Analyzing Interactive QA Dialogues using Logistic Regression Models

Manuel Kirschner[1], Raffaella Bernardi[1], Marco Baroni[2], and Le Thanh Dinh[1,3]

[1] KRDB, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy
{kirschner, bernardi}@inf.unibz.it
[2] Center for Mind/Brain Sciences, University of Trento, Italy
marco.baroni@unitn.it
[3] Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics, Charles University in Prague, Czech Republic
lethanh.dinh@stud-inf.unibz.it

Abstract. With traditional Question Answering (QA) systems having reached nearly satisfactory performance, an emerging challenge is the development of successful Interactive Question Answering (IQA) systems. Important IQA subtasks are the identification of a dialogue-dependent typology of Follow Up Questions (FU Qs), automatic detection of the identified types, and the development of different context fusion strategies for each type. In this paper, we show how a system relying on shallow cues to similarity between utterances in a narrow dialogue context and other simple information sources, embedded in a machine learning framework, can improve FU Q answering performance by implicitly detecting different FU Q types and learning different context fusion strategies to help re-ranking their candidate answers.

1 Introduction

It is widely acknowledged that answering Follow Up Questions (FU Qs), viz., questions uttered after some other interaction, is a different task than answering isolated questions. Hence, Interactive Question Answering (IQA) systems have to tackle different challenges than Question Answering (QA) systems. The latter can rely only on the question to extract the relevant keywords. The former should take the previous interactions into consideration and achieve some form of context fusion, i.e., identify the information in the previous interactions that are relevant for processing the FU Q and answering it properly [1]. A first crucial context-dependent distinction is among topic shift and topic continuation FU Qs. These types of FU Qs might require different processing strategies. Hence, important sub-tasks within the IQA community are the identification of a typology of questions, automatic detection of the identified types, and the development of different context fusion strategies for each type [2, 3, 1].

In this paper, we aim to show how a system based on shallow cues to similarity between utterances in a narrow dialogue context and other simple information sources, embedded in a machine learning framework, can improve FU Q answering performance, and how such system can also implicitly detect different FU Q types and learn different answer ranking strategies to cope with them.

A further innovative aspect of our work is that instead of using the artificial Text Retrieval Conference (TREC) data most IQA systems are evaluated on (TREC 2001, TREC 2004), we train and test our system on real user questions collected via an on-line chatter Bot on a closed domain.

Section 2 places our proposal in the broader picture of IQA research. We describe our dialogue corpus in Section 3, and we introduce our general modeling framework and the features we use in Section 4. Versions of the model that do and do not take context into account are evaluated in Section 5. We conclude with an error analysis that points at directions for future improvements in Section 6.

2 Related Work

The importance of evaluating IQA against real user questions and the need to consider preceding system answers has already been emphasized [2–4, 1]. In these earlier studies, dialogue corpora were collected via Wizard of Oz experiments and/or by giving specific tasks to the users. The corpus of dialogues we deal with consists of real logs in which the users were chatting with a Bot to obtain information in a help-desk scenario.

[4] and [5] look for salient transitions among utterances in a given context. To this end, they exploit deep semantic analyses to detect Argument-Predicate structure [4] and Centering Theories features [5]. Both [4] and [2] take Argument-Predicate structures as the base of semantic networks used to model context interaction and guide context fusion. In [2], the system relies also on deep and chunk reasoning on an external ontology. In this paper, we avoid any form of deep analysis of this sort.

Fine grained typologies of questions have been suggested [2, 3, 5], and different processing strategies have been proposed for the identified types. We consider the basic distinction between topic shift and topic continuation, and we propose a generalized linear model framework in which this distinction is automatically detected and used to improve the answering performance.

Our work is closely related to [1], that presents a question classifier that detects topic shifts and topic continuations by exploiting utterance similarity measures. However, we go two steps further by not requiring training data annotated for question type, and directly using the question classification cues to improve answer re-ranking performance.

