System Initiative Prediction for Multi-turn Conversational Information Seeking

Meng, C.; Aliannejadi, M.; de Rijke, M.

DOI
10.1145/3583780.3615070

Publication date
2023

Document Version
Final published version

Published in
CIKM '23

System Initiative Prediction for Multi-turn Conversational Information Seeking

Chuan Meng
University of Amsterdam
Amsterdam, The Netherlands
c.meng@uva.nl

Mohammad Aliannejadi
University of Amsterdam
Amsterdam, The Netherlands
m.aliannejadi@uva.nl

Maarten de Rijke
University of Amsterdam
Amsterdam, The Netherlands
m.derijke@uva.nl

ABSTRACT

Identifying the right moment for a system to take the initiative is essential to conversational information seeking (CIS). Existing studies have extensively studied the clarification need prediction task, i.e., predicting when to ask a clarifying question, however, it only covers one specific system-initiative action. We define the *system initiative prediction* (SIP) task as *predicting whether a CIS system should take the initiative at the next turn*. Our analysis reveals that for effective modeling of SIP, it is crucial to capture dependencies between adjacent user–system initiative-taking decisions. We propose to model SIP by CRFs. Due to their graphical nature, CRFs are effective in capturing such dependencies and have greater transparency than more complex methods, e.g., LLMs. Applying CRFs to SIP comes with two challenges: (i) CRFs need to be given the unobservable system utterance at the next turn, and (ii) they do not explicitly model multi-turn features. We model SIP as an *input-incomplete* sequence labeling problem and propose a *multi-turn system initiative predictor* (MuSIc) that has (i) *prior-posterior inter-utterance encoders* to eliminate the need to be given the unobservable system utterance, and (ii) a *multi-turn feature-aware CRF layer* to incorporate multi-turn features into the dependencies between adjacent initiative-taking decisions. Experiments show that MuSIc outperforms LLM-based baselines including LLaMA, achieving state-of-the-art results on SIP. We also show the benefits of SIP on clarification need prediction and action prediction.

CCS CONCEPTS

• **Information systems** → Retrieval tasks and goals; *Users and interactive retrieval.*

KEYWORDS

Conversational information seeking; System initiative prediction; Mixed initiative

ACM Reference Format:
Chuan Meng, Mohammad Aliannejadi, and Maarten de Rijke. 2023. System Initiative Prediction for Multi-turn Conversational Information Seeking. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management (CIKM '23), October 21–25, 2023, Birmingham, United Kingdom.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3583780.3615070

CIKM '23, October 21–25, 2023, Birmingham, United Kingdom

ACM ISBN 979-8-4007-0124-5/23/10.
https://doi.org/10.1145/3583780.3615070

1 INTRODUCTION

An essential part of conversational information seeking (CIS) is to identify the right moment for a CIS system to take the initiative [6, 72], given that system initiative-taking risks frustrating the user and hurting the user experience [64, 65, 72, 75, 76]. Various system-initiative actions can be taken by a CIS system to take the initiative, e.g., asking a clarifying question or requesting feedback [58, 60]. Existing work has extensively studied the clarification need prediction task, that is, predicting when to ask a clarifying question in an information-seeking conversation [2, 3, 5, 64, 65, 68]. However, as shown in Fig. 1, asking a clarifying question is only one of several possible system-initiative actions [1, 7, 72].

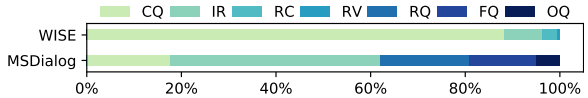


Figure 1: Distribution of system-initiative actions in two realistic CIS training datasets, WISE and MSDialog. CQ: clarifying question (called *clarify* in WISE); IR: information request (called *request* in WISE); RV: revise; RC: recommendation (ask users if they would like something); OQ: original question; RQ: repeat question; and FQ: follow up question.

Task and motivation. We define *system initiative prediction* (SIP) task, which is to predict whether the CIS system should take the initiative at the next turn in an information-seeking conversation. To the best of our knowledge, no existing studies explicitly model this problem. SIP has three benefits for CIS systems: (i) SIP can improve the controllability of the overall initiative level of the system to balance utility and user experience [53]. (ii) SIP can enable knowledge sharing among various system-initiative actions; the shared knowledge learned through SIP can be transferred to improve the prediction of a specific system-initiative action by transfer learning, e.g., by fine-tuning a model, pre-trained on SIP, on clarification need prediction. And (iii) SIP is a high-level decision, and downstream tasks, such as action prediction, depend on SIP; SIP can boost the prediction performance on downstream tasks by reducing the decision space; e.g., on action prediction, the action *requesting feedback* is performed only if the SIP result is initiative. One could argue that existing action prediction methods [70] are sufficiently effective for SIP. However, our experiments show that using action prediction methods for SIP leads to *suboptimal* results, but conversely, SIP significantly improves downstream action prediction.

Our empirical analysis of two CIS datasets [43, 47] reveals that a system's initiative-taking decision at the next turn is not isolated but depends on the user's previous initiative-taking decision. Fig. 2a shows that the system is more likely to take the initiative

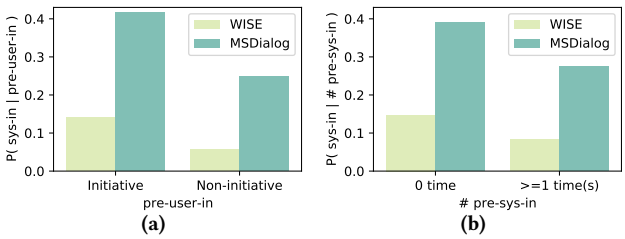


Figure 2: The probability of system initiative-taking (sys-in) conditioned on the user's initiative-taking decision at the preceding turn (pre-user-in) and the number of times the system has taken the initiative (# pre-sys-in) on the WISE and MSDialog training sets.

immediately after the user has taken the initiative in a conversation; thus, capturing the *dependencies between adjacent user–system initiative-taking decisions* is critical for modeling SIP.

A natural way to capture such structural dependencies is to use probabilistic graphical models, such as conditional random fields (CRFs) [32]. We propose to use linear-chain CRFs [32, 56] to model SIP for three reasons: (i) they have been shown to be effective in capturing dependencies between adjacent output decisions [56]; (ii) linear-chain CRFs for SIP can guarantee the best initiative-taking decision at the next turn by decoding the optimal sequence of initiative-taking decisions in context ($1:T-1$ in Fig. 3a) and the next turn (T in Fig. 3a), instead of outputting the decision at the next turn independently [32, 56]; and (iii) due to CRFs' graphical nature, they have been shown to exhibit better interpretability and transparency than other methods [20, 30], such as emergent large language models (LLMs) [14, 57, 74].

Challenges. When adopting linear-chain CRFs to the SIP task we face two challenges: (i) They cannot be directly applied to SIP because we face an *input-incomplete* sequence labeling problem. Linear-chain CRFs are designed for sequence labeling problems that have a one-to-one correspondence between input observations and output decisions. As shown in Fig. 3a, to output initiative-taking decisions in context and at the next turn, linear-chain CRFs need to be given a complete input sequence of utterances in context and at the next turn. However, given the nature of SIP, as shown in Fig. 3b, the system utterance at the next turn is unobservable, leading to an *input-incomplete sequence labeling* problem. And (ii) linear-chain CRFs do not explicitly model *multi-turn features*. Our empirical analysis shows that an initiative-taking decision depends on multi-turn features. We define a multi-turn feature as a variable that varies across turns. Consider, e.g., *the number of times the system has taken the initiative*; Fig. 2b shows that a system is much less likely to take the initiative once again if it has already taken the initiative before. But linear-chain CRFs do not consider this feature as it is beyond the dependency between adjacent initiative decisions.

Approach. To address the challenges, we cast SIP as an *input-incomplete* sequence labeling problem and propose a *multi-turn system initiative predictor* (MuSIc). We propose (i) *prior-posterior inter-utterance encoders* to adapt linear-chain CRFs to the *input-incomplete* sequence labeling problem and eliminate the need to be given the unobservable system utterance, and (ii) a *multi-turn feature-aware conditional random field* (CRF) layer to explicitly capture *the impact of multi-turn features on an initiative-taking decision* by

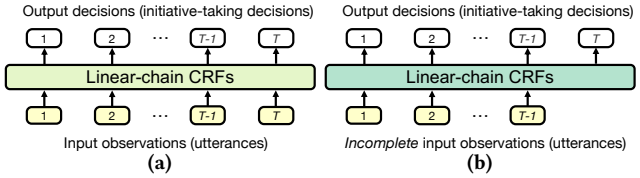


Figure 3: A comparison between a sequence labeling problem (a) and an *input-incomplete* sequence labeling problem (b). $1:T$ denote turn numbers and T is the next system turn.

conditioning the dependencies between adjacent initiative-taking decisions on multi-turn features. MuSIc can use an arbitrary number of multi-turn features; we consider three essential ones: (i) role transition direction, (ii) the number of times the system has taken the initiative, and (iii) the distance to the last system initiative turn.

Experiments. We annotate the initiative-taking decision at each turn on two multi-turn CIS datasets, WISE [47] and MSDialog [42]. Experiments on both datasets show that MuSIc achieves state-of-the-art performance on SIP, outperforming strong clarification need prediction, action prediction, and LLM-based (LLaMA [57]) baselines. We get two more insights: (i) LLMs do not show promising performance on SIP where scaling up LLMs is not an effective way to solve SIP; and (ii) probabilistic graphical modeling is still competitive and effective for this task and it should not be ignored in the era of LLMs. Furthermore, a visual analysis indicates that the transition matrices learned through the MuSIc exhibit meaningful transition patterns and explicitly show how MuSIc models the dependencies, showing great interpretability and transparency. Moreover, we fine-tune MuSIc pre-trained on SIP on the clarification need prediction task, achieving the state-of-the-art clarification need prediction performance on ClariQ [2, 3], indicating that the knowledge shared among various system-initiative actions learned through SIP can be used to improve the prediction of a specific system-initiative action. Finally, we construct a SIP-aware action prediction framework where action prediction is fed with SIP results returned by MuSIc. The action prediction performance is significantly improved, indicating the effectiveness of SIP in benefiting downstream tasks.

