DECIDING WHETHER TO ASK CLARIFYING QUESTIONS IN LARGE-SCALE SPOKEN LANGUAGE UNDERSTANDING

Joo-Kyung Kim, Guoyin Wang, Sungjin Lee, Young-Bum Kim

Amazon Alexa AI

ABSTRACT

A large-scale conversational agent can suffer from understanding user utterances with various ambiguities such as ASR ambiguity, intent ambiguity, and hypothesis ambiguity. When ambiguities are detected, the agent should engage in a clarifying dialog to resolve the ambiguities before committing to actions. However, asking clarifying questions for all the ambiguity occurrences could lead to asking too many questions, essentially hampering the user experience. To trigger clarifying questions only when necessary for the user satisfaction, we propose a neural self-attentive model that leverages the hypotheses with ambiguities and contextual signals. We conduct extensive experiments on five common ambiguity types using real data from a large-scale commercial conversational agent and demonstrate significant improvement over a set of baseline approaches.

***Index Terms*—** spoken language understanding, clarifying questions, user satisfaction, self-attention

1. INTRODUCTION

The spoken language understanding (SLU) task of a large-scale conversational agent goes through a set of components such as automatic speech recognition (ASR), natural language understanding (NLU) including domain classification, intent classification, and slot-filling [1], and skill routing (SR) [2, 3], which is responsible for selecting an appropriate service provider to handle the user request.

When a user interacts with the agent, the underlying systems may not be able to understand what the user actually wants if the utterance is ambiguous. Ambiguity comes from ASR when audio cannot be recognized correctly (*e.g.*, audio quality issues can cause ASR to confuse "Five minute timer" and "Find minute timer"); it comes from NLU when the user's request cannot be interpreted correctly (*e.g.*, "Garage door" could mean open it or close it); it comes from SR when it is not possible to confidently select the best experience between multiple valid service providers (*e.g.*, "Play frozen" can mean playing `video`, `soundtrack`, or `game`), and so on. Ignoring such ambiguities from upstream components can pass incorrect signals to the downstream and lead to an unsatisfactory user experience.

ASR	Recognized Dialog	Label
Correct	**U:** Set a timer for 15 minutes **S:** 15 minutes, right? **U:** Yes. **S:** 15 minutes, starting now.	**Unnecessary**
Incorrect	**U:** Set a timer for 50 minutes **S:** 50 minutes, right? **U:** No, 15 minutes **S:** 15 minutes, starting now.	**Necessary**

Fig. 1: Examples of Unnecessary and Necessary clarifying turns when a user actually said "Set a timer for **15** minutes" but ASR ambiguity exists between 15 and 50, where the most confident ASR is (top: correct (15), bottom: incorrect (50).)

Thus, when the agent is unsure due to ambiguity, it should engage in a clarifying dialog before taking actions. However, asking clarifying questions for all the detected ambiguous utterances can end up spamming users with too many redundant questions, resulting in poor user experiences as shown in Fig. 1.

In SLU systems, the ambiguities are typically detected by applying thresholds to the system components' confidence scores along with certain heuristics [4, 5, 6, 7, 8, 9, 10]. However, using thresholds to properly filter ambiguities is a practically difficult task in large-scale SLU. First, it is non-trivial for humans to define fine-grained thresholds for the upstream components or rules for deciding clarification considering dialog and ambiguity context information. It is more complicated to make the decisions when multiple ambiguities from different components co-occur since each ambiguity is entangled to the other ambiguities in many cases. Therefore, relying on the thresholding based approach is not scalable in recent general-purposed conversational agents such as Amazon Alexa, Google Assistant, and Apple Siri since a huge number of various dialog contexts and ambiguity situations should be considered. Second, each system component is differently supervised and the confidence thresholding is separately decided in large-scale pipelined SLU systems [11]. For example, an ASR prediction with softmax outputs can be considered ambiguous when the second best ASR confidence is above 30% while an intent classification with sigmoid outputs

is defined to be ambiguous when the second best intent's confidence is above 80%. In addition, each upstream component can be independently updated in large-scale systems while previous studies such as [8] and [10] assume that each component is fixed or the components can be holistically manageable. Therefore, many of the detected ambiguities are indeed false positives (*i.e.*, unambiguous) with a different ratio for each SLU component. Lastly, even if obvious ambiguities exist, clarifying dialogs are unnecessary if the top prediction is correct in terms of user satisfaction. It was shown that more than 60% of ASR errors do not need clarification since many of the errors do not influence the end-to-end performance in a speech-to-speech translation system [12] and we also observe a similar tendency from our work in Section 4.

