

Exploring the Influence of Dialog Input Format for Unsupervised Clinical Questionnaire Filling

Farnaz Ghassemi Toudeshki^{&,#}, Anna Liednikova^{&,\dagger}, Philippe Jolivet[&], Claire Gardent^{\gamma}

[&] ALIAE, [#] IDMC, Université de Lorraine

^{\dagger} Loria, Université de Lorraine, ^{\gamma} CNRS

{farnaz.ghassemi, anna.liednikova, philippe.jolivet}@aliae.io
claire.gardent@loria.fr

Abstract

In the medical field, we have seen the emergence of health-bots that interact with patients to gather data and track their state. One of the downstream application is automatic questionnaire filling, where the content of the dialog is used to automatically fill a pre-defined medical questionnaire. Previous work has shown that answering questions from the dialog context can successfully be cast as a Natural Language Inference (NLI) task and therefore benefit from current pre-trained NLI models. However, NLI models have mostly been trained on text rather than dialogs, which may have an influence on their performance. In this paper, we study the influence of content transformation and content selection on the questionnaire filling task. Our results demonstrate that dialog pre-processing can significantly improve the performance of zero-shot questionnaire filling models which take health-bots dialogs as input.

1 Introduction

Work on Question Answering (QA) and Machine Reading Comprehension (MRC) mostly focuses on wh-questions of arbitrary types (who, what, where etc.) whose answer can be found in text. The answer can be extractive where a short span of the text is identified as the answer (Pearce et al., 2021) or it can be abstractive where a free-form answer is generated from the question and some support document (Bauer et al., 2018).

Here, we focus instead on a QA setting where questions are restricted to polar (yes/no) and Agreement Likert Scale (ALS) questions and where answers are contained in a dialog rather than a paragraph text. As illustrated in Figure 1, this setting is useful for automatic questionnaire filling (AQF) in the medical field. Given a dialog between a patient and a health bot, the goal of automatic questionnaire filling is to answer a set of predefined questions from a medical questionnaire (here the Pain

Dialog
bot: What is the most difficult for you about your sleep ? patient: I have back pain that prevents me from sleeping. bot: I'm sorry to hear that. How long have you had back pain? patient: Since I've been working out, I've had constant back pain at night. bot: Do you think pain can last for long? patient: I think it will stop once I stop playing sports. bot: Should we let time fix the pain? patient: My doctor thinks that I need to get used to doing sports and that the pain will disappear after a while.
Questionnaire
(1) My pain is a temporary problem in my life. CQ: <input type="checkbox"/> No <input checked="" type="checkbox"/> Yes <input type="checkbox"/> NA ALS: <input type="checkbox"/> Totally disagree <input type="checkbox"/> Rather disagree <input type="checkbox"/> Agree <input checked="" type="checkbox"/> Totally agree <input type="checkbox"/> NA

Figure 1: An example of a dialog and a question from the PBPI Questionnaire, answered in CQ and ALS format

Beliefs and Perceptions Inventory (PBPI) questionnaire (Williams and Thorn, 1989)) based on the dialog content.

In previous work, Toudeshki et al. (2021) compared three ways of deriving answers to questions from dialogs: Natural Language Inference, Question Answering and Text Classification. For polar and ALS questions, they found that Natural Language Inference (NLI) performs best. One possible limitation of their approach however is that they apply NLI models to dialogs while NLI models are trained on non-dialogic text.

In this paper, we propose different ways of transforming and selecting dialog content before applying NLI to answer questions, and we analyse the impact of these operations on NLI-based questionnaire filling. Our hypothesis is that transforming the input dialog into a format closer to the text format on which NLI models are trained, should help these models perform better. Our experimental results confirm this hypothesis: it demonstrates that, in a zero-shot setting, transforming and select-

ing dialog content yields significant improvements over a baseline which takes the full dialog content as input.

2 Related work

We briefly situate our work with respect to three tasks which have similarities with Automatic Questionnaire Filling namely, Machine Reading Comprehension, Question Answering and Aspect-Based Sentiment Analysis (ABSA).

MRC/QA. Given a text and a question, MRC and QA models aim to derive the answer to that question from some input document (Zeng et al., 2020).

Similar to our approach, Ren et al. (2020) focus on filling in medical questionnaires consisting of polar questions about medical terms. However, in their case, the input to the model is a text (patient records) rather than a dialog. Furthermore, QA is modeled as a classification task which restricts the approach to a limited set of possible questions and answers. Finally, the questions are restricted to polar questions about terms whereas we consider polar and ALS questions about full sentences.

Recently, some work has focused on answering questions from dialogs rather than text. A simple approach for modeling a multi-turn dialog is to concatenate all turns (Zhang et al., 2019; Adiwardana et al., 2020). However, for retrieval-based response selection, Zhang et al. (2018); Yuan et al. (2019) showed that turns-aware aggregation methods can achieve a better understanding of dialogs compared to considering all turns equally. Similarly for MRC on dialogs, turns-aware approach have been proposed which select turns in the conversation that are related to the input question: Zhang et al. (2021) uses embedding-based similarity to select such turns while Li et al. (2020) uses a pre-trained language model fine-tuned on NLI tasks. Their results showed that eliminating irrelevant turns effectively improves results. Our work extends on this work showing that both content selection and content transformation help improve MRC on dialogs.