Contributions. Our main contributions are as follows:

- We introduce the task of *system initiative prediction* (SIP) for CIS, which has not been explicitly modeled in prior work.
- We propose a *multi-turn system-initiative predictor* (MuSIc), which formalizes SIP as an *input-incomplete* sequence labeling problem and jointly considers *dependencies between adjacent user–system initiative-taking decisions* and *the impact of multi-turn features on an initiative-taking decision.*
- We conduct experiments on two multi-turn CIS datasets, showing state-of-the-art performance of MuSIc on SIP.
- We fine-tune MuSIc pre-trained on SIP on the clarification need prediction task, achieving state-of-the-art clarification need prediction performance on the ClariQ dataset.
- We propose a SIP-aware action prediction framework, showing the effectiveness of MuSIc in downstream action prediction.

2 RELATED WORK

2.1 Conversational information seeking

We focus on modeling mixed-initiative CIS systems [16, 21, 39–41, 72]. Mixed initiative is a key aspect in CIS [45]: the user and system

can both take the initiative at different times in a conversation. Mixed-initiative CIS systems can ask clarifying questions [3, 4, 49, 71], elicit user preferences [44, 50], ask for feedback [58, 60], initiate a conversation [62] and so on. Existing work focuses on when a CIS system should take the initiative [6] and response generation/selection given a decided system-initiative action [4, 12, 49, 66, 71]. We focus on the former. In this direction, Avula et al. [6] run a user study to investigate it. Besides, much work has studied the prediction of when to perform a specific system-initiative action, asking a clarifying question (a.k.a. clarification need prediction) [2, 3, 5, 64, 65, 68].

Clarification need prediction. Zou et al. [75, 76] show that asking a clarifying question is not always necessary, and inappropriate requests for clarification can hurt user experience. Xu et al. [68] propose a binary classification model to identify whether clarification is needed given the conversational context. Aliannejadi et al. [2, 3] fine-tune pre-trained language models fed with user queries to return a clarification need score. Wang and Ai [64, 65] propose a binary classification model that further takes into account clarifying question and answer candidates returned by retrieval models. Arabzadeh et al. [5] utilize the coherency of items retrieved for the user query: the more coherent the retrieved items are, the less ambiguous the query is, and the need for clarification decreases.

Our work differs from these studies, as SIP covers a broader range of system-initiative actions, while these studies are limited to one initiative type (i.e., asking a clarifying question).

System action prediction. Radlinski and Craswell [45] define a system action space and emphasize the need for system action prediction in CIS, i.e., a CIS system should predict an appropriate action from an action space at the right time. Azzopardi et al. [8] define a more detailed taxonomy of user/system actions in CIS. Schneider et al. [48] conduct user study to reveal action flow patterns in CIS. Ghosh et al. [22] first identify the user action used in the previous user utterance and then use that to benefit the system action prediction. In this paper, we are concerned with the more challenging multi-action system action prediction task, i.e., the system performs multiple actions concurrently per turn [73]. Beyond CIS, multi-action system action prediction has been well studied for task-oriented conversations, where it is typically formulated as a multi-label classification [25, 34, 67] or sequence generation problem [28, 34, 52, 63]. Ye et al. [70] propose a sequence generation-based method, called Co-Gen, achieving leading performance in terms of response generation and action prediction.

Our work differs from action prediction because SIP is a higher-level decision on which the action prediction depends.

2.2 Linear-chain conditional random fields

Linear-chain CRFs are discriminative probabilistic graphical models for sequence labeling problems that assign output decisions to all of the observations in a sequence jointly [32]. The output decisions are arranged in a sequence/linear chain where adjacent output decisions are dependent according to the first-order Markov assumption, enabling linear-chain CRFs to effectively capture dependencies between adjacent output decisions [56]. We focus on neural linear-chain CRFs [26, 27], where parameters can be trained end-to-end. They have been widely used for sequence labeling tasks, e.g., POS tagging [26], named entity recognition [26, 33] and dialogue act recognition [13, 17, 31, 46, 51].

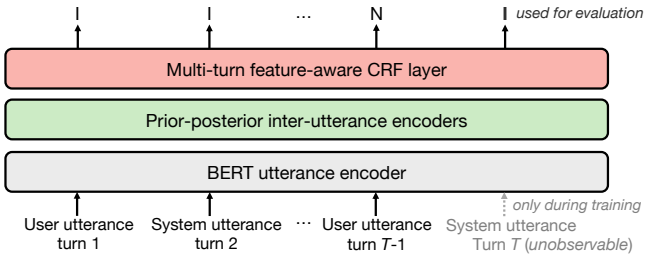


Figure 4: Overview of MuSIc. Its target is to predict the optimal sequence of initiative-taking decisions in the context $1:T-1$ and at the next turn T given the utterances over turns $1:T-1$. I/N at the top denotes *Initiative*/*Non-initiative*.

None of the work listed above can be directly applied to SIP due to the *input-incomplete* sequence labeling problem. Another line of research captures the dependencies between adjacent output decisions by dynamically generating transition matrices [24, 27, 54, 55]. MuSIc differs as it explicitly incorporates multi-turn features into the adjacent dependencies. While some work [11, 51] injects features (e.g., emotion shifts) into the adjacent dependencies for sequence labeling, MuSIc is for *input-incomplete* sequence labeling and considers CIS-specific features that have not been studied yet.

3 TASK DEFINITION

Suppose that we have an information-seeking conversation $X = (x_1, x_2, \ldots, x_{|X|-1}, x_{|X|})$ with a sequence of $|X|$ utterances, where x is an utterance uttered by either a user or system. The conversation X comes with a sequence of ground-truth initiative-taking decisions $Y = (y_1, y_2, \ldots, y_{|X|-1}, y_{|X|})$, i.e., each utterance x in the conversation has a corresponding initiative-taking decision $y \in \{\textit{Initiative}, \textit{Non-initiative}\}$. Given the context $X_{1:T-1} = (x_1, x_2, \ldots, x_{T-1})$, where $T-1$ is a user turn, the *system initiative prediction* (SIP) task is to predict the system's initiative-taking decision y_T at the next turn T. We formulate SIP as an *input-incomplete* sequence labeling problem: we model the conditional probability $P(Y_{1:T} \mid X_{1:T-1})$ of the sequence of initiative-taking decisions in the context $Y_{1:T-1}$ and at the next turn y_T given the sequence $X_{1:T-1}$ of utterances in the context. Only the system's initiative-taking decision y_T at the next turn T is used for evaluation.

4 METHOD

4.1 Limitations of linear-chain CRFs

Linear-chain CRFs predict a sequence of output decisions based on emission and transition scores (see [26, 33] for details), and have two main limitations when applied "as is" to SIP: (i) They model $P(Y_{1:T} \mid X_{1:T})$ to output the sequence $Y_{1:T}$: they use the sequence $X_{1:T}$ of utterances in the context and at the next turn to calculate emissions scores over $\{\textit{Initiative}, \textit{Non-initiative}\}$ over turns $1:T$; there is a one-to-one correspondence between $X_{1:T}$ and emission scores over turns $1:T$. However, x_T, the utterance at the next turn, is unobservable for SIP (see Fig. 3b), leading to the absence of the emission scores at turn T. (ii) They use a transition matrix that contains transition scores from one initiative-taking decision to itself (e.g., *Initiative* to *Initiative*) or the other (e.g., *Initiative* to

Non-initiative) to capture dependencies between adjacent initiative–taking decisions. An initiative-taking decision y_{t+1} is also impacted by a multi-turn feature $s_{t:t+1}$ that changes across turns, e.g., *the number of times the system has taken the initiative* (see Fig. 2b). However, the transition matrix is unique and shared across all turns; thus, the transition scores cannot be adjusted across turns to capture the impact of a multi-turn feature $s_{t:t+1}$ effectively.

4.2 Overview of MuSIc

We propose MuSIc for SIP, which consists of three parts: (i) a *BERT utterance encoder*, (ii) *prior-posterior inter-utterance encoders*, and (iii) a *multi-turn feature-aware CRF layer*. See Fig. 4. The *BERT utterance encoder* is used to encode each utterance into a latent representation. *Prior-posterior inter-utterance encoders* enable MuSIc to model the *input-incomplete* sequence labeling by approximating the absent emission scores at turn T. We model $P(Y_{1:T} \mid X_{1:T})$ during training (see Fig. 5a) as we can access the unobservable system utterance x_T at the next turn T; we pass $X_{1:T}$ through the BERT encoder and a posterior inter-utterance encoder to calculate emission scores over turns $1:T$; we define them as posterior emission scores. Similarly, we pass $X_{1:T-1}$ through BERT and a prior inter-utterance encoder; we use the output of the prior inter-utterance encoder to calculate prior emission scores that are forced to approximate the posterior emission scores at T via an MSE loss. During inference (see Fig. 5b), we model $P(Y_{1:T} \mid X_{1:T-1})$ and regard the approximate (prior) emission scores as the absent emission scores at turn T, eliminating the need to be given the unobservable system utterance x_T. The *multi-turn feature-aware CRF layer* incorporates three multi-turn features and conditions transition scores (dependencies) between adjacent initiative-taking decisions on multi-turn features. We extend the single transition matrix in linear-chain CRFs to multiple ones, corresponding to different multi-turn features. For a pair of adjacent initiative-taking decisions between turn t and $t+1$, we adjust the transition score between them by selecting the transition matrix corresponding to the multi-turn features from turn t to $t+1$.