Similarly to other work in QA [6, 7], we use corpus-based similarity measures, with the important innovation that we extend them to similarity with previous utterances in the context. Finally, there is a large literature on using supervised machine learning for various aspects of QA, including question re-ranking [8]. Again, as far as we know we are the first to propose a supervised classifier that takes the previous dialogue into account for answering FU Qs.

3 Data

Most IQA systems have been trained and evaluated over the TREC (2001, 2004) data-sets, that consist of several sessions of related questions, the first of which sets the topic. Hence, there are no topic shifts, apart from the artificial ones at the first question of each session, if one considers the whole set as a single interaction [1]. Furthermore, there are no answers to rely on, and the questions were collected by TREC evaluators, i.e. they are not questions asked by users genuinely interested in the interaction. Hence, their nature is rather artificial. To overcome these limitations, we have been collecting a corpus of human-machine interactions with a Bot that provides information about a university library, picking a canned-text answer from a set of 484 information statements produced by the librarians (thus, the system faces the task that is often called *answer/passage re-ranking* in the QA literature).

The corpus consists of 139 4-turn snippets of human-machine interactions. We limit ourselves to four turns since there is evidence [2, 9] that in most cases the previous two turns (Q_1 and A_1) contain enough information to process the FU Q (Q_2) and select its answer (A_2). Moreover, this makes our classifier well suited for practical applications, as it only relies on cues extracted from a fixed window of 3 utterances (Q_1, A_1 and Q_2). We refer to these 3 utterances as “the context”. Note that A_1 correctly answers Q_1 in all snippets kept for this study.

Contexts in the corpus were manually classified by one of the authors as either topic shifts or topic continuations (57 and 82 cases, respectively). Below we report two sample topic continuation and one topic shift snippet examples. In a closed domain, the difference between the two types is rather fine-grained. We have considered topic continuations all those cases where the FU Qs are natural continuations of the conversation. The interactions are “task-related” information seeking dialogues, hence if two questions pertain to tasks where one is a typical development of the other, they are counted as “topic continuations” (example 1, where the actions are “borrow” and “search”). Questions that are about an entity introduced in Q_1 or A_1 (example 2) are also topic continuations. Shift/continuation annotation is a delicate task, making unsupervised, implicit detection of context (as in the experiments below) particularly desirable.

Example 1: Topic continuation (related tasks)

User How many items can I borrow at once? (Q_1)

System You may borrow up to 40 items at time, independently of the type of medium. (A_1)

User Is there a help for searching books? (Q_2)

System To find books I’d recommend searching in our on-line catalogue (OPAC). (A_2)

Example 2: Topic continuation (entities in A_1)

User How can I borrow a book? (Q_1)

System You can either get the item yourself or request it via our delivery service. (A_1)

User What are the requirements for it? (Q_2)

System If you want an item to be sent to another Library branch simply click on the blue ORDER button to reserve it [. . .] (A_2)

Example 3: Topic shift

User Where can I find a public phone? (Q_1)

System Public phones are situated in the main entrance area. (A_1)

User I have lost the Library Card. What can I do? (Q_2)

System If you've lost your Library Card please tell my colleagues at the information desk. They will lock your account. (A_2)

Like in TREC data, the topic continuation FU Qs can contain ellipses, e.g., Q_1: *Where can I find design books?* Q_2: *and dvd?*. Differently from TREC, both Q_1 and Q_2 could be just keywords, may contain noisy information such as typos or bad grammar, and could be very similar: either the user is trying to refine the question (the answer is correct but not what they want to know) or the topic is further explored by moving the focus of attention on a new related entity or a new related action: Q_1: *Could you recommend me some book?* Q_2: *Could you recommend me some novel?*. These kinds of interactions seem typical of real user data and they have been noticed also in other corpora of this type [2, 1].

