Dr. Summarize: Global Summarization of Medical Dialogue by Exploiting Local Structures

Anirudh Joshi
Stanford University
anirudhjoshi@stanford.edu

Namit Katariya
Curai

Xavier Amatriain
Curai

Anitha Kannan
Curai

Abstract

Understanding a medical conversation between a patient and a physician poses unique natural language understanding challenge since it combines elements of standard open-ended conversation with very domain-specific elements that require expertise and medical knowledge. Summarization of medical conversations is a particularly important aspect of medical conversation understanding since it addresses a very real need in medical practice: capturing the most important aspects of a medical encounter so that they can be used for medical decision making and subsequent follow ups.

In this paper we present a novel approach to medical conversation summarization that leverages the unique and independent local structures created when gathering a patient's medical history. Our approach is a variation of the pointer generator network where we introduce a penalty on the generator distribution, and we explicitly model negations. The model also captures important properties of medical conversations such as medical knowledge coming from standardized medical ontologies better than when those concepts are introduced explicitly. Through evaluation by doctors, we show that our approach is preferred on twice the number of summaries to the baseline pointer generator model and captures most or all of the information in 80% of the conversations making it a realistic alternative to costly manual summarization by medical experts.

DR: You mentioned **having a cough for 2 days** and a **fever since last night** along with **being short of breath**. Is that correct?
PT: **yes** , correct
DR: I appreciate your concern for preventing spread. Do you feel like you are unable to move around as usual?
PT: I'm **definitely weaker and low energy** the **fever went down to 99 this morning**
DR: Have you taken any medications or tried anything else to help you with your symptoms?
PT: **lots of fluids and vitamin c. lozenges to minimize coughing**
DR: do you **have any medical conditions** or have you been on any medications
PT: **no**, none
DR: alright. When you had a fever, did you **take medicine** like **tylenol** to bring the fever down?
PT: I **didn't**

Model Output Summary

- mentioned **having a cough** for 2 days and **a fever** since last night along with being **short of breath**.
- unable to move around as usual. **definitely weaker and low energy fever went down to 99**
- **lots of fluids and vitamin c. lozenges** to minimize coughing with symptoms .
- **no medical conditions.** none have any medical conditions.
- **didn't take medicine** like **tylenol** to bring the fever down.

Figure 1: Example medical dialogue and summary generated by our proposed model. Note that the summary captures **affirmatives**, **negatives** and **medical concepts**. For more examples, see supplement.

1 Introduction

Telemedicine is a rapidly growing medium of interaction with the healthcare system (Mann et al., 2020). With the COVID-19 pandemic limiting in person medical visits, healthcare systems are seeing greater than 100% increase in virtual urgent care visits and greater than 4000% increase in virtual non-urgent care visits (Mann et al., 2020). Telemedicine systems today involve either direct voice and video chat or text based chat interfaces. At the end of a history taking conversation with the patient (*i.e.* gathering of presenting symptoms, patient concerns and the past medical, psychological and social history), a doctor or nurse typically summarizes the information from the dialogue in order

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 3755–3763
November 16 - 20, 2020.

to pass it on to other care providers or as a means of recording the interaction. Even in more traditional in-person medical settings, the advent of electronic health records (EHR) as a way to record medical information has created the need to summarize any patient conversation with healthcare professionals. In all of the above cases, requiring human experts to summarize a potentially long medical conversation is costly and limits the scalability of the healthcare system. Furthermore, nurses or doctors who are required to do these tasks feel burnt out since the task feels repetitive and mechanical (Shanafelt et al., 2016).

Medical Dialogue summarization possesses unique characteristics and goals that are not present in other domains. Notes written by domain experts need to capture important parts of the conversation needed for clinical decision making, and not the summary of the entire conversation. As an example from Figure 1 we observe that from the conversation it is important to (1) capture all the medical conditions and terminology described in the dialogue (cough, fever, shortness of breath etc.) (2) discern all the affirmatives and negatives on medical conditions correctly (no allergies, having a cough for 2 days) and (3) bias towards copying from the source text while not being completely extractive. We observe that the majority of the information that is needed in a summary note is present in the medical dialogue with some novel words introduced to stitch phrases together. Unlike open domain dialogue between peers which involves long term memory dependencies between turns in the dialogue, patient history taking possesses an inherent local structure.