4.2.1 BERT utterance encoding. We use a BERT encoder [18] to encode an utterance x_t ($t = 1, \ldots, T$ during training, $t = 1, \ldots, T-1$ during inference) into an utterance representation $\mathbf{H}^{x_t} \in \mathbb{R}^{|x_t| \times d}$, after which an average pooling operation [10] is used to get a condensed representation $\mathbf{h}^{x_t} \in \mathbb{R}^{1 \times d}$, where $|x_t|$ and d denote the number of tokens in x_t and the hidden size, respectively.

4.2.2 Prior-posterior inter-utterance encoding. [1] We have a *prior encoder* fed with $\{\mathbf{h}^{x_t}\}_{t=1}^{T-1}$, returning prior utterance representations $\{\mathbf{h}_{pri}^{x_t}\}_{t=1}^{T-1}$, as shown in Fig. 5. Also, we have a *posterior encoder* fed with $\{\mathbf{h}^{x_t}\}_{t=1}^{T}$ ($\{\mathbf{h}^{x_t}\}_{t=1}^{T-1}$ during inference), outputting posterior utterance representations $\{\mathbf{h}_{pos}^{x_t}\}_{t=1}^{T}$ ($\{\mathbf{h}_{pos}^{x_t}\}_{t=1}^{T-1}$ during inference).

4.2.3 Multi-turn feature-aware CRF layer. During training, we feed the unobservable system utterance x_T to MuSIc and model the conditional probability $P(Y_{1:T} \mid X_{1:T})$ of the sequence $Y_{1:T}$ of initiative-taking decisions in the context and at the next turn given the sequence $X_{1:T}$ of utterances in the context and at the next turn.

We consider three multi-turn features $\mathbf{S} = \{s_{t:t+1}^r, s_{t:t+1}^n, s_{t:t+1}^d\}_{t=1}^{T-1}$ as additional input:

(1) $s_{t:t+1}^r$ represents the *role transition direction* from turn t to $t+1$, i.e., $s_{t:t+1}^r = u2s/s2u$ means that the role transition is from the user to the system/the system to the user from turn t to $t+1$.
(2) Given $s_{t:t+1}^r = u2s$ ("user to system"),[2] $s_{t:t+1}^n$ represents *the number of times the system takes the initiative* before the next system turn at $t+1$. Table 1 shows that the average number of system initiative utterances in a conversation in training sets is less than 1. To make full use of the sparse training data, we only consider the cases $s_{t:t+1}^n = 0$ and > 0, which means that the system has not taken the initiative and has taken the initiative once or more before the next system turn at $t+1$, respectively.
(3) Given $s_{t:t+1}^r = u2s$ (again, "user to system") and $s_{t:t+1}^n > 0$, $s_{t:t+1}^d$ represents *the distance to the last system initiative turn* from the next system turn at $t+1$. Similarly, to make full use of the sparse data, we only consider $s_{t:t+1}^d = 2$ and > 2,[3] which means that the distance to the last system initiative turn from the next system turn at $t+1$ is 2 and more than 2 turns, respectively.

After considering the three multi-turn features, MuSIc models:

$$\begin{aligned} P(Y_{1:T} \mid X_{1:T}, \mathbf{S}) &= \frac{\exp(\psi(X_{1:T}, Y_{1:T}, \mathbf{S}))}{\sum_{\tilde{Y}_{1:T}} \exp(\psi(X_{1:T}, \tilde{Y}_{1:T}, \mathbf{S}))}, \\ \psi(Y_{1:T}, X_{1:T}, \mathbf{S}) &= \sum_{t=1}^{T} e(y_t, X_{1:T}) + \\ &\quad \sum_{t=1}^{T-1} g(y_t, y_{t+1}, s_{t:t+1}^r, s_{t:t+1}^n, s_{t:t+1}^d), \end{aligned} \tag{1}$$

where $\tilde{Y}_{1:T}$ denotes one of all possible sequences of initiative-taking decisions, $e(y_t, X_{1:T})$ is the emission scoring function to calculate the posterior emission scores based on $X_{1:T}$, and $g(y_t, y_{t+1}, s_{t:t+1}^r, s_{t:t+1}^n, s_{t:t+1}^d)$ is the transition score function to calculate the transition scores conditioned on multi-turn features $\mathbf{S}$.

Computing emission scores. $e(y_t, X_{1:T})$ calculates the posterior emission scores $\{\mathbf{e}_{pos}^{x_t}\}_{t=1}^{T}$ based on the posterior utterance representations $\{\mathbf{h}_{pos}^{x_t}\}_{t=1}^{T}$; see Fig. 5a. The calculation at each turn is modeled as:

$$\begin{aligned} e(y_t, X_{1:T}) &= \mathbf{e}_{pos}^{x_t, y_t} \in \mathbb{R}^{1 \times 1} \\ \mathbf{e}_{pos}^{x_t} &= \text{MLP}(\mathbf{h}_{pos}^{x_t}) \in \mathbb{R}^{1 \times 2}, \end{aligned} \tag{2}$$

where $t = 1, 2, \ldots, T$, $\mathbf{e}_{pos}^{x_t} \in \mathbb{R}^{1 \times 2}$ are posterior emission scores over $\{$*Initiative*, *Non-initiative*$\}$, and $\text{MLP}(\cdot)$ denotes a multilayer perceptron (MLP). In parallel, we calculate the prior emission scores $\mathbf{e}_{pri}^{x_{T-1}} \in \mathbb{R}^{1 \times 2}$ based on the last output (at turn $T-1$) of the prior inter-utterance encoder $\mathbf{h}_{pri}^{x_{T-1}}$ (see Fig. 5a):

$$\mathbf{e}_{pri}^{x_{T-1}} = \text{MLP}(\mathbf{h}_{pri}^{x_{T-1}}) \in \mathbb{R}^{1 \times 2}. \tag{3}$$

The prior emission scores $\mathbf{e}_{pri}^{x_{T-1}} \in \mathbb{R}^{1 \times 2}$ would learn to approximate the posterior emission scores $\mathbf{e}_{pos}^{x_T} \in \mathbb{R}^{1 \times 2}$ at turn T (see Fig. 5a and Eq. 8). The parameters of the MLP in Eq. 2 and Eq. 3 are not shared.

Computing transition scores. Linear-chain -CRFs do not condition a transition score on any multi-turn features:

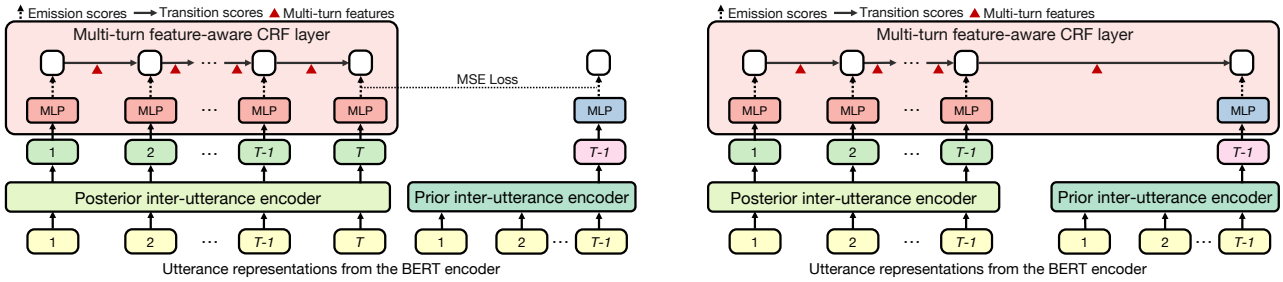
$$g(y_t, y_{t+1}) = G_{y_t, y_{t+1}} \in \mathbb{R}^{1 \times 1}, \tag{4}$$

where $G \in \mathbb{R}^{2 \times 2}$ is a transition matrix shared across all turns, and $G_{y_t, y_{t+1}}$ is the transition score from the decision y_t to y_{t+1}. Our transition scoring function $g(y_t, y_{t+1}, s_{t:t+1}^r, s_{t:t+1}^n, s_{t:t+1}^d)$ does

[1] We implement inter-utterance encoders by BiLSTMs, which got better performance than Transformers in our preliminary experiments.

[2] We consider other features only given "user to system" for simplicity;"user to system" is more critical as the role transition from turn $T-1$ to T is always from "user to system".

[3] We also experimented with more fine-grained cases, such as $s_{t:t+1}^n = 1, 2, 3, 4$ and $s_{t:t+1}^d = 4, 6, 8$, but no further improvements were obtained.



(a) Training (b) Inference

Figure 5: *Prior-posterior inter-utterance encoders* and *multi-turn feature-aware CRF layer* during (a) training and (b) inference. The system utterance at the next turn T can be accessed by the posterior inter-utterance encoder *only* during training.

condition the computation of the transition scores between adjacent initiative-taking decisions on the multi-turn features $s^r_{t:t+1}$, $s^n_{t:t+1}$, and $s^d_{t:t+1}$. We define separate transition matrices corresponding to different combinations of multi-turn features. For a pair of adjacent initiative-taking decisions between turn t and $t+1$, we select the transition matrix corresponding to the multi-turn features from turn t to $t+1$. If the transition score is only conditioned on the multi-turn feature *role transition direction* $s^r_{t:t+1}$, it is calculated as:

$$g(y_t, y_{t+1}, s^r_{t:t+1}) = (1 - I(s^r_{t:t+1})) \cdot \mathbf{G}^{s2u}_{y_t,y_{t+1}} + I(s^r_{t:t+1}) \cdot \mathbf{G}^{u2s}_{y_t,y_{t+1}}, \quad (5)$$

where $I(s^r_{t:t+1})$ is an indicator function that equals 1 if $s^r_{t:t+1} = u2s$ and 0 otherwise, and $\mathbf{G}^{s2u} \in \mathbb{R}^{2\times 2}$ and $\mathbf{G}^{u2s} \in \mathbb{R}^{2\times 2}$ are transition matrices corresponding to "from system to user" and "from user to system," respectively.

