A Dependency-Aware Utterances Permutation Strategy to Improve Conversational Evaluation

Guglielmo Faggioli[1][0000−0002−5070−2049], Marco Ferrante[1][0000−0002−0894−4175], Nicola Ferro[1][0000−0001−9219−6239], Raffaele Perego[2][0000−0001−7189−4724], and Nicola Tonellotto[3][0000−0002−7427−1001]

[1] University of Padova, Padova, Italy
[2] ISTI-CNR, Pisa, Italy
[3] University of Pisa, Pisa, Italy

Abstract. The rapid growth in the number and complexity of conversational agents has highlighted the need for suitable evaluation tools to describe their performance. The main evaluation paradigms move from analyzing conversations where the user explores information needs following a scripted dialogue with the agent. We argue that this is not a realistic setting: different users ask different questions (and in a diverse order), obtaining distinct answers and changing the conversation path. We analyze what happens to conversational systems performance when we change the order of the utterances in a scripted conversation while respecting temporal dependencies between them. Our results highlight that the performance of the system widely varies. Our experiments show that diverse orders of utterances determine completely different rankings of systems by performance. The current way of evaluating conversational systems is thus biased. Motivated by these observations, we propose a new evaluation approach based on dependency-aware utterance permutations to increase the power of our evaluation tools.

1 Introduction

The conversational search domain has recently drawn increasing attention from the Information Retrieval (IR) community. A conversational agent, by definition, is expected to interact seamlessly with the user through natural language, either written (i.e. text chat-bots) or spoken (i.e. vocal assistants). Following the development of conversational systems, the evaluation of such systems is receiving a lot of attention. Following the best practices proposed by TREC CAsT [6, 7], the principal evaluation campaign in the conversational domain, the evaluation process is very similar to the one used in ad-hoc retrieval. It follows the Cranfield paradigm, with a corpus of passage documents, a set of conversations representing various information needs, and a set of relevance judgements. Each conversation is a sequence of utterances – i.e., phrases issued by the user during the conversation – and the relevance judgements are collected for each utterance. Several works [1, 9, 19–21, 26, 38] have already recognized the drawbacks of using traditional evaluation approaches in a (multi-turn) conversational setup. Among

the difficulties that make traditional evaluation techniques hardly applicable in the conversational domain, we can list the following: – *Lack of generalizability* [19]: conversations in the current evaluation collections represent a single interaction between a user and the ideal system. Therefore, when we evaluate using a conversation represented as a sequence of utterances, we consider a snapshot of reality. Since we have a unique sequence of utterances, we cannot generalize to conversations on the same topic that could have happened between the user and the system but are not in the collection. – *Lack of comparability* [19]: conversations have different lengths, they can contain chains of anaphoras or might have multiple self-contained utterances. Evaluation procedures should account for such diversity. – *Interdependency between utterances* [9]: utterances in conversational search are intrinsically dependent, differently from topics in ad-hoc retrieval. Therefore, cannot be treated as independent and identically distributed events. This work aims at providing a new perspective on the first aspect: low generalizability. We show a series of experiments meant to demonstrate the poor generalizability of results obtained using offline evaluation collections. Our work can be formalized with the following research questions:

RQ1 How can we shuffle utterances of a conversation by maintaining their original meaning and inter-dependencies?
RQ2 What is the effect of including dependency-aware permuted conversations in the comparison between systems?
RQ3 Can we improve conversational agents evaluation using permuted dialogues?

By answering the first question, we obtain a sound process to permute utterances of a conversation, producing new conversations to test conversational systems. We, therefore, use such conversations to compare models under the current evaluation paradigm, highlighting and measuring its flaws. Finally, we propose a new strategy to include the permuted conversations in the evaluation methodology. We do not propose a new evaluation measure – as done for example in [9, 19] – but show how, by adapting our current instruments, we could partially mitigate the limitations associated with the evaluation of the conversational systems. Our main contributions are the following. We show that:

- Modeling a conversation using a single sequence of utterances only favours some systems, while penalizing others;
- If we consider multiple valid permutations of the conversations, the performance of conversational agents moves from point estimations to distributions of performance (in which the default sequence is an arbitrary point);
- By including multiple permutations in the evaluation, we obtain more reliable and generalizable statistical inference.

Our work is organized as follows: Section 2 describes the current state in conversational evaluation. In Section 3 we describe our experimental methodology. Section 4 details on the experimental results observed. Finally, Section 5 describes the insights of our and outlines the next steps.