4 Model

Our goal is, given a FU Q (Q_2 in our dialogue snippets), to pick the best answer from the fixed A_2 candidate set, by assigning a score to each candidate, and ranking them by this score. Different context types might require different answer picking strategies. Thus, we specify both A_2 *(identification) features*, aiming at selecting the correct A_2 among candidates, and *context (identification) features*, that aim at characterizing the context. The A_2 identification features measure the similarity between an utterance in the context (e.g., Q_2) and a candidate A_2. Context features measure the similarity between pairs of utterances in the context (e.g., Q_1 and Q_2). They do not provide direct information about A_2, but might cue a special context (say, an instance of topic shift) where we should pay more attention to different A_2 identification features (say, less attention to the relation between Q_2 and A_2, and more to the one between A_1 and A_2).

We implement these ideas by estimating a generalized linear model from training data to predict the probability that a certain A_2 is correct given the context. In this model, we enter A_2 features as main effects, and context features in interactions with the former, allowing for differential weight assignment to the same A_2 features depending on the values of the context features.

4.1 A_2 features

Most of our A_2 features measure the similarity between a context utterance and A_2. The intuition is that the correct A_2 is similar to the context.

Lexical Similarity (*lexsim*): If two utterances (e.g., Q_2 and A_2) share some terms, they are similar; the more *discriminative* the terms they share, the more similar the utterances. Implemented by representing the utterances as vectors with the words they contain as dimensions. The value of each dimension is the *tf.idf* [10] of the corresponding word in the general ukWaC corpus,[4] calculated as:

$$\text{tf.idf}(w) = \sqrt{count(w)}\sqrt{\log \frac{|D|}{|D_w|}}$$

where *count(w)* returns the number of occurrences of the word in the corpus, $|D|$ is the number of corpus documents, and $|D_w|$ the number of documents containing the word. Weighting by *tf.idf* favours more discriminative terms, that occur in a restricted number of documents (e.g., *library* is less frequent but more discriminative than *long*). Similarity is quantified by the cosine of the angle between the vectors representing the two utterances being compared:

$$cos(\boldsymbol{u}_1, \boldsymbol{u}_2) = \frac{\boldsymbol{u}_1 \cdot \boldsymbol{u}_2}{||\boldsymbol{u}_1||\;||\boldsymbol{u}_2||}$$

Distributional Similarity (*distsim*): Two utterances are similar not only if they share the same terms, but also if they share similar terms (e.g., *book* and *journal*). Term similarity is estimated on the ukWaC corpus, by representing each content word (noun, verb, adjective) as a vector that records its corpus co-occurrence with other content words within a 5-word span. Raw co-occurrence counts are weighted by pointwise mutual information, that dampens the impact of frequent words [10]:

$$\text{mi}(w_1, w_2) = \log_2 \frac{p(w_1 \& w_2)}{p(w_1)\; p(w_2)}$$

where $p(w_1 \& w_2)$ is estimated by the proportion of $w_1 \& w_2$ co-occurrences over the total co-occurrence count for all pairs, and $p(w_*)$ from the marginal frequencies. Distributionally similar words, such as *book* and *journal* will have similar vectors [11]. An utterance is represented by the sum of the normalized distributional vectors of the words it contains. Similarity between utterances is again quantified by the cosine of their vector representations. We tried a few variants (20-word spans, raw frequency or log-likelihood ratio instead of mutual information, max similarity between nouns or verbs instead of summed vectors), but the resulting models were either highly correlated to the one we are reporting here, or they performed much worse in preliminary experiments. We leave it to further work to devise more sophisticated ways to measure the overall distributional similarity among utterances [12].

Semantic similarity (*semsim*): We try to capture the same intuition that similar utterances contain similar words, but we measure similarity using WordNet [13].[5] We experimented with most of the WordNet similarity measures that

[4] http://wacky.sslmit.unibo.it/

[5] We use the WordNet::Similarity package: http://wn-similarity.sourceforge.net/.

were used by [1], settling for the Lin measure, that gave the best results across the board:

$$\text{linsim}(w_1, w_2) = \frac{2\ ic(lcs(w_1, w_2))}{ic(w_1) + ic(w_2)}$$

where $lcs(w_1, w_2)$ is the lowest common subsumer of synsets containing the two words in the WordNet hierarchy (we pick the synsets maximizing the score), and $ic(x)$ (the information content) is given by $-\log(p(x))$. Probabilities are estimated from the sense-annotated SemCor corpus coming with the WordNet::Similarity package. Following [1], the similarity between two utterances is determined by matching the words so as to maximize pairwise similarities, while normalizing for sentence length.