Aspect-Based Sentiment Analysis. Aspect based sentiment analysis (ABSA) is the process of determining sentiment polarity for a specific aspect in a given context. An aspect term is generally a word or a phrase which describes an aspect of an entity (Jiang et al., 2019). For

instance, (Jang et al., 2021; Sun, 2022) investigate aspect-based sentiment analysis on user tweets related to COVID-19. While AQF could be viewed as an ABSA task where each item should be labelled with one of three (polar question) or five (ALS question) sentiment value (agree, disagree, etc.), two key differences between ABSA and AQF is that (i) labels apply to sentences rather than aspect terms and (ii) contrary to these terms, the questions used in medical questionnaire can be very similar semantically (e.g., “Is your pain constant?” “Is your pain a temporary problem?”) making it harder to extract the correct answer from the input dialog.

Closest to our work, Toudeshki et al. (2021) showed that pre-trained NLI models can be used to fill in questionnaires from dialogs in a zero-shot setting. We depart from their work in that we propose different ways of transforming and selecting dialog content and investigate how this impact zero-shot, dialog-based, automatic questionnaire filling.

3 Automatic Questionnaire Filling (AQF)

Task. Given a dialog D and a questionnaire Q , the Automatic Questionnaire Filling task consists in providing an answer a_i for each question $q_i \in Q$.

We address the task in a zero-shot setting (no training data). For evaluation, we provide a test set consisting of 100 dialogs and their associated questions and answers.

Questionnaire. We consider two types of questions: Closed Questions (CQ) and Agreement Likert Scale (ALS) questions. CQ have three possible answers (yes, no or Not Applicable, i.e. the dialog does not address the question) and ALS has five (totally disagree, rather disagree, agree, totally agree, NA). As illustrated in Figure 1, questions are reformulated as declarative statements with multiple choice answers. With the emergence of health-bots, AQF can help transform human-bot dialogs into structured data which can be used by physicians to track patients condition. In particular, it can be used to fill in questionnaires such as the Pain Beliefs and Perceptions Inventory (PBPI, (Williams and Thorn, 1989)) questionnaire which includes 16 questions and is standardly used in the context of clinical studies.

Collecting dialogs that include information for all of these questions is a difficult task however. To facilitate data collection for the creation of the test set, we therefore decrease the number of questions

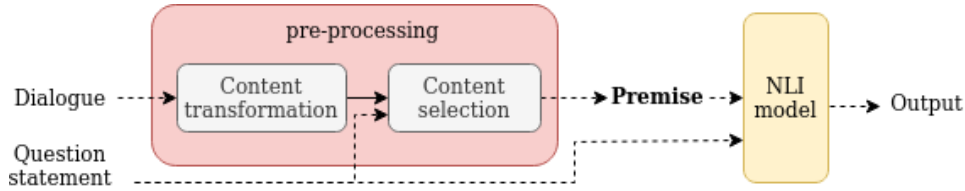


Figure 2: Dialog pre-processing schema

by selecting five questions out of sixteen. Because the questions in the PBPI are often very similar, and knowing the answer to one of them allows deriving the answer to others, we chose questions that are semantically distinct from one another. The list of all PBPI questions is given in Appendix C and the five selected questions are indicated in bold.

Test Data. To evaluate our approach, we create a test set of 100 dialogs and their associated question/answer pairs.

The creation of the test data involves first, collecting human-bot dialogs and second, extracting answers to the PBPI questions from the collected dialogs.

Collecting Dialogs. We collect the dialogs using the Amazon Mechanical Turk platform and asking Turkers to interact with the ComBot health bot (Liednikova et al., 2021) while behaving as if they had chronic pain issues. To avoid Turkers introducing the PBPI questions verbatim in the dialog, they were given a list of topics to be mentioned rather than the questions themselves (See details in Appendix D). In this way, we ensure that the collected dialogs address the questions to be answered while encouraging their diversified paraphrasing during the conversation. Turkers received bonuses each time they mention a topic. Turkers were also given the ability to modify the bot utterance in order to redirect the conversation more easily: they could reject the current candidate in which case, the turn with the next highest confidences score would be displayed by the bot. More information about Turkers payments is provided in the Ethic section (Sec. A). Details of the instructions given to the Turkers and a screenshot of the annotation interface are given in the Appendix.

Identifying Question Answers. Two annotators with good English proficiency were asked to select the correct answer for each of the five selected questions based on each of the 100 collected dialogs. We computed agreement between the two annotators on all Q/A pairs and all 100 dialogs.

The Kappa score is 0.94 for CQ and 0.86 for ALS question type. Thereafter, we used adjudication to decide on the final answer for all cases where the two annotators disagreed. The annotators were the first two authors of this paper.

The final test corpus consists of 100 dialogs, each associated with 10 questions (5 yes/no questions and 5 ALS questions) and their answers. Dialog length varies from 4 to 70 turns and from 47 to 593 tokens, with 17.1 turns and 218.7 tokens on average.