2 Related Work

Conversational agents are commonly divided into chit-chat bots [36, 37] and task-oriented systems [3, 14, 26]. Chit-chat bots are used to entertain the users, while the latter guide them to satisfy a goal, such as buy or discover something, through a dialogue. Task-driven conversational systems can be categorized into systems that retrieve and rank answers [14, 31, 35] and systems that build them through summarization techniques such as T5 [27]. While the latter are traditionally evaluated through traditional NLP and machine translation measures, such as BLEU [25] or METEOR [2], the former still relies on traditional IR evaluation measures such as Precision or Normalized Discounted Cumulated Gain (nDCG) [15], with typically a very small cutoff [6]. Finally, conversational systems can be divided into single-answer systems and multi-turn conversational systems. Among the former, we can list current commercial vocal assistants, handling very short - often scripted - sequences of interactions. The latter should ideally deal with a sequence of interactions of unspecified length. One of the most peculiar aspects related to the multi-turn conversational task is the role played by the concept of "context" [18, 23, 33]. The context corresponds to the system's internal representation of the conversation state that evolves through time. Correctly maintaining and updating such internal beliefs is essential to approach effectively the multi-turn conversational task. In this work, we focus on the evaluation of *Multi-turn Task-driven Conversational search systems*. Multi-turn conversational search is also the main focus of the TREC Conversational Assistance Track (CAsT) campaign [6, 7]. Currently, the track has reached its third edition: a further demonstration of the interest shown by the community. The evaluation aspect of conversational agents is consequently drawing increasing interest [1, 9, 19–21, 26, 38]. Even though several efforts aimed at developing proper techniques to evaluate conversational systems [9, 38], there is a consensus on the fact that we still lack the properer statistical tools to correctly evaluate such systems. Faggioli et al. [9] propose to model a conversation through a graph: utterances in a conversation are linked if they concern the same entities. Authors argue that current evaluation approaches introduce biases on systems comparison, by considering utterances as independent events. Faggioli et al. [10] do not tackle the problem linked to the low generalizability, due to predefined conversations available in current offline collections. Lipani et al. [19] start the low generalizability that affects the current offline evaluation of conversational systems. Lipani et al. [19] propose to simulate users through a stochastic process, similarly to what done in [38]. In particular, each topic is modelled as a set of subtopics (collected manually and using the available experimental collections). Using crowd assessors, Lipani et al. [19] define a Markov chain process that should model how users present utterances to the system when interacting with a conversational agent. This allows producing new simulated conversations. Such a solution partially solves the low generalizability problem. Nevertheless, the need for online data makes it infeasible for purely offline scenarios, where no users are available.

3 Methodology

In this section, we describe the experimental methodology to answer the research questions. In Subsection 3.1 we describe a permutation process capable of preserving the dependencies between utterances (RQ1). Finally, Section 3.2 defines a methodology to use ANOVA to evaluate conversational systems, using permuted conversation utterances (RQ3).

3.1 RQ1: a Dependence-aware Utterance Permutation Strategy

Several works [9, 19, 26, 38] recognize the need of increasing the variety of conversations to improve the generalizability of offline conversational evaluation. As observed by [19], when conversing with a system about a specific topic, distinct users tend to traverse subtopics in different orders. Generalization would ask to observe how distinct users interact with the systems to investigate a specific topic: this is not possible in an offline scenario. A possible approach to simulate how users would experience a system would be permuting the utterances of a given conversation, and measuring how it performs. We cannot however permute utterances completely randomly. In fact, we might lose temporal dependency between the moment the entity is mentioned in an utterance for the first time and referenced later. To solve this limitation, we would have to re-gather the relevance judgements to fit the newly defined anaphoras in the randomly built conversation. This is prohibitive and not suited to an offline evaluation scenario. A better permutation strategy consists in permuting utterances by respecting the temporal dependencies. To this end, we could rely on classification labels (we dub this approach `class-based` permutation) to identify such dependencies. Similarly to what done in [24], we can manually annotate the data using three classes of utterances:

- Self-Explanatory (`SE`) utterances: utterances that do not contain any semantic omission. Non-contextual retrieval systems can answer such utterances.
- Utterances that depend on the First Topic of the conversation (`FT`): they contain an - often implicit - reference to the general topic of the conversation, subsumed by the first utterance.
- Utterances that depend on a Previous Topic (`PT`): the previous `SE` utterance contains the entity to solve the semantic omission in the current one.

