Improving Conversational Evaluation via a Dependency-Aware Permutation Strategy

Guglielmo Faggioli[1], Marco Ferrante[1], Nicola Ferro[1], Raffaele Perego[2] and Nicola Tonellotto[3]

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

Abstract

The rapid growth in number and complexity of conversational agents has highlighted the need for suitable evaluation tools to describe their performance. Current offline conversational evaluation approaches rely on collections composed of multiturn conversations, each including a sequence of utterances. Such sequences represent a snapshot of reality: a single dialog between the user and a hypothetical system on a specific topic. We argue that this paradigm is not realistic enough: multiple users will ask diverse questions in variable order, even for a conversation on the same topic. In this work[1] we propose a dependency-aware utterances sampling strategy to augment data available in conversational collections while maintaining temporal dependencies within conversations. Using the sampled conversations, we show that the current evaluation framework favours specific systems while penalizing others, leading to biased evaluation. We further show how to exploit dependency-aware utterances permutations in our current evaluation framework and increase the power of statistical evaluation tools such as ANOVA.

1. Introduction

The conversational search domain has recently drawn increasing attention from the Information Retrieval (IR) community. A conversational agent 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, also their evaluation is receiving a lot of attention. According to the best practices proposed by TREC CAsT [2], 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 [3, 4, 5] have already recognized the drawbacks of using traditional evaluation approaches in a (multi-turn) conversational setup. Conversations in the current evaluation collections represent a single interaction between a user and a hypothetical system. Therefore, when we evaluate using a conversation represented as a sequence of utterances, we consider a snapshot of reality [4]. Therefore, since we have a unique sequence of utterances, we cannot generalize to conversations

[1]This is an extended abstract of Faggioli et al. [1]

SEBD 2022: The 30th Italian Symposium on Advanced Database Systems, June 19-22, 2022, Tirrenia (PI), Italy

on the same topic not present in the collection that could have happened between the user and the system. 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 What is the effect of including dependency-aware permuted conversations in the comparison between systems?

RQ2 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 [5, 4] – 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: i) Modeling a conversation using a single sequence of utterances only favours some systems, while penalizing others; ii) 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); iii) By including multiple permutations in the evaluation, we obtain more reliable and generalizable statistical inference.

2. Related Work

In this work, we focus on the evaluation of *Multi-turn Task-driven Conversational search systems*. One of the most peculiar aspects related to the multi-turn conversational task is the role played by the concept of "context" [6, 7]. The context corresponds to the system's internal representation of the conversation state that evolves through time. Correctly maintaining and updating such internal belief is essential to approach effectively the multi-turn conversational task. Multi-turn conversational search is also the main focus of the TREC Conversational Assistance Track (CAsT) campaign [2]. 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 [8, 3, 4, 5]. Even though several efforts aimed at developing proper techniques to evaluate conversational systems [5, 3], there is a consensus on the fact that we still lack the properer statistical tools to correctly evaluate such systems. Faggioli et al. [5] 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. Lipani et al. [4] propose to simulate users through a stochastic process, similarly to what done in [3]. 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. [4] 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. Dependence-aware Utterance Permutation Strategy

Several works [8, 3, 5, 4] recognize the need of increasing the variety of conversations to improve the generalizability of offline conversational evaluation. As observed by [4], 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 consists in permuting the utterances of a given conversation, and measure how the system performs on the new scenario. We cannot however permute utterances completely randomly. In fact, we might lose temporal dependency between the moment an entity is mentioned for the first time and its subsequent references. 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. Following the work by Mele et al. [9], we manually annotate the data using four classes of utterances. We also identify constraints to permute utterances of a conversation, while preserving temporal dependencies. The utterance classes and constraints are the following:

- First utterance: it expresses the main topic of the conversation and cannot be moved.
- Self-Explanatory (SE) utterances do not contain any semantic omission. Non-contextual retrieval systems can answer such utterances. Being independent from other utterances, they can appear in any position inside the conversation.
- 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. Since the FT utterances depend only on the global topic of the dialogue, they can be issued at any moment after the first one.
- Utterances that depend on a Previous Topic (PT): the previous SE utterance contains the entity to solve the semantic omission in the current one. PT utterances have to appear immediately after their SE utterance but they can be permuted with other PT utterances referring to the same SE.