Given $s^r_{t:t+1} = u2s$, if the transition score is further conditioned on the feature $s^n_{t:t+1}$, *the number of times the system takes the initiative* before the next system turn at $t+1$, it is calculated as:

$$\begin{aligned} g(y_t, y_{t+1}, s^r_{t:t+1}, s^n_{t:t+1})) = (1 - I(s^r_{t:t+1})) \cdot \mathbf{G}^{s2u}_{y_t,y_{t+1}} + \\ I(s^r_{t:t+1}) \cdot [(1 - I(s^n_{t:t+1})) \cdot \mathbf{G}^{u2s,n=0}_{y_t,y_{t+1}} + I(s^n_{t:t+1}) \cdot \mathbf{G}^{u2s,n>0}_{y_t,y_{t+1}}], \end{aligned} \quad (6)$$

where $I(s^n_{t:t+1})$ is an indicator function that equals 1 if $s^n_{t:t+1} > 0$ and 0 otherwise, and $\mathbf{G}^{u2s,n=0} \in \mathbb{R}^{2\times 2}$ and $\mathbf{G}^{u2s,n>0} \in \mathbb{R}^{2\times 2}$ are transition matrices corresponding to "the system has not take the initiative" and "the system has taken the initiative once or more" before the next system turn at $t+1$, respectively.

Given $s^r_{t:t+1} = u2s$ and $s^n_{t:t+1} > 0$, if the transition score is further conditioned on the feature $s^d_{t:t+1}$, *the distance to the last system's initiative turn* from the next system turn at $t+1$, it is calculated as:

$$\begin{aligned} g(y_t, y_{t+1}, s^r_{t:t+1}, s^n_{t:t+1}, s^d_{t:t+1})) = (1 - I(s^r_{t:t+1})) \cdot \mathbf{G}^{s2u}_{y_t,y_{t+1}} + \\ I(s^r_{t:t+1}) \cdot \{(1 - I(s^n_{t:t+1})) \cdot \mathbf{G}^{u2s,n=0}_{y_t,y_{t+1}} + \\ I(s^n_{t:t+1}) \cdot [(1 - I(s^d_{t:t+1})) \cdot \mathbf{G}^{u2s,n>0,d=2}_{y_t,y_{t+1}} + \\ I(s^d_{t:t+1}) \cdot \mathbf{G}^{u2s,n>0,d>2}_{y_t,y_{t+1}}]\}, \end{aligned} \quad (7)$$

where $I(s^d_{t:t+1})$ is an indicator function that equals 1 if $s^d_{t:t+1} > 2$ and 0 otherwise, and $\mathbf{G}^{u2s,n>0,d=2} \in \mathbb{R}^{2\times 2}$ and $\mathbf{G}^{u2s,n>0,d>2} \in \mathbb{R}^{2\times 2}$ are transition matrices for "the distance to the last system's initiative turn is 2 turns" and "the distance to the last system's initiative turn is more than 2 turns" from the next system turn at $t+1$, respectively.

Training objectives. Our final loss function is defined as $\mathcal{L} = \mathcal{L}_{crf} + \mathcal{L}_{mse}$. We not only minimize the negative log-likelihood of the sequence $Y_{1:T}$ of ground-truth initiative-taking decisions in the context and at the next turn, but also force $\mathbf{e}^{x_{T-1}}_{pri}$ to learn to approximate $\mathbf{e}^{x_T}_{pos}$ via an MSE loss (see Fig. 5a):

$$\begin{aligned} \mathcal{L}_{crf} &= -\log P(Y_{1:T} \mid X_{1:T}, \mathbf{S}) \\ \mathcal{L}_{mse} &= -(\mathbf{e}^{x_{T-1}}_{pri} - \mathbf{e}^{x_T}_{pos})^2. \end{aligned} \quad (8)$$

Inference phase. MuSIc models the conditional probability $P(\tilde{Y}_{1:T} \mid X_{1:T-1}, \mathbf{S})$ of a possible sequence $\tilde{Y}_{1:T}$ of initiative-taking decisions in the context $(1:T-1)$ and at the next turn T only given the sequence $X_{1:T-1}$ of utterances in the context (see Fig. 5b):

$$\begin{aligned} P(\tilde{Y}_{1:T} \mid X_{1:T-1}, \mathbf{S}) &= \frac{\exp(\psi(X_{1:T-1}, \tilde{Y}_{1:T}, \mathbf{S}))}{\sum_{\tilde{Y}_{1:T}} \exp(\psi(X_{1:T-1}, \tilde{Y}_{1:T}, \mathbf{S}))}, \\ \psi(\tilde{Y}_{1:T}, X_{1:T-1}, \mathbf{S}) &= \sum_{t=1}^{T} e(\tilde{y}_t, X_{1:T-1}) + \\ &\quad \sum_{t=1}^{T-1} g(\tilde{y}_t, \tilde{y}_{t+1}, s^r_{t:t+1}, s^n_{t:t+1}, s^d_{t:t+1}), \end{aligned} \quad (9)$$

where $e(\tilde{y}_t, X_{1:T-1}) = \mathbf{e}^{x_{T-1}, \tilde{y}_T}_{pri}$ if $t = T$ and $\mathbf{e}^{x_t, \tilde{y}_t}_{pos}$ otherwise (see Fig. 5b). The optimal sequence $Y^*_{1:T}$ of initiative-taking decisions in context and at the next turn is decoded by the Viterbi algorithm [61]:

$$Y^*_{1:T} = \arg\max_{\tilde{Y}_{1:T}} P(\tilde{Y}_{1:T} \mid X_{1:T-1}, \mathbf{S}). \quad (10)$$

5 EXPERIMENTAL SETUP

Research questions. (**RQ1**) To what extent does MuSIc improve performance on the SIP task compared to state-the-art baselines? (**RQ2**) What is the effect of multi-turn features on the performance of MuSIc? (**RQ3**) To what extent does knowledge shared among various system-initiative actions learned through SIP benefit the clarification need prediction task? (**RQ4**) To what extent does the SIP task benefit the downstream action prediction task?

Datasets. We consider two multi-turn CIS datasets with annotations of actions for utterances, WISE [47] and MSDialog [42, 43, 69]. Based on the action annotations, we annotate the initiative-taking decision for each utterance. WISE is collected through crowdsourcing; it consists of mixed-initiative conversations between two workers playing the role of user and system. All utterances are annotated with actions. We use the data split from [47]. MSDialog consists of mixed-initiative conversations between users who ask for technical help and expert users or staff (i.e., system) who help to solve problems. This dataset has two versions: the complete set and a labeled subset. Each utterance in the labeled subset is annotated with actions; We use the data split of the labeled subset from [43].

Pre-processing. Following [59, 64, 65], we merge consecutive utterances from either the user or system into one utterance by concatenation; their corresponding actions are merged by a union

Table 1: Statistics of the WISE and MSDialog datasets after preprocessing; conv. is short for "conversation."

	WISE			MSDialog		
	train	valid	test	train	valid	test
# conversations	705	200	1,000	1,760	220	219
# utterances	12,184	3,811	18,828	6,305	752	747
# system utterances	5,949	1,868	9,246	2,938	352	354
# system initiatives	691	324	1,457	1,085	131	143
Max. # turns/conversation	38	38	42	10	10	10
Avg. # turns/conversation	17.28	19.06	18.83	3.58	3.42	3.41
Max. # actions/system turn	3	2	3	6	6	7
Avg. # actions/system turn	1.02	1.02	1.02	1.67	1.77	1.80
Avg. # system initiatives/conv.	0.98	1.62	1.46	0.62	0.60	0.65
Avg. # clarifying questions/conv.	0.87	1.23	1.17	0.15	0.18	0.15

operation too. See Table 1 for the statistics of the datasets. The average numbers of turns in both datasets are less than the numbers in the original papers [43, 47] due to the merging operation.

Annotation of initiative-taking decision labels. For both datasets, we derive the initiative annotations by mapping the manual annotations of actions to initiative or non-initiative labels. An utterance is annotated as *initiative* if it is annotated with any of the actions showing initiative[4] and *non-initiative* otherwise.

Baselines. We compare MuSIc with recently proposed LLM-based baselines, and three other groups of state-of-the-art baselines for the SIP task: (i) clarification need prediction, (ii) system action prediction, and (iii) linear-chain CRF-based methods.

We consider **LLaMA-7B/13B/33B/65B** [57] using in-context learning [9, 19] as the LLM-based baselines. Mao et al. [35] prompt LLMs for conversational query rewriting and we adapt their designed prompt to SIP. We prepend the SIP task instruction at the beginning of the prompt, followed by two groups of demonstrations: (i) a few complete conversations randomly sampled from the training set, and (ii) utterances in the context $X_{1:T-1}$ prior to the next turn T. Given the prompt, LLaMA generates the system-initiative decision at the next system turn T. WISE is a Chinese language dataset; however, the original LLaMA has a limited ability to encode and decode Chinese text [15]. Cui et al. [15] release Chinese-LLaMA-Plus-7B and -13B at the time of writing. These LLaMA variants use the extended Chinese vocabulary and are further trained on Chinese data. We report the performance of both [15] on WISE.