Action sequence (*action*): The binary action feature indicates whether two turns are associated with the same action, and thus represent an action sequence. For identifying the action associated with each A_2, we hand-annotated each of the 484 answer candidates with one of 25 relevant actions (borrowing, delivering, etc.). The action(s) associated with the other turns (Q_1, A_1, Q_2) are automatically assigned by looking for strings that match words that we think represent one of the 25 actions.

For each feature type, we compute its value for both the $A_1.A_2$ and $Q_2.A_2$ interplays: we will refer to them below as the *far* and *near* features, respectively (far and near in terms of distance of the compared utterance from A_2. We ignore $Q_1.A_2$ features for now). By crossing the measure types and the considered interplays, we obtain 8 A_2 features ($far.lexsim$, $near.lexsim$, $far.distsim$, etc.).

4.2 Context features

Topic shifts across turns have been generally recognized as the main contextual factor affecting the relative role of context in FU Q answering [1]. If Q_2 continues the previous topic, then the previous context should still be relevant to A_2. If the topic shifted, the A_2 selection strategy should focus on the most recent turn only (i.e., Q_2 itself).

In order to verify that adding topic continuation information to the model does indeed help A_2 prediction, we used our manual coding of contexts for whether they contain a topic shift or not (*topshift*). This feature is of limited practical utility in a real life system: topic change or persistence should be detected by automated means.

A simple way to capture the notion of topic continuity is in terms of similarity between Q_2 and each of the preceding utterances (the less similar Q_2 is to Q_1 or A_1, the more likely it is that the topic shifted). Thus, the same utterance similarity measures that, when used to compare other utterances to A_2, serve as A_2 identification features, can be treated as continuous approximations to a topic shift when applied to the $Q_1.Q_2$ and $A_1.Q_2$ interplays. Since we defined 3 similarity measures (lexsim, distsim and semsim), we obtain 6 more context identification features ($Q_1.Q_2.lexsim$, $A_1.Q_2.lexsim$, $Q_1.Q_2.distsim$, etc.).

We tried various strategies to combine the topic approximation cues into composite measures (e.g., by defining "profiles" in terms of high and low $Q_1.Q_2$ and $A_1.Q_2$ scores, or by defining multiple interaction terms), but they did not improve on the simpler context features, and we do not report their performance here.

4.3 Logistic regression

Logistic regression models (LRMs) are generalized linear models that describe the relationship between features (independent variables) and a binary outcome [14]. Our logistic regression equations, which specify the probability for a particular answer candidate A_2 being correct, depending on the learned intercept β_0, the other β coefficients (representing the contribution of each feature to the total answer correctness score), and the feature values $x_1, \ldots, x_k$ (which themselves depend on a combination of Q_1, A_1, Q_1 or A_2) have the form:

$$\text{Prob}\{\text{answerCorrect}\} = \frac{1}{1 + \exp(-X\hat{\beta})}$$

$$\text{where } X\hat{\beta} = \beta_0 + (\beta_1 x_1 + \cdots + \beta_k x_k)$$

Context typology is implicitly modeled by *interaction* terms, given by the product of an A_2 feature and a context feature (when we enter an interaction term with a context feature, we also always introduce the corresponding main effect). An interaction term provides an extra β to assign a differential weight to an A_2 feature depending on the value(s) of a context feature. In the simplest case of interaction with a binary 0-1 feature, the interaction β weight is only added when the binary feature has the 1-value.

We estimate the model parameters (the beta coefficients $\beta_1, \ldots, \beta_k$) using maximum likelihood estimation. Moreover, we put each model we construct under trial by using an iterative backward elimination procedure that takes off all those terms whose removal does not cause a significant drop in goodness-of-fit. All the results we report below are obtained with models that underwent this trimming procedure.

