

Query Refinement Using Conversational Context: a Method and an Evaluation Resource

Maryam Habibi and Andrei Popescu-Belis

Idiap Research Institute and École Polytechnique Fédérale de Lausanne (EPFL)
Rue Marconi 19, 1920 Martigny, Switzerland
{maryam.habibi, andrei.popescu-belis}@idiap.ch

Abstract. This paper introduces a query refinement method applied to queries asked by users during a meeting or a conversation. Current approaches suffer from poor quality to achieve this goal, but we argue that their performance could be improved by focusing on the local context of the conversation. The proposed technique first represents the local context by extracting keywords from the transcript of the conversation. It then expands the queries with keywords that best represent the topic of the query (e.g. pairs of expansion keywords together with a weight indicating their topical similarity to the query). Moreover, we present a dataset called AREX and an evaluation metric. We compared our query expansion approach with other methods, on topics extracted from the AREX dataset and based on relevance judgments collected in a crowdsourcing experiment. The comparisons indicate the superiority of our method on both manual and ASR transcripts of the AMI Meeting Corpus.

Keywords: Query Refinement, Speech-based Information Retrieval, Crowdsourcing, Evaluation.

1 Introduction

We introduce a query refinement technique for explicit queries addressed by users to a system during a conversation. Retrieval based on these queries can be erroneous, due to their inherent ambiguity or to automatic speech recognition (ASR) mistakes. We propose a technique for query refinement using the local context of the conversation, in order to both properly answer the users’ information needs and to prevent from distracting users from the main topics of their discussion without asking for additional clarifications.

For instance, in the example discussed in Section 5.4 below, in which people are talking about the design of remote control, a participant needs more information about “LCD”. What would then be the most helpful Wikipedia pages to answer users’ information needs in the context of remote control, and how would a system determine them?

Previous query refinement techniques mostly enrich queries either interactively, or automatically, by adding relevant specifiers obtained from an external

data source. However, interacting with users for query refinement may distract them from their current conversation, while using an external data source outside the users’ local context may cause misinterpretations. For example, the word “LCD” can be interpreted as “a mathematic function”, “an American dance band”, “a politician at Lesotho Congress for Democracy” in addition to “Liquid-Crystal Display” which is the true interpretation in the example provided in Section 5.4.

Other previous techniques have attempted to use the local context of users’ activities, without requiring user interaction [1, 8]. However, as we will show, these are less suitable for a conversational environment, because of the nature of the vocabulary and the errors introduced by the ASR such as ‘recap’, ‘sleek’ and ‘snowman’ in this example.

In this paper, the local context of an explicit query is represented by a keyword set that is automatically obtained from the conversation fragment preceding each query. We assign a weight value to each keyword, based on its topical similarity to the explicit query, to reduce the effect of the ASR noise, and to recognize appropriate interpretations of the query. In order to evaluate the improvement brought by this method, we made available the AREX dataset (AMI Requests for Explanations and Relevance Judgments for their Answers). This dataset contains a set of explicit queries inserted in several conversations of the AMI Meeting Corpus [9], along with a set of human relevance judgments over sample retrieval results from Wikipedia for each query, and an automatic evaluation metric based on Mean Average Precision (MAP). The results show the superiority of our technique over previous ones and its robustness with respect to unrelated keywords or ASR noise.

The paper is organized as follows. In Section 2, we review existing methods for query refinement. We then describe the proposed query refinement method using conversational context in Section 3. Section 4 explains how the AREX dataset was constructed and specifies the evaluation metric. Section 5 presents and discusses the experimental results obtained both with ASR output and human-made transcripts of the AMI Meeting Corpus.

2 Related Work

Several methods for the refinement of explicit queries asked by users have been proposed in the field of information retrieval, and are often classified into query expansion techniques and relevance feedback ones [11]. Query expansion generates one or more hypotheses for query refinement by recognizing possible interpretations of a query, based on knowledge coming either directly from the corpus [2, 29, 24, 10, 3] or from Web data or personal profiles in the case of Web search [30, 13, 12, 21]. Query expansion techniques select suggestions for query refinement either interactively or automatically [11]. For instance, relevance feedback gathers judgments obtained from the users on sample results obtained from an initial query [25, 26, 19].

These methods are not ideal for refinement of explicit queries asked during a conversation, because they require users to interrupt their conversation. On the contrary, our overall goal is to estimate users’ information needs from their explicit queries as discretely as possible. Moreover, using the local context for query refinement instead of external, non-contextual resources has the potential to improve retrieval results [8].