Another important challenge of end-to-end medical dialogue summarization is the lack of large scale annotated datasets. Annotations of medical dialogue needs trained doctors, which is expensive and slow. It is important to design modeling strategies and data capturing processes in a way that enables learning important biases as described above from sparse data.

In this paper, we propose a method to automate the generation of summary notes from the original patient/provider dialogue. Given the lack of existing public datasets with specific medical dialogues, we build our own dataset using conversations from a telemedicine platform and obtain reference summaries from healthcare professionals.

Our approach is based on the following key insights:

1. The learning problem for patient history taking summarization can be posed in the form of summarizing local dialogue turns (snippets) which are composed of smaller sections of the conversation. For example in Figure 1, the doctor's question about medical conditions and the patient's response can be considered a local snippet.
2. Some specific characteristics of the medical conversation that are important for a doctor such as concept negations need to be explicitly modeled to avoid information loss.

Our proposed model leverages the pointer generator network (See et al., 2017) to capture the inductive bias present in our data. We extend the baseline pointer generator network by introducing a penalty to the generator distribution to guarantee that the network defaults to extractive summarization when necessary. We also model the unique domain specific challenges of medical data by introducing explicit modeling of negations and introducing medical concepts. Given our smaller dataset we found transformer based models produce poor outputs and were not pursued further. Recent work in medical summarization has also shown that Pointer Generator Networks produce strong performance (Zhang et al., 2018; Krishna et al., 2020a).

We propose a set of automated metrics that evaluate the dialogue on those particular challenges. We show through those automated metrics and doctor evaluations that the modified pointer generator network with the generator penalty is able to capture domain nuances and provide useful summarizations in over 80% of the cases. While explicitly modeling negations introduces subtle improvements, those are not captured by our experts, and explicitly introducing medical concepts does not improve the performance of our model.

The main contributions of our paper can be summarized in:

1. A novel end-to-end approach to medical dialogue summarization that is competitive with expensive and not scalable manual summaries.

2. A simple extension of the pointer generator network that shows that adding a penalty for using the generator distribution has significant

advantages in dialogues with high domain expertise like medicine

3. A thorough evaluation of different summarization models that shows the correlation (and lack of such) between automated metrics and expert judgement

2 Related Work

Neural Summarization: Emergence of sequence to sequence models and attention mechanisms (Sutskever et al., 2014; Nallapati et al., 2016) has led to rapid progress on extractive (Nallapati et al., 2017) , abstractive (Nallapati et al., 2016) and hybrid models (See et al., 2017; Gu et al., 2016) for summarization. Much of the recent work has shown these models to generate near-human coherent summaries while retaining reasonable factual correctness. Of interest, is the class of hybrid models, that has inductive bias for being more extractive while possessing the ability to be abstractive for document text summarization tasks. Notably, (Boutkan et al., 2019) introduced the idea of using dropout mechanism and pointer losses for this trade-off. In medical conversation summarization, harnessing the inductive bias of these hybrid models lead to more factually correct summaries, as we study in this paper.

Dialog Summarization: While most neural summarization has focused on news corpora, recent work has tried to tackle unique challenges associated with summarizing dialogues. (Goo and Chen, 2018) proposes using dialogue history encoders labelers based on type of dialogue section to inform the generation. (Liu et al., 2019a) propose using key points as a means of categorizing sections of dialogue.

Medical Summarization: (Alsentzer and Kim, 2018) explore the upper bounds on extractive summarization in medical text and find that a purely extractive approach may not provide sufficient recall. Incorporating medical knowledge into sequence to sequence summarization was studied by (Zhang et al., 2018) by encoding background information to condition the decoder. (Liu et al., 2019b; Krishna et al., 2020b) study spoken dialogue summarization in the medical domain with pointer generator networks however they don't explicitly model for the properties of medical data and do not report doctor evaluations of the outputs.