Using this utterance classification, we define a sampling process to randomly permute utterances of a conversation, while preserving temporal dependencies. We define the following rules for the generation of utterance permutations:

- The first utterance in any conversation expresses the main topic of the conversation. It cannot be moved to other positions.
- `SE` utterances, being independent by definition, can appear in any order inside the conversation.

- `PT` utterances have to appear immediately after their `SE` utterance. More in detail, after a `SE` utterance, in CAsT 2019 conversations, we have an arbitrary number of `PT` utterances (usually between 0 and 4): such utterances can appear in any order, as long as they occur after the associated `SE` utterance.
- `FT` utterances, depending on the global topic of the dialogue can be issued at any moment, since the first utterance cannot be moved.

3.2 RQ3: Exploiting Permuted Conversation Utterances

As a final methodological remark, we show how to embed utterances permutations in the evaluation. To have a common ground with current evaluation strategy, we consider to compare different retrieval models using ANalysis Of VAriance (ANOVA). If we were to apply ANOVA in the current evaluation setup, we would likely rely on the following model:

$$y_{ik} = \mu_{..} + \tau_i + \alpha_k + \varepsilon_{ik} \tag{MD0}$$

Where y_{ik} is the mean performance of all utterances for the conversation i, using the retrieval model k. $\mu_{..}$ is the grand mean, τ_i is the contribution to the performance of the i-th conversation, while α_k is the effect of the k-th system. Finally, ε_{ik} is the unexplained portion of the performance variation using the ANOVA model MD0. This is the traditional two-way ANOVA model used on IR data to recognize statistical differences between systems [5, 13, 34].

If we also include multiple permutations for each conversation, Model MD0 cannot be applied satisfactorily anymore. The different permutations behave as a nested factor. We need to resort to a three-way ANOVA, that includes the different permutations. A specific permutation is, trivially, a permutation only of one conversation: we cannot treat it as a permutation of others. The variation in the performance due to a permutation should contribute only to the variation in performance of the conversation it represents. Including multiple permutations, which behave as replicates [28], allows computing the interaction factor between retrieval models and conversations in the ANOVA model. In ad-hoc retrieval, such interaction has a medium-to-large size effect [4, 12, 34] and, if included, allows more powerful inferential analyses. We leave this analysis for future works. We use the following ANOVA model:

$$y_{i(j)k} = \mu_{..} + \tau_i + \nu_{j(i)} + \alpha_k + \varepsilon_{ijk} \tag{MD1}$$

Where, compared to Model MD0, $\nu_{j(i)}$ represent the effect of the j-th permutation of the i-th conversation.

4 Experimental Analysis

In our experimental analysis, we consider the Conversational Assistance Track (CAsT) 2019 [6]. Such collection contains 50 multi-turn conversations, each composed of 9 utterances on average. The utterances in their original formulation

contain semantic omissions - anaphoras, ellipsis and co-references. Among the 50 conversations, 30 were used for training and have smaller pools of relevance judgements. The remaining 20 are the test set. In our subsequent analyses, we consider only test conversations, being their relevance judgements much more significant. The corpus is composed of approximately 38 million paragraphs from the TREC Complex Answer Retrieval Paragraph Collection (CAR) [8] and the MS MARCO collection[4]. Regarding the relevance judgements, CAsT 2019 contains graded judgements on a scale from 0 to 4. We adopt nDCG with cutoff at 3, being the most widely diffused evaluation measure for this specific scenario [6]. To ease the reproducibility the code is publicly available[5].

4.1 Conversational Models

As commonly done [10, 11, 19], we select as baselines a set of models that represent different families of approaches to the multi-turn conversational task. Notice that, for all the rewriting strategies, we used BM25 as ranker.

Non-contextual baseline Models We consider three non-contextual baseline models, used as a comparison with other approaches. We compute the runs using the okapi BM25 model with default terrier parameters ($k = 1.2$ and $b = 0.75$). The second baseline is Query Language Model with Bayesian Dirichlet smoothing and $\mu = 2500$. Finally, we include results from a Pseudo-Relevance feedback RM3 rewriting model [22], which considers the 10 most popular terms of the 10 documents ranked the highestd.

Concatenation-Based Models A simple approach to enrich utterances with context to address the multi-turn conversational challenges consists in concatenating them with one (or more) of the previous ones. We propose three concatenation-based strategies, previously adopted as baselines in the literature [24]:

- First Utterance (`FU`): each utterance u_j is concatenated with u_1, the first utterance of the conversation.
- Context Utterance (`CU`): each utterance u_j is concatenated with u_1 and u_{j-1}, the previous utterance.
- Linear Previous (`LP`): we concatenate u_j with u_{j-1} linearly weighting the terms: $q_j = \lambda * u_j + (1 - \lambda) * u_{j-1}$, with $\lambda \in [0, 1]$. In particular, we observed empirically the best results for $\lambda = 0.6$.