4 Approach

Following Toudeshki et al. (2021), we model question answering as an NLI task where the premise is derived from the dialog, the hypothesis from the question and the answer from the NLI result. Given a question and a dialog, our model, illustrated in Figure 2, answers the question in three steps as follows.

Deriving an NLI Premise from the dialog. The NLI premise is derived from the input dialog using first, Content Transformation and second, Content Selection. As detailed in Section 5, we experiment with different ways of transforming and selecting content.

Deriving an NLI hypothesis from a question. To derive an NLI hypothesis from a question, we simply represent questions as statements (E.g., "I have pain regularly" instead of "Do you have pain regularly?"). Since the PBPI questionnaire questions are already in the form of a statement, we did not make any changes to them and used them as they are.

Deriving the answer. We use RoBERTa large (Liu et al., 2019)¹ fine-tuned on the MNLI dataset (Williams et al., 2018) to determine the entailment relation. We then derive the answer from the entailment relation between dialog and question as

¹<https://huggingface.co/roberta-large-mnli>



Figure 3: Map NLI scores to ALS answer types

follows.

For Close Questions, we set the answer to "Yes" if NLI returns an entailment, "No" if it returns a contradiction and "NA" if it returns "neutral".

For ALS questions, we map the NLI result to agreement choices as follows. If "neutral" has the highest score, the answer is "NA". Else, the contradiction score is subtracted from the entailment score. The subtraction result lies in a range of (-1,1) which is uniformly divided into 5 segments corresponding to the 5 ALS answer types, as shown in figure 3.

5 NLI-oriented Dialog Pre-processing

We consider different ways of transforming and selecting dialog content.

We also study the impact of the NLI model used, comparing DeBERTa, the model used in Toudeshki et al. (2021), with RoBERTa (Liu et al., 2019), the model used in our approach.

The DeBERTa model (He et al., 2020)² extends the BERT architecture with two innovative techniques: disentangled attention mechanism and an enhanced mask decoder. We compare AQF models with and without pre-processing and based on RoBERTa vs. DeBERTa, and find that whereas, when no pre-processing is applied, a DeBERTa model generally outperforms a RoBERTa-based model, the reverse is true when pre-processing is applied. This shows that while the improved DeBERTa-based, NLI model helps bridge the gap between dialog and text, explicit pre-processing still yields better results.

5.1 Content transformation

Null Transformation (CT_{null}) A null transformation baseline where we simply concatenate the turns of the input dialog. To encode the speaker information in each turn, the utterance is accompanied by the speaker role (patient/bot) at the beginning.

Summary (CT_{sum}) Pairs of adjacent turns are summarized, and the resulting summaries are con-

catenated. In this way, the input dialog is transformed into a sequence of two-turn summaries. We also tried summarizing the whole dialog in one go but found that applying summarization on each two turns rather than on the whole dialog gives better results. We use the **BART-large** model³ (Lewis et al., 2020) fine-tuned on the News summarization corpus XSUM (Narayan et al., 2018) and on the dialog summarization corpus SAMSum (Gliwa et al., 2019). The model achieves ROUGE-L score of 0.44 on SAMSum test set⁴.

Long Answers (CT_{answer}) In information seeking dialog, adjacent turns often are question-answer pairs. Based on this observation, we map each pair of adjacent turns in the dialog into a single declarative sentence assuming that the first turn is a question (e.g., "Which drug did you take?"), the second is a short answer to that question (e.g., "Doliprane") and the sentence derived from the mapping is a long answer to the question (e.g., "I took Doliprane"). To learn this mapping, we fine-tune T5 (Raffel et al., 2019), a pre-trained encoder-decoder model, on two datasets of (question, incomplete answer, full answer) triples, one for wh- and one for yes-no (YN) questions. For wh-questions, we use 3,300 entries of the dataset consisting of (question, answer, declarative answer sentence) triples gathered by Demszky et al. (2018) using Amazon Mechanical Turk workers. For YN questions, we used the SAMSum corpus, (Gliwa et al., 2019) which contains short dialogs in chat format. We created 1,100 (question, answer, full answer) triples by automatically extracting YN (question, answer) pairs from this corpus and manually associating them with the corresponding declarative answer. Data was splitted into train and test (9:1) and the fine-tuned model achieved 0.90 ROUGE-L score on the test set.