Our work differs by leveraging the unique local structures created when gathering a patient's medical history. We also explicitly incorporate into the learning process several properties of medical conversation that are important in summarization. While the pointer generator model shows a bias towards copying when trained on news corpora, we find that this is not true when trained on dialogue and our work on explicitly modulating generation probabilities to encourage copying has broad applicability to dialogue summarization in domains where factual correctness is important.

3 Model

We are interested in a model that has two main properties. First, it encourages copying from the snippet to preserve the integrity of the symptoms and medical issues being discussed. Second, it can handle out-of-vocabulary terms, such as medically relevant terms, that are used by patients and doctors.

The pointer generator (See et al., 2017) is naturally suited as they imbibe these properties by providing a hybrid of extractive and abstractive summarization, with more emphasis on extraction (Boutkan et al., 2019). We use pointer generator as the base model to build upon and encode medical conversation specific properties.

3.1 Base model: Pointer Generator Network

Pointer generator network (See et al., 2017) is a recurrent neural network based sequence model with attention and a soft switch variable p_{gen} to orchestrate between copy and generation. At each time step of decoding, the model uses p_{gen} to either copy words from the source text using a pointer mechanism or generate words from a fixed vocabulary using the decoder probability distribution,

$$P(w) = p_{\text{gen}} P_{\text{vocab}}(w) + (1 - p_{\text{gen}}) \sum_{i:w_i = w} a_i^t \quad (1)$$

$$\text{loss}_t = -\log P(w_t^*) + \lambda \sum_i \min(a_i^t, c_i^t) \quad (2)$$

P_{vocab} is a probability distribution over all words in the vocabulary. a^t is the attention/probability distribution over the source words that tells the decoder where to look to produce the next word. Thus, $P(w)$ the probability distribution over the extended vocabulary that is the union of fixed vocabulary and the words that appear in the source, enabling the the model to copy out of vocabulary words. The loss in equation 2 encompasses the cross entropy and coverage loss described in (See et al., 2017)

3.2 Incorporating medical knowledge

In the absence of sufficient amount of labeled data, transfer learning from a pre-trained model such as pointer generator may also be impoverished in distinguishing important (*e.g* medical concepts) and unimportant out-of-vocabulary words. To infuse some of this knowledge, we leverage compendium of medical concepts, known as unified medical language systems(UMLS).

During training, we use a one-hot vector m^t that is same dimension as the source snippet. This vector encodes the presence of UMLS medical concepts that are in both source snippet and in reference. The requirement for presence of concepts in reference is to make sure that only those concepts that are relevant for the snippet is taken into account. m^t influences the attention distribution $a^t = \text{softmax}(e^t)$, through the functional form:

$$e_i^t = v^t \tanh(W_h h_i + W_s s_t + w_c c_i^t + w_m m_i^t + b_{\text{attn}})$$

where h_i is the encoder hidden state, s_t is the decoder hidden state and c_t is the coverage vector described in (See et al., 2017). Since the concepts are encoded based on whether they are present in the reference, this acts as a form of teacher forcing where the concepts are encoded and supervised during training time but at test time, these encodings are not available to the model.

Analogous to the coverage mechanism introduced by (See et al., 2017), we propose to model this both, in the attention mechanism as well as in the loss function to directly supervise the model such that higher attention weights are placed on positions where concepts are present, by adding additional term $\lambda_m(1-\sum_i m_i^t \cdot a_i^t)$ to the loss function described in equation 2. λ_m is the scaling factor on the concept loss term and $a \cdot b$ is the dot product between a and b.

3.3 Modeling negations

We take two complementary approaches to 'supervise' modeling of negations - attention mechanism on negation words, and by explicitly modeling a switching variable that induces a mixture model over copy, generate and negate.