Pseudo-Relevance Feedback Based Models We consider two approaches based on pseudo-relevance feedback (PRF) that account for the "multi-turn" aspect:

- RM3-previous (`RM3p`): it concatenates the current utterance and the RM3 expansion of the previous one (using BM25 as first stage retrieval model).

[4] http://www.msmarco.org/
[5] https://github.com/guglielmof/utterance_permutations

Table 1: Number of unique permutations that can be observed for each conversation in CAsT 2019, according to the `class-based` permutation.

Conv. id	31	32	33	34	37	40	49	50	54	56
unique perm.	72	15184	720	720	240	120	5039	120	25676	720
Conv. id	58	59	61	67	68	69	75	77	78	79
unique perm.	720	121	720	289	4996	480	721	48	48	241

– RM3-sequential (`RM3s`): it takes the relevance feedback considering the ranked list retrieved for the previous utterance, and uses it to expand the current.

The difference between the two models is that, for `RM3p`, the ranked list depends only on the previous utterance and the one at hand. Conversely, the latter considers the sequence of utterances observed up to the current one. In both cases, for the first query, we apply directly BM25, without rewriting it.

Language Model-Based Models Among the neural language models, we consider coref-spanBERT (`anCB`). This method relies on the Higher-order Coreference Resolution model, as defined in [17], but employs the spanBERT [16] embeddings to represent the words. In particular, we use the pre-trained version of the approach available in the AllenNLP framework[6].

4.2 RQ1: Permuting Conversations

Following the sampling process described in Subsection 3.1 we randomly permute the CAsT 2019 conversations. Table 1 reports the number of unique permutations obtained for each of the conversations in CAsT 2019.

The majority of the conversations have the `class-based` permutations in the order of tens to thousands. There are two main exceptions: conversations `54`, `32`. The larger number of permutations is due to the different structures of such conversations. For example, conversation `54` contains 3 `SE` utterances plus the first utterance and 5 `FT` utterances. Given these characteristics, we need to enforce only the first and third constraints to obtain valid `class-based` permutations, producing a larger space of valid permutations[7].

4.3 RQ2: Conversational Systems Performance on Permuted Conversations

Table 2 reports the nDCG@3 observed for the different archetypal conversational retrieval baselines either by considering only the original order of the utterances

[6] https://docs.allennlp.org

[7] If we consider all the random permutations, for an average 9-utterances conversation, we would have approximately 3.6×10^5 permutations: 10 times more than the maximum number of permutations observed using the `class-based` strategy.

Table 2: Performance measured with nDCG@3 for the baselines and PRF conversational models. Baselines results do not depend on the order of the utterances. We report the mean for both standard order of the utterances, and over all permuted conversations. Concerning permuted conversations, we also report the minimum and maximum mean over all conversations that can be observed, using different permutations.

		nDCG@3			
		orig. order	permutations		
	model		min	mean	max.
baselines	BM25	0.0981	0.0981	0.0981	0.0981
	DLM	0.0794	0.0794	0.0794	0.0794
	RM3	0.1064	0.1064	0.1064	0.1064
concatenation-based	`FU`	0.1692	0.1692	0.1692	0.1692
	`CU`	0.1687	0.1185	0.1481	0.1809
	`LP`	0.1464	0.0906	0.1279	0.1671
PRF-based	`RM3p`	0.1451	0.1019	0.1353	0.1709
	`RM3s`	0.1639	0.1108	0.1482	0.1857
neural LM based	`anCB`	0.1640	0.1410	0.1553	0.1645

as defined in CAsT 2019 or considering the average over multiple permutations for each conversation. To grant a fair comparison between different conversations, since they can have a different number of valid `class-based` permutations, we sample only 100 permutations for each of them. The most interesting insight that Table 2 is that the best performing system is the "First Utterance" (`FU`). We explain this because the first utterance of the original conversation is often the most generic. If we concatenate it with other utterances it can boost their recall, helping them obtain better results. The `FU` approach obtains the same results even when we permute conversations. Since we forced the first utterance to remain in its position, the order does not influence this algorithm. Therefore, we do not include it in subsequent analyses that measure the impact of permutations on conversational models. If we consider the result achieved with permuted conversations, we observe a general decrease in the average performance, due to the increased variance caused by the permutations. If we consider the maximum performance achievable, interestingly, all the methods can outperform the results achieved with the original order, indicating that there are situations in which different orders are preferable. The change in performance occurs due to the different information flow. The conversational models selected – as the majority of common conversational strategies – exploit the context to solve the anaphoras and rewrite the utterances. Such context derives from previous turns. By changing the previous turns, we also change the context, and thus the information used by the system. This aims at mimicking a real-world scenario, where we do not know if previous utterances provided good context. Furthermore, such context might change depending on the path followed by the user.

