, all these components we propose both play an important role in the summarization capability of HMNet.

4.6 Human Evaluation

We conduct a human evaluation of the meeting summary to assess its readability and relevance. Readability measures how fluent the summary language is, including word and grammatical error rate. Relevance measures how well the summary sums up the main ideas of the meeting.

As MM model (Li et al., 2019) does not have summarization text or trained model available, we compare the results of HMNet and UNS (Shang et al., 2018). For each meeting in the test set of AMI and ICSI, we have 5 human evaluators from Amazon Mechanical Turk label summaries from HMNet and UNS. We choose labelers with high approval rating (>98%) to increase the credibility of results.

Each annotator is presented with the meeting transcript and the summaries. The annotator needs to give a score from 1 to 5 (higher is better) for readability (whether the summary consists of fluent and coherent sentences and easy to understand) and likewise for relevance (whether the summary contains important information from the meeting). The annotators need to read both the meeting transcript and the summary to give evaluations. To reduce bias, for each meeting, the two versions of summaries are randomly ordered.

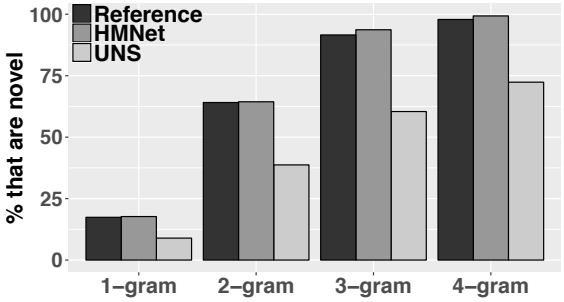

Figure 2: Percentage of novel n-grams in the reference and the summaries generated by HMNet and UNS (Shang et al., 2018) in AMI's test set.

Dataset	AMI	
Source	HMNet	UNS
Readability	**4.17** (.38)	2.19 (.57)
Relevance	**4.08** (.45)	2.47 (.67)
Dataset	ICSI	
Source	HMNet	UNS
Readability	**4.24** (.20)	2.08 (.20)
Relevance	**4.02** (.55)	1.75 (.61)

Table 4: Average scores (1-5) of readability and relevance of summaries on AMI and ICSI's test sets. Each summary is judged by 5 human evaluators. Standard deviation is shown in parenthesis.

Table 4 shows that HMNet achieves much higher scores in both readability and relevance than UNS in both datasets. And the scores for HMNet are all above 4.0, indicating that it can generate both readable and highly relevant meeting summaries.

5 Insights

5.1 How abstractive is our model?

An abstractive system can be innovative by using words that are not from the transcript in the summary. Similar to See et al. (2017), we measure the abstractiveness of a summary model via the ratio of novel words or phrases in the summary. A higher ratio could indicate a more abstractive system.

Fig. 2 displays the percentage of novel n-grams, i.e. that do not appear in the meeting transcript, in the summary from reference, HMNet, and UNS.

As shown, both reference and HMNet summaries have a large portion of novel n-grams ($n > 1$). Almost no 4-grams are copied from the transcript. In contrast, UNS has a much lower ratio of novel n-grams, because it generates a summary mainly from the original word sequence in transcripts.

5.2 Error Analysis

We qualitatively examine the outputs of HMNet and summarize two major types of errors:

1. Due to the nature of long meeting transcripts, the system sometimes summarizes salient information from parts of the meeting different from the reference summaries.

2. Our system sometimes summarizes meetings at a high level (e.g. topics, decisions) and not to cover all detailed items as in the reference.