To the best of our knowledge, two previous systems have utilized the local context for the augmentation of explicit queries. The JIT-MobIR system for mobile devices [1] used contextual features from the physical and the human environment for the refinement of explicit queries. However, the content of the activities itself was not used as a feature. The WATSON system [8] refined explicit queries by concatenating them with keywords extracted from the documents being edited or viewed by the user. However, in order to apply this method to a retrieval system for which the local context is a conversation, the keyword lists must avoid considering irrelevant topics from ASR errors. Moreover, unlike written documents which follow generally a planned and focused structured, in a conversation users often turn from one topic to another, and adding such a variety of keywords to a query might deteriorate the retrieval results [4, 11].

3 Content-based Query Refinement

The system that we have been building is the Automatic Content Linking Device [23, 22], which monitors a conversation between people, such as a business meeting, and allows them to formulate explicit spoken queries in order to retrieve documents from the Web or a specific repository – in this paper, Wikipedia. The users can simply address the system by using a pre-defined unambiguous name, which is robustly recognized by a real-time ASR component [14]. More sophisticated strategies for addressing a system in a multi-party dialogue context have been studied [6, 28], but they are beyond the scope of this paper, which is concerned with the processing of the query itself by the system. Once the results are generated by the system, they are displayed on a shared projection screen or on each user’s device.

To answer an explicit query Q , the process of query refinement starts by modeling the local context using the transcript of the conversation fragment preceding the query. We use a fixed length for all the fragment, though more sophisticated strategies are under consideration too. From this context, we extract a set of keywords C using a diverse keyword extraction technique which we previously proposed [15], by maximizing the coverage of the fragment’s topics with keywords. We then weigh the extracted keywords by using a filter that assigns a weight m_i , with $0 \leq m_i < 1$, to each keyword $kw_i \in C \setminus Q$ based on the normalized topical similarity of the keyword to the explicit query, as formulated in the following equation:

$$m_i = \frac{\sum_{z \in Z} p(z|Q)p(z|kw_i)}{\sqrt{\sum_{z \in Z} p(z|kw_i)^2} \sqrt{\sum_{z \in Z} p(z|Q)^2}} \quad (1)$$

In this equation, Z is the set of abstract topics which correspond to latent variables inferred using a topic modeling technique over a large collection of documents, and $p(z|kw_i)$ is the distribution of topic z in relation to the keyword kw_i . Similarly, $p(z|Q) = \frac{1}{|Q|} \sum_{q \in Q} p(z|q)$ is the averaged distribution of topic z in relation to the query Q , made of words q . These topic distributions are created using the LDA topic modeling technique [5], implemented in the Mallet toolkit [20]. The topic models are learned over a large subset of the English Wikipedia with around 125,000 randomly sampled documents [18]. Following several previous studies, we fixed the number of topics at 100 [7, 18].

Note that we do not directly select the expansion keywords from all the words of the fragments sorted in proportion to their topical similarity to the query, in order to avoid expanding it with words which are relevant to one of its aspects but not to the main topics of the fragments.

Each query Q is refined by adding additional keywords extracted from the fragment, generating a parametrized refined query $RQ(\lambda)$ which is defined as a set of weighted keywords, i.e. pairs of (word, weight), as shown below:

$$RQ(\lambda) = \{(q_1, 1), \dots, (q_{|Q|}, 1), (kw_1, m_1^\lambda), \dots, (kw_{|C|}, m_{|C|}^\lambda)\} \quad (2)$$

In other words, the refined query contains the words from the explicit query with weight 1, and the expansion keywords with a weight proportional to their topic similarity to the query.

The λ parameter has the following role. If $\lambda = \infty$, the refined query is the same as the initial explicit query (with no refinement). By setting λ to 0, the query is like the one used in the Watson system [8], giving the same weight to the query words and to the keywords representing the local context. Because the keywords are related to topics that have various relevance values to the explicit query, we will set the intermediate value $\lambda = 1$ in our experiments, to weigh each keyword based on its relevance to the topics of the query. The value of λ could be optimized if more training data were available.

4 Dataset and Evaluation Method

Our experiments are conducted on the AREX dataset (“AMI Requests for Explanations and Relevance Judgments for their Answers”) which we make available at www.idiap.ch/dataset/arex. The dataset contains a set of explicit queries, inserted at various locations of the conversations from the AMI Meeting Corpus [9], as explained in Section 4.1. The dataset also includes relevance judgments gathered using a crowdsourcing platform over the documents retrieved for four query expansion methods described in Sections 4.2 and 5.