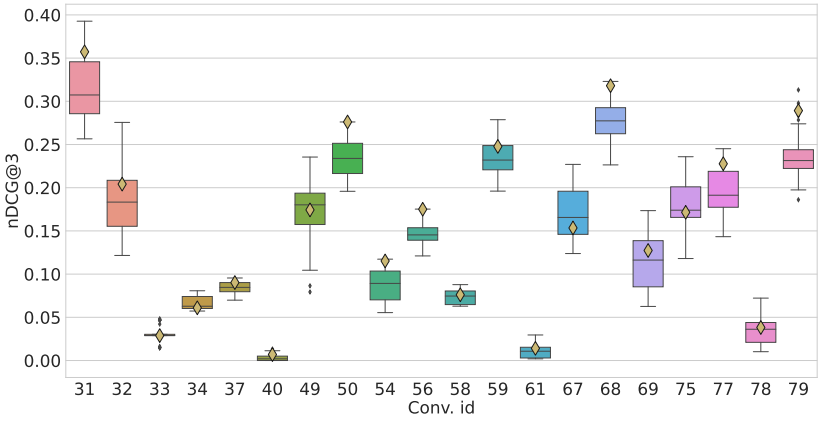

Fig. 1: Distributions of the average systems performance over different permutations of the conversations, considering original CAsT 2019 utterances. The yellow diamond is the average performance achieved using the original order of utterances. Observe that, in most cases the original order of the utterances does not have the best performance.

Table 3: Maximum distance observed between models, using different permutations. On the diagonal, the maximum average distance from all other systems. The absence of negative numbers indicates that it is always possible to make any model "the best".

model	CU	LP	RM3p	RM3s	anCB
CU	0.0727	0.1030	0.0958	0.0810	0.0882
LP	0.0644	0.0432	0.0460	0.0470	0.0805
RMp	0.0646	0.0577	0.0396	0.0476	0.0803
RMs	0.0955	0.1226	0.1147	0.0937	0.1148
anCB	0.0420	0.0668	0.0593	0.0402	0.0250

Figure 1 plots, for each CAsT 2019 conversation, the distribution over the permutations of the average performance of all systems. The yellow diamond represents the mean performance using the default order of the utterances. It is insightful noticing that the default order rarely gives the best performance: using a different order of utterances strongly influences performance. Such a pattern is also observable for each system singularly[8].

To further investigate the effect of permutations, we select the permutation that maximizes the difference in nDCG@3 between each pair of systems. We repeat this for each conversation. We also select the permutation that maximizes the average difference in performance between a system and all the others. Table 3 reports the results of such analysis. It is always possible to cherry-pick conversations permutations to make any model the best in a pairwise compar-

[8] We do not report the figure for each system, to avoid clutter.