There are various studies about how to compose clarifying questions[1] and their effectiveness in SLU systems when ambiguities exist [5, 13, 14, 15]. Also, clarifying questions on other tasks such as Q&A [16, 17, 18, 19] and information retrieval [20, 21, 22] are being actively studied. However, none of them specifically focus on initiating clarifying interactions only when necessary in the interest of preventing user experience degradation, which is crucial in large-scale SLU systems.

To address the issue of deciding whether to ask clarifying questions in large-scale SLU systems, we propose a unified neural self-attentive model that makes a global decision on whether to trigger a clarifying question considering ambiguity occurrence information and various contextual signals. We show that the self-attentive representations of the top hypothesis and the aggregated alternative hypotheses from a hypothesis reranker [23, 24, 2, 3] are effective for dealing with the ambiguities from SLU.

Given the fact that a large-scale conversational system supports various devices, languages, and application components, providing access to a wide variety of skills [25, 26, 27], it is not scalable to rely on manually annotated data to train and evaluate the model. Instead, we leverage a user satisfaction model, which has recently attracted significant attention in both academia and industry [28, 29, 30], to generate ground-truth labels at scale. The user satisfaction model we use [30] marks defective turns by examining the input utterance, the system response, and the user's implicit/explicit feedback of their following turns. Having turn-level defect labels, our model is supervised to not trigger a clarifying question when the agent is likely to deliver satisfying experience even if ambiguities are detected from the upstream SLU components.

In this paper, we define five ambiguity types that are popular in the SLU task (see Section 2.2), conduct extensive experiments using real data from a large-scale commercial conversational system, and demonstrate significant improvements over several baseline approaches in reducing unnecessary clarifying interactions.

[1] e.g., asking either reprise questions for targeted clarification or generic questions such as repeat or rephrase requests dependent on the occurred ambiguity contexts.

#	ASR	ASR Conf	Intent	Intent Conf	Slots	Hyp Conf
1	harry potter	0.9	PlayVideo	0.95	videotitle	0.9
2	harry potter	0.9	ReadBook	0.75	booktitle	0.8
3	harry potter	0.9	SoundTrack	0.94	albumtitle	0.6
4	harry potter	0.9	GetWiki	0.7	entity	0.4
5	larry potter	0.6	ReadBook	0.65	booktitle	0.1

Table 1: An example of the ranked hypothesis list from HypRank given *harry potter* as the utterance.

2. AMBIGUITIES IN SLU

Our task is to determine whether to ask clarifying questions when ambiguity signals are captured by any SLU components in the user utterance. In this section, we describe how an input utterance is interpreted with different hypotheses in SLU, the ambiguity types we are dealing with, and how ground-truths of the ambiguous utterances are assigned in our work.

2.1. Hypothesis representations

In large-scale SLU, it is common to represent various possible interpretations of an input utterance as *hypotheses*, each of which contains the outputs and the confidence scores of upstream components such as ASR, domain, intent, and slot-filling results [23, 24, 2, 3]. Given the hypothesis list as the input, a hypothesis ranker (HypRank) is used to rank the input hypotheses. Table 1 shows an example of the hypotheses from HypRank.

Then, given the HypRank output, which is a ranked hypothesis list, the top ranked hypothesis is chosen as the final SLU decision for unambiguous utterances. For ambiguous utterances, since it is unconfident whether the top hypothesis would be promising, clarification interactions allow choosing a non-top/alternative hypothesis. In this work, we focus on deciding whether clarification is necessary or not for the ambiguous utterances.

2.2. Ambiguity Types

We first define five common types of ambiguities in SLU as follows:

- **ASR**: Two ASR outcomes are regarded as ambiguous when their edit distance is 1, their ASR confidences are close, and they produce different slot values. *e.g.*, "thirteen minutes" vs "thirty minutes".
- **Similar Intent Confidences** (IC): The intent of an utterance is ambiguous. *e.g.*, "turn on off" is ambiguous since both `TurnOnIntent` and `TurnOffIntent` can have high confidences by the intent classifier.