We train and test two clarification need prediction models on SIP: (i) **CtxPred (BERT)** uses a BERT encoder to encode the context and predict whether to take the initiative at the next turn [2, 3, 68]. (ii) **Risk-aware Conversational Search agent with Q-learning (RCSQ)** is fed with the context, clarifying question and answer candidates returned by retrievers, and is trained with a user simulator by reinforcement learning [64, 65]. To adapt it to SIP,[5] we replace the clarifying question and answer candidates with initiative and non-initiative system utterance candidates retrieved by bi-encoders;[6] we also replace Q-learning with supervised learning using the annotations of initiative-taking decisions.

We also compare MuSIc with the state-of-art system action prediction method **Co-Gen** [70]. Co-Gen generates actions and responses concurrently — the two generators share a common latent space. We consider two variants of Co-Gen:[7] (i) **Co-Gen (action prediction)** is trained with action and response generation; the model outputs actions based on which we derive initiative-taking decisions using our action-initiative mapping. (ii) **Co-Gen (SIP)** is trained with SIP and response generation; the action generator in the original paper directly learns SIP to output the initiative-taking decision at the next turn.

Linear-chain CRF-based methods cannot be directly applied to SIP as they need to be given the unobservable utterance at the next turn. Based on the same BERT utterance encoder and prior-posterior inter-utterance encoders as in MuSIc, we implement the following: (i) **VanillaCRF** only uses a unique transition matrix (see Eq. 4). (ii) **VanillaCRF+features** feeding the three multi-turn features into the prior-posterior inter-utterance encoders by encoding the multi-turn features as one-hot vectors at each turn and concatenating the vectors with the BERT utterance representation. (iii) **DynamicCRF** uses adjacent input observations x_t, x_{t+1} to generate a dynamic transition matrix $G^{x_t,x_{t+1}}$ to model the dependency between the corresponding output decisions y_t, y_{t+1} [24, 27, 54, 55]. x_T is unseen so G^{x_{T-1},x_T} cannot be computed. Like the calculation of the prior/posterior emissions scores in MuSIc, we use the output of the prior inter-utterance encoder $\mathbf{h}_{pri}^{x_{T-1}}$ to generate a prior transition matrix $G^{x_{T-1}}$ for the output decisions y_{T-1}, y_T; $G^{x_{T-1}}$ approximates a posterior matrix G^{x_{T-1},x_T} generated by the output of the posterior encoder $\mathbf{h}_{pos}^{x_{T-1}}, \mathbf{h}_{pos}^{x_T}$ via an MSE loss.

Evaluation metrics. Because SIP is a binary classification problem, we use macro-averaged F1, precision, recall, and accuracy.

Implementation details. For all models except LLaMA, we use BERT encoders (BERT-base) on all datasets, set the hidden size to 768, batch whole conversations instead of individual turns, set the overall learning rate to 0.00002, use the Adam optimizer [29], and pick the best checkpoint in terms of F1 on the validation set.[8] For LLaMA with all sizes, we randomly sample 2 complete conversations from the training set of WISE/MSDialog as demonstrations since other numbers lead to degraded performance. Note that all methods need to predict initiative-taking decisions for all system turns in all conversations in a dataset. Our code and data resources are available at https://github.com/ChuanMeng/SIP.

6 RESULTS AND ANALYSIS

6.1 Performance comparison

To answer **RQ1**, the results of MuSIc and all baselines on WISE and MSDialog are presented in Table 2. We have five observations.

First, LLaMA-7B/13B gets the worst result on WISE; on MSDialog, LLaMA-13B outperforms CtxPred (BERT), and is comparable to VanillaCRF and DynamicCRF, showing the effectiveness of LLMs. However, LLaMA with a larger parameter size even performs worse

[4] The WISE dataset has different taxonomies for user and system actions; system actions showing initiative have been shown in Fig. 1; user actions showing initiative are *reveal*, *request*, and *revise*. MSDialog has the same taxonomy for user and system actions; actions showing initiative have been shown in Fig. 1. [5] We use the code from the author: https://github.com/zhenduow/conversationalQA [6] We implement the bi-encoders based on BERT, as MuSIc and most of the baselines use BERT.

[7] We use the code released by the author and adapt Co-Gen to SIP by making three changes: (i) we replace the GRU encoder with a BERT encoder like MuSIc has; (ii) Co-Gen requires a state vector (belief state and database records) that does not exist in CIS, so we replace the state vector with one-hot vectors encoding the current multi-turn features; and (iii) we remove reinforcement learning in Co-Gen as the rewards (task completion) do not exist in both CIS datasets. [8] We found that F1 can better show the ability of a model to deal with the class imbalance problem according to experimental results on the WISE and MSDialog validation sets.

Table 2: Performance comparison on SIP. Significant improvements over the best baseline results are marked with * (t-test, $p < 0.05$). The significance test is only performed on accuracy because it gives a score for each individual example, while other metrics evaluate the performance over all examples. Chinese versions of LLaMA-33B/65B are unavailable at the time of writing.

Methods	WISE (%)				MSDialog (%)			
	F1	Precision	Recall	Accuracy	F1	Precision	Recall	Accuracy
LLaMA-7B	46.96	46.69	47.57	75.45	60.22	60.40	60.13	62.15
LLaMA-13B	26.91	55.01	54.28	26.96	62.54	62.73	63.21	62.99
LLaMA-33B	–	–	–	–	58.11	58.24	58.53	58.76
LLaMA-65B	–	–	–	–	55.30	62.33	60.44	55.93
CtxPred (BERT)	68.47	69.66	67.52	84.16	60.17	60.25	60.12	61.86
RCSQ	70.11	71.57	68.96	85.07	63.68	63.86	64.38	64.12
Co-Gen (action prediction)	67.65	69.89	66.14	84.40	53.76	55.23	54.35	58.47
Co-Gen (SIP)	69.47	71.37	68.09	85.01	63.13	63.62	62.97	65.25
VanillaCRF	69.06	71.38	67.46	85.04	62.31	63.24	62.17	64.97
DynamicCRF	69.21	71.25	67.75	84.97	62.01	61.95	62.20	62.99
VanillaCRF+features	69.32	71.85	67.61	85.25	63.29	64.19	63.10	65.82
MuSIc	**71.40**	**73.53**	**69.84**	**85.98***	**65.37**	**65.79**	**65.19**	**67.23***

in most cases, e.g., 7B vs. 13B on WISE and 33B vs. 65B on MSDialog. This problem is also known as *inverse scaling* [36]. McKenzie et al. [36] identify four potential causes of it and highlight that there's still much to uncover in understanding it. Further investigation of this problem on SIP is left for future work.

Second, MuSIc and the linear-chain CRF-based methods outperform CtxPred (BERT). In terms of F1, VanillaCRF outperforms CtxPred (BERT) by 0.59% and 2.14% on WISE and MSDialog, respectively. The gains indicate that it is beneficial for SIP to capture dependencies between adjacent initiative-taking decisions.

Third, both MuSIc and VanillaCRF+features outperform VanillaCRF and DynamicCRF, indicating that it is beneficial for SIP to take into account the impact of multi-turn features on an initiative-taking decision. Also, in terms of F1, MuSIc outperforms VanillaCRF+features by more than 3% on both datasets, underlining the importance of introducing such impact in the CRF layer.

Fourth, Co-Gen (action prediction) performs poorly, indicating that SIP cannot be effectively inferred from the predicted system actions. This could be due to the large action space, making the model prone to action prediction errors, which would propagate to SIP. It also implies the potential of SIP to reduce the decision space of action prediction, which we discuss in response to **RQ4**. Co-Gen (SIP) outperforms Co-Gen (action prediction), suggesting that sharing a common latent space between SIP and response generation is beneficial, however, MuSIc does not use that information.

Fifth, MuSIc outperforms RCSQ, which uses system initiative and non-initiative utterance candidates returned by retrieval models, whereas MuSIc does not have access to such information. MuSIc outperforms RCSQ in terms of F1 by 2.51% and 2.89% on WISE and MSDialog, respectively, confirming the effectiveness of MuSIc.

6.2 Visualisation of transition matrices

We show MuSIc's transition matrices $\mathbf{G}^{s2u}$, $\mathbf{G}^{u2s,n=0}$, $\mathbf{G}^{u2s,n>0,d=2}$ and $\mathbf{G}^{u2s,n>0,d>2}$ on WISE and MSDialog in Fig. 6. We see different patterns in each transition matrix, indicating that different transition patterns are associated with different cases: (i) $\mathbf{G}^{u2s,n=0}$ shows that the user's initiative tends to transition to the system's initiative when the system has not taken the initiative before. This corresponds to cases where the system tends to take the initiative for the first time to ask a clarifying question after the user has asked a question. (ii) $\mathbf{G}^{u2s,n>0,d=2}$ shows that the user's initiative tends to transition to the system's non-initiative if the system has taken the initiative at the last system turn. In other words, the system is less likely to take the initiative in two consecutive system turns if the user takes the initiative in the middle. (iii) According to $\mathbf{G}^{u2s,n>0,d>2}$, we see that compared to $\mathbf{G}^{u2s,n>0,d=2}$, if the system has not taken the initiative at the last system turn, the possibility of system initiative increases, especially when the user takes the initiative (on MSDialog). This corresponds to cases where the system takes the initiative once again to ask for feedback after answering a question from the user. The complexities of the patterns described above indicate that MuSIc effectively captures the impact of multi-turn features on an initiative-taking decision.