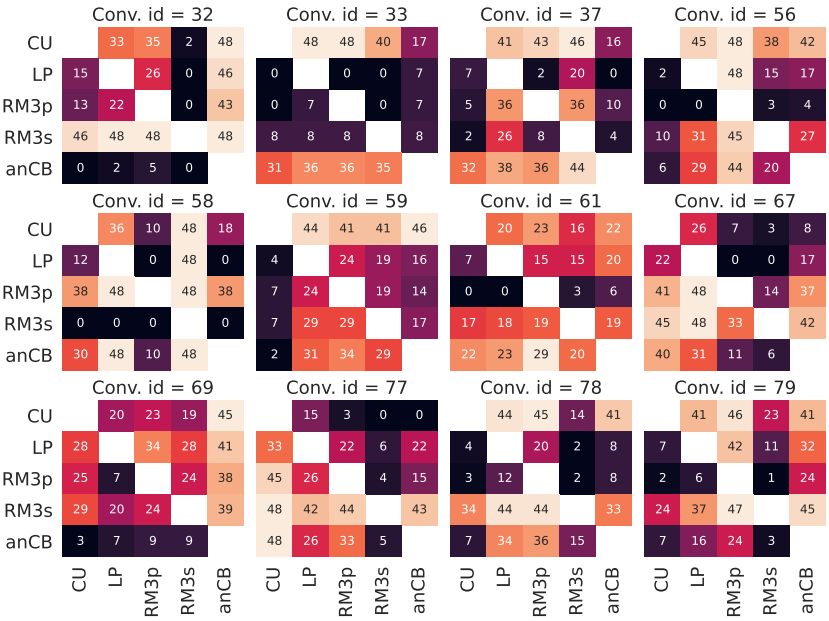

Fig. 2: Conversation-wise comparisons between pairs of systems. Number of times the row system is preferred over the column one, over different permutations of the conversation. Permuting the utterances order changes what system is deemed better: limiting ourselves to only one permutation might lead us to wrong conclusions. 12 out of 20 Conversations have been randomly selected, for the sake of presentation.

ison. When using a collection with a single sorting of the utterances for each conversation, we need to ask ourselves: is a system better than another or is it an artefact of the collection at hand? Can we trust our results to be generalizable on previously unseen conversations? The difference can be as large as 12%: it is huge if we consider the scale of our performance - see Table 2. Not only it is possible to make any model the best in a pairwise comparison, but we are also able to maximize the distance in terms of performance from any other model, to make an arbitrary system the best in absolute (see diagonal of Table 3). Figure 2 describes how often, conversation by conversation, we would change our opinion over which system is the best, if we present them with different utterances permutations. More in detail, for each conversation, in each cell we report how often the row system is deemed better than column one, over different utterances permutations. When we consider pairwise comparisons between systems, there is seldom a clear winner. For example, consider Conversation 59; in the majority of the pairwise comparisons, there is a 50% chance that one model is better than the other if we select a specific permutation of the utterances. A system wins over another on every permutation only in a few cases.

Table 4: Summary statistics for ANOVA MD0. This models considers only one permutation for each conversation (the original one, presented in CAsT 2019). Different models do not show significant differences. ω^2_{model} is not reported, being ω^2 ill-defined for non-significant factors.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$
topic	1.052	19	0.055	17.454	0.0000	0.758
model	0.010	4	0.002	0.762	0.5532	—
Error	0.241	76	0.003			
Total	1.302	99				

4.4 RQ3: Comparing Systems via ANOVA

Relying on the methodology proposed in Subsection 3.2, we now describe the ANOVAs on different conversational models, when either we consider or not multiple permutations of the utterances for each conversation. Notice that, since we are interested in evaluating the effect of the permutations and `FU` is not influenced by them, we exclude it from subsequent analyses.

Table 4 reports the summary statistics for ANOVA when applied to CAsT 2019 conversations, using the Model MD0. For each factor, we report the Sum of Squares (SS), the Degrees of Freedom (DF), the Mean Squares (MS), the F statistics, the p-value and the Strength of Association (SOA), measured according to the ω^2 measure.

We observe that the effect of the "conversation" factor is significant and large-sized ($\omega^2 \geq 0.14$). This pattern is often observed in many IR scenarios, such as ad-hoc retrieval [4, 13, 34] or Query Performance Prediction (QPP) [10]. Conversely, the effect of the Model factor is not significant: none of the models is significantly the best. We are not particularly surprised by that: both Table 3 and Figure 2 have shown that considering only a single permutation of the utterances, we would likely say something false by saying that a specific system is the best! This indicates the low discriminative power associated with this evaluation approach. If we were to consider state-of-the-art systems, possibly even more complex (and similar) than the ones we used, would we be able to state which system is statistically the best? Being able to discriminate between systems is a fundamental requirement for any evaluation approach [29, 30, 32]: could we deem ourselves satisfied with what we can achieve with the current evaluation setup in multi-turn conversational search?

Table 5 reports the summary statistics for ANOVA with model MD1. By looking at Table 5 we can see the first huge advantage of including permutations in our evaluation framework: the Model factor is now significant - although small ($0.01 < \omega^2 < 0.06$). As a side note, Tukey's post-hoc analysis shows that `anCB` is the best model, followed by `RM3s` which belong to the same tier. Subsequently, we have `RM3p` and `CU`, which again are statistically not different from each other, but worse than the previous ones. Finally, `LP` is the only member of the worst-quality tier. We have moved from having all models equal in Table 4 to a four-tiers

Table 5: Summary statistics for ANOVA MD1. This models considers 100 unique permutations for each conversation plus the original one. Observe that now all the factors have a significant effect.