This fine-tuned model was applied to each two subsequent turns of the input dialogs, and the resulting declarative sentences were then concatenated to

²<https://github.com/microsoft/DeBERTa>

³<https://huggingface.co/Salesforce/bart-large-xsum-samsum>

⁴<https://paperswithcode.com/sota/abstractive-text-summarization-on-samsum>

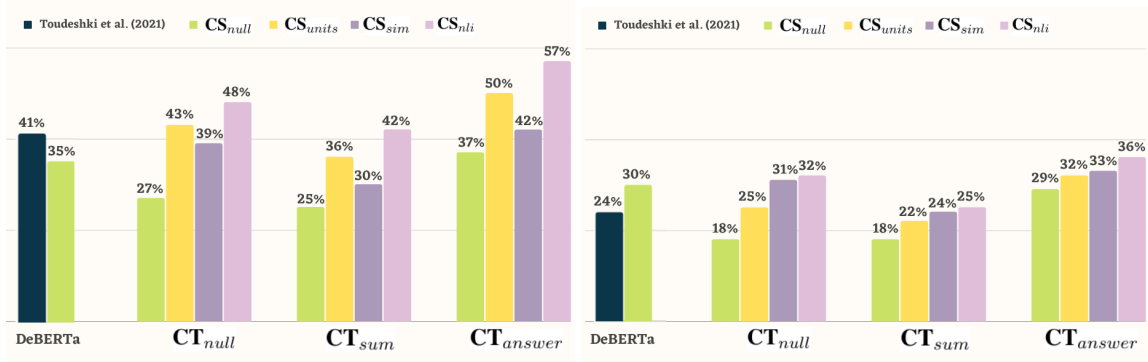


Figure 4: F1 macro average for Close Questions (on the left) and ALS questions (on the right) for the RoBERTa variant of our model. The two most left columns indicate the performance of (Toudeshki et al., 2021)’s model on their (dark blue) and our (light green) test set. The best results are obtained by the CT_{answer} , CS_{nli} model.

form the declarative transform of the whole dialog.

5.2 Content selection

The transformation operations described in the previous section yield sequences of dialog turns, two-turn summaries or full answers. We call these "input units" and consider three ways of pre-selecting the input units that will be used as premise when testing for entailment.

Null Content Selection (CS_{null}) A null content selection baseline where the premise is the concatenation of all the input units produced by the content transformation operations (dialog turns, sequence of two turn summaries, sequence of full form answers).

Unit-Based (CS_{units}). Each question is assessed against each input item. Given an input sequence I_n of length n , the answer a_i to a question q is then determined by aggregating the resulting entailment probabilities as follows:

- $a_i = NA$ if for all input items $i \in I_n$, the NA probability is highest.
- $a_i = Yes$ (resp. $a_i = No$) if for at least one item $i \in I_n$, the Yes (resp. No) probability is highest and the highest Yes (resp. No) probability is higher than the highest No (resp. Yes) probability.

Similarity (CS_{sim}). For each question q , we select a subset of input units that are semantically similar to q . We encode question and input units using SBERT⁵ (Reimers and Gurevych, 2019) and

⁵<https://huggingface.co/sentence-transformers/paraphrase-distilroberta-base-v2>

compute *cosine similarity* for each $(q, \text{input unit})$ pair. We then select items whose similarity score is higher than 0.5, concatenate them and use the result as the NLI premise.

NLI (CS_{nli}). For each question q in the questionnaire, we select the input units that are related to q using the NLI model (RoBERTa-Large). Specifically, we select sentences which have an entailment or contradiction score higher than 0.5. All selected sentences are then concatenated to form the NLI premise.

5.3 Baseline and Comparison

Our baseline is the null method ($CT_{null} + CS_{null}$) i.e., the approach where question answering applies to the untransformed, unfiltered dialog. To compare our approach with Toudeshki et al. (2021), we also report the performance of their model on both their test set (10 dialogs) and on ours (100 dialogs).

6 Results

We evaluate our approach using macro and weighted F1 score.

6.1 How much does pre-processing help improve performance ?

Figure 4 shows the results for all combinations of our content transformation and selection methods⁶.

Improvement over the baseline. Comparing our best model (CT_{answer}, CS_{nli}) with the no-preprocessing CT_{null}, CS_{null} baseline, we see (Figure 4) that pre-processing can multiply the

⁶We first focus on the results of our RoBERTa based model and delay the comparison with DeBERTa based models to Section 6.4.

Two turns
bot: do you feel anxiety or stress during nights awakenings ?
patient: I feel stressed during night awakenings although I am not feeling guilty about being in pain.
Generated summary Patient feels stressed during night awakenings although he's not in pain.

Table 1: An example of the summarization model performance on two subsequent turns, showing missing and **inconsistent** information in the output summary

macro and weighted F1 scores by two. The best pre-processing method combines a question+answer to sentence transformation (CT_{answer}) with the entailment-based content selection method (CS_{nli}).

Content transformation The CT_{answer} question+answer transform, which merges pairs of adjacent dialog turns into declarative statements, consistently yields the best results. A possible explanation is that this transform yields an input, a declarative sentence, which is consistent with the format of the training data used for NLI models.

Conversely, summarization (CT_{sum}) has the lowest performance. This could be due to errors such as hallucinations or omissions known to be produced by summarization systems (Zhao et al., 2020). Table 1 shows an example of such errors when applying the CS_{sum} transformation.