Ambiguity	User utterance	Potential clarifying question	User response	System response or <action>
ASR	Set a thirty minute timer	Do you mean thirteen or thirty?	Thirteen	Start a thirteen minute timer
IC	Turn on off	Do you mean turn on or turn off?	Off	<Turn off the agent>
HC	Get me a ride	Do you want Uber or Lyft?	Uber	Finding a uber driver
SNR	Turn on the ligXXX (noisy)	Sorry, could you repeat it?	Turn on the light (clear)	<Turn on the light>
TRUNC	Turn on the	Sorry, turn on what?	The fan	<Turn on the fan>

Table 2: Clarifying dialog examples

- **Similar Hypothesis Confidences** (HC): The final hypothesis confidences from HypRank [2, 3] are similar. *e.g.*, "get me a ride" can have similar confidences for the hypotheses associated with `UBER` and `LYFT` as the service providers.
- **Signal to Noise Ratio** (SNR): When the acoustic noise level is very high, it is not clear whether we can trust the ASR output even if the ASR confidence is sufficiently high.
- **Utterance Truncation** (TRUNC): An utterance can be recognized missing the later tokens due to slow speaking or ASR errors. *e.g.*, if a user said "Music composed by Mozart" but only "Music composed by" are recognized, the missed token should be clarified. In this work, we regard utterances ending with articles ("a", "an", "the"), some possessive pronouns (*e.g.*, "my"), or some prepositions (*e.g.*, "by") as truncated.

Table 2 shows the clarifying dialog examples for different ambiguity types, which demonstrates how clarifying questions can help resolve the ambiguities.

2.3. Ground-truth Labeling

It is difficult to decide whether a clarifying question would be helpful or not when ambiguities exist since each ambiguity is with a different occurrence condition, multiple ambiguities can co-occur, and the top predictions are correct in many cases even if ambiguities exist. In this work, we regard ambiguous utterances with unsatisfactory results as those need clarifications, and vice versa. The rationale is that if a user is unsatisfied with the top predicted hypothesis from the HypRank when ambiguities exist, the user could have been satisfied by allowing the user to choose another hypothesis.

We use the log data from a conversational agent system, where each utterance is assigned to be either satisfactory or unsatisfactory by a user satisfaction metric [30, 28, 29]. Specifically, we use a model-based metric described in [30], which utilizes the current turn's utterance and the response as well as the follow-up turn's utterance to judge whether the current turn is satisfactory or not.

Our labeling method is a weak supervision approach assuming no clarifying questions exist in the log data. If the log data already include turns with clarifying questions, we can identify whether the questions were helpful or not. For example, if a user selected the top hypothesis in the clarification, it was unnecessary to ask since the top hypothesis could be chosen without the clarification. Oppositely, if the user chose the other hypothesis, then the clarification was useful evading unwanted top hypothesis. Formally speaking, the ground-truths can be set with counterfactual learning using the logged data, but this is beyond the scope of this work.

3. MODEL

Figure 2 shows the overall architecture of the proposed model deciding whether to ask clarifying questions or not.

The input to our proposed model is a subset of the HypRank hypotheses described in Section 2.1. The top predicted hypothesis is always included in our model input since it is what to be compared with the alternative hypotheses through clarification dialogs. Then, for each occurred ambiguity, we add the most confident alternative hypothesis corresponding to the ambiguity. For the example in Table 1, assuming 0.8 is the threshold to represent ASR, IC, and HC ambiguity occurrences, and SNR and TRUNC ambiguities did not occur, hypothesis #2 is with the highest confidence among the hypotheses corresponding to HC ambiguity and #3 is the one corresponding to IC ambiguity. Since the hypothesis with different transcript has lower ASR confidence (0.6) than the threshold (0.8), we do not use it as an alternative hypothesis. Therefore, the input sequence to the proposed model consists of hypothesis #1 as the top hypothesis, and #2 and #3 as the alternative hypotheses. [2]

As the model input, each hypothesis is represented as a concatenated vector of ASR, ASR confidence, intent confidence, domain, intent, slots, and ambiguity type.[3] ASR is represented by the output summation of a single layered standard multi-head transformer encoder [31][4], where the word embedding is initialized with GloVe [32]. ASR confidence

[2]SNR and Trunc ambiguities do not have corresponding alternative hypotheses since those ambiguities could be resolved by generating new hypotheses based on additional information from clarification interactions. Therefore, we represent the corresponding hypothesis with $\langle unk \rangle$ vector for each hypothesis elements.