Source	SS	DF	MS	F	p-value	$\hat{\omega}^2_{\langle fact \rangle}$
topic	38.594	19	2.031	657.983	>1e-3	0.722
perm(topic)	2.438	940	0.003	0.840	0.999	—
model	0.472	4	0.118	38.230	>1e-3	0.030
Error	11.842	3836	0.003			
Total	53.347	4799				

sorting of the models in Table 5. The Permutation factor is not significant. This suggests that there is not a single permutation that allows every system to work better, but rather there is an interaction between the systems and permutations: distinct models behave differently according to the permutation at hand. Table 5 shows that, if we use the permutations as additional evidence of the quality of a model, we discriminate better between them. Furthermore, we do not know in which order the user will pose their utterances. Including permutations allows us to model better the reality: what we observe in our offline experiment is likely to generalize more to a real-world scenario. Permutations allow robust statistical inference, without requiring to gather new conversations, utterances and relevance judgements.

5 Conclusions and Future Works

In this work, we showed that traditional evaluation is seldom reliable when applied to the conversational search. We proposed a methodology to permute the utterances of the conversations used to evaluate conversational systems, enlarging conversational collections. We showed that it is hard to determine the best system when considering multiple conversation permutations. Consequently, any system can be deemed the best, according to specific permutations of the conversations. Finally, we showed how to use permutations of the evaluation dialogues, obtaining by far more reliable and trustworthy systems comparisons.

As future work, we plan to study how to estimate the distribution of systems performance without actually having the permutations and the models at hand. We plan to investigate how to use the performance distributions to compare multi-turn conversational models.

Acknowledgments Nicola Tonellotto was partially supported by the Italian Ministry of Education and Research (MIUR) in the framework of the CrossLab project (Departments of Excellence).

References

1. Anand, A., Cavedon, L., Joho, H., Sanderson, M., Stein, B.: Conversational Search (Dagstuhl Seminar 19461). In: Dagstuhl Reports, vol. 9 (2020)
2. Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pp. 65–72 (2005)
3. Bangalore, S., Di Fabbrizio, G., Stent, A.: Learning the Structure of Task-Driven Human–Human Dialogs. IEEE Transactions on Audio, Speech, and Language Processing **16**(7), 1249–1259 (2008)
4. Banks, D., Over, P., Zhang, N.F.: Blind Men and Elephants: Six Approaches to TREC data. Information Retrieval Journal **1**(1-2), 7–34 (1999)
5. Culpepper, J.S., Faggioli, G., Ferro, N., Kurland, O.: Topic difficulty: Collection and query formulation effects. ACM Transactions on Information Systems (TOIS) **40**(1), 1–36 (2021)
6. Dalton, J., Xiong, C., Callan, J.: TREC CAsT 2019: The Conversational Assistance Track Overview. In: Proceedings of TREC (2020)
7. Dalton, J., Xiong, C., Callan, J.: TREC CAsT 2020: The Conversational Assistance Track Overview. In: Proceedings of TREC (2021)
8. Dietz, L., Verma, M., Radlinski, F., Craswell, N.: TREC Complex Answer Retrieval Overview. In: Proceedings of TREC (2017)
9. Faggioli, G., Ferrante, M., Ferro, N., Perego, R., Tonellotto, N.: Hierarchical Dependence-Aware Evaluation Measures for Conversational Search. In: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 1935–1939 (2021)
10. Faggioli, G., Zendel, O., Culpepper, J.S., Ferro, N., Scholer, F.: An Enhanced Evaluation Framework for Query Performance Prediction. In: Porceedings of the 43rd European Conference on Information Retrieval, pp. 115–129 (2021)
11. Ferro, N., Harman, D.: CLEF 2009: Grid@CLEF Pilot Track Overview. In: Multilingual Information Access Evaluation I. Text Retrieval Experiments, pp. 552–565 (2009)
12. Ferro, N., Sanderson, M.: Improving the Accuracy of System Performance Estimation by Using Shards. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 805–814 (2019)
13. Ferro, N., Silvello, G.: A General Linear Mixed Models Approach to Study System Component Effects. In: Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 25–34 (2016)
14. Gu, J.C., Ling, Z.H., Liu, Q.: Utterance-to-Utterance Interactive Matching Network for Multi-Turn Response Selection in Retrieval-Based Chatbots. IEEE/ACM Trans. Audio, Speech and Lang. Proc. **28**, 369–379 (Jan 2020)
15. Järvelin, K., Kekä läinen, J.: Cumulated Gain-Based Evaluation of IR Techniques. ACM Trans. on Information Systems **20**(4), 422–446 (2002)