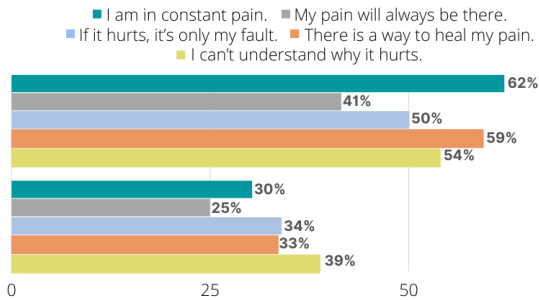


Figure 5: Break down of F1 macro average scores for each question based on out-performed model ($CT_{answer} + CS_{nli}$) results

Content selection The NLI-based content selection method (CS_{nli}) consistently outperforms other content selection approaches. This is consistent with Toudeshki et al. (2021)’s findings that for automatic questionnaire filling in a medical setting, NLI models performed better on average on polar and ALS question types.

We also see that the second best performing content selection method varies depending on the question type. As CS_{unit} first filters question/item pairs with highest probability, the method works well on CQ questions but struggles to handle more nuanced ALS questions which leads to an overall drop in performance on ALS questions.

6.2 Impact of pre-processing on different question/answer types

Table 2 shows the results for all combinations of pre-processing steps for each question/answer type.

Agreement answers (Yes, Totally agree) have the highest accuracy (about 70% in the best case) in both CQ and ALS questions, which suggests that the NLI model is better at confirming rather than rejecting a statement.

On *CQ questions*, various content selection methods have different impacts on each answer type. CS_{sim} shows much lower (3-4 times lower) performance on ‘No’ class than on ‘NA’ or ‘Yes’, CS_{null} has higher accuracy for the ‘NA’ class than for ‘Yes’ or ‘No’ classes and CS_{nli} performs better on ‘Yes’ and ‘No’ answers than on ‘NA’. Both CS_{nli} and CS_{units} gives the most balanced F1 distribution across classes.

For *ALS questions*, CS_{nli} and CS_{sim} show the best results. While the CS_{nli} model is best at identifying ‘Totally agree’ and ‘Totally disagree’ classes, CS_{sim} distinguishes well whether the answer is absent (‘NA’) or whether it belongs to the ‘Totally agree’ class.

Performance on ALS questions is always lower. This can be explained by choice of threshold that distinguishes classes ‘Totally agree’ and ‘Agree’ as well as ‘Totally disagree’ and ‘Rather disagree’. As mentioned above, CS_{units} favors the extreme classes which leads to a higher performance drop in comparison with CS_{sim} on ALS.

6.3 Break down of results for each question

Figure 5 presents the results of our best model ($CT_{answer} + CS_{nli}$) for each PBPI question separately.

The question “I am in constant pain.” obtains highest score in CQ, while it performs poorly in ALS, demonstrating that the model is effective at detecting the presence of consistent pain but bad at predicting the level of agreement. The same behavior can be seen for the question “There is a way to heal my pain”. On the other hand, for question “My pain will always be there” gets lowest score

	CQ					ALS						
support	NA	YES	NO	macro	weighted	NA	TD	RD	A	TA	macro	weighted
	142	228	130			142	54	79	115	110		
CT_{null}												
CS_{null}	0.39	0.15	0.27	0.27	0.25	0.28	0.11	0.26	0.07	0.16	0.18	0.18
CS_{units}	0.33	0.48	0.46	0.43	0.43	0.33	0.25	0.02	0.07	0.58	0.25	0.25
CS_{sim}	0.52	0.55	0.10	0.39	0.42	0.54	0.07	0.09	0.23	0.60	0.31	0.36
CS_{nli}	0.34	0.60	0.48	0.48	0.50	0.34	0.29	0.08	0.21	0.67	0.32	0.34
CT_{sum}												
CS_{null}	0.41	0.11	0.23	0.25	0.23	0.36	0.12	0.21	0.11	0.10	0.18	0.20
CS_{units}	0.32	0.33	0.43	0.36	0.35	0.32	0.23	0.06	0.02	0.44	0.22	0.23
CS_{sim}	0.49	0.40	0.02	0.30	0.32	0.51	0.00	0.05	0.21	0.46	0.24	0.30
CS_{nli}	0.37	0.43	0.46	0.42	0.42	0.31	0.28	0.10	0.09	0.48	0.25	0.26
CT_{answer}												
CS_{null}	0.45	0.28	0.37	0.37	0.35	0.41	0.27	0.27	0.17	0.33	0.29	0.30
CS_{units}	0.40	0.59	0.51	0.50	0.52	0.41	0.29	0.17	0.16	0.57	0.32	0.33
CS_{sim}	0.53	0.60	0.13	0.42	0.46	0.55	0.10	0.20	0.23	0.59	0.33	0.38
CS_{nli}	0.45	0.70	0.57	0.57	0.59	0.42	0.35	0.16	0.23	0.65	0.36	0.38

Table 2: F1-Scores for RoBERTa for closed (CQ) and agreement Likert scale (ALS) question types; TD - totally disagree, RD - rather disagree, A - agree, TA - totally agree. CT: content transformation, CS: content selection.