[3]We do not include the hypothesis confidence as an input feature since there exist utterances without the scores due to rule-based or shortlister only hypothesis decision.

[4]We empirically find 4 attention heads shows the best performance.

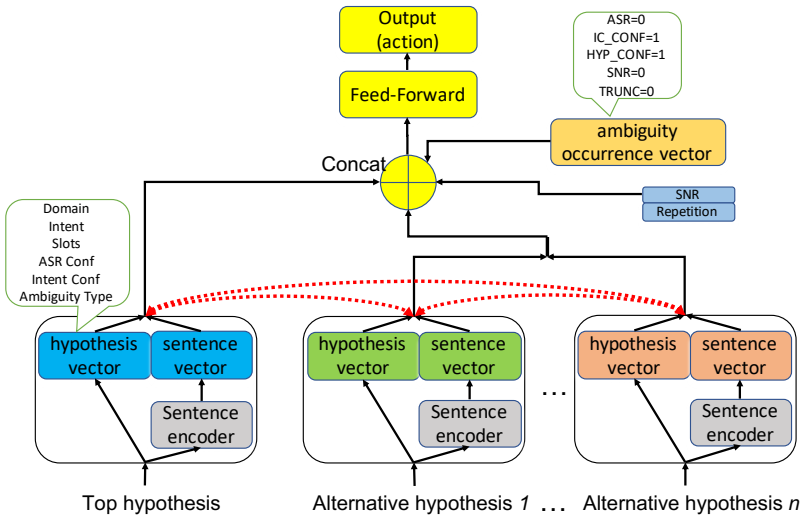


Fig. 2: The model architecture. The top hypothesis and the alternative hypotheses corresponding to the occurred ambiguities are extracted from the HypRank output list and used as the input sequence. On top of the vector sequence, self-attention mechanism is used to produce contextualized hypotheses representations. Then the top hypothesis output, the summation of the alternative hypothesis outputs, the ambiguity occurrence vector, SNR value, and repetition value are concatenated to represent the entire information and it is transformed to an output vector for deciding 'ask' or 'not'.

is a scalar value normalized to be between 0 and 1. A vector for slots is represented as a sum of matched slot key vectors similarly to [2]. Domain, intent, and ambiguity type[5] are also vectorized with embeddings.[6] On top of the hypothesis vector sequence, we obtain a contextualized vector sequence using self-attention with a transformer encoder. For this self-attention, inspired by Set Transformer [33], we do not utilize position encoding since the order of alternative hypotheses for different ambiguities is not informative for the model's decision.

From the contextualized hypotheses, we obtain the top hypothesis's representation and the sum of the alternative hypothesis representation to be used as the inputs to the final prediction layer. Summation of the alternatives is necessary since the number of the alternative hypotheses (*i.e.* # occurred ambiguities) varies for each utterance and they should be aggregated to be used as an input representation. While the majority of self-attentive models for other tasks use single representation aggregated over all the elements in the given sequence, we observe that separating the top hypothesis representation and the aggregated representation over the alternative hypotheses performs better due to different aspects of the top hypothesis and the alternative hypotheses in terms of deciding clarification or not.

In addition to the hypothesis representations, we also use other signals: SNR, which is a scalar value normalized to be between -1 and 1, ambiguity occurrence vector, which is a concatenation of binary values representing the occurred ambiguities, and a binary signal representing repetition of the previous user utterance, which is a common indicator that the same utterance was wrongly recognized or unsatisfactory previously. All these vectors are concatenated and transformed to an output vector through a feed-forward network.

4. EXPERIMENTS

4.1. Datasets

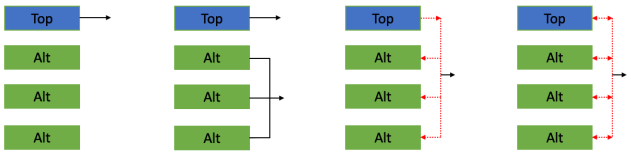
To the best of our knowledge, there is no existing public dataset for asking clarification questions including ASR related features such as ASR confidences and SNR values. Based on an assessment of a randomly sampled ambiguous utterances from a conversational AI system, we estimate that about 23% ambiguous traffic should be resolved for user satisfaction through a clarification dialog. To show statistical significance on the evaluation results and to make the data split similar to real deployment scenarios, we construct a large test set by selecting the second half of the data based on time stamp. We then randomly split the first half of data to training/validation sets with 9:1 ratios. The detailed statistics for each ambiguity type are summarized in Table 3. For example, there are total 4.6M utterances with ASR ambiguity in the test set. However, only 780K of them are worth to clarify for better user satisfaction and the remaining 3.8M do not need to clarify since these are satisfactory to the users even though they are ambiguous. The ratio of 'ask' labels varies for different ambiguity types due to different criteria and thresholds in ambiguity detection.