4.1 Explicit Queries in the Dataset

The AMI Meeting Corpus contains conversations on designing remote controls, in series of four scenario-based meetings each, for a total of 138 meetings. Our

dataset is made of a set of explicit queries with the time of their occurrence in the AMI Corpus. Since the number of naturally-occurring queries in the corpus is insufficient for evaluating our system, we artificially generated a set of queries using the following procedure.

First, utterances containing an acronym X are automatically detected, because acronyms are one of the typical items which are likely to require explanations because of their potential ambiguity, and also they cover the queries which appeared in the AMI Corpus naturally. Of course, our query expansion technique is applicable to any explicit query. We formulate an explicit query such as “I need more information about X ”, and add it after these utterances. Seven acronyms, all-but-one related to the domain of remote controls, are considered: *LCD* (liquid-crystal display), *VCR* (videocassette recorder), *PCB* (printed circuit board), *TFT* (thin-film-transistor liquid-crystal display), *NTSC* (National Television System Committee), *IC* (integrated circuit), and *RSI* (repetitive strain injury). These acronyms occur 74 times in the scenario-based meetings of the AMI Corpus and are accompanied by 74 different conversation fragments.

We used both manual and ASR transcripts of the fragments from the AMI Corpus in our experiments. The ASR transcripts were generated by the AMI real-time ASR system for meetings [14], with an average word error rate (WER) of 36%. In addition, for experimenting with a variable range of WER values, we have simulated the potential speech recognition mistakes by applying to the manual transcripts of these conversation fragments three different types of the ASR noise: deletion, insertion and substitution. In a systematic manner, i.e. altering all occurrences of a word type, we randomly selected the conversation words, as well as the words to be inserted, from the vocabulary of the English Wikipedia [16]. The percentage of simulated ASR noise varied from 10% to 30%, as the best recognition accuracy reaches around 70% in conversational environments [17]. However, noise was never applied to the explicit query itself.

4.2 Evaluation Using the Dataset

Ground truth relevance judgments. Following a classical approach for evaluating information retrieval [27], we build a reference set of retrieval results by merging the lists of the top 10 results from four different methods for query expansion. Expanded queries are processed by the Apache Lucene search engine over the English Wikipedia. Three of the methods are listed in Sections 3 and 5, and the last one consists of only the keywords extracted from conversation fragments, with no words from the queries. We found that each explicit query had at least 31 different results for all the 74 fragments, and we decided to limit the reference set to 31 documents for each query.

Each fragment is about 400 words long, for the following reason. We computed the sum of the weights assigned to the keywords extracted from each fragment by $RQ(1)$ which weighs keywords based on their relevance to the query topics. Then we averaged them over 25 queries, which were randomly selected from the AREX dataset to serve as a development set for tuning our hyperparameters. The values obtained from five repetitions of the experiment with

the fragment lengths varying from 100 to 500 words in increments of 100 were, respectively: 2.14, 2.32, 2.08, 2.08, and 2.08. Since there is no variation in these values for the last three values, we set fragment size to 400 words. We have also limited the weighting to the first 10 keywords extracted from each fragment, following several previous studies [11], thus speeding up the query processing.

We designed a set of tasks to gather relevance judgments for the reference set from human subjects. We showed to the subjects the transcript of the conversation fragment ending with the query: “I need more information about X” with ‘X’ being one of the acronyms considered here. This was followed by a control question about the content of the conversation, and then by the list of 31 documents from the reference set. The subjects had to decide on the relevance value of each document by selecting one of the three options among ‘irrelevant’, ‘somewhat relevant’ and ‘relevant’ (noted below as $A = \{a_0, a_1, a_2\}$).

We collected judgments for the 74 queries of our dataset from 10 subjects per query. The tasks were crowdsourced via Amazon’s Mechanical Turk, each judgment becoming a “human intelligence task” (HIT). The average time spent per HIT was around 2 minutes. For qualification control, we only accepted subjects with greater than 95% approval rate and with more than 1000 previously approved HITs, and we only kept answers from the subjects who answered correctly the control questions. We applied furthermore a qualification control factor to the human judgments, in order to reduce the impact of “undecided” cases, inferred from the low agreement of the subjects. We compute the following measure of the uncertainty of subjects regarding the relevance of document j : $H_{tj} = -\sum_{a \in A} (s_{tj}(a) \ln(s_{tj}(a)) / \ln |A|)$, where $s_{tj}(a)$ is the proportion in which the 10 subjects have selected each of the allowed options $a \in A$ for the document j and the conversation fragment t . Then, the relevance value assigned to each option a is computed as $s'_{tj}(a) = s_{tj}(a) \cdot (1 - H_{tj})$, i.e. the raw score weighted by the subjects’ uncertainty.