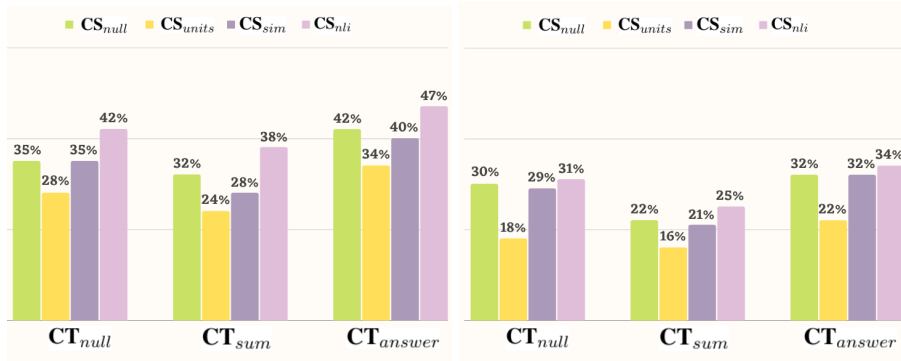


Figure 6: F1 macro average for the DeBERTa variant of our model on Closed Questions (CQ) on the left and Agreement Likert Scale (ALS) on the right. Test set of 100 dialogs with 10 questions each (5 yes/no questions and 5 ALS questions).

for both question types. The presence of the term “always” in the question turns it into a strong statement and consequently the model mostly rejects the statement unless it has been explicitly mentioned in the dialog.

6.4 Comparison with previous work and a different classifier (RoBERTa vs. DeBERTa)

Our model differs from previous work by Toudeshki et al. (2021) in two ways: it includes a pre-processing phase and uses the RoBERTa classifier whereas Toudeshki et al. (2021) applies DeBERTa to the whole input dialog. We compare our model with (i) the same model using DeBERTa and (ii) Toudeshki et al. (2021)’s model both on their and our test set.

Comparison with previous work In Figure 4, the two columns on the far left show the performance of Toudeshki et al. (2021)’s model on two test sets: the test set they used (10 instances and 16 questions) and our test set (100 instances and 5 questions).

Unsurprisingly, Toudeshki et al. (2021)’s results vary with the test set: while they report F1 score of 41 for CQ and 24 for ALS questions on their test set, these change to 35 and 30 on ours.

We also see that Toudeshki et al. (2021)’s DeBERTa-based, no pre-processing model outperforms our RoBERTa-based, null-preprocessing model (CT_{null} , CS_{null}) on both test sets. We conjecture that this difference can be explained by DeBERTa’s improved attention mechanism, which se-

support	CQ					ALS						
	NA	YES	NO	macro	weighted	NA	TD	RD	A	TA	macro	weighted
	142	228	130			142	54	79	115	110		
CT_{null}												
CS_{null}	0.43	0.33	0.31	0.35	0.35	0.41	0.23	0.19	0.22	0.47	0.30	0.32
CS_{units}	0.15	0.29	0.40	0.28	0.28	0.15	0.21	0.07	0.00	0.45	0.18	0.17
CS_{sim}	0.51	0.45	0.09	0.35	0.37	0.54	0.03	0.05	0.24	0.60	0.29	0.35
CS_{nli}	0.34	0.51	0.40	0.42	0.43	0.29	0.23	0.17	0.21	0.63	0.31	0.32
CT_{sum}												
CS_{null}	0.40	0.29	0.26	0.32	0.31	0.37	0.16	0.11	0.16	0.31	0.22	0.25
CS_{units}	0.20	0.18	0.33	0.24	0.23	0.20	0.17	0.11	0.05	0.26	0.16	0.16
CS_{sim}	0.48	0.34	0.01	0.28	0.29	0.49	0.00	0.00	0.17	0.40	0.21	0.27
CS_{nli}	0.39	0.39	0.35	0.38	0.38	0.37	0.20	0.07	0.15	0.45	0.25	0.27
CT_{answer}												
CS_{null}	0.44	0.48	0.35	0.42	0.43	0.43	0.26	0.14	0.19	0.57	0.32	0.34
CS_{units}	0.19	0.45	0.39	0.34	0.36	0.19	0.20	0.07	0.10	0.55	0.22	0.23
CS_{sim}	0.52	0.51	0.16	0.40	0.42	0.53	0.09	0.15	0.20	0.61	0.32	0.37
CS_{nli}	0.40	0.60	0.42	0.47	0.50	0.41	0.30	0.21	0.16	0.63	0.34	0.36

Table 3: F1-Scores for DeBERTa for closed (CQ) and agreement Likert scale (ALS) question types; TD - totally disagree, RD - rather disagree, A - agree, TA - totally agree. CT: content transformation, CS: content selection.

lects relevant information in the input dialog with respect to the hypothesis.

However, our best model outperforms Toudeshki et al. (2021)’s approach by 22 points F1 for CQ questions and 6 points for ALS questions which indicates that pre-processing better helps bridge the gap between dialog and NLI-based QA.

DeBERTa vs. RoBERTa figure 6 and Table 3 show the result of our model when using DeBERTa instead of RoBERTa.

When using pre-processing, we see that the best RoBERTa model (CT_{answer}, CS_{nli}) outperforms the best DeBERTa model by 10 points F1 for CQ questions and 2 points for ALS questions.

