Dialogue and Discourse 3(2) (2012) 125–146 doi: 0.5087/dad.2012.206

# Creating Conversational Characters Using Question Generation Tools

**Xuchen Yao** *(Now at Johns Hopkins University)* XUCHEN@CS.JHU.EDU
**Emma Tosch** *(Now at the University of Massachusetts Amherst)* ETOSCH@GMAIL.COM
**Grace Chen** *(Now at California State University, Long Beach)* GRACE.CHEN@STUDENT.CSULB.EDU
**Elnaz Nouri** NOURI@ICT.USC.EDU
**Ron Artstein** ARTSTEIN@ICT.USC.EDU
**Anton Leuski** LEUSKI@ICT.USC.EDU
**Kenji Sagae** SAGAE@ICT.USC.EDU
**David Traum** TRAUM@ICT.USC.EDU
*University of Southern California*
*Institute for Creative Technologies*
*12015 Waterfront Drive, Playa Vista CA 90094-2536, USA*

**Editors:** Paul Piwek and Kristy Elizabeth Boyer

## Abstract

This article describes a new tool for extracting question-answer pairs from text articles, and reports three experiments that investigate how suitable this technique is for supplying knowledge to conversational characters. Experiment 1 demonstrates the feasibility of our method by creating characters for 14 distinct topics and evaluating them using hand-authored questions. Experiment 2 evaluates three of these characters using questions collected from naive participants, showing that the generated characters provide full or partial answers to about half of the questions asked. Experiment 3 adds automatically extracted knowledge to an existing, hand-authored character, demonstrating that augmented characters can answer questions about new topics but with some degradation of the ability to answer questions about topics that the original character was trained to answer. Overall, the results show that question generation is a promising method for creating or augmenting a question answering conversational character using an existing text.

## 1. Introduction

Virtual question-answering characters (Leuski et al., 2006a) are useful for many purposes, such as communication skills training (Traum et al., 2007), informal education (Swartout et al., 2010), and entertainment (Hartholt et al., 2009). Question answering ability can also form the basis for characters with capabilities for longer, sustained dialogue (Roque and Traum, 2007; Artstein et al., 2009). In order to answer questions from users, a character needs to know the required answers; this has proved to be a major bottleneck in creating and deploying such characters because the requisite knowledge needs to be authored manually. There is research into ways of generating characters from corpora (Gandhe and Traum, 2008, 2010), but this requires large amounts of relevant conversational data, which are not readily available. On the other hand, plenty of information is available on many topics in text form, and text has been successfully transformed into dialogue that is acted out by conversational agents (Piwek et al., 2007; Hernault et al., 2008). We would like to be able to leverage textual resources in order to create a conversational character that responds to users – think

 Submitted 3/11; Accepted 1/12; Published online 3/12

of it as a dialogue system which can be given an article or other piece of text, and is then able to answer questions based on that text.

There are essentially two ways to use textual resources to provide knowledge to a virtual character: the resources can be precompiled into a knowledge base that the character can use, or the character can consult the text directly on an as-needed basis, using generic software for answering questions based on textual material. Automatic question answering has been studied extensively in recent years, retrieving answers from information databases (Katz, 1988) as well as unstructured text collections, as in the question-answering track at the Text REtrieval Conference (TREC) (Voorhees, 2004); interactive question answering uses a dialogue system as a front-end for adapting and refining user questions (Quarteroni and Manandhar, 2007). We are aware of one effort of integrating automatic question answering with a conversational character (Mehta and Corradini, 2008): when the character encounters a question for which it does not know an answer, it uses a question-answering system to query the web and find an appropriate answer.

Our approach follows the alternative route, of taking a textual resource and compiling it into a format that the character can use. This approach is extremely practical, as the virtual characters are created using an existing, publicly available toolkit – the ICT Virtual Human Toolkit[1] – which uses a knowledge base in the form of linked question-answer pairs (Leuski and Traum, 2010). Instead of authoring the knowledge base by hand, we populate it with question-answer pairs derived from a text through the use of question generation tools – Question Transducer (Heilman and Smith, 2009) and our own reimplementation, called OpenAryhpe.[2] Such question generation algorithms were initially developed for the purpose of reading comprehension assessment and practice, but turn out to be general enough that we can use them to create a character knowledge base. The extracted knowledge base can be used as a stand-alone conversational character or added to an existing one. The approach has the additional advantage that all the knowledge of the resulting character resides in a single, unified database, which can be further tweaked manually for improved performance.