[5]The top hypothesis's ambiguity type is denoted as `TOP` to differentiate it with the alternative hypotheses' ambiguity types such as `ASR`, `IC`, and `HC`.

[6]The effectiveness of these features is shown in Section 4.5.

Ambiguity type	Train			Valid			Test		
	Total	Ask	No ask	Total	Ask	No ask	Total	Ask	No ask
ASR	4.3M	763K	4.2M	477K	82K	395K	4.6M	780K	3.8M
IC	4.2M	612K	3.6M	464K	67K	397K	4.5M	491K	4M
HC	265K	77K	188K	29K	8K	21K	343K	92K	251K
SNR	16.6M	3.4M	13.2M	1.8M	375K	1.4M	18.2M	3.8M	14.4M
TRUNC	2.6M	1.6M	1M	292K	178K	114K	2.8M	1.7M	1.1M
Total	28.6M	6.5M	22.2M	3M	710K	2.3M	30.4M	6.9M	23.6M

Table 3: Ambiguity dataset sizes.



(a) *no-alt* (b) *alt-avg* (c) *crs-att* (d) *self-att*

Fig. 3: Attention structure illustration. Solid arrows and dotted arrows denote using the outputs and using the attentions, respectively, and `Top` and `Alt` denote the top hypothesis and each alternative hypothesis, respectively: (a) using the top only (b) using the average alternatives (c) the top hypothesis as the key for the cross attention over the alternatives (d) self-attention over all the hypotheses.

4.2. Experiment Setup

In a high level, we consider three approaches in our experiments, (*i*) asking questions for every ambiguity occurrence, denoted as *Always*; (*ii*) utilizing the top hypothesis and the context information as the input to the output layer, denoted as *No-alt* in Figure 3a; (*iii*) using all the top and the alternative hypotheses and the context information.

In the third category, we try different types of alternative hypothesis aggregation. The simplest one is using the top hypothesis vector and simple average over the alternative hypothesis vectors, denoted as *Alt-avg* in Figure 3b. To more effectively represent contextualized information of the hypotheses, we try two types of attention mechanisms: (1) representing alternative hypotheses with cross attention given the top hypothesis as the key, denoted as *Crs-att* in Figure 3c. (2) a complete self-attention over all the hypotheses is *Self-att* in Figure 3d.

When using both the top hypothesis and the alternative hypothesis aggregation, the previous approaches concatenate the top and the alternative vectors to be used as the input to the final layer. We also try their summation, denoted as *Self-sum* to check whether separately representing the top and the alternatives in the final layer is better or not.

4.3. Implementation Details

We train each model with ADAM optimizer [34] for 20 epochs and select the best model based on performance on validation set. The dimensionality of hypothesis components such as domain, intent, and slot vectors and utterance vectors are set to 100. The other hyperparameters for self-attention and positional embedding are identical to the default values in [31].

4.4. Experiment Results

Note that precision can evaluate the model's ability to avoid unnecessary clarifications and recall can measure the ability to ask clarifications when necessary. Hence, we use F1 score, which balances the two metrics, to evaluate the model performance. To make a thorough evaluation, we evaluate both F-1 for each ambiguity type and the overall F-1 for all the types.

The relative F-1 scores over all aforementioned models are summarized in Table 4.[7] Since *Always* always asks clarifying questions for the ambiguous utterances, its F-1 score is the lowest due to low precision even though recall is 100%. As aforementioned, about 23% of all the ambiguous utterances need clarification in our experiment setting, thereby the F-1 score of all the ambiguities is around 37%, where each ambiguity's F-1 is between 20% to 70%. Therefore, using any of the tested models shows significantly better performance.