Negation word attention: Similar to modeling of medical concept, negation attention directly supervises the model in the attention distribution and in the loss function. For this, a small set of negative unigrams ('no' , 'nope', 'doesn't', 'not') are manually curated. An additional binary vector $\mathbf{n}^t$ of the same length as that of the source snippet encodes $n_i^t = 1$ when t^{th} location in the source has one of these negative unigrams. The attention distribution is modified to focus the attention distribution on such terms.

$$e_i^t = v^t \tanh(W_h h_i + W_s s_t + w_c c_i^t + w_m m_i^t + w_n n_i^t + b_{\text{attn}})$$

The loss function is augmented with $\lambda_n(1 - \sum_i n_i^t \cdot a_i^t$

Negation as a switching variable: In addition to the snippet-level summary, we also collect explicit labels in the form of a special token '[NO]' for parts of the snippet that are negated. While the [NO] token can be added to the fixed vocabulary, we note that the model would need to learn when to generate the [NO] token in the final summary using the decoder, and thereby influencing the likelihood of p_{gen} in other abstractive parts of the summary. Instead, we use this additional signal to formulate the probability distribution over extended vocabulary as a convex combination:

$$P(w) = p_{\text{gen}} P_{\text{vocab}}(w) + p_{\text{copy}} \sum_{i:w_i=w} a_i^t + p_{\text{neg}} P_{[\text{NO}]}$$

where p_{neg} controls the generation of [NO]. This extends (See et al., 2017) with additional switching variable p_{neg}:

$$p_{\text{gen}},\ p_{\text{copy}},\ p_{\text{neg}} = \text{softmax}(w_{h^*}^T h_t^* + w_s^T s_t + w_x^T x_t + b_{ptr})$$

where h_t^* is the context vector, s_t is the decoder state and x_t is the decoder input. In positions where p_{neg} is 1, $p_{gen} + p_{copy}$ needs be 0, and vice versa so that [NO] token is correctly incorporated into the summary during decoding. We explicitly supervise this behavior by adding an additional L1 loss term to encourage $|p_{neg} - (p_{gen} + p_{copy})|$ to be maximal. The L1 loss is weighted by a scalar factor γ to modulate its contribution.

3.4 Controlling generation probability

We explicitly want summaries to be copied as much as possible from the source dialogue since factual errors in a medical setting are unacceptable. In order to do this copying, we need copy distributions that can shift rapidly back and forth between doctor and patient turns, while still *generating* to stitch between them. See table. 3 for an example.

As the pointer generator has flexibility to switch between generation and copying, it has the (dis)advantage of using p_{gen} to compensate for lack of flexibility to rapidly shift between copy and generated by mostly depending on generation. In fact, we found this behavior to be true empirically: In (See et al., 2017), at inference time, for text summarization task, p_{gen} is below 0.2. In zero-shot setup, average p_{gen} on our medical dialogue dataset is 0.2. However after fine-tuning on our dataset, we observe that the average p_{gen} is 0.4 at inference time validating that the model depends a lot more on generation for conversation summarization.

This is detrimental since the model can choose to hallucinate medical concepts that are not part of the snippet. We propose a penalty on the model for using the generator distribution instead of the copy distribution to force it to learn to use the copy mechanism effectively. We add $\delta\ p_{\text{gen}}$ term to the overall loss where δ is a scalar constant. This term will be large if the model is using the generator more during decoding.

4 Evaluation

We evaluate models using automated metrics and manual evaluation from doctors. Multiple studies have shown that automated metrics in NLP do not always correlate well to human judgments as they may not fully capture coherent sentence structure and semantics (Roller et al., 2020; Kryściński et al., 2019). Since medical dialogue summarization would be used to assist health care, it is important for doctors to evaluate the quality of the output.

4.1 Automated metrics

While we measure model performance on standard metrics of ROUGE (Lin, 2004), we also wanted to specifically measure a model's effectiveness in capturing the medical concepts that are of importance, and the negations. Therefore, we propose a new set of automated metrics that directly measure medically relevant information in the summary.