This article describes a set of experiments which demonstrate the viability of this approach. We start with a sample of encyclopedia texts on a variety of topics, which we transform into sets of question-answer pairs using the two question generation tools mentioned above. These question-answer pairs form knowledge bases for a series of virtual question answering characters. Experiment 1 tests 14 such characters using a small, hand-crafted set of test questions, demonstrating that the method is workable. Experiment 2 uses a large test set of questions and answers collected from a pool of naive participants in order to provide a more thorough evaluation of three of the characters from the previous experiment. Overall, the characters work reasonably well when asked questions for which the source text has an answer, giving a full or partial answer more than half the time. Experiment 3 adds some of the automatically extracted knowledge bases to an existing, hand-authored character, and tests it on questions about topics it was originally designed to answer as well as from the newly added topics. Performance on the new topics is similar to that of the corresponding stand-alone characters, though there is some degradation of the character's ability to respond to questions about its original topics.

The remainder of the article gives an overview of question generation and the tools we use, describes in detail the setup and results for the three experiments, and concludes with broader implications and directions for further study.

1. http://vhtoolkit.ict.usc.edu
2. http://code.google.com/p/openaryhpe

## 2. Question generation and OpenAryhpe

### 2.1 Background

Question Generation is the task of generating reasonable questions and corresponding answers from a text. The source text can be a single sentence, a paragraph, an article and even a fact table or knowledge base. Current approaches use the source text to provide answers to the generated questions. For instance, given the sentence *Jack went to a Chinese restaurant yesterday*, a question generator might produce *Where did Jack go yesterday?* but it would not ask *Did he give tips?* This behavior of asking for known information makes question generation applicable in areas such as intelligent tutoring systems (to help learners check their accomplishment) and closed-domain Question Answering systems (to assemble question-answer pairs automatically).

A taxonomy of questions used in the question answering track of the Text REtrieval Conference divides questions into three types: factoid, list, and definition (Voorhees, 2004). Factoid questions have a single correct answer, for example *How tall is the Eiffel Tower?* List questions ask for a set of answer terms, for example *What are army rankings from top to bottom?* Finally, definition questions are more open-ended, for example *What is mold?* In later editions of the question answering track, definition questions were replaced by "other" questions, specifically asking for other information on a topic beyond what was asked in previous questions.

In terms of target complexity, question generation can be divided into *shallow* and *deep* (Graesser et al., 2009). Fact-based question generation produces typical shallow questions such as *who, when, where, what, how many, how much* and yes/no questions. Automatic question generators usually achieve this by employing various named entity recognizers, or retrieving corresponding entity annotations from ontologies and semantic web resources; the generators then formulate questions which ask for these named entities. The types of questions that are generated depend on the named entities that the generator is able to recognize in the input sentence.

A special type of shallow question involves an interrogative pronoun restricted by a head noun, for example *what stone*, *which instrument* or *how many rivers*. To generate such questions, generators typically use information about hierarchical semantic relations, gleaned from lexical resources such as WordNet. For example, given a sentence *Kim plays violin*, a question generator may use the information that *instrument* is a hypernym of *violin* in order to come up with a question such as *What instrument does Kim play?* This type of question is particularly prone to overgeneration if a lexical item is ambiguous and the generator cannot determine the appropriate sense. For example, given the source sentence *A sword is a hand-held weapon made of metal*, and knowing that metal is a genre of music, one of our generators formulated the question *What music type is a sword a hand-held weapon made?* (see Table 1 below).

Deep questions, such as *why, how, why not* and *what if* questions, are considered a more difficult task than shallow questions because they typically involve inference that goes beyond reordering and substitution of words in the surface text. Such inference is beyond the capabilities of most current question generators. One exception is Mostow and Chen (2009), which generates *what*, *why* and *how* questions about mental states following clues of a set of pre-defined modal verbs.

### 2.2 Question Transducer

Current approaches to question generation may be based on templates (Mostow and Chen, 2009), syntax (Wyse and Piwek, 2009; Heilman and Smith, 2009), or semantics (Schwartz et al., 2004).

Our experiments are based on a tool called Question Transducer (Heilman and Smith, 2009). The core idea of Question Transducer is to transduce a syntactic tree of a declarative sentence into that of an interrogative following a set of hand-crafted rules. The algorithm uses Tregex (a tree query language) and Tsurgeon (tree manipulation language) to manipulate sentences into questions in three stages: selecting and forming declarative sentences using paraphrasing rules and word substitutions, syntactically transforming the declarative sentences into questions, and scoring and ranking the resulting questions. Question Transducer employs a set of named entity recognizers (NERs) to identify the target term and determine the type of questions to be asked. Thus correctness of the question types relies on the NERs, and the grammaticality of the generated sentences depends on the transformation rules.