Compared to *No-alt* model, which does not utilize the alternative hypothesis information, *Alt-avg* does not significantly improve the performance. However, the other models using attention mechanisms to represent the alternative hypotheses significantly outperform *No-alt* model. This indicates that properly represented alternative hypothesis information is an important factor in the model decision. Also, using self-attention, *Self-att*, is again significantly better than using cross-attention with the top hypothesis as the key, *Crs-att*. This shows that the fully contextualized hypothesis representations are helpful. We further deepen the model by considering 2-layer of self-attention, which further improves the performance. Such good performance verifies the effectiveness of our self-attention based model in asking clarification question task. In addition, relatively poor performance of *Self-sum* reflects that the proposed architecture, which separately represents the top hypothesis and the ag-

[7] Due to internal confidentiality policy, we report relative F-1 scores, $(f_x - f_{always}) / f_{always}$, where f_x and f_{always} denote F-1 scores of model x and model $Always$, respectively.

Model	All	ASR	IC	HC	SNR	TRUNC
Always	0	0	0	0	0	0
No-alt	77.79	71.63	99.88	201.86	9.28	88.76
Alt-avg	77.81	72.43	98.88	212.28	8.90	74.89
Crs-att	78.05	72.35	99.23	212.18	9.09	79.02
Self-sum	78.21	70.67	101.49	214.31	9.30	89.43
Self-att	79.61	73.70	103.03	214.01	9.42	**90.42**
Self-att2	**81.09**	**75.91**	**105.34**	**216.80**	**9.79**	87.36

Table 4: Relative F-1 % improvements of different model approaches compared to *Always* scores. *Self-att2* denotes two layers of self-attention.

Model	All	ASR	IC	HC	SNR	TRUNC
No-hyp	-37.74	-49.81	-32.90	-28.84	-31.11	-10.08
ASR	-35.90	-46.72	-34.30	-24.30	-30.11	-7.96
No-sent	-16.01	-21.69	-31.70	-9.55	-5.68	-31.18
Diff-att	-3.09	-4.27	-4.61	-2.85	-1.32	-4.72
No-rpt	-1.51	-2.45	-2.18	-1.48	-0.44	0.59

Table 5: Ablation study results by excluding specific features from *Self-att2* model. *No-hyp* refers to no hypothesis, *ASR* refers to no hypothesis but keeping ASR confidence, *No-sent* refers to no sentence, *Diff-att* denotes using different self-attention weights for the sentence and the other features, and *No-rpt* denotes excluding the repetition feature. Each score is relative to the corresponding *Self-att2* score in Table 4.

gregated alternative hypotheses with concatenation, is more proper for our task than singly aggregated representation, which is more common in self-attentive architectures. This empirically demonstrates the top hypothesis and the alternative hypotheses play different roles in the model decision, thereby separating them is more helpful.

4.5. Ablation Study

We further explore the impact of the input features and architecture settings from our best model, *Self-att2*. We represent each hypothesis as the concatenation of two vectors: the hypothesis vector and the sentence vector. Among the features of the hypothesis vector, ASR confidence is expected to be closely related to ASR and SNR ambiguities. Therefore, we conduct the following experiments: excluding whole hypothesis features (*i.e.*, all the input features except the sentences), excluding hypothesis features but ASR confidence, and excluding sentence vectors. Another architecture decision is using single self-attention weight for the concatenation of the hypothesis vector and the sentence vector. Since those two vectors are significantly different views of a hypothesis, we check if using single attention weight is better than having separate attention weights for the two different vectors. In addition, we check the effectiveness of using the repetition feature since we hypothesize that whether the current utterance is repeated or not is helpful for the model decision.

The ablation study results are shown in Table 5. Excluding the hypothesis vector features (*No-hyp*) results in a big drop in the overall performance. Hence, the hypothesis features are critical in the model decision. Using only ASR confidence signal and the sentence vector (*ASR*) is shown slightly helpful for most ambiguities except IC, but the improvements for ASR and SNR ambiguities are not very high. This means that the contextual features unrelated to the acoustics are also significantly influential to the decision for the acoustics related ambiguities. Excluding the sentence vector (*No-sent*) also causes lower performance but the drop is less compared excluding the hypothesis vector. This indicates that both the sentence and the hypotheses are important but the hypothesis features would provide more information for the model decision. Using different self attention weights (*Diff-att*) shows worse performance, which demonstrates that holistic self attention for the concatenation of the hypothesis vector and the sentence vector is not only simpler but also empirically more effective. One possible reason of the result is that the attention over different sentence vectors would be less influential because sentence vectors in different hypotheses are different only when ASR ambiguity exists. Excluding the repetition feature (*No-rpt*) also shows significant degradation, which reflects its effectiveness.