Scoring a list of documents. Using the ground truth relevance of each document in the reference set, weighted by the subjects’ uncertainty, we will measure the MAP score at rank n of a candidate document result list. We start by computing gr_{tj} , the global relevance value for the conversation fragment t and the document j by giving a weight of 2 for each “relevant” answer (a_2) and 1 for each “somewhat relevant” answer (a_1).

$$gr_{tj} = \frac{s'_{tj}(a_1) + 2s'_{tj}(a_2)}{s'_{tj}(a_0) + s'_{tj}(a_1) + 2s'_{tj}(a_2)} \quad (3)$$

Then we calculate $AveP_{tk}(n)$ the Average Precision at rank n for the conversation fragment t and the candidate list of results of a system k as follows:

$$AveP_{tk}(n) = \sum_{i=1}^n P_{tk}(i) \Delta r_{tk}(i) \quad (4)$$

where $P_{tk}(i) = \sum_{c=1}^i gr_{tl_{tk}(c)} / i$ is the precision at cut-off i in the list of results l_{tk} , $\Delta r_{tk}(i) = gr_{tl_{tk}(i)} / \sum_{j \in l_t} gr_{tj}$ is the change in recall from document in rank $i - 1$ to rank i over the list l_{tk} , and l_t is the reference set for fragment t .

Finally, we compute $MAP_k(n)$, the MAP score at rank n for a system k by averaging the Average Precisions of all the queries at rank n as follows, where $|T|$ is the number of queries: $MAP_k(n) = \sum_{t=1}^{|T|} AveP_{t,k}(n)/|T|$.

Comparing two lists of documents. We compare two lists of documents obtained by two systems k_1 and k_2 through the percentage of the relative MAP at rank n improvement, defined as follows:

$$\%RelativeScore_{k_1,k_2}(n) = (MAP_{k_1}(n) - MAP_{k_2}(n))/(MAP_{k_2}(n)) \times 100. \quad (5)$$

5 Experimental Results

We defined in Section 3 three methods for expanding queries based on the values of λ in Equation 2. The first method has $\lambda = \infty$ and is therefore noted $RQ(\infty)$ – it only uses explicit query keywords, with no refinement. The second one refines explicit queries using the method of the Watson system [8], with $\lambda = 0$, hence noted $RQ(0)$. The third method has $\lambda = 1$ and is noted $RQ(1)$ – this is the novel method proposed here, which expands the query with keywords from the conversation fragment based on their topical similarity to the query. Comparisons are performed over the human-made transcripts and the ASR output, using as a test set the remaining 49 queries not used for development.

5.1 Variation of Fragment Length

We study first the effect of the conversation fragment length on the retrieval results of the three methods, $RQ(1)$, $RQ(\infty)$, and $RQ(0)$. Keyword sets used for expansion are extracted here from the manual transcript of the conversation fragments preceding the 49 queries of the testset. The fragments have a fixed-length per experiment, but we ran our experiments over lengths from 100 to 500 words.

The relative MAP scores of $RQ(1)$ over $RQ(\infty)$ for different ranks n from $n = 1$ to $n = 4$ are provided in Figure 1a, demonstrating the superiority of $RQ(\infty)$ at $n = 1$. However, $RQ(1)$ surpasses $RQ(\infty)$ for ranks 2, 3 and 4. The improvement over $RQ(\infty)$ slightly decreases by increasing the conversation fragment length, likely because of the topic drift in longer fragments. Indeed, when increasing the fragment length, the proposed method $RQ(1)$ behaves more similarly to $RQ(\infty)$ by assigning small weight values (close to zero) to the candidate expansion keywords.

The relative MAP scores of $RQ(1)$ over $RQ(0)$ are reported at ranks $n = 1$ and $n = 2$ in Figure 1b. We do not report values for higher ranks, because of the lack of enough judgments for the retrieval results of $RQ(0)$ among the reference set. The improvements over $RQ(0)$ at rank $n = 1$ are approximately the same for different fragment lengths. They, nevertheless, vary a lot with the length of fragments when looking at rank $n = 2$. The improvement is minimum at length 200 words, likely due to more relevant candidate expansion keywords at this

length compared to the others. As shown above, the average sum of the weights of the expansion keywords is maximized by our method, $RQ(1)$, at length 200 words. When the length decreased or increased from 200 words, the query topics are not completely covered, or the topics are changed respectively. Therefore, the improvement over $RQ(0)$ is increased by decreasing or increasing the length from 200 words at rank $n = 2$, thus showing that $RQ(1)$ is more robust to out-of-topic keywords than $RQ(0)$.