Medical Concept Coverage: The concept coverage set of metrics captures the encapsulation of the medical terms in the model's output summary to the ground truth reference. In particular, let $\mathcal{C}$ be the set of medical concepts in the reference summary and $\hat{\mathcal{C}}$ be the set of concepts in the summary output by the model. Then, Concept recall $= \frac{\sum_{n=1}^{N} |\hat{\mathcal{C}}^{(n)} \cap \mathcal{C}^{(n)}|}{\sum_{n=1}^{N} |\mathcal{C}^{(n)}|}$ and Concept precision $= \frac{\sum_{n=1}^{N} |\hat{\mathcal{C}}^{(n)} \cap \mathcal{C}^{(n)}|}{\sum_{n=1}^{N} |\hat{\mathcal{C}}^{(n)}|}$

We use these to compute a Concept F1. We use an inhouse medical entity extractor to match concepts in the summary to UMLS. Medical concepts in the decoded summary that weren't present in the original conversation would be false positives and vice versa for false negatives.

Negation Correctness: To measure the effectiveness of the model to identify the negated status of medical concepts, we use Negex (Harkema et al., 2009) to determine negated concepts. Of the concepts present in the decoded summary, we evaluate precision and recall on whether the decoded negations were accurate for the decoded concepts and compute a Negation F1.

4.2 Doctor Evaluation

We also had two doctors, who serve patients on our telehealth platform, evaluate the summaries produced by the models. Given the local dialogue snippets and the generated summary, we asked them to evaluate the extent to which the summary captured factually correct and medically relevant information from the snippet. Depending on what percentage of the concepts were correctly mentioned in the decoded summary of the provided snippet, the doctors graded the summaries with *All* (100%), *Most* (at least 75%), *Some* (at least 1 fact but less than 75%), *None* (0%) labels. We also formulated a comparison task where given two summaries generated by different models and the associated dialogue, they were asked which summary was better. The doctors also had the ability to use "both" and "none" depending on if both models captured a good summary or if none of them did. To avoid bias, the doctors do not know the model that produced the summary in both the experiments. In the comparison task, the two summaries were provided in randomized order so that there is no bias in the order of presentation of the summaries.

5 Dataset construction

We collected a random subset of dialogue of 25,000 conversations from a telemedicine platform. We split the dialogue into a series of local dialogue snippets using a simple heuristic: the turns between two subsequent question by the physician corresponds to a snippet. The length of these snippets ranged anywhere from two turns (a physician question and patient response) to ten turns.

We had medical doctors summarize a random sample of 3000 snippets. These are the same doctors who practice on the same telemedicine platform. The doctors were asked to summarize the sections as they would for a typical clinical note by including all the relevant information. Further if the summary included a negated medical term, eg. “doesn’t have fever”, the doctors were asked to use a [NO] token in front of that particular sentence in the summary. If a local snippet did not contain any relevant information they were excluded from annotations. For example in the beginning or end of conversations there may be turns that are purely greetings and not part of the patient history taking process, or purely educational in nature.

At the end of the labeling, we used the 1690 number of local snippets that the doctors labeled as containing pertinent information for history gathering. We used 1365 as the training set, 158 as a validation set and 167 as a held out test set. The test set was made sure to be from distinct conversations that were taken from a different date range on the platform compared to the training or validation sets. This was done to ensure that different local snippets from a certain conversation weren’t part of the training and test sets. The data was preprocessed by removing the names of the actors in the dialogue (“Doctor”, “Patient”) and concatenating all the turns within a snippet together.

6 Experiments

Model variants: We study the following variants:

- **2M-BASE** : Pretrained pointer generator model fine-tuned on medical dialogue summarization
- **2M-PGEN** : 2M-BASE + generator loss to control generation probability (§ 3.4)
- **2M-PGEN-NEG** : 2M-PGEN + negation attention mechanism and loss (§ 3.3)
- **3M** : Pretrained pointer generator model finetuned on medical dialogue summarization using negation as a switching variable to form a 3 mixture final probability distribution (§ 3.3)
- **3M-NEG** : 3M + negation attention loss (§ 3.3)
- **3M-PGEN-NEG-CONCEPT** : 3M-NEG + the losses (§ 3.4) to control generation probability and § 3.2 to improve medical concept coverage.

Training details: All the models use a vocabulary size of 50k with 128 dimensional embeddings and 256 dimensional hidden states. The training parameters followed (See et al., 2017) with a learning rate of 0.15 and Adagrad as the optimizer. The coverage mechanism as described in (See et al., 2017) was used for all our models. Models were first pretrained on the CNN-Daily Mail corpus and finetuned on conversational data from our in-house chat-based telehealth platform (§ 5).