4. Experimental Analysis

In our experimental analysis, we consider the Conversational Assistance Track (CAsT) 2019 [2]. 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 all conversations, we consider only the 20 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) [10] and the MS MARCO collection. Regarding the relevance judgements, CAsT 2019 contains graded judgements on a scale from 0 to 4. We adopt Normalized Discounted Cumulated Gain (nDCG) with cutoff at 3, being the most widely diffused evaluation measure for this specific scenario [2].

4.1. Conversational Models

As commonly done [4], we select as a set of archetypal conversational models to observe what happens with conversations permutations. If not differently specified, we used BM25 as ranker.
Non-contextual baseline Models We consider three non-contextual baseline models, used as a comparison with other approaches: okapi BM25 model with default terrier parameters ($k = 1.2$ and $b = 0.75$); Query Language Model with Bayesian Dirichlet smoothing and $\mu = 2500$; a model based on Pseudo-Relevance feedback RM3 rewriting [11] that considers the 10 most popular terms of the 10 documents ranked the highest.

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 [9]. 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 $\lambda \in [0, 1]$. We use $\lambda = 0.6$, since it provides the best empirical results.

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); 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.

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 [12], but employs the spanBERT [13] embeddings to represent the words. In particular, we use the pre-trained version of the approach available in the AllenNLP framework[1].

4.2. RQ1: Conversational Systems Performance on Permuted Conversations

Table 1 reports the nDCG@3 observed for the different archetypal conversational retrieval baselines either by considering only the original order of the utterances 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. An

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

Table 1
Performance measured with nDCG@3 for the selected archetypal 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.

		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

interesting insight that can be drawn by Table 1 is that the best performing system is the "First Utterance" (FU). The first utterance of the original conversation is often the most generic: f we concatenate it with other utterances, it boosts 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.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.

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[2]. Notice that, with the new permuted conversations, it is always possible to cherry-pick conversations permutations to make any model the best in a pairwise comparison.

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

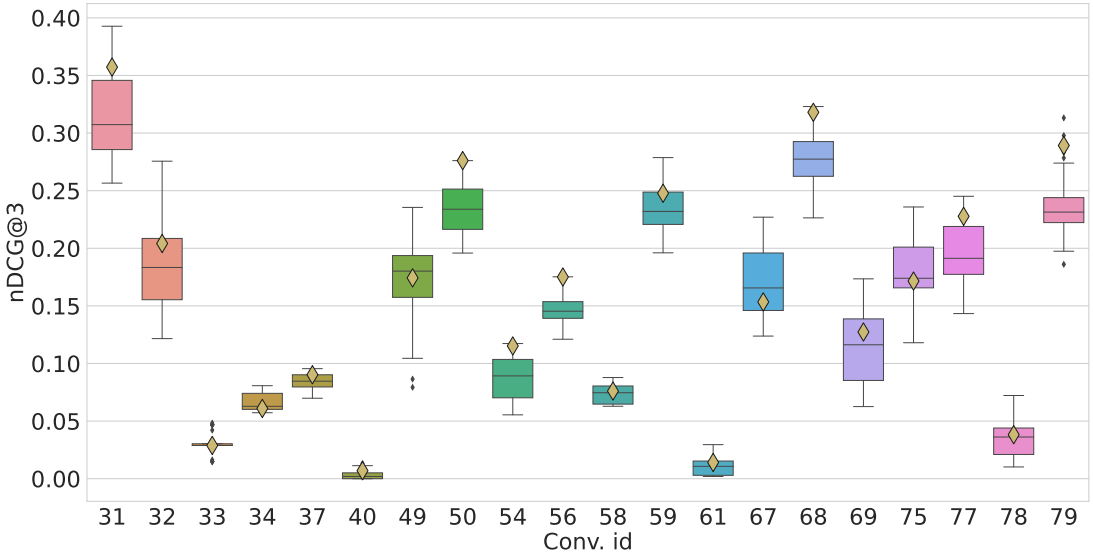

Figure 1: Distributions of the average systems performance over different permutations of the conversations, considering original CAsT 2019 utterances. The yellow diamond is the average performance achieved using the original order of utterances. In most cases the original order of the utterances does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

Table 2
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				

4.3. RQ2: Comparing Systems via ANOVA

We are now interested in assessing the effect of using utterances permutations in the current evaluation scenario. We therefore 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 error. Table 2 reports the summary statistics for ANOVA when applied to CAsT 2019 conversations, using the Model MD0. We observe that the effect of the "conversation" factor is significant and large-sized ($\omega^2 \geq 0.14$).