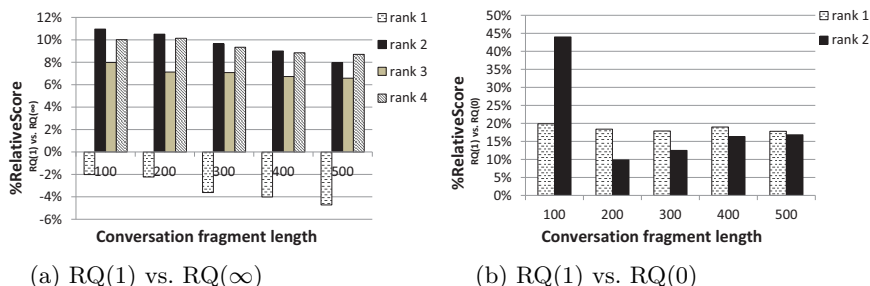


Fig. 1: Relative MAP scores of $RQ(1)$ over the two methods: (a) $RQ(\infty)$ up to rank 4 and (b) $RQ(0)$ up to rank 2, obtained using manual transcripts with fragment lengths of 100, 200, 300, 400 and 500 words. $RQ(1)$ outperforms the other two methods, except for $RQ(\infty)$ at rank $n = 1$.

5.2 Comparisons on Manual Transcripts

We now compare the proposed method $RQ(1)$ with two methods, $RQ(0)$ and $RQ(\infty)$ over the manual transcripts of the 49 conversation fragments, for ranks n from $n = 1$ to $n = 8$, with fragments of 400 words preceding each query. The improvements obtained by $RQ(1)$ over the two others are represented in Figure 2 (the results for 400 words from Fig. 1 are reused in this figure).

The relative MAP scores of $RQ(1)$ over $RQ(\infty)$, except at rank $n = 1$, demonstrate the significant superiority of $RQ(1)$ over $RQ(\infty)$ (between 7% to 11%) up to rank $n = 6$ on average. There are also on average small improvements around 2% over $RQ(\infty)$ at ranks $n = 7$ and 8, because of retrieving the documents which are relevant to both the queries and the fragments by $RQ(\infty)$ (which does not disambiguate the query) at ranks $n = 1, 7$ and 8.

The relative MAP scores of $RQ(1)$ over $RQ(0)$ show significant improvements of more than 15% for ranks $n = 1$ and $n = 2$. Although the scores decrease from rank 2, they remain considerably high at around 7%.

5.3 Comparisons on ASR Transcripts

We applied the explicit query expansion methods to our dataset using the ASR transcripts of the conversations, in order to consider the effect of ASR noise on

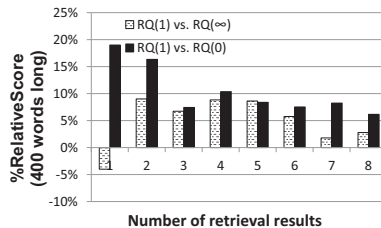


Fig. 2: The relative MAP scores of $RQ(1)$ over the two methods $RQ(\infty)$ and $RQ(0)$ up to rank 8, obtained over the manual transcript of the 49 fragments with 400 words long from the AMI Corpus. $RQ(1)$ surpasses both methods for ranks 2 to 8.

the retrieval results of the expanded queries. We experimented with real ASR transcripts with an average word error rate of 36% and with simulated ones with a noise level varying from 10% to 30%. We computed the average of the scores over five repetitions of the experiment with simulated ASR transcripts, which are randomly generated, and provide below the relative MAP scores of $RQ(1)$ over $RQ(\infty)$ up to rank 3, and over $RQ(0)$ up to rank 2. Moreover, upon manual inspection, we found that there are many relevant documents retrieved in the presence of ASR noise, which have no judgment in the AREX dataset, because they do not overlap with the 31 documents obtained by pooling four methods.