5 Evaluation

We match each of the 139 contexts (Q_1, A_1 and Q_2 sequences) in our dialogue corpus with each of the 484 A_2s in our pre-canned answer repository. Since the corpus had been pre-annotated for what is the (single) correct A_2 for each context, this produces 483 negative examples and 1 positive example for each context, that can be used to estimate a LRM.[6] We rank the $Q_1.A_1.Q_2.A_2$ 4-tuples constructed in this way in terms of the probability they are assigned by

[6] Our experiments with random sampling of the majority class (i.e., the negative training examples) did not improve model performance.

the estimated LRM, and we look at the rank of the *correct* A_2 for each context. We use "leave-one-out" cross validation by predicting the ranks of the A_2s given each context with a model that was trained on the remaining 138 contexts. The lower the average rank of the correct A_2 across contexts, the better the model predictions. In the tables below, we report these average ranks. When we report statistical results about the relative quality of models, these are based on paired Mann-Whitney tests across the 139 ranks. Statistical significance claims are based on the $p < 0.05$ level, after correction for multiple comparisons.

We will first look at models that only look at the relation between context utterances and A_2 ("main effects only" models), and then at models that also exploit information about the relation between context utterances, to approximate a typology of contexts ("interaction" models).

5.1 Main effects only models

We enter each feature from Section 4.1 at a time in separate models (e.g., the **.semsim* models) and combine them (the **.combined* models). Moreover, we look at $Q_2.A_2$ features (*near.**, in the sense that we look at the nearest context element with respect to A_2), $A_1.A_2$ features (*far.**), and both (*complete.**). Table 1 summarizes our first set of experiments. All *near* and *far* single feature models, except $far.action$, perform significantly better than *baseline*, and in general the near context is more informative than the far one (group 1 vs. group 2). Combining different knowledge sources helps: *near.combined* is significantly better than the best non-combined model, *near.distsim*. Combining features only helps marginally when we look at the *far* setting (compare groups 2 and 4). The *complete.combined* model (group 6) significantly outperforms the corresponding best single feature model of group 5 *complete.distsim*, but its performance is not distinguishable from the one of *near.combined*.

This first batch of analyses shows that the proposed features have a significant impact on correct A_2 prediction, that combining different knowledge sources considerably improves performance and that, if we do not consider an interaction with context type, the far features (comparing A_1 and A_2) are not helpful. In the next step, we will work with the two best main-effects-only models obtained so far, namely *near.combined* and *complete.combined* (groups 3 and 6 in Table 1), and we will investigate their behaviour when we add interaction terms that try to capture different context types.

5.2 Models with an interaction

Table 2 reports the results obtained with models that add an interaction term that should capture contextual development patterns, and in particular the presence of a topic shift. As discussed in Section 4.2 above, depending on whether there is a topic shift, we should assign different weights to *near* and *far* features. Thus, we predict that the presence of an interaction term marking topic development should help *complete.*$*$ models (that encode both *near* and *far* features), but not *near.*$*$ models.

Group	Description	Model name	Mean rank	SD
0	A_2 picked at random	*baseline*	235.0	138.2
1	Single $Q_2.A_2$ feature	*near.lexsim*	80.4	106.0
		near.distsim	74.4	113.5
		near.semsim	101.2	115.2
		near.action	178.1	156.8
2	Single $A_1.A_2$ feature	*far.lexsim*	164.6	138.3
		far.distsim	157.3	145.3
		far.semsim	152.3	135.3
		far.action	231.5	153.5
3	Combined $Q_2.A_2$ features	*near.combined*	**57.6**	93.4
4	Combined $A_1.A_2$ features	*far.combined*	141.7	130.5
5	Single $Q_2.A_2$ and $A_1.A_2$ features	*complete.lexsim*	75.3	109.4
		complete.distsim	72.6	108.9
		complete.semsim	103.2	111.0
		complete.action	(= *near.action*)	
6	Combined $Q_2.A_2$ and $A_1.A_2$ features	*complete.combined*	**58.6**	97.4