The idea of syntactic transformation from declarative sentences into questions is shared by some work from the Question Answering community. Given a question, question answering attempts to find the most probable answer, or the sentence that contains the answer. Thus, a measure of how a question is related to a sentence containing the answer is defined over syntactic transformation from the sentence to the question. This transformation is done, for instance, in the noisy channel approach of Echihabi and Marcu (2003) via a distortion model (specifically, IBM Model 4 in Brown et al. 1993) and in Wang et al. (2007) via the Quasi-Synchronous Grammar formalism of Smith and Eisner (2006).

### 2.3 OpenAryhpe

In addition to using the Question Transducer tool, we also developed a partial reimplementation in the framework of OpenEphyra (Schlaefer et al., 2006) which we call OpenAryhpe. The difference between the two tools is that while Question Transducer only uses the Stanford Named Entity Recognizer (Finkel et al., 2005), OpenAryhpe also includes NERs based on ontologies (to expand to new domains) and regular expressions (to recognize time, distance, measurement more precisely). Thus, OpenAryhpe is able to produce more questions by recognizing more terms. Also, OpenAryhpe is able to ask more specific questions by utilizing the hypernyms provided in the ontology list. For instance, OpenAryhpe includes 90 lists of common newspapers, films, animals, authors, etc. OpenAryhpe is able to ask *what newspaper* or *what film* questions, providing more hints to the user about the particular type of answer the system is seeking. However, these hypernyms are not disambiguated, which leads to overgeneration as discussed above. Moreover, OpenAryhpe does not implement the question ranking module that Question Transducer uses to output scored questions (Heilman and Smith, 2010). This is not a limitation for the purpose of the experiments reported in this article, because the experiments do not use the ranking of question-answer pairs.

OpenAryhpe and Question Transducer work on individual sentences of the original text, so they only generate question-answer pairs that are contained in a single sentence. Table 1 gives some examples of question-answer pairs extracted from a single source sentence (the list is not exhaustive – many more pairs were extracted from the same sentence). This sample shows that the two question generation tools are not equivalent, as each tool gives different pairs (though there is some overlap). Looking at all the questions and answers, we see that there is a many-to-many mapping – a single extracted question is paired with multiple answers, and a single answer is paired with multiple questions. The question generation tools also add knowledge that is not available in the source – Question Transducer has figured out that a sword is a *kind* of weapon, and OpenAryhpe knows that metal is a genre of music (though it is not aware that this is not the relevant meaning

| Source: A sword is a hand-held weapon made of metal. | | |
|---|---|---|
| OA[a] | | What is a hand-held weapon made of metal? — A sword. |
| OA | QT[b] | What is a sword? — A sword is a hand-held weapon made of metal. |
| OA | QT | What is a sword? — A hand-held weapon made of metal. |
| OA | | What music type is a sword a hand-held weapon made? — A hand-held weapon made of metal. |
| OA | | What music type is a sword a hand-held weapon made? — Of metal. |
| | QT | What kind of weapon is a sword? — A sword is a hand-held weapon made of metal. |
| | QT | What kind of weapon is a sword? — A hand-held weapon made of metal. |

[a]OpenAryhpe [b]Question Transducer

Table 1: Sample question-answer pairs extracted from a single source

in this case). The music question also shows us that the questions are not always grammatically well-formed; the same holds for the answers, though it is not apparent from this sample.

We did not perform a direct evaluation on the quality of the questions generated by the two systems, as in our experiments these questions are actually hidden from the humans talking to the conversational characters. However, we do evaluate how these generated questions affect the performance of the generated conversational characters (see Experiment 1 and Table 4).

## 3. Experiment 1: Hand-authored questions

### 3.1 Method

Our first experiment was intended to investigate whether a character knowledge base in question-answer format, created automatically from a source text using a question generator, could provide good answers to questions about the source text. The experiment involved the following steps.

1. Select texts to serve as the raw source for the character knowledge base, and create a set of questions and answers based on the texts to serve as a "gold standard" test set.

2. Create a character knowledge base in question-answer format, using question generation tools to extract question-answer pairs from the source text.

3. Present the test questions to the character, and evaluate the quality of resulting responses using the answers in the test set as a reference.

The above three steps are presented in the following sections.