First we compared the two contextual expansion methods, $RQ(0)$ and $RQ(1)$, in terms of the proportion of noisy keywords that each method added to the refined queries. This proportion was computed by summing up the weight value of the keywords used for query refinement that were in fact ASR errors (their set is noted N_j), normalized by the sum of the weight value of all keywords used for the refinement of the query j , as follows:

$$pn_j = \frac{\sum_{kw_i \in (C_j \cap N_j)} m_i^\lambda}{\sum_{kw_i \in C_j} m_i^\lambda} \times 100\% \quad (6)$$

We averaged these values over the 49 explicit queries and the five experimental runs with different random ASR errors. The results shown in Table 1 reveal that the proposed method, $RQ(1)$, is more robust to the ASR noise than $RQ(0)$. We also represent the relative scores of $RQ(1)$ over $RQ(0)$ in Figure 3b. The improvement over $RQ(0)$ increases when the percentage of noise added to the fragments increases, and shows that our method exceeds $RQ(0)$ considerably.

Moreover, we compare the retrieval results of $RQ(1)$ and $RQ(\infty)$ (which does not consider context) in noisy conditions, in Figure 3a. Although the improvement over $RQ(\infty)$ slightly decreases with the noise level, $RQ(1)$ still outperforms $RQ(\infty)$ in terms of relevance, and is generally more robust to ASR noise.

Table 1: Effect of ASR noise on the two query refinement methods $RQ(1)$ and $RQ(0)$ over the 49 explicit queries from our dataset, for a noise level varying from 10% to 30%. $RQ(1)$ is clearly more robust to noise than $RQ(0)$.

Average Percentage of the ASR noise added to queries (%)			
ASR noise	10%	20%	30%
$RQ(1)$	0.78	1.30	2.27
$RQ(0)$	5.64	12.07	21.07

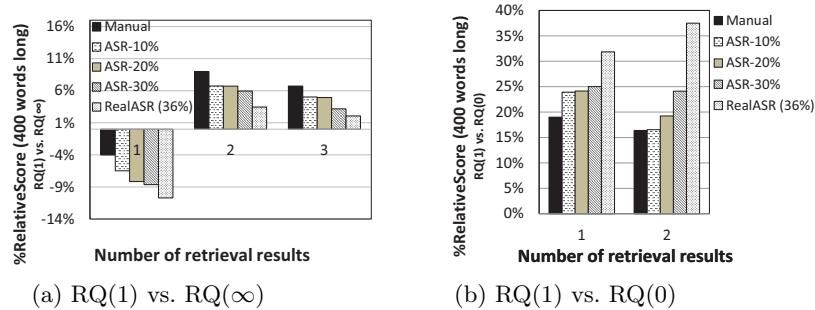


Fig. 3: The relative MAP scores of $RQ(1)$ vs. the other two: (a) $RQ(\infty)$ up to rank 3 and (b) $RQ(0)$ up to rank 2, obtained over the real and simulated ASR transcripts of the AMI Meeting Corpus. The results show $RQ(1)$ outperforms the others.

5.4 Examples of Expanded Queries and Retrieval Results

To illustrate how $RQ(1)$ surpasses the other techniques, we consider an example from one of the queries of our dataset expanded using the ASR transcript of the conversation fragment given in Appendix A. The query bears on the acronym “LCD”. The list of keywords extracted for this fragment is $C = \{\text{“interface”, “design”, “decision”, “recap”, “user”, “control”, “final”, “remote”, “discuss”, “sleek”, “snowman”}\}$. Three keywords (“recap”, “sleek”, and “snowman”) are ASR noise. The proposed method $RQ(1)$ assigns in this case a weight of zero to keywords from ASR noise or those unrelated to the conversation topics. So its corresponding expanded query is $RQ(1) = \{(\text{lcd}, 1.0), (\text{control}, 0.7), (\text{remote}, 0.4), (\text{design}, 0.1), (\text{interface}, 0.1), (\text{user}, 0.1)\}$. $RQ(0)$ assigns a weight 1 to each keyword of the list C and use all of them for expansion regardless of considering their importance to the query, and $RQ(\infty)$ does not expand the query.

The retrieval results up to rank 8 obtained for the three methods are displayed in Table 2. All the results of $RQ(1)$ are related to “liquid-crystal display” which is the correct interpretation of the query, while $RQ(\infty)$ provides three irrelevant documents: “Lowest common denominator” (a mathematic function), “LCD Soundsystem” (an American dance band), and “Pakalitha Mosisili” (a politician at Lesotho Congress for Democracy). None of the results provided by $RQ(0)$ addresses “liquid-crystal display” directly due to irrelevant keywords

added to the query from the out of query topics of the users conversation or the ASR noise.

Table 2: Example of retrieved Wikipedia pages using the three expansion methods. Results of $RQ(1)$ are more relevant to both the query and conversation topics.