Conversely, when no pre-processing is applied, the DeBERTa variant of our model outperforms the RoBERTa variant which is consistent with the results discussed in the previous paragraph. For the DeBERTa variant, we observe that the CS_{null} baseline is no longer the lowest performing content selection approach, while the performance of CS_{units} and CS_{sim} becomes lower than the baseline (CS_{null}). This highlights the fact that the DeBERTa model performs better without weak content selection approaches. On the other hand, it can be seen that the impact of content selection and transformation approaches is significant in RoBERTa, although using a weaker classifier, and our model outperforms previous work. This shows that the proposed select-and-transform

pre-processing approach improves results in both RoBERTa and DeBERTa, though this improvement is more significant in RoBERTa, suggesting that this latter model is more sensitive to the form and size of the input content.

7 Conclusion

In this paper, we studied how dialog pre-processing can impact the task of filling medical questionnaires based on patient-bot interactions. Our experimental results show that converting pairs of adjacent turns to declarative sentences and selecting input units based on their entailment relation with the question can significantly enhance performance.

References

- Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. 2020. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*.
- Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. *arXiv preprint arXiv:1809.06309*.
- Dorottya Demszky, Kelvin Guu, and Percy Liang. 2018. Transforming question answering datasets into natural language inference datasets. *arXiv preprint arXiv:1809.02922*.

- Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. Samsun corpus: A human-annotated dialogue dataset for abstractive summarization. *arXiv preprint arXiv:1911.12237*.
- Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. DeBERTa: Decoding-enhanced BERT with disentangled attention. *arXiv preprint arXiv:2006.03654*.
- H Jang, E Rempel, D Roth, G Carenini, and NZ Janjua. 2021. Tracking COVID-19 discourse on twitter in North America: Topic modeling and aspect-based sentiment analysis. *Journal of Medical Internet Research*.
- Qingnan Jiang, Lei Chen, Ruifeng Xu, Xiang Ao, and Min Yang. 2019. A challenge dataset and effective models for aspect-based sentiment analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6280–6285.
- 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.
- Junlong Li, Zhuosheng Zhang, and Hai Zhao. 2020. Knowledgeable dialogue reading comprehension on key turns. *arXiv preprint arXiv:2004.13988*.
- Anna Liednikova, Philippe Jolivet, Alexandre Durand-Salmon, and Claire Gardent. 2021. Gathering information and engaging the user combot: A task-based, serendipitous dialog model for patient-doctor interactions. In *Proceedings of the Second Workshop on Natural Language Processing for Medical Conversations*, pages 21–29.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- 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. *arXiv preprint arXiv:1808.08745*.
- Kate Pearce, Tiffany Zhan, Aneesh Komanduri, and Justin Zhan. 2021. A comparative study of transformer-based language models on extractive question answering. *arXiv preprint arXiv:2110.03142*.
- 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*.
- Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.
- Jiangtao Ren, Naiyin Liu, and Xiaojing Wu. 2020. Clinical questionnaire filling based on question answering framework. *International Journal of Medical Informatics*, 141:104225.
- Mary Sun. 2022. *Natural Language Processing for Health System Messages: Deep Transfer Learning Approach to Aspect-Based Sentiment Analysis of COVID-19 Content*. Ph.D. thesis, Harvard University.
- Farnaz Ghassemi Toudeshki, Philippe Jolivet, Alexandre Durand-Salmon, and Anna Liednikova. 2021. Zero-shot clinical questionnaire filling from human-machine interactions. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 51–62.
- Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.
- David A. Williams and Beverly E. Thorn. 1989. An empirical assessment of pain beliefs. *Pain*, 36(3):351–358.
- Chunyuan Yuan, Wei Zhou, Mingming Li, Shangwen Lv, Fuqing Zhu, Jizhong Han, and Songlin Hu. 2019. Multi-hop selector network for multi-turn response selection in retrieval-based chatbots. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 111–120.
- Changchang Zeng, Shaobo Li, Qin Li, Jie Hu, and Jianjun Hu. 2020. A survey on machine reading comprehension—tasks, evaluation metrics and benchmark datasets. *Applied Sciences*, 10(21):7640.
- Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019. DialoGPT: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.
- Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling multi-turn conversation with deep utterance aggregation. *arXiv preprint arXiv:1806.09102*.

Zhuosheng Zhang, Junlong Li, and Hai Zhao. 2021. Multi-turn dialogue reading comprehension with pivot turns and knowledge. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1161–1173.

Zheng Zhao, Shay B. Cohen, and Bonnie Webber. 2020. Reducing quantity hallucinations in abstractive summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2237–2249, Online. Association for Computational Linguistics.

A Ethics

Regarding Regulation (EU) 2017/745, described software is intended for general uses, even when used in a healthcare environment, it is intended for uses relating to lifestyle or well-being that do not constitute any a medical prediction and medical prognosis function without doctors validation or correction.

We gathered dialogs for experiments using Amazon Mechanical Turk. Because of the task's difficulty and estimated completion time, we set the initial reward at 1\$. We assigned 0.5\$ bonus for each key point mentioned by the user during the dialogue. If the user was successful in mentioning all five key points, he was awarded a bonus of 2.5\$ in total.