#### 3.1.1 MATERIALS

As raw material for generating the character knowledge bases we selected 14 text excerpts from Simple English Wikipedia[3] (Table 2). These texts were chosen to represent a variety of topics. We chose the Simple English version because it contains a smaller vocabulary and simpler sentence structure than the regular English version, and therefore lends itself better for processing by the

3. http://simple.wikipedia.org

| Source Article | Length (words) | Question-Answer Pairs | | | |
|---|---|---|---|---|---|
| | | Test | Extracted | | |
| | | | OA[a] | QT[b] | Tot.[c] |
| Albert_Einstein | 385 | 12 | 162 | 295 | 405 |
| Australia | 567 | 9 | 240 | 308 | 500 |
| Beer | 299 | 10 | 146 | 231 | 315 |
| Chicago_Blackhawks | 338 | 8 | 134 | 164 | 291 |
| Excalibur | 342 | 10 | 146 | 256 | 379 |
| Greenhouse_gas | 378 | 7 | 164 | 206 | 345 |
| Ludvig_van_Beethoven | 765 | 14 | 338 | 635 | 889 |
| Movie | 543 | 12 | 162 | 246 | 393 |
| River | 368 | 13 | 108 | 262 | 339 |
| Roman_Empire | 466 | 12 | 270 | 310 | 550 |
| Rugby_football | 478 | 14 | 238 | 260 | 473 |
| Scientific_theory | 408 | 9 | 134 | 234 | 333 |
| Sword | 363 | 12 | 168 | 268 | 407 |
| United_States | 426 | 10 | 194 | 270 | 436 |

[a]OpenAryhpe [b]Question Transducer [c]The total is less than the sum of OA and QT due to overlap.

Table 2: Wikipedia text excerpts and question-answer pairs

question generation tools (limitations of current question generation technology, and specifically the tools we used, mean that source texts for creating virtual characters need to be chosen with care; such restrictions are likely to be relaxed as general question generation technology matures and improves). Articles were retrieved on June 10, 2010, and text excerpts were manually copied and pasted from a web browser into a text editor. The lengths of the individual texts ranged from 299 to 765 words (mean 438, standard deviation 122).

For each text the third author constructed a set of questions with answers that can be found in the text, to serve as a test set for the generated characters. The number of question-answer pairs per text ranged from 7 to 14, for a total of 152 (mean 10.9, standard deviation 2.2). The test set concentrated on the types of questions that the characters were expected to handle, with an overwhelming majority of *what* questions (Table 3) (our *what* category includes also *what* and *which* restricted by a head noun or preposition phrase, for example *what instrument* or *which of the generals*).

### 3.1.2 QUESTION GENERATION

We created question-answer pairs from the texts using the two question generation tools described in section 2: Question Transducer (Heilman and Smith, 2009) and OpenAryhpe. Table 2 shows the number of question-answer pairs extracted from each text by each tool. The number of extracted pairs is large because both tools are biased towards overgeneration, creating multiple questions and answers for each identified keyword. Question Transducer overgenerates because the generation step is followed by ranking the question-answer pairs to find the best ones. We did not use the rank-

| Question Type | Total | | Albert_Einstein | Australia | Beer | Chicago_Blackhawks | Excalibur | Greenhouse_gas | Ludvig_van_Beethoven | Movie | River | Roman_Empire | Rugby_football | Scientific_theory | Sword | United_States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | N | % | | | | | | | | | | | | | | |
| What | 95 | 62 | 10 | 6 | 8 | 1 | 3 | 3 | 6 | 11 | 13 | 5 | 9 | 6 | 11 | 3 |
| Who | 23 | 15 | . | 1 | . | 4 | 4 | . | 5 | 1 | . | 6 | . | 1 | . | 1 |
| How much | 8 | 5 | . | . | . | 1 | . | 1 | . | . | . | . | 1 | . | . | 5 |
| How | 7 | 5 | . | . | 1 | . | . | 2 | . | . | . | 1 | 1 | 1 | . | 1 |
| When | 6 | 4 | . | . | 1 | 1 | . | . | 2 | . | . | . | 1 | . | 1 | . |
| Yes/No | 5 | 3 | 2 | . | . | . | . | 1 | 1 | . | . | . | . | 1 | . | . |
| Where | 5 | 3 | . | 1 | . | 1 | 2 | . | . | . | . | . | 1 | . | . | . |
| Why | 3 | 2 | . | 1 | . | . | 1 | . | . | . | . | . | 1 | . | . | . |

Table 3: Test set question types

ing in our experiments (and OpenAryhpe did not even implement the ranking model), but the results in section 3.2 suggest that overgeneration is also useful for our method of creating conversational characters, because overgeneration provides many options from which the engine that drives the characters is able to identify the appropriate answers.