	Document Results		
Rank	$RQ(1)$	$RQ(\infty)$	$RQ(0)$
1	Liquid-crystal display	Liquid-crystal display	User interface
2	Backlight	Backlight	X Window System
3	Liquid-crystal display television	Liquid-crystal display television	Usability
4	Thin-film transistor	Lowest common denominator	Wii Remote
5	LCD projector	LCD Soundsystem	Walkman
6	LG Display	LCD projector	Information hiding
7	LCD shutter glasses	Pakalitha Mosisili	Screensaver
8	Universal remote	LG Display	Apple IIc

6 Conclusion

The best method for contextual query refinement appears to be the proposed method $RQ(1)$ over both manual and ASR transcripts. Although, $RQ(\infty)$ outperforms $RQ(1)$ at rank $n = 1$, the scores of $RQ(1)$ show a significant improvement up to rank $n = 8$ over manual transcripts and up to rank $n = 3$ over ASR ones. Moreover, $RQ(1)$ outperforms $RQ(0)$ on both manually-made and ASR transcripts. The scores also demonstrate that the proposed method $RQ(1)$ is robust to various ASR noise levels and to the length of the conversation fragment used for expansion. The dataset accompanying these experiments, AREX, is public and can be used for future comparisons of conversational query-based retrieval systems.

In future work, we plan to setup experiments with human subjects in a scenario that encourages them to use spoken queries during a task-oriented conversation, and confirm the superiority of our proposal with respect to the state-of-the-art through evaluation on a deployed system.

Acknowledgments. The authors are grateful to the Swiss National Science Foundation (SNSF) for its financial support through the IM2 NCCR on Interactive Multimodal Information Management (see www.im2.ch), as well as to the Hasler Foundation for the REMUS project (n. 13067, Re-ranking Multiple Search Results for Just-in-Time Document Recommendation).

References

1. Alidin, A.A., Crestani, F.: Context modelling for Just-In-Time mobile information retrieval (JIT-MobIR). *Pertanika Journal of Science & Technology* 21(1), 227–238 (2013)
2. Attar, R., Fraenkel, A.S.: Local feedback in full-text retrieval systems. *Journal of the ACM (JACM)* 24(3), 397–417 (1977)
3. Bai, J., Song, D., Bruza, P., Nie, J.Y., Cao, G.: Query expansion using term relationships in language models for information retrieval. In: *Proc. of the 14th ACM Int. Conf. on Information and Knowledge Management*. pp. 688–695 (2005)
4. Bhogal, J., Macfarlane, A., Smith, P.: A review of ontology based query expansion. *Information Processing and Management* 43(4), 866–886 (2007)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3, 993–1022 (2003)
6. Bohus, D., Horvitz, E.: Models for multiparty engagement in open-world dialog. In: *Proc. of the SIGDIAL 2009 Conference: The 10th Annual Meeting of the Special Interest Group on Discourse and Dialogue*. pp. 225–234 (2009)
7. Boyd-Graber, J., Chang, J., Gerrish, S., Wang, C., Blei, D.: Reading tea leaves: How humans interpret topic models. In: *Proc. of 23rd Annual Conf. on Neural Information Processing Systems*. pp. 288–296 (2009)
8. Budzik, J., Hammond, K.J.: User interactions with everyday applications as context for Just-In-Time information access. In: *Proc. of the 5th int. conference on intelligent user interfaces*. pp. 44–51 (2000)
9. Carletta, J.: Unleashing the killer corpus: Experiences in creating the multi-everything AMI Meeting Corpus. *Language Resources and Evaluation Journal* 41(2), 181–190 (2007)
10. Carpineto, C., De Mori, R., Romano, G., Bigi, B.: An information-theoretic approach to automatic query expansion. *ACM Transactions on Information Systems (TOIS)* 19(1), 1–27 (2001)
11. Carpineto, C., Romano, G.: A survey of automatic query expansion in information retrieval. *ACM Computing Surveys (CSUR)* 44(1), 1–50 (2012)
12. Chirita, P.A., Firan, C.S., Nejdl, W.: Personalized query expansion for the web. In: *Proc. of 30th annual int. ACM SIGIR conf. on Research and development in IR*. pp. 7–14 (2007)
13. Diaz, F., Metzler, D.: Improving the estimation of relevance models using large external corpora. In: *Proc. of 29th annual int. ACM SIGIR conf. on Research and development in IR*. pp. 154–161 (2006)
14. Garner, P.N., Dines, J., Hain, T., El Hannani, A., Karafiat, M., Korchagin, D., Lincoln, M., Wan, V., Zhang, L.: Real-time ASR from meetings. In: *Proc. of the 10th Annual Conference of the International Speech Communication Association*. pp. 2119–2122 (2009)
15. Habibi, M., Popescu-Belis, A.: Diverse keyword extraction from conversations. In: *Proc. of the 51st Annual Meeting of the Association for Computational Linguistics*. pp. 651–657 (2013)
16. Habibi, M., Popescu-Belis, A.: Keyword extraction and clustering for document recommendation in conversations. *IEEE/ACM Transactions on Audio Speech and Language Processing* 23(4), 746 – 759 (2015)
17. Hain, T., Burget, L., Dines, J., Garner, P.N., El Hannani, A., Huijbregts, M., Karafiat, M., Lincoln, M., Wan, V.: The AMIDA 2009 meeting transcription system. In: *Proc. of INTERSPEECH*. pp. 358–361 (2010)