16. Joshi, M., Chen, D., Liu, Y., Weld, D.S., Zettlemoyer, L., Levy, O.: Spanbert: Improving pre-training by representing and predicting spans. Trans. of the Association for Computational Linguistics **8**, 64–77 (2020)
17. Lee, K., He, L., Zettlemoyer, L.: Higher-order Coreference Resolution with Coarse-to-fine Inference. In: Proceedings of the 2018 Conference of the NAACL-HLT, pp. 687–692 (2018)
18. Li, J., Liu, C., Tao, C., Chan, Z., Zhao, D., Zhang, M., Yan, R.: Dialogue history matters! personalized response selection in multi-turn retrieval-based chatbots. ACM Trans. on Information Systems **39**(4) (2021)
19. Lipani, A., Carterette, B., Yilmaz, E.: How Am I Doing?: Evaluating Conversational Search Systems Offline. ACM Trans. on Information Systems **39**(4) (2021)
20. Liu, C.W., Lowe, R., Serban, I.V., Noseworthy, M., Charlin, L., Pineau, J.: How NOT To Evaluate Your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation (2017)
21. Liu, Z., Zhou, K., Wilson, M.L.: Meta-Evaluation of Conversational Search Evaluation Metrics. ACM Trans. on Information Systems **39**(4) (2021)
22. Lv, Y., Zhai, C.: Positional relevance model for pseudo-relevance feedback. In: Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 579–586 (2010)
23. Mele, I., Muntean, C.I., Nardini, F.M., Perego, R., Tonellotto, N., Frieder, O.: Topic Propagation in Conversational Search. In: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, pp. 2057–2060 (2020)
24. Mele, I., Muntean, C.I., Nardini, F.M., Perego, R., Tonellotto, N., Frieder, O.: Adaptive utterance rewriting for conversational search. Information Processing & Management **58**(6), 102682 (2021)
25. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pp. 311–318 (2002)
26. Penha, G., Hauff, C.: Challenges in the evaluation of conversational search systems. In: Workshop on Conversational Systems Towards Mainstream Adoption, KDD-Converse (2020)
27. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020)
28. Rutherford, A.: ANOVA and ANCOVA. A GLM Approach. John Wiley & Sons, New York, USA, 2nd edn. (2011)
29. Sakai, T.: Evaluating Evaluation Metrics based on the Bootstrap. In: Proceedings of the 29th International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 525–532 (2006)
30. Smucker, M.D., Allan, J., Carterette, B.A.: A Comparison of Statistical Significance Tests for Information Retrieval Evaluation. In: Proceedings of the Sixteenth ACM Conference on Conference on Information and Knowledge Management, pp. 623–632 (2007)

31. Tao, C., Wu, W., Xu, C., Hu, W., Zhao, D., Yan, R.: Multi-representation fusion network for multi-turn response selection in retrieval-based chatbots. In: Proceedings of the 12th ACM International Conference on Web Search and Data Mining, pp. 267–275 (2019)
32. Urbano, J., Lima, H., Hanjalic, A.: Statistical Significance Testing in Information Retrieval: An Empirical Analysis of Type I, Type II and Type III Errors. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 505–514 (2019)
33. Vakulenko, S., Longpre, S., Tu, Z., Anantha, R.: Question rewriting for conversational question answering. In: Proceedings of the fourth ACM international conference on Web search and data mining (WSDM), pp. 355—-363 (2021)
34. Voorhees, E.M., Samarov, D., Soboroff, I.: Using Replicates in Information Retrieval Evaluation. ACM Trans. Information Systems **36**(2), 1–21 (2017)
35. Wu, Y., Wu, W., Xing, C., Zhou, M., Li, Z.: Sequential Matching Network: A New Architecture for Multi-turn Response Selection in Retrieval-Based Chatbots. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, pp. 496–505 (2017)
36. Yan, R.: " chitty-chitty-chat bot": Deep learning for conversational ai. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI), vol. 18, pp. 5520–5526 (2018)
37. Yu, Z., Xu, Z., Black, A.W., Rudnicky, A.: Strategy and policy learning for non-task-oriented conversational systems. In: Proceedings of the 17th annual meeting of the special interest group on discourse and dialogue, pp. 404–412 (2016)
38. Zhang, S., Balog, K.: Evaluating conversational recommender systems via user simulation. In: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, p. 1512–1520 (2020)