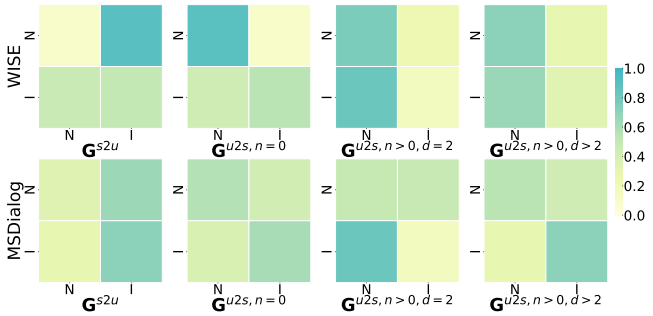


Figure 6: MuSIc's transition matrices learned on WISE and MSDialog. N and I denote *non-initiative* and *initiative*, respectively. See Section 4.2.3 for more information about each transition matrix. Transition scores are normalized across columns. Darker colors indicate higher scores.

6.3 Effect of different multi-turn features

To answer **RQ2**, we evaluate MuSIc with multi-turn features on WISE and MSDialog. We consider four settings: (i) (r, n, d) is our final model considering all features (Eq. 7); (ii) (r, n) does not consider

Table 3: Effect of multi-turn features in MuSIc. Notation for features explained in Section 6.3. * means (r, n, d) is significantly better than (–).

featu.	WISE (%)				MSDialog (%)			
	F1	Prec.	Recall	Acc.	F1	Prec.	Recall	Acc.
r, n, d	**71.40**	**73.53**	**69.84**	**85.98***	**65.37**	**65.79**	**65.19**	**67.23***
r, n	70.71	73.00	69.08	85.75	64.80	64.96	64.70	66.38
r	69.98	72.03	68.49	85.31	62.84	63.10	62.72	64.69
–	69.06	71.38	67.46	85.04	62.31	63.24	62.17	64.97

the distance to the last system's initiative turn (Eq. 6); (iii) (r) does not consider *the number of times the system has taken the initiative* (Eq. 5); (iv) – does not consider any feature, degrading to VanillaCRF (Eq. 4). See Table 3. All proposed multi-turn features contribute to the success of MuSIc. On WISE, the MuSIc performance shows the biggest drop (0.92%) in terms of F1 score after removing role transition direction ((r) vs. –). On MSDialog, MuSIc's F1 score shows the biggest drop (1.96%) after removing the number of times the system has taken the initiative ((r, n) vs. (r)).

6.4 Benefits of SIP on other tasks

We have demonstrated the effectiveness of MuSIc on SIP. Next, we illustrate two applications of SIP.

Improving clarification need prediction via transfer learning. To answer **RQ3**, we examine the benefits of SIP to clarification need prediction (CNP) [2, 3, 5, 64, 65, 68]. We examine whether knowledge shared among system-initiative actions learned through SIP on a dataset (MSDialog) can be reused to improve clarification need prediction on the single-turn ClariQ dataset [2, 3]. We adopt MuSIc and the two strong clarification need prediction baselines CtxPred (BERT) [2, 3, 68] and RCSQ [64, 65] in two settings: (i) a supervised setting (CNP, ClariQ), where we only train models on the ClariQ training dataset, and (ii) a transfer learning setting (SIP, MS. → CNP, ClariQ), where we first get the best checkpoints pre-trained on SIP on the MSDialog training set and then fine-tune them on the ClariQ training dataset. We also introduce MiniLm-ANC [5], an unsupervised learning method for clarification need prediction. We follow [5] to binarize the graded clarification need scores ranging from 1 (no need for clarification) to 4 (clarification is necessary) on ClariQ. Unlike [5], where scores are split in the middle, we only regard score 1 as not asking a clarifying question because the author of ClariQ states that clarification is still needed for scores 2 and 3 but not as much as score 4.[9] We present the results in Table 4.

MuSIc outperforms strong baselines on the single-turn ClariQ dataset in the supervised setting; it outperforms MiniLm-ANC and RCSQ (CNP, ClariQ) that use retrieved documents by 6.88% and 3.07% in terms of F1 score, respectively. Transfer learning from SIP to clarification need prediction benefits MuSIc and the baselines: performance increases with knowledge shared among system-initiative actions acquired from SIP. MuSIc (SIP, MS. → CNP, ClariQ) shows an increase (3.77%) in terms of F1 compared to MuSIc (CNP, ClariQ), significantly exceeding all baselines in the transfer learning setting and achieving state-of-the-art performance on ClariQ.

Because the MSDialog training set contains system utterances of clarifying questions, pre-training on SIP on the MSDialog dataset already includes the pre-training of clarification need prediction. Is the improvement of transfer learning because the model learns knowledge shared among various system-initiative actions on the SIP task or because the model is just augmented with more training examples of clarification need prediction on MSDialog? In order to determine this, we introduce MuSIc (CNP, MS. → CNP, ClariQ), which is only pre-trained on clarification need prediction on the MSDialog training dataset, i.e., pre-trained on the partial SIP training examples containing clarifying questions. The performance of MuSIc (SIP, MS. → CNP, ClariQ) shows an increase (2%) in terms of F1 score compared to the performance of MuSIc (CNP, MS. → CNP, ClariQ), confirming that shared knowledge of various system-initiative actions learned through SIP benefits the model.

Table 4: Performance on clarification need prediction on ClariQ. (CNP, ClariQ) indicates models in the supervised setting, where we only train the models on the ClariQ training dataset; (SIP, MS. → CNP, ClariQ) indicates models in the transfer learning setting, where we further fine-tune the best checkpoints, pre-trained on SIP, on the ClariQ training dataset; MuSIc (CNP, MS. → CNP, ClariQ), pre-trained on the SIP examples only containing clarifying questions on the MSDialog training dataset. Significant improvements over the best baseline results are marked with * (t-test, $p < 0.05$).

Method	ClariQ (%)			
	F1	Prec.	Recall	Acc.
MiniLm-ANC	54.38	54.12	54.95	77.05
CtxPred (CNP, ClariQ)	50.59	50.66	50.59	78.69
RCSQ (CNP, ClariQ)	58.19	58.73	57.78	81.97
MuSIc (CNP, ClariQ)	61.26	64.64	59.67	85.25
CtxPred (SIP, MS. → CNP, ClariQ)	56.84	56.84	56.84	80.33
RCSQ (SIP, MS. → CNP, ClariQ)	61.26	64.64	59.67	85.25
MuSIc (CNP, MS. → CNP, ClariQ)	63.03	69.74	60.61	86.89
MuSIc (SIP, MS. → CNP, ClariQ)	**65.03**	**78.16**	**61.56**	**88.52***

Improving downstream action prediction. To answer **RQ4**, we propose a SIP-aware action prediction framework where action prediction is fed with the initiative-taking decision predicted by MuSIc. In our scenario, the system can take multiple actions per turn. Multi-action system action prediction is typically modeled as multi-label classification [25, 34, 67] or sequence generation [28, 34, 52, 63]. We adopt two typical models for both types and a state-of-art system action prediction method, Co-Gen [70]: (i) following [25, 34, 67], we construct a multi-label classification model by using a BERT encoder to encode the context and feeding the [CLS] token to an MLP followed by sigmoid activation function to perform binary classification for each action; (ii) following [28, 34, 52, 63], we construct a sequence generation model by using BERT to encode the context and feeding the [CLS] token to a GRU decoder to sequentially decode actions step by step; and (iii) Co-Gen is a sequence generation model, and we use Co-Gen (action prediction) (see Section 5) to generate actions. To inject initiative-taking decisions into these models, we first embed an initiative-taking decision (annotated during training and predicted by MuSIc during inference) to a 768-dimensional vector. For the models under (i) and (ii) we concatenate the vector with the [CLS] token and feed the concatenation to an

[9] https://github.com/aliannejadi/ClariQ

Table 5: Performance on the downstream task. Methods used: mlc (multi-label classification), sg (sequence generation), and Co-Gen (state-of-the-art action prediction). + MuSIc: inject the initiative-taking decision predicted by MuSIc; + oracle: inject ground-truth initiative-taking decisions. Significant improvements over results of methods without using SIP results are marked with * (t-test, $p < 0.05$).

Meth.	WISE (%)				MSDialog (%)			
	F1	Prec.	Recall	Acc.	F1	Prec.	Recall	Acc.
mlc	21.59	25.80	20.24	48.78	18.23	20.41	18.06	48.83
+ MuSIc	**23.05**	**25.87**	**22.71**	**51.98***	**19.61**	**24.00**	**18.53**	**50.11***
+ oracle	24.78	27.53	24.82	54.69	21.77	29.08	19.84	56.51
sg	21.92	22.77	23.01	54.28	19.36	21.65	19.31	45.87
+ MuSIc	**23.28**	**25.92**	**24.07**	**56.68***	**21.12**	**22.94**	**21.00**	**49.87***
+ oracle	29.40	29.29	32.09	61.50	27.88	31.43	26.61	53.71
Co-Gen	24.17	24.14	25.77	55.02	21.34	22.98	20.95	48.94
+ MuSIc	**26.26**	**28.95**	**26.86**	**58.54***	**23.38**	**24.39**	**23.08**	**51.76***
+ oracle	30.49	31.49	32.23	62.32	28.37	29.14	28.27	57.47

MLP/GRU decoder. For Co-Gen, we concatenate the vector with the context representation (see [70]).

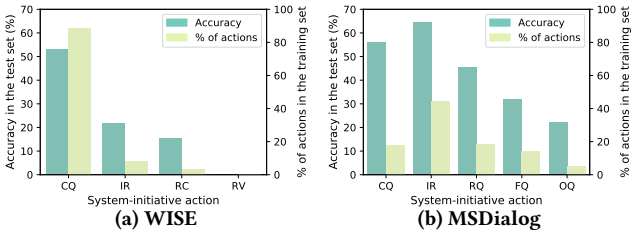
For evaluation, we adopt the same metrics as the previous sections except for accuracy. Accuracy here is measured by the Hamming score (a.k.a. the intersection over the union) [23] that is widely used in multi-label classification evaluation [43]. Table 5 shows the results. The performance of three action prediction models fed with the initiative-taking decision predicted by MuSIc (+ MuSIc) is significantly improved compared to models without using SIP results. We think that this is because SIP, when effective, can reduce the action space of the downstream action prediction models. However, the downstream action prediction model cannot solve the SIP task (see Section 6.1). It shows that action prediction cannot replace SIP, reiterating the effectiveness of SIP in benefiting downstream tasks.