Table 3
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. 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.

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				

This pattern is often observed in many IR scenarios, such as ad-hoc retrieval [14] or Query Performance Prediction (QPP) [15]. Conversely, the effect of the Model factor is not significant: none of the models is significantly the best. This indicates the low discriminative power associated with this evaluation approach.

If we include permutations for each conversation, we can 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, $\nu_{j(i)}$ is the nested factor that represents the effect of the j-th permutation of the i-th conversation. Table 3 reports the summary statistics for ANOVA with model MD1. By looking at Table 3 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 2 to a four-tiers sorting of the models in Table 3. The Permutation factor is not significant an this highlights that there is not a single best permutation for every system, but rather there is an interaction between the systems and permutations: distinct models behave differently according to the permutation at hand. Table 3 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.

References

1. G. Faggioli, M. Ferrante, N. Ferro, R. Perego, N. Tonellotto, A Dependency-Aware Utterances Permutation Strategy to Improve Conversational Evaluation, in: Proc. ECIR, 2022.
2. J. Dalton, C. Xiong, J. Callan, TREC CAsT 2019: The Conversational Assistance Track Overview, in: TREC, 2020.
3. S. Zhang, K. Balog, Evaluating conversational recommender systems via user simulation, in: Proc. SIGKDD, 2020, p. 1512–1520.
4. A. Lipani, B. Carterette, E. Yilmaz, How Am I Doing?: Evaluating Conversational Search Systems Offline, TOIS 39 (2021).
5. G. Faggioli, M. Ferrante, N. Ferro, R. Perego, N. Tonellotto, Hierarchical Dependence-Aware Evaluation Measures for Conversational Search, in: Proc. SIGIR, 2021, p. 1935–1939.
6. J. Li, C. Liu, C. Tao, Z. Chan, D. Zhao, M. Zhang, R. Yan, Dialogue history matters! personalized response selection in multi-turn retrieval-based chatbots, TOIS 39 (2021).
7. I. Mele, C. I. Muntean, F. M. Nardini, R. Perego, N. Tonellotto, O. Frieder, Topic Propagation in Conversational Search, in: Proc. SIGIR, 2020, pp. 2057–2060.
8. G. Penha, C. Hauff, Challenges in the evaluation of conversational search systems, in: Workshop on Conversational Systems Towards Mainstream Adoption, KDD-Converse, 2020.
9. I. Mele, C. I. Muntean, F. M. Nardini, R. Perego, N. Tonellotto, O. Frieder, Adaptive utterance rewriting for conversational search, IPM 58 (2021) 102682.
10. L. Dietz, M. Verma, F. Radlinski, N. Craswell, TREC Complex Answer Retrieval Overview., in: TREC, 2017.
11. Y. Lv, C. Zhai, Positional relevance model for pseudo-relevance feedback, in: Proc. SIGIR, 2010, p. 579–586.
12. K. Lee, L. He, L. Zettlemoyer, Higher-order Coreference Resolution with Coarse-to-fine Inference, in: Proc. NAACL-HLT, 2018, pp. 687–692.
13. M. Joshi, D. Chen, Y. Liu, D. S. Weld, L. Zettlemoyer, O. Levy, Spanbert: Improving pre-training by representing and predicting spans, TACL 8 (2020) 64–77.
14. D. Banks, P. Over, N.-F. Zhang, Blind Men and Elephants: Six Approaches to TREC data, IRJ 1 (1999) 7–34.
15. G. Faggioli, O. Zendel, J. S. Culpepper, N. Ferro, F. Scholer, An Enhanced Evaluation Framework for Query Performance Prediction, in: Proc. ECIR, 2021, pp. 115–129.