These findings show that the utilized signals and the architecture decisions are helpful for making proper model predictions.

5. CONCLUSION

In this work, we have introduced five common ambiguities in SLU, where empirically 23% of the utterances with these ambiguities need be clarified. To decide whether asking a clarifying question would be helpful, we have proposed a scalable neural self-attentive model, where the top and the alternative hypotheses, ambiguity occurrence information, and the other contextual information are used as the input representation and then the model predicts whether to ask clarifying questions or not. The model is supervised leveraging a user satisfaction model in order to ask a clarifying question when it would be helpful. The proposed model utilizing self-attention for hypothesis representations and ambiguity related contextual information has showed significantly improved performance compared to various baseline approaches evaluated on the user log data from a conversational agent system.

As future work, we will study how logged clarifying interactions can be utilized for the fine-tuning to further improve user satisfaction with clarification as briefly described in Section 2.3. Also, we will look at how the clarifying questions should be composed for each ambiguity type for effective and natural user engagement in the large-scale setting.

6. REFERENCES

[1] Gokhan Tur and Renato de Mori, *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*, New York, NY: John Wiley and Sons, 2011.

[2] Young-Bum Kim, Dongchan Kim, Joo-Kyung Kim, and Ruhi Sarikaya, "A scalable neural shortlisting-reranking approach for large-scale domain classification in natural language understanding," in *NAACL*, 2018, pp. 16–24.

[3] Joo-Kyung Kim and Young-Bum Kim, "Pseudo labeling and negative feedback learning for large-scale multi-label domain classification," in *ICASSP*, 2020, pp. 7964–7968.

[4] Kazunori Komatani and Tatsuya Kawahara, "Flexible mixed-initiative dialogue management using concept-level confidence measures of speech recognizer output," in *COLING*, 2000, pp. 467–473.

[5] Malte Gabsdil, "Clarification in spoken dialog systems," in *AAAI Workshop on Natural Language Generation in Spoken and Written Dialogue*, 2003, pp. 28–35.

[6] Gabriel Skantze, "GALATEA: A discourse modeller supporting concept-level error handling in spoken dialogue systems," in *SIGdial Workshop on Discourse and Dialogue*, 2005, pp. 178–189.

[7] Dan Bohus and Alexander I. Rudnicky, "A principled approach for rejection threshold optimization in spoken dialog systems," in *Interspeech*, 2005, pp. 2781–2784.

[8] Necip Fazil Ayan, Arindam Mandal, Michael Frandsen, Jing Zheng, Peter Blasco, Andreas Kathol, Frederic Bechet, Benoit Favre, Alex Marin, Tom Kwiatkowski, Mari Ostendorf, Luke Zettlemoyer, Philipp Salletmayr, Julia Hirschberg, and Svetlana Stoyanchev, "'Can you give me another word for hyperbaric?': Improving speech translation using targeted clarification questions," in *ICASSP*, 2013, pp. 8391–8395.

[9] Svetlana Stoyanchev and Michael Johnston, "Localized error detection for targeted clarification in a virtual assistant," in *ICASSP*, 2015, pp. 5241–5245.

[10] Ingrid Zukerman, Su Nam Kim, Thomas Kleinbauer, and Masud Moshtaghi, "Employing distance-based semantics to interpret spoken referring expressions," *Computer Speech and Language*, vol. 34, pp. 154–185, 2015.

[11] Ryuichi Takanobu, Qi Zhu1, Jinchao Li, Baolin Peng, Jianfeng Gao, and Minlie Huang, "Is your goal-oriented dialog model performing really well? Empirical analysis of system-wise evaluation," in *SIGDIAL*, 2020, pp. 297–310.

[12] Svetlana Stoyanchev, Alex Liu, and Julia Hirschberg, "Clarification questions with feedback," in *Interdisciplinary Workshop on Feedback Behaviors in Dialog*, 2012, pp. 73–76.

[13] Alex Liu, Rose Sloan, Mei-Vern Then, Svetlana Stoyanchev, Julia Hirschberg, and Elizabeth Shriberg, "Detecting inappropriate clarification requests in spoken dialogue systems," in *SIGDIAL*, 2014, pp. 238–242.