6.5 Error analysis

We conduct an error analysis of SIP. We group system initiative utterances in the test sets of WISE and MSDialog according to their annotated system-initiative actions; utterances in each group share the same system-initiative action. See Fig. 7. MuSIc can still perform well on some system-initiative actions that only take up a limited proportion of the training sets. E.g., on MSDialog, the percentage of CQ is far less than the percentage of IR in the training set, but the performance of MuSIc is comparable in terms of CQ and IR in the test set. SIP enables knowledge sharing among various system-initiative actions, benefiting individual system-initiative actions. For revise (RV), there are only 4 and 3 system utterances of this type in the WISE training and test sets, respectively, numbers that are too small to properly evaluate the performance.

7 CONCLUSIONS AND FUTURE WORK

We have introduced the task of *system initiative prediction* (SIP), which is to predict whether a CIS system should take the initiative at the next turn. We found that it is natural to utilize probabilistic graphical models for SIP but we faced two main challenges: solving the *input-incomplete* sequence labeling problem and explicitly modeling multi-turn features. To solve the challenges, we proposed



(a) WISE **(b) MSDialog**

Figure 7: SIP accuracy over utterance groups (utterances in one group share the same system-initiative action) in the test sets and percentages of system-initiative actions in the training sets. Abbreviations are explained in Figure 1.

MuSIc, which has (i) *prior-posterior inter-utterance encoders* to adapt CRFs to *input-incomplete* sequence labeling by eliminating the need to be given the unobservable system utterance at the next turn, and (ii) a *multi-turn feature-aware CRF layer* to jointly consider *dependencies between adjacent user–system initiative-taking decisions* and *the impact of multi-turn features on an initiative-taking decision.*

Experiments on two CIS datasets show that MuSIc outperforms various baselines including LLMs and achieves state-of-the-art performance on SIP. A visual analysis shows how the learned transition matrices exhibit MuSIc's interpretability and transparency. Transferring knowledge shared among system-initiative actions learned through SIP to the clarification need prediction task greatly benefits it; MuSIc achieves state-of-the-art performance on ClariQ. Lastly, SIP significantly improves the downstream action prediction task by the proposed SIP-aware action prediction framework.

As to MuSIc's limitations and future work, MuSIc does not utilize retrieved documents to improve SIP. Recent research into query performance prediction (QPP) on conversational search [37, 38] has shown that QPP can model retrieved documents and has the potential to help a CIS system take appropriate action at the next turn [37, 38]. We plan to incorporate QPP-based features into our model. Clearly, splitting out SIP as a separate task adds complexity to CIS systems. Pre-training a model on SIP to learn knowledge shared among system-initiative actions and then fine-tuning the model on other tasks does not change the model architecture, but only increases training time without affecting inference time. Our proposed SIP-aware action prediction framework models SIP and action prediction as a two-stage process, which carries additional computational costs at inference time. We plan to improve the efficiency in the future, e.g., by modeling SIP and action prediction jointly in one stage.

ACKNOWLEDGMENTS

We would like to thank our reviewers for their feedback. This research was partially supported by the China Scholarship Council (CSC) under grant number 202106220041, and by the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, https://hybrid-intelligence-centre.nl, and project LESSEN with project number NWA.1389.20.183 of the research program NWA ORC 2020/21, which is (partly) financed by the Dutch Research Council (NWO). All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Mohammad Aliannejadi, Leif Azzopardi, Hamed Zamani, Evangelos Kanoulas, Paul Thomas, and Nick Craswell. 2021. Analysing Mixed Initiatives and Search Strategies during Conversational Search. In *CIKM*. 16–26.

[2] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2020. ConvAI3: Generating Clarifying Questions for Open-Domain Dialogue Systems (ClariQ). *arXiv preprint arXiv:2009.11352* (2020).

[3] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2021. Building and Evaluating Open-Domain Dialogue Corpora with Clarifying Questions. In *EMNLP*. 4473–4484.

[4] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking Clarifying Questions in Open-Domain Information-Seeking Conversations. In *SIGIR*. 475–484.

[5] Negar Arabzadeh, Mahsa Seifikar, and Charles L.A. Clarke. 2022. Unsupervised Question Clarity Prediction Through Retrieved Item Coherency. In *CIKM*. 3811–3816.

[6] Sandeep Avula, Bogeum Choi, and Jaime Arguello. 2023. Why and When: Understanding System Initiative during Conversational Collaborative Search. *arXiv preprint arXiv:2303.13484* (2023).

[7] Leif Azzopardi, Mohammad Aliannejadi, and Evangelos Kanoulas. 2022. Towards Building Economic Models of Conversational Search. In *ECIR*. 31–38.

[8] Leif Azzopardi, Mateusz Dubiel, Martin Halvey, and Jeffery Dalton. 2018. Conceptualizing Agent-human Interactions during the Conversational Search Process. In *CAIR*.

[9] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models are Few-Shot Learners. In *NeurIPS*. 1877–1901.

[10] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sunga, Brian Stropea, and Ray Kurzweil. 2018. Universal Sentence Encoder. *arXiv preprint arXiv:1803.11175* (2018).

[11] Chun-Yu Chen, Yun-Shao Lin, and Chi-Chun Lee. 2022. Emotion-Shift Aware CRF for Decoding Emotion Sequence in Conversation. *Interspeech* (2022), 1148–1152.

[12] Maximillian Chen, Xiao Yu, Weiyan Shi, Urvi Awasthi, and Zhou Yu. to appear. Controllable Mixed-Initiative Dialogue Generation through Prompting. In *ACL*.

[13] Zheqian Chen, Rongqin Yang, Zhou Zhao, Deng Cai, and Xiaofei He. 2018. Dialogue Act Recognition via CRF-Attentive Structured Network. In *SIGIR*. 225–234.

[14] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. PaLM: Scaling Language Modeling with Pathways. *arXiv preprint arXiv:2204.02311* (2022).

[15] Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and Effective Text Encoding for Chinese LLaMA and Alpaca. *arXiv preprint arXiv:2304.08177* (2023).

[16] J. Shane Culpepper, Fernando Diaz, and Mark D. Smucker. 2018. Research Frontiers in Information Retrieval: Report from the Third Strategic Workshop on Information Retrieval in Lorne (SWIRL 2018). In *ACM SIGIR Forum*, Vol. 52. 34–90.

[17] Yang Deng, Wenxuan Zhang, Wai Lam, Hong Cheng, and Helen Meng. 2022. User Satisfaction Estimation with Sequential Dialogue Act Modeling in Goal-oriented Conversational Systems. In *WWW*. 2998–3008.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL*. 4171–4186.

[19] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, and Zhifang Sui. 2023. A Survey on In-context Learning. *arXiv preprint arXiv:2301.00234* (2023).

[20] Yao Fu, Chuanqi Tan, Bin Bi, Mosha Chen, Yansong Feng, and Alexander Rush. 2020. Latent Template Induction with Gumbel-CRFS. *NeurIPS* 33 (2020), 20259–20271.

[21] Jianfeng Gao, Chenyan Xiong, Paul Bennett, and Nick Craswell. 2022. Neural Approaches to Conversational Information Retrieval. *arXiv preprint arXiv:2201.05176* (2022).

[22] Souvick Ghosh, Satanu Ghosh, and Chirag Shah. 2023. Toward Connecting Speech Acts and Search Actions in Conversational Search Tasks. *arXiv preprint arXiv:2305.04858* (2023).

[23] Shantanu Godbole and Sunita Sarawagi. 2004. Discriminative Methods for Multi-labeled Classification. In *PAKDD*. Springer, 22–30.

[24] Ashim Gupta, Pawan Goyal, Sudeshna Sarkar, and Mahanandeeshwar Gattu. 2019. Fully Contextualized Biomedical NER. In *ECIR 2019*. 117–124.

[25] Wanwei He, Yinpei Dai, Yinhe Zheng, Yuchuan Wu, Zheng Cao, Dermot Liu, Peng Jiang, Min Yang, Fei Huang, Luo Si, Jian Sun, and Yongbin Li. 2022. Galaxy: A Generative Pre-trained Model for Task-oriented Dialog with Semi-supervised Learning and Explicit Policy Injection. In *AAAI*, Vol. 36. 10749–10757.

[26] Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF Models for Sequence Tagging. *arXiv preprint arXiv:1508.01991* (2015).

[27] Abhyuday N. Jagannatha and Hong Yu. 2016. Structured Prediction Models for RNN-based Sequence Labeling in Clinical Text. In *EMNLP*. 856–865.

[28] Megha Jhunjhunwala, Caleb Bryant, and Pararth Shah. 2020. Multi-Action Dialog Policy Learning with Interactive Human Teaching. In *SIGdial*. 290–296.

[29] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

[30] Daphne Koller and Nir Friedman. 2009. *Probabilistic Graphical Models: Principles and Techniques*. MIT press.

[31] Harshit Kumar, Arvind Agarwal, Riddhiman Dasgupta, and Sachindra Joshi. 2018. Dialogue Act Sequence Labeling Using Hierarchical Encoder With CRF. In *AAAI*, Vol. 32.

[32] John D. Lafferty, Andrew McCallum, and Fernando C.N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML 2001*. 282–289.