Pretraining took approximately 2 days on a single NVIDIA Titan Xp GPU and finetuning took under 2 hours. The concept and negation attention modifications added 512 parameters each to the base pointer generator model with coverage. For details on hyperparameters and validation results see supplement.

6.1 Main Results

Table 1 presents the key results, with a side-by-side comparison between automated metrics and doctor evaluation. We chose 2M-PGEN as the default improvement over 2M-BASE as it is the model with the simplest improvement over 2M-BASE . The subsequent models build on 2M-PGEN by modeling negations and explicit concept attention. We make the following observations:

- 2M-PGEN improves concept F1 score over 2M-BASE at the cost of drop in negation score. Even when we consider snippets where 2M-BASE and 2M-PGEN have identified the same set of concepts, we find that 2M-PGEN generates better summaries. This is also evidenced by the corresponding doctor evaluation, where we see 2M-PGEN preferred on twice the number of examples compared to the 2M-BASE (37.1% vs 18.5%).

  Qualitatively, consider the first and third examples in Table 2 in which both models have identified the medical concepts like “phelgm” and “cycle” however 2M-PGEN clearly provides more descriptive and coherent summaries explaining the wide margin on human evaluation.
- With 2M-PGEN-NEG model that extends 2M-PGEN with negation attention, we can see that the negation F1 improves, with a dip in concept attention. While in doctor evaluation, both 2M-PGEN-NEG and 2M-PGEN perform comparably, in comparison to 2M-BASE , we do find that on difficult negations such as the

Models	Metrics			Doctor Evaluation			
	Negation F1	Concept F1	ROUGE-L F1	Model	Baseline	Both	None
2M-BASE	70.1±0.8	69.1±1.3	52.6±0.9	-	-	-	-
2M-PGEN	67.3±3.3	72.8±0.8	55.4±0.9	37.1%	18.5 %	38.9 %	5.3%
2M-PGEN-NEG	72.2±3.6	70.9±2.2	53.5±0.7	37.7%	22.7%	34.1%	5.4%
3M-PGEN-NEG-CONCEPT	78.0±4.2	70.6±1.4	55.2±1.2	26.9%	25.7%	42.5%	4.2%

Table 1: Automated and Doctor evaluation.

Snippet	Model Summaries
DR: what color is the phlegm? **PT**: dark green dark green	**2M-BASE** : the color is the phlegm.
	2M-PGEN : the phlegm is dark green dark green
	2M-PGEN-NEG : phlegm color is dark green dark green.
	3M-PGEN-NEG-CONCEPT : has dark green dark green color.
DR: thanks for letting me know. does your back hurt? **PT**: nope my back feels fine	**2M-BASE** : back pain does not back feels fine.
	2M-PGEN : back is back . thanks for letting me.
	2M-PGEN-NEG : no back hurt . fine back feels fine.
	3M-PGEN-NEG-CONCEPT [NO] no back hurt . back feels fine.
DR: What was your last period date? **PT**: feb 9th to 13th i have an average cycle of about 32-33 days.	**2M-BASE** : last period is feb 9th to 13th. has an average cycle of 32-33 days.
	2M-PGEN : period date feb 9th to 13th. average cycle of about 32-33 days
	2M-PGEN-NEG : period feb 9th to 13th. has an average cycle of about 32-33 days
	3M-PGEN-NEG-CONCEPT : has an average cycle of 32-33 about feb 9th to 13th. has an average cycle of 32-33 days.

Table 2: Snippets along with multiple model outputs

second example in Table 2, models that explicitly incorporate negations prove better.

- 3M-PGEN-NEG-CONCEPT performs best on the negation metric while maintaining comparable performance on concept metric to 2M-BASE . However, it is not our best performing model on human evaluation. We analyze this in detail in § 6.5. From the first and third examples in Table 2 we can see that sentence structure and coherency reduces on 3M-PGEN-NEG-CONCEPT despite capturing concepts and negations.
- Across all models, we find that there is only 5% of the snippets where no model produces good summary. On closer investigation, we find that these are snippets where there is a lack of coherent response from the patient.