[14] Svetlana Stoyanchev, Alex Liu, and Julia Hirschberg, "Towards natural clarification questions in dialogue systems," in *AISB*, 2014.

[15] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen, "Toward voice query clarification," in *SIGIR*, 2018, pp. 1257–1260.

[16] Sudha Rao and Hal Daumé III, "Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information," in *ACL*, 2018, pp. 2737–2746.

[17] Jingjing Xu, Yuechen Wang, Duyu Tang, Nan Duan, Pengcheng Yang, Qi Zeng, Ming Zhou, and Xu Sun, "Asking clarification questions in knowledge-based question answering," in *EMNLP*, 2019, pp. 1618–1629.

[18] Vaibhav Kumar and Alan W Black, "ClarQ: A large-scale and diverse dataset for clarification question generation," in *ACL*, July 2020, pp. 7296–7301.

[19] Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer, "AmbigQA: Answering ambiguous open-domain questions," in *EMNLP*, 2020, pp. 5783–5797.

[20] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft, "Asking clarifying questions in open-domain information-seeking conversations," in *SIGIR*, 2019, p. 475–484.

[21] Hamed Zamani, Gord Lueck, Everest Chen, Rodolfo Quispe, Flint Luu, and Nick Craswell, "MIMICS: A large-scale data collection for search clarification," in *CIKM*, 2020.

[22] Aishwarya Padmakumar and Raymond J. Mooney, "Dialog policy learning for joint clarification and active learning queries," in *AAAI*, 2021.

[23] Jean-Philippe Robichaud, Paul A. Crook, Puyang Xu, Omar Zia Khan, and Ruhi Sarikaya, "Hypotheses ranking for robust domain classification and tracking in dialogue systems," in *Interspeech*, 2014, pp. 145–149.

[24] Omar Zia Khan, Jean-Philippe Robichaud, Paul A. Crook, and Ruhi Sarikaya, "Hypotheses ranking and state tracking for a multi-domain dialog system using multiple asr alternates," in *Interspeech*, 2015.

[25] Anjishnu Kumar, Arpit Gupta, Julian Chan, Sam Tucker, Bjorn Hoffmeister, Markus Dreyer, Stanislav Peshterliev, Ankur Gandhe, Denis Filiminov, Ariya Rastrow, Christian Monson, and Agnika Kumar, "Just ASK: Building an Architecture for Extensible Self-Service Spoken Language Understanding," in *NIPS Workshop on Conversational AI*, 2017.

[26] Young-Bum Kim, Dongchan Kim, Anjishnu Kumar, and Ruhi Sarikaya, "Efficient large-scale neural domain classification with personalized attention," in *ACL*, 2018, pp. 2214–2224.

[27] Joo-Kyung Kim and Young-Bum Kim, "Supervised domain enablement attention for personalized domain classification," in *EMNLP*, 2018, pp. 894–899.

[28] Chikara Hashimoto and Manabu Sassano, "Detecting absurd conversations from intelligent assistant logs by exploiting user feedback utterances," in *WWW*, 2018, pp. 147–156.

[29] Braden Hancock, Antoine Bordes, Pierre-Emmanuel Mazaré, and Jason Weston, "Learning from dialogue after deployment: Feed yourself, chatbot!," in *ACL*, 2019, pp. 3667–3684.

[30] Dookun Park, Hao Yuan, Dongmin Kim, Yinglei Zhang, Spyros Matsoukas, Young-Bum Kim, Ruhi Sarikaya, Edward Guo, Yuan Ling, Kevin Quinn, Pham Hung, Benjamin Yao, and Sungjin Lee, "Large-scale hybrid approach for predicting user satisfaction with conversational agents," in *NeurIPS Workshop on Human in the Loop Dialogue Systems*, 2020.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukas Kaiser, and Illia Polosukhin, "Attention is all you need," in *NIPS*, 2017, pp. 5998–6008.

[32] Jeffrey Pennington, Richard Socher, and Christopher D Manning, "GloVe: Global vectors for word representation," in *EMNLP*, 2014, pp. 1532–1543.

[33] Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh, "Set transformer: A framework for attention-based permutation-invariant neural networks," in *ICML*, 2019, pp. 3744–3753.

[34] Diederik P. Kingma and Jimmy Lei Ba, "ADAM: A method for stochastic optimization," in *ICLR*, 2015.