[33] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural Architectures for Named Entity Recognition. In *NAACL*. 260–270.

[34] Ziming Li, Julia Kiseleva, and Maarten de Rijke. 2020. Rethinking Supervised Learning and Reinforcement Learning in Task-Oriented Dialogue Systems. In *Findings of EMNLP*. 3537–3546.

[35] Kelong Mao, Zhicheng Dou, Haonan Chen, Fengran Mo, and Hongjin Qian. 2023. Large Language Models Know Your Contextual Search Intent: A Prompting Framework for Conversational Search. *arXiv preprint arXiv:2303.06573* (2023).

[36] Ian R McKenzie, Alexander Lyzhov, Michael Pieler, Alicia Parrish, Aaron Mueller, Ameya Prabhu, Euan McLean, Aaron Kirtland, Alexis Ross, Alisa Liu, et al. 2023. Inverse Scaling: When Bigger Isn't Better. *arXiv preprint arXiv:2306.09479* (2023).

[37] Chuan Meng, Mohammad Aliannejadi, and Maarten de Rijke. 2023. Performance Prediction for Conversational Search Using Perplexities of Query Rewrites. In *QPP++2023*. 25–28.

[38] Chuan Meng, Negar Arabzadeh, Mohammad Aliannejadi, and Maarten de Rijke. 2023. Query Performance Prediction: From Ad-hoc to Conversational Search. In *SIGIR*. 2583–2593.

[39] Chuan Meng, Pengjie Ren, Zhumin Chen, Christof Monz, Jun Ma, and Maarten de Rijke. 2020. RefNet: A Reference-aware Network for Background Based Conversation. In *AAAI*.

[40] Chuan Meng, Pengjie Ren, Zhumin Chen, Zhaochun Ren, Tengxiao Xi, and Maarten de Rijke. 2021. Initiative-Aware Self-Supervised Learning for Knowledge-Grounded Conversations. In *SIGIR*. 522–532.

[41] Chuan Meng, Pengjie Ren, Zhumin Chen, Weiwei Sun, Zhaochun Ren, Zhaopeng Tu, and Maarten de Rijke. 2020. DukeNet: A Dual Knowledge Interaction Network for Knowledge-Grounded Conversation. In *SIGIR*. 1151–1160.

[42] Chen Qu, Liu Yang, W. Bruce Croft, Johanne R. Trippas, Yongfeng Zhang, and Minghui Qiu. 2018. Analyzing and Characterizing User Intent in Information-seeking Conversations. In *SIGIR*. 989–992.

[43] Chen Qu, Liu Yang, W. Bruce Croft, Yongfeng Zhang, Johanne R. Trippas, and Minghui Qiu. 2019. User Intent Prediction in Information-seeking Conversations. In *CHIIR*. 25–33.

[44] Filip Radlinski, Krisztian Balog, Bill Byrne, and Karthik Krishnamoorthi. 2019. Coached Conversational Preference Elicitation: A Case Study in Understanding Movie Preferences. In *SIGDial*. 353–360.

[45] Filip Radlinski and Nick Craswell. 2017. A Theoretical Framework for Conversational Search. In *CHIIR*. 117–126.

[46] Vipul Raheja and Joel Tetreault. 2019. Dialogue Act Classification with Context-Aware Self-Attention. In *NAACL 2019*. 3727–3733.

[47] Pengjie Ren, Zhongkun Liu, Xiaomeng Song, Hongtao Tian, Zhumin Chen, Zhaochun Ren, and Maarten de Rijke. 2021. Wizard of Search Engine: Access to Information Through Conversations with Search Engines. In *SIGIR*.

[48] Phillip Schneider, Anum Afzal, Juraj Vladika, Daniel Braun, and Florian Matthes. 2023. Investigating Conversational Search Behavior for Domain Exploration. In *ECIR*. 608–616.

[49] Ivan Sekulić, Mohammad Aliannejadi, and Fabio Crestani. 2022. Exploiting Document-Based Features for Clarification in Conversational Search. In *ECIR*. 413–427.

[50] Anna Sepliarskaia, Julia Kiseleva, Filip Radlinski, and Maarten de Rijke. 2018. Preference Elicitation as an Optimization Problem. In *RecSys*. 172–180.

[51] Guokan Shang, Antoine Tixier, Michalis Vazirgiannis, and Jean-Pierre Lorré. 2020. Speaker-change Aware CRF for Dialogue Act Classification. In *COLING*. 450–464.

[52] Lei Shu, Hu Xu, Bing Liu, and Piero Molino. 2019. Modeling Multi-Action Policy for Task-Oriented Dialogues. In *EMNLP*. 1304–1310.

[53] Clemencia Siro, Mohammad Aliannejadi, and Maarten de Rijke. 2022. Understanding User Satisfaction with Task-Oriented Dialogue Systems. In *SIGIR 2022*. 2018–2023.

[54] Yixuan Su, Deng Cai, Yan Wang, David Vandyke, Simon Baker, Piji Li, and Nigel Collier. 2021. Non-Autoregressive Text Generation with Pre-trained Language Models. In *EACL 2021*. 234–243.

[55] Zhiqing Sun, Zhuohan Li, Haoqing Wang, Di He, Zi Lin, and Zhihong Deng. 2019. Fast Structured Decoding for Sequence Models. In *NeurIPS*, Vol. 32.

[56] Charles Sutton and Andrew McCallum. 2012. An Introduction to Conditional Random Fields. *Foundations and Trends in Machine Learning* 4, 4 (2012), 267–373.

[57] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971* (2023).
[58] Johanne R. Trippas, Damiano Spina, Paul Thomas, Mark Sanderson, Hideo Joho, and Lawrence Cavedon. 2020. Towards a Model for Spoken Conversational Search. *Information Processing & Management* 57, 2 (2020), 102162.
[59] Svitlana Vakulenko, Evangelos Kanoulas, and Maarten de Rijke. 2021. A Large-Scale Analysis of Mixed Initiative in Information-Seeking Dialogues for Conversational Search. *ACM Transactions on Information Systems* 39, 4 (2021), 1–32.
[60] Svitlana Vakulenko, Kate Revoredo, Claudio Di Ciccio, and Maarten de Rijke. 2019. QRFA: A Data-Driven Model of Information-Seeking Dialogues. In *ECIR*. 541–557.
[61] Andrew Viterbi. 1967. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory* 13, 2 (1967), 260–269.
[62] Somin Wadhwa and Hamed Zamani. 2021. Towards System-Initiative Conversational Information Seeking. In *DESIRES*. 102–116.
[63] Kai Wang, Junfeng Tian, Rui Wang, Xiaojun Quan, and Jianxing Yu. 2020. Multi-Domain Dialogue Acts and Response Co-Generation. In *ACL*. 7125–7134.
[64] Zhenduo Wang and Qingyao Ai. 2021. Controlling the Risk of Conversational Search via Reinforcement Learning. In *WWW*. 1968–1977.
[65] Zhenduo Wang and Qingyao Ai. 2022. Simulating and Modeling the Risk of Conversational Search. *ACM Transactions on Information Systems* 40, 4 (2022), 1–33.
[66] Zhenduo Wang, Yuancheng Tu, Corby Rosset, Nick Craswell, Ming Wu, and Qingyao Ai. 2023. Zero-Shot Clarifying Question Generation for Conversational Search. In *WWW*. 3288–3298.
[67] Chien-Sheng Wu, Steven C.H. Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained Natural Language Understanding for Task-Oriented Dialogue. In *EMNLP*. 917–929.
[68] Jingjing Xu, Yuechen Wang, Duyu Tang, Nan Duan, Pengcheng Yang, Qi Zeng, Ming Zhou, and Xu Sun. 2019. Asking Clarification Questions in Knowledge-Based Question Answering. In *EMNLP*. 1618–1629.
[69] Liu Yang, Minghui Qiu, Chen Qu, Jiafeng Guo, Yongfeng Zhang, W Bruce Croft, Jun Huang, and Haiqing Chen. 2018. Response Ranking with Deep Matching Networks and External Knowledge in Information-seeking Conversation Systems. In *SIGIR*. 245–254.
[70] Chenchen Ye, Lizi Liao, Fuli Feng, Wei Ji, and Tat-Seng Chua. 2022. Structured and Natural Responses Co-generation for Conversational Search. In *SIGIR*. 155–164.
[71] Hamed Zamani, Susan T. Dumais, Nick Craswell, Paul N. Bennett, and Gord Lueck. 2020. Generating Clarifying Questions for Information Retrieval. In *WWW*. 418–428.
[72] Hamed Zamani, Johanne R. Trippas, Jeff Dalton, and Filip Radlinski. 2022. Conversational Information Seeking. *arXiv preprint arXiv:2201.08808* (2022).
[73] Shuo Zhang, Junzhou Zhao, Pinghui Wang, Yu Li, Yi Huang, and Junlan Feng. 2022. "Think Before You Speak": Improving Multi-Action Dialog Policy by Planning Single-Action Dialogs. In *IJCAI*. 4510–4516.
[74] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A Survey of Large Language Models. *arXiv preprint arXiv:2303.18223* (2023).
[75] Jie Zou, Mohammad Aliannejadi, Evangelos Kanoulas, Maria Soledad Pera, and Yiqun Liu. 2022. Users Meet Clarifying Questions: Toward a Better Understanding of User Interactions for Search Clarification. *ACM Transactions on Information Systems* 41, 1 (2022), Article 16.
[76] Jie Zou, Aixin Sun, Cheng Long, Mohammad Aliannejadi, and Evangelos Kanoulas. 2023. Asking Clarifying Questions: To benefit or to disturb users in Web search? *Information Processing & Management* 60, 2 (2023), 103176.