Table 1. Mean ranks of correct A_2 out of 484 answer candidates in main effects only models

Group	Description	Model name	Mean rank	SD
7	*near.combined*: int. with manual *topshift*	*near.combined* × *topshift*	57.5 (= *near.combined*)	93.4
8	*complete.combined*: int. with manual *topshift*	*complete.combined* × *topshift*	**54.3**	93.2
9	*complete.combined*: interaction with approximation feature	*complete.combined* × $Q_1.Q_2$*.lexsim*	55.5	91.7
		complete.combined × $A_1.Q_2$*.lexsim*	56.7	93.8
		complete.combined × $Q_1.Q_2$*.distsim*	57.1	98.2
		complete.combined × $A_1.Q_2$*.distsim*	58.4	96.4
		complete.combined × $Q_1.Q_2$*.semsim*	60.0	100.8
		complete.combined × $A_1.Q_2$*.semsim*	**54.3**	90.9
		complete.combined × $Q_1.Q_2$*.action*	55.8	94.4
		complete.combined × $A_1.Q_2$*.action*	56.9	96.1

Table 2. Mean ranks of correct A_2 out of 484 answer candidates in interaction models

Predictor	β coef.	SE	z value	Pr(>|z|)	Predictor (cont'd)	β coef.	SE	z value	Pr(>|z|)
intercept	-11.6	0.7	-15.9	0.000	*far.distsim*	2.7	1.1	2.3	0.019
near.lexsim	6.5	1.0	6.4	0.000	*far.action*	0.0	0.3	-0.1	0.923
near.distsim	1.5	0.9	1.7	0.092	*topshift*	1.1	1.1	1.0	0.317
near.semsim	2.3	0.5	4.8	0.000	*topshift* × *near.distsim*	2.7	1.2	2.4	0.018
near.action	0.9	0.2	4.3	0.000	*topshift* × *far.distsim*	-3.2	1.6	-2.0	0.040
far.lexsim	3.2	0.6	5.1	0.000	*topshift* × *far.action*	-1.4	0.6	-2.5	0.014

Table 3. Retained predictors, model *complete.combined* ∗ *topshift* (Table 2, group 8)

The prediction is fully confirmed by comparing the model in group 3 of Table 1 to the one in group 7 of Table 2, on the one hand, and the model in group 6 of Table 1 to the model in group 8 of Table 2 on the other. In both cases, we are adding interaction terms for each of the main effects (the A_2 prediction features) crossed with the binary feature recording manual annotation of the presence of a topic shift. For the *near* model (that only takes the $Q_2.A_2$ relations into account), there is *no* improvement whatsoever from adding the interaction. Indeed, the backward elimination procedure we use when estimating the model drops all interaction terms, leading to an estimated model that is identical to the one in group 3 (main effects only). Vice versa, some interaction terms are preserved in the model of group 8, that improves from the corresponding interaction-less model, from a mean rank of 57 to a mean rank of 54 (although the rank improvement itself is not statistically significant).

Table 3 reports the coefficients of the estimated group 8 model (trained, for these purposes only, on the complete corpus). The four *near* features have a major positive effect on the odds of an answer candidate being correct. Also, the first two far features listed in the table have positive effects. We interpret the retained interaction terms with $topshift$ (the last three rows) as follows. If Q_2 is a topic shift, the weight given to *near.distsim* (a term measuring the similarity with Q_2) has an extra positive effect, while the weight given to semantic similarity and the repetition of the same action between A_1 and A_2 is significantly decreased, since in a topic shift the earlier context should be (nearly) irrelevant. These effects are in line with our hypothesis about topic shift FU Q processing. Thus, we confirm that knowing about topic shifts helps, and that it helps because we can assign different weights to *far* and *near* relations depending on whether the topic continues or changes.