6 Related Work

Meeting Summarization. There are a number of studies on generating summaries for meetings and dialogues (Zhao et al., 2019; Liu and Chen, 2019; Chen and Metze, 2012; Liu et al., 2019b,a). Mehdad et al. (2013) uses utterance clustering, an entailment graph, a semantic word graph and a ranking strategy to construct meeting summaries. Murray et al. (2010) and Wang and Cardie (2013) focus on various aspects of meetings such as decisions and action items. Oya et al. (2014) employs multi-sentence fusion to construct summarization templates for meetings, leading to summaries with higher readability and informativeness. Recently, Shang et al. (2018) leverages a multi-sentence compression graph and budgeted submodular maximization to generate meeting summaries. In general, these multi-step methods make joint optimization intractable. Li et al. (2019) proposes an encoder-decoder structure for end-to-end multi-modal meeting summarization, but it depends on manual annotation of topic segmentation and visual focus, which may not be available in practice. In comparison, our model only requires meeting transcripts directly from speech recognition.

Document Summarization. Rush et al. (2015) first introduces an attention-based seq2seq (Sutskever et al., 2014) model to the abstractive sentence summarization task. However, the quality of the generated multi-sentence summaries for long documents is often low, and out of vocabulary (OOV) words cannot be efficiently handled. To tackle these challenges, See et al. (2017) proposes a pointer-generator network that can both produce words from the vocabulary via a generator and copy words from the source text via a pointer. Paulus et al. (2018) further adds reinforcement learning to improve the result. Gehrmann et al. (2018) uses a content selector to over-determine phrases in source documents that helps constrain the model to likely phrases and achieves state-of-the-art results in several document summarization datasets. Recently several works on using large-scale pretrained language models for summarization are proposed and achieves very good performance (Liu, 2019; Zhu et al., 2019; Raffel et al., 2019; Lewis et al., 2019; Zhang et al., 2019).

Hierarchical Neural Architecture. As a variety of NLP data (e.g., conversation, document) has an internal hierarchical structure, there have been many works applying hierarchical structures in NLP tasks. Li et al. (2015) proposes a hierarchical neural auto-encoder for paragraph and document reconstruction. It applies two levels of RNN: one on tokens within each sentence and the other on all sentences. Lin et al. (2015) applies a hierarchical RNN language model (HRNNLM) to document modeling, which similarly encodes token-level and turn-level information for better language modeling performance. Serban et al. (2016) puts forward a hierarchical recurrent encoder-decoder network (HRED) to model open-domain dialogue systems and generate system responses given the previous context. Nallapati et al. (2016) proposes the hierarchical attention mechanism on word-level and turn-level in the encoder-decoder structure for abstractive document summarization.

7 Conclusion

In this paper, we present an end-to-end hierarchical neural network, HMNet, for abstractive meeting summarization. We employ a two-level hierarchical structure to adapt to the long meeting transcript, and a role vector to represent each participant. We also alleviate the data scarcity problem by pretraining on news summarization data. Experiments show that HMNet achieves state-of-the-art performance in both automatic metrics and human evaluation. Through an ablation study, we show that the role vector, hierarchical architecture, and pretraining all contribute to the model's performance.

For future work, we plan to utilize organizational chart, knowledge graph and topic modeling to generate better meeting summaries, which can better

capture salient information from the transcript.

Acknowledgement

We thank William Hinthorn for proof-reading this paper. We thank the anonymous reviewers for their valuable comments.

References

Tao Chen, Ruifeng Xu, Yulan He, Yunqing Xia, and Xuan Wang. 2016. Learning user and product distributed representations using a sequence model for sentiment analysis. *IEEE Computational Intelligence Magazine*, 11(3):34–44.

Yun-Nung Chen and Florian Metze. 2012. Integrating intra-speaker topic modeling and temporal-based inter-speaker topic modeling in random walk for improved multi-party meeting summarization. In *Thirteenth Annual Conference of the International Speech Communication Association*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Nikhil Garg, Benoit Favre, Korbinian Reidhammer, and Dilek Hakkani-Tür. 2009. Clusterrank: a graph based method for meeting summarization. *Tenth Annual Conference of the International Speech Communication Association,*.

Sebastian Gehrmann, Yuntian Deng, and Alexander M Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109.

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

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing,*, pages 1373–1378.