18. Hoffman, M.D., Blei, D.M., Bach, F.: Online learning for Latent Dirichlet Allocation. In: Proc. of 24th Annual Conference on Neural Information Processing Systems. pp. 856–864 (2010)
19. Lavrenko, V., Croft, W.B.: Relevance based language models. In: Proc. of 24th annual int. ACM SIGIR conf. on Research and development in IR. pp. 120–127 (2001)
20. McCallum, A.K.: MALLET: A machine learning for language toolkit (2002), <http://mallet.cs.umass.edu>
21. Park, L.A.F., Ramamohanarao, K.: Query expansion using a collection dependent probabilistic latent semantic thesaurus. In: Advances in Knowledge Discovery and Data Mining, pp. 224–235 (2007)
22. Popescu-Belis, A., Yazdani, M., Nanchen, A., Garner, P.N.: A speech-based just-in-time retrieval system using semantic search. In: Proc. of the 49th Annual Meeting of the ACL, Demonstrations. pp. 80–85 (2011)
23. Popescu-Belis, A., Boertjes, E., Kilgour, J., Poller, P., Castronovo, S., Wilson, T., Jaimes, A., Carletta, J.: The AMIDA Automatic Content Linking Device: Just-in-time document retrieval in meetings. In: Proc. of MLMI 2008 (Machine Learning for Multimodal Interaction), LNCS 5237. pp. 272–283. Utrecht (2008)
24. Robertson, S.E., Walker, S., Beaulieu, M., Willett, P.: Okapi at TREC-7: automatic ad hoc, filtering, VLC and interactive track. NIST Special Publication SP pp. 253–264 (1999)
25. Rocchio, J.J.: Relevance feedback in information retrieval. In: Salton, G. (ed.) The SMART Retrieval System: Experiments in Automatic Document Processing, chap. 14, pp. 313–323. Prentice-Hall, Englewood Cliffs, NJ (1971)
26. Salton, G., Buckley, C.: Improving retrieval performance by relevance feedback. Readings in Information Retrieval 24, 5 (1997)
27. Voorhees, E.M., Harman, D.K. (eds.): TREC: Experiment and Evaluation in Information Retrieval. MIT Press, Cambridge, MA (2005)
28. Wang, D., Hakkani-Tur, D., Tur, G.: Understanding computer-directed utterances in multi-user dialog systems. In: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. pp. 8377–8381 (2013)
29. Xu, J., Croft, W.B.: Query expansion using local and global document analysis. In: Proc. of 19th annual int. ACM SIGIR conf. on Research and development in IR. pp. 4–11 (1996)
30. Xu, J., Croft, W.B.: Improving the effectiveness of information retrieval with local context analysis. ACM Trans. on Information Systems (TOIS) 18(1), 79–112 (2000)

Appendix A : Transcript of a Conversation fragment from the AMI Meeting Corpus

The following transcript of a four-party conversations (speakers noted A through D) around 150 words used to refine the explicit query which is asked at the end of the conversation fragment by the participant. They are talking about designing a remote control.

A: Okay well All sacked Right Oh i see a kind of detailed design meeting Um We're gonna discuss the the look-and-feel design user interface design and We're gonna evaluate the product And For The end result of this meeting has to be a

decision on the details of this remote control like a sleek final decision Uh-huh Um i'm then i'm gonna have to specify the final design In the final report.

B: Yeah So um just from from last time To recap So we're gonna have a snowman shaped remote control with no LCD display new need for tap bracket so if you're gonna be kinetic power and battery Uh with rubber buttons maybe park lighting the buttons with um Internal LEDs to shine through the casing Um hopefully a job done and incorporating the slogan somewhere as well I think i missed Okey Um so Uhuh If you want to present your prototype Go ahead.

C: I need more information about LCD.