6.2 Independent model evaluation by doctors

In Figure 2, we study doctor's evaluations when models were evaluated independently. The model's output summary was graded on whether it included "All", "Most", "Some", "None" of the relevant facts, 2M-PGEN gets all or most of the facts on 80% of examples compared to 60% for 2M-BASE . From Table 2 and Table 3 this is evident with the integrity of the patient response being preserved when using a generator penalty.

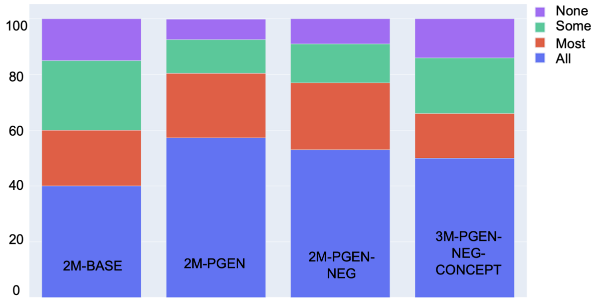

Figure 2: Doctor evaluation of amount of relevant information covered by summaries

6.3 Role of explicitly modulating $p_{\mathbf{gen}}$

Table 3 provides a qualitative comparison between 2M-BASE and 2M-PGEN . We observe that 2M-BASE relies more on the generator to create the summaries and at times does not copy from the patient's answers. For instance, in the second example in Table 3, 'no' is produced by the generator in 2M-BASE . In contrast, 2M-PGEN is able to learn a copy mechanism that shift attention distributions over to patient answers and back to doctor questions as opposed to linear copying. For the same example, 'no' is copied from the patient.

We can also see examples in Table 3 where 2M-BASE is more erroneous than 2M-PGEN . Additionally, while 2M-BASE correctly identified all three concepts (acne, bradycardia and isotretenoin), 2M-PGEN captured the semantics of summarizing that snippet, even though the resultant summary had only two of the three concepts (acne, isotretenoin). This example also sheds light on why automated metrics are not as reliable in measuring the efficacy of the models.

Snippet	2M-BASE	2M-PGEN
DOCTOR : Thanks for letting me know. How severe is the pain on a scale of 1-10, considering 10 being the worst and severe? PATIENT: 4-5	pain is severe . pain is 10 on a scale of 1-10.	the pain is 4-5 on a scale of 1 - 10 being the worst and severe
DOCTOR: Thanks for sharing. Have you been on any medications for the same? PATIENT: I've been on isotretenoin for the acne but off of it for the last 5 months nothing for the bradycardia	not on isotretenoin for acne but off of bradycardia for last 5 months .	been on isotretenoin for the acne but off of it for the last 5 months
DOCTOR: does your chest pain get worse while taking deep breaths ? PATIENT: no	no chest pain . worse while taking breaths	no chest pain get worse while taking deep breaths

Table 3: Modulating p_{gen}: Directly modulating the p_{gen} allows for the copy mechanism to learn how to copy between doctor questions and patient answers instead of just sequentially copying. Pink is used to show words copied from the Doctor, green for words copied from the Patient and orange for words generated.

6.4 Role of explicit negation modeling

To determine which mode of explicit negation modeling has greater effect we compare 3M and 3M-NEG against 2M-BASE (Table 4). We observe that extending 2M-BASE to 3M with the p_{neg} soft switch improves Negation F1 (76.9 vs 70.1). Further extending this to 3M-NEG by incorporating negation attention we see an even larger improvement in Negation F1 (81.5). We also find that on difficult negations such as the second example in Table 2 where the patient responds "nope" followed by an affirmative "fine", models that explicitly incorporate negations produce better summaries. However from the human evaluation and qualitative examples (Table 2) we see that coherency reduces even though the quantitative metrics improve. See supplement for qualitative comparison.