Adam Janin, Don Baron, Jane Edwards, Dan Ellis, David Gelbart, Nelson Morgan, Barbara Peskin, Thilo Pfau, Elizabeth Shriberg, Andreas Stolcke, et al. 2003. The icsi meeting corpus. *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03),*, 1:I–I.

Yichen Jiang and Mohit Bansal. 2018. Closed-book training to improve summarization encoder memory. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4067–4077.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Jiwei Li, Minh-Thang Luong, and Dan Jurafsky. 2015. A hierarchical neural autoencoder for paragraphs and documents. *arXiv preprint arXiv:1506.01057*.

Manling Li, Lingyu Zhang, Heng Ji, and Richard J Radke. 2019. Keep meeting summaries on topic: Abstractive multi-modal meeting summarization. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics,*, pages 2190–2196.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out,*.

Rui Lin, Shujie Liu, Muyun Yang, Mu Li, Ming Zhou, and Sheng Li. 2015. Hierarchical recurrent neural network for document modeling. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing,*, pages 899–907.

Chunyi Liu, Peng Wang, Jiang Xu, Zang Li, and Jieping Ye. 2019a. Automatic dialogue summary generation for customer service. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1957–1965.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2020. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*.

Yang Liu. 2019. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*.

Zhengyuan Liu and Nancy Chen. 2019. Reading turn by turn: Hierarchical attention architecture for spoken dialogue comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5460–5466, Florence, Italy. Association for Computational Linguistics.

Zhengyuan Liu, Angela Ng, Sheldon Lee, Ai Ti Aw, and Nancy F Chen. 2019b. Topic-aware pointer-generator networks for summarizing spoken conversations. *arXiv preprint arXiv:1910.01335*.

Iain McCowan, Jean Carletta, Wessel Kraaij, Simone Ashby, S Bourban, M Flynn, M Guillemot, Thomas Hain, J Kadlec, Vasilis Karaiskos, et al. 2005. The ami meeting corpus. *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research,*, 88:100.

Yashar Mehdad, Giuseppe Carenini, Frank Tompa, et al. 2013. Abstractive meeting summarization with

entailment and fusion. *Proceedings of the 14th European Workshop on Natural Language Generation,*, pages 136–146.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. *Proceedings of the 2004 conference on empirical methods in natural language processing,*.

Gabriel Murray, Giuseppe Carenini, and Raymond Ng. 2010. Generating and validating abstracts of meeting conversations: a user study. *Proceedings of the 6th International Natural Language Generation Conference,*, pages 105–113.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807.

Tatsuro Oya, Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. 2014. A template-based abstractive meeting summarization: Leveraging summary and source text relationships. *Proceedings of the 8th International Natural Language Generation Conference (INLG),*, pages 45–53.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Korbinian Riedhammer, Dan Gillick, Benoit Favre, and Dilek Hakkani-Tür. 2008. Packing the meeting summarization knapsack. *Ninth Annual Conference of the International Speech Communication Association,*.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

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

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. *Thirtieth AAAI Conference on Artificial Intelligence,*.

Guokan Shang, Wensi Ding, Zekun Zhang, Antoine Tixier, Polykarpos Meladianos, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2018. Unsupervised abstractive meeting summarization with multi-sentence compression and budgeted submodular maximization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 664–674.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems,*, pages 3104–3112.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems,*, pages 5998–6008.

Lu Wang and Claire Cardie. 2013. Domain-independent abstract generation for focused meeting summarization. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers),*, 1:1395–1405.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *arXiv preprint arXiv:1912.08777*.

Zhou Zhao, Haojie Pan, Changjie Fan, Yan Liu, Linlin Li, Min Yang, and Deng Cai. 2019. Abstractive meeting summarization via hierarchical adaptive segmental network learning. In *The World Wide Web Conference*, pages 3455–3461.

Chenguang Zhu, Ziyi Yang, Robert Gmyr, Michael Zeng, and Xuedong Huang. 2019. Make lead bias in your favor: A simple and effective method for news summarization. *arXiv preprint arXiv:1912.11602*.