B Experiment time estimation

The experiments were conducted with a laptop having Intel® Core™ i7-10610U CPU @ 1.80GHz * 8 and NVIDIA Quadro P520.

C Questionnaire

PBPI questionnaire statements are provided in table 4.

D Data Collection

Instructions used for data collection in Amazon Mechanical Turk and the interface are shown in figures 7, 8 and 9.

We requested the Turkers to converse with the heath-bot for at least 10 turns in total.

Id	Question
1	No one is able to tell me why it hurts.
2	I thought my pain could be healed, but now I'm not so sure.
3	There are times when it doesn't hurt.
4	My pain is difficult for me to understand.
5	My pain will always be there.
6	I am in constant pain.
7	If it hurts, it's only my fault.
8	I don't have enough information about my pain.
9	My pain is a temporary problem in my life.
10	I feel like I wake up with pain and fall asleep with it.
11	I am the cause of my pain.
12	There is a way to heal my pain.
13	I blame myself when it hurts.
14	I can't understand why it hurts.
15	One day, again, I won't have any pain at all.
16	My pain varies in intensity but it is always present with me.

Table 4: List of questions in PBPI questionnaire

Task Description

In this task, you are going to talk to a chatbot about health and quality of your life.

You are supposed to **play the role of a chronic pain patient**, and **share your pain with the bot**.

What is chronic pain? Doctors often define chronic pain as any pain that lasts for 3 to 6 months or more. Chronic pain can have real effects on day-to-day life and mental health.

It is very important that you **mention about all following key points during your conversation** (in a seamless way):

1. (Constantly/Temporarily) in pain
2. (Having/Losing) hope for getting healed
3. (Feeling/Not feeling) guiltiness that the pain is your fault
4. (Possibility/Impossibility) of healing
5. (Understanding/Not understanding) the reason of having pain

Try to **give an implicit and seamless reference to these keypoints** in the dialogue (with considering the flow of conversation). **Prevent using the same wording** in your messages.

**** BONUS ****

Playing the role of a chronic pain patient and mentioning each keypoint will get 0.5 \$ bonus. By mentioning all keypoints you will get 2.5 \$ bonus (do not use the same wording).

note1: It is an information seeking conversation and **you are not expected to ask questions from the bot**.

note2: Wait for the bot message to be appeared completely, and then reply.

note3: When it is your time to reply, **reply only once**.

Figure 7: Instructions (part 1)

Lead the conversation

The chatbot is not developed to ask you explicitly about these key points. Therefore, you have to mention them creatively during the dialog flow.

To make it easier for you, we have given you the authority of **controlling chatbot messages**. You can direct the conversation by **changing chatbot reply**. To do that, you can **click on the "next" button** (below the bot message) to change the chatbot utterance and if you found it good enough you can just continue the conversation.

Annotating each user reply

After you entered your answer, you will notice **5 checkpoints appear below your answer**. Each check point refers to each of the keypoints. **If one or multiple keypoints have been mentioned in your answer (implicitly or explicitly) choose the related checkpoints.**

Please keep in my mind that you have to **fill check points before entering your next response**. They would be disabled afterwards.

End the conversation

To end the conversation, you can click on **green "Submit" button**. But before that, **wiat for the bot message to be appeared completely, and then press the submit button**.

If you click on the button before reaching to the minimum number of turns (5 messages each user), you will receive an alert error message and be taken back to the conversation to complete the task.

Figure 8: Instructions (part 2)

Talk to the chatbot about quality of life!

Task Description

In this task, you are going to talk to a chatbot about health and quality of your life.

You are supposed to **play the role of a chronic pain patient, and share your pain with the bot**. What is chronic pain? Doctors often define chronic pain as any pain that lasts for 3 to 6 months or more. Chronic pain can have real effects on day-to-day life and mental health.

It is very important that you **mention about all following key points during your conversation** (in a seamless way):

1. (Constantly/Temporarily) in pain
2. (Having/Losing) hope for getting healed
3. (Feeling/Not feeling) guiltiness that the pain is your fault
4. (Possibility/Impossibility) of healing
5. (Understanding/Not understanding) the reason of having pain

Try to **give an implicit and seamless reference to these keypoints** in the dialogue (with considering the flow of conversation). **Prevent using the same wording** in your messages.

**** BONUS ****

Playing the role of a chronic pain patient and mentioning each keypoint will get 0.5 \$ bonus. By mentioning all keypoints you will get 2.5 \$ bonus (do not use the same wording).

Combobot: Hi, how are you ?

Worker: I am doing ok I supposed but I have a lot of pain.

Check the key points mentioned in your reply, if there is none, then leave it as it is

- 1. (Constantly/Temporarily) in pain
- 2. (Having/Losing) hope for getting healed
- 3. (Feeling/Not feeling) guiltiness that the pain is your fault
- 4. (Possibility/Impossibility) of healing
- 5. (Understanding/Not understanding) the reason of having pain

Combobot E: I'm sorry to hear that. I hope you feel better soon. What kind of pain?

To change bot reply, click on the next button.

Please enter here...

Figure 9: Interface