Having established this, we now ask whether the manually coded *topshift* variable can be replaced by an automatically computed feature that captures topic shifting in terms of the similarity between context utterances (Q_1 vs. Q_2 or A_1 vs. Q_2). By looking at the results in group 9 of Table 2, we see that in fact one of these models (interaction with $A_1.Q_2.semsim$) performs as well as the model using the hand-annotated $topshift$ feature, while outperforming the *complete.combined* model (barely missing full statistical significance, with $p = 0.05457$). Interpretation of the coefficients is harder in this case, since we deal with a continuous interaction term, but the main patterns are as for the model in group 8. Not only keeping track of topic development helps answer (re-)ranking, but a simple automatically assigned feature (WordNet-based similarity between A_1 and Q_2) is as good as manual annotation to cue basic topic development.

6 Conclusion

From our quantitative evaluation via LRM we can conclude that to answer FU Qs asked in a real help-desk setting, some form of shallow context detection and fusion should be considered. In particular, the system answer preceding the FU Q seems to play an important role, especially because its similarity to the FU

Q can cue a topic shift, that in turn requires a different context fusion strategy (more weight to FU Q, less to the preceding context).

Our shallow cues, though promising, need further refinement, in particular to deal with the following problems particular to real user interactions: (I) some Q_1 and Q_2 are quite similar, which could happen for example when users are not satisfied with the answer (even if it was correct for the question that was asked) and hence rephrase the question by, e.g., using a more specific entity, or they even repeat the same question in the hope to obtain a better answer; (II) Q_2 contains only *WH VERB ENTITY*, and the verb is a factotum verb. Both (I) and (II) require further investigation and seem to ask for more structured cues than those explored in this paper.

References

1. Yang, F., Feng, J., Di Fabbrizio, G.: A data driven approach to relevancy recognition for contextual question answering. In: Interactive Question Answering Workshop. (2006)
2. Bertomeu, N.: A Memory and Attention-Bases Approach to Fragment Resolution and its Application in a Question Answering System. PhD thesis, Universität des Saarlandes (2007)
3. Van Schooten, B.w., Op den Akker, R., Rosset, S., Galibert, O., Max, A., Illouz, G.: Follow-up question handling in the imix and ritel systems: A comparative study. Nat. Lang. Eng. **15**(1) (2009) 97–118
4. Chai, J.Y., Jin, R.: Discourse structure for context question answering. In: Proceedings of the Workshop on Pragmatics of Question Answering at HLT-NAACL 2004. (2004)
5. Sun, M., Chai, J.: Discourse processing for context question answering based on linguistic knowledge. Know.-Based Syst. **20**(6) (2007) 511–526
6. Burek, G., De Roeck, A., Zdrahal, Z.: Hybrid mappings of complex questions over an integrated semantic space. In: Proceedings of the 16th International Workshop on Database and Expert Systems Applications (DEXA'05), IEEE (2005)
7. Tomás, D., Vicedo, J., Bisbal, E., Moreno, L.: Experiments with lsa for passage re-ranking in question answering. In: CLEF Proceedings. (2006)
8. Moschitti, A., Quarteroni, S.: Kernels on linguistic structures for answer extraction. In: Proceedings of ACL-08: HLT, Short Papers. (2008) 113–116
9. Kirschner, M., Bernardi, R.: An empirical view on iqa follow-up questions. In: Proc. of the 8th SIGdial Workshop on Discourse and Dialogue. (2007)
10. Manning, C.D., Schütze, H.: Foundations of statistical natural language processing. MIT Press, Cambrdige (1999)
11. Sahlgren, M.: The Word-Space Model. Dissertation, Stockholm University (2006)
12. Mihalcea, R., Corley, C., Strapparava, C.: Corpus-based and knowledge-based measures of text semantic similarity. In: Proceedings of AAAI. (2006)
13. Fellbaum, C., ed.: WordNet: An electronic lexical database. MIT Press, Cambrdige (1998)
14. Agresti, A.: Categorical data analysis. Wiley, New York (2002)