Model	Negation F1	ROUGE-L F1
2M-BASE	70.1±0.8	52.6 ± 0.9
3M	76.9±3.6	56.4 ± 0.3
3M-NEG	81.5±4.7	54.5 ± 1.3

Table 4: Negation ablation

6.5 Role of encoding medical concepts

Given data sparsity our hypothesis was that directly using medical concepts to guide the attention mechanism would help performance on the concept metric. We see improvement in this metric when adding concept attention to 2M-BASE (concept F1 72.0 vs 69.1) however once p_{gen} loss is introduced we notice these gains no longer hold. On local snippets, we observe that the increased copying ability compensates for the removal of concept attention and adding concept attention can reduce performance. In both cases the attention on medical concepts in the copy distribution increases 5% however this doesn't amount to consistent increase on the automated metrics. We leave this as an open research direction for longer dialogue snippets where enhanced copying may need to be coupled with concept attention.

7 Conclusions

In this paper, we presented a novel approach to medical conversation summarization. This is an important application for text summarization since medical professionals rely on good conversation summarizations for medical decision making and follow up. Medical conversations, however, have traditionally posed a challenge for vanilla machine learning approaches because of the importance of domain knowledge and syntactic nuances such as negation. We extend a deep learning approach, pointer generator networks, and show that for domains like medicine where integrity of the source is critical, encouraging copying in the learning process produces the best model (2M-PGEN) on human evaluation. This approach represents a viable alternative to human summarization since experts report that up to 80% of the relevant information is present in 2M-PGEN summaries with only 5% of summaries containing no relevant information. Even if the system implementing this approach could not operate completely automated, it is clear that it could speed up the summarization process by reducing the amount of human intervention needed.

For future work, we would like to see if our findings generalize well on other datasets and other domains. Particularly, we would like to see if the anecdotal evidence that explicit negation modeling matters, can be captured by the metrics or the expert human evaluation.

References

Emily Alsentzer and Anne Kim. 2018. Extractive summarization of EHR discharge notes. *CoRR*, abs/1810.12085.

Freek Boutkan, Jorn Ranzijn, David Rau, and Eelco van der Wel. 2019. Point-less: More abstractive summarization with pointer-generator networks. *CoRR*, abs/1905.01975.

Chih-Wen Goo and Yun-Nung Chen. 2018. Abstractive dialogue summarization with sentence-gated modeling optimized by dialogue acts. *CoRR*, abs/1809.05715.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany. Association for Computational Linguistics.

Henk Harkema, John N. Dowling, Tyler Thornblade, and Wendy W. Chapman. 2009. Context: An algorithm for determining negation, experiencer, and temporal status from clinical reports. *Journal of Biomedical Informatics*, 42(5):839 – 851. Biomedical Natural Language Processing.

Kundan Krishna, Sopan Khosla, Jeffrey P. Bigham, and Zachary C. Lipton. 2020a. Generating soap notes from doctor-patient conversations.

Kundan Krishna, Sopan Khosla, Jeffrey P. Bigham, and Zachary C. Lipton. 2020b. Generating soap notes from doctor-patient conversations.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Evaluating the factual consistency of abstractive text summarization.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

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*, KDD '19, page 1957–1965, New York, NY, USA. Association for Computing Machinery.

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

Devin M Mann, Ji Chen, Rumi Chunara, Paul A Testa, and Oded Nov. 2020. COVID-19 transforms health care through telemedicine: evidence from the field. *Journal of the American Medical Informatics Association*. Ocaa072.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3075–3081. AAAI Press.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar GuÌ‡lçehre, 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, Berlin, Germany. Association for Computational Linguistics.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M. Smith, Y-Lan Boureau, and Jason Weston. 2020. Recipes for building an open-domain chatbot.

Abigail See, Peter Liu, and Christopher Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Association for Computational Linguistics*.

Tait D. Shanafelt, Lotte N.Dyrbye, Christine Sinsky, Omar Hasan, Daniel Satele, Jeff Sloan, and Colin P. West. 2016. Relationship between clerical burden and characteristics of the electronic environment with physician burnout and professional satisfaction. *Mayo Clinic Proceedings*, 91:836–848.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.

Yuhao Zhang, Daisy Yi Ding, Tianpei Qian, Christopher D. Manning, and Curtis P. Langlotz. 2018. Learning to summarize radiology findings. *CoRR*, abs/1809.04698.