The generated question-answer pairs were imported as a character knowledge base into NPCEditor (Leuski and Traum, 2010), a text classification system that drives virtual characters and is available for download as part of the ICT Virtual Human Toolkit (see footnote 1). NPCEditor is trained on a knowledge base of linked question-answer pairs, and is able to answer novel questions by selecting the most appropriate response from the available answers in the knowledge base. For each new input question, NPCEditor computes a language model for the ideal answer using the linked training data; it then compares the language model of the ideal answer to those of all of the answers in the knowledge base, and selects the closest available answer based on a similarity metric between language models. The use of language models allows NPCEditor to overcome some variation in the phrasing of questions, and retrieve appropriate responses for questions it has not seen in the training data. The training questions are only an avenue for selecting the answer and are never seen by the user interacting with the character; it is therefore not crucial that they be grammatically well-formed, only that they provide useful information for selecting an appropriate answer.

For each source text we used NPCEditor to train 3 characters: one was trained on the question-answer pairs extracted by OpenAryhpe, another on those extracted by Question Transducer, and a third character was trained on the pooled set of question-answer pairs.

### 3.1.3 EVALUATION

We evaluated the character knowledge bases by presenting each of the test questions to the appropriate character in NPCEditor and rating the resulting response against the predetermined correct answer. This was done separately for the three sets of training data – the question-answer pairs

| Q-A generator | Mean | N | Distribution | | | | |
|---|---|---|---|---|---|---|---|
| | | | 0 | 0.5 | 1 | 1.5 | 2 |
| OpenAryhpe | 1.27 | 152 | 48 | 3 | 9 | 2 | 90 |
| Question Transducer | 1.33 | 152 | 44 | 3 | 9 | 1 | 95 |
| Combined | 1.49 | 152 | 30 | 2 | 14 | 0 | 106 |

Table 4: Rating of character answers (scale 0–2, two annotators)

extracted by OpenAryhpe, those extracted by Question Transducer, and the pooled set. Two raters (the second and third authors) rated all of the responses on the following three-point scale.

0 Incorrect answer.

1 Partly correct answer.

2 Fully correct answer.

Agreement between the annotators was very high: $\alpha = 0.985$,[4] indicating that the rating is fairly straightforward: of the 456 responses rated, the annotators disagreed only on 11, and in all of those instances the magnitude of the disagreement was just 1. We therefore proceeded with the analysis using the mean of the two raters as a single score.

### 3.2 Results

Table 4 shows the distribution of ratings for the character answers, broken down by the source of the question-answer pairs. The success rate is comparable for OpenAryhpe and Question Transducer, and somewhat better when the character is trained on the combined output of the two tools. The difference, however, does not appear to be significant: for the distribution data in Table 4, $\chi^2(8) = 9.6$, $p = 0.30$. We do find a marginally significant effect of question generation tool when we run an ANOVA modeling the individual mean ratings as an effect of tool and source text (a $3 \times 14$ design): $F(2,414) = 2.68$, $p = 0.07$. To the extent that the difference is meaningful, we conjecture that the pooled knowledge bases provide the best responses because they contain the largest number of question-answer pairs in the training data, and thus offer the most choices for selecting an appropriate answer.

The analysis does not show a significant interaction between question-generation tool and source text ($F(26,414) = 0.59$, $p > 0.9$), but a highly significant main effect of source text ($F(13,414) = 3.25$, $p < 0.001$), indicating that the source text and the questions asked during testing have a profound effect on the success of the character. The mean answer rating for characters trained on the pooled output from the two question generation tools ranged from 1.00 for the Australia topic to 2.00 for the Chicago Blackhawks topic (Table 5).

4. Krippendorff's $\alpha$ (Krippendorff, 1980) is a chance-corrected agreement coefficient, similar to the more familiar K statistic (Siegel and Castellan, 1988). Like K, $\alpha$ ranges from $-1$ to 1, where 1 signifies perfect agreement, 0 obtains when agreement is at chance level, and negative values show systematic disagreement. We chose to use $\alpha$ because it allows a variety of distance metrics between the judgments; here we used the interval metric as an approximation of the notion that partly correct answers fall somewhere between incorrect and fully correct answers.

| Source text | Mean | N | Distribution | | | | |
|---|---|---|---|---|---|---|---|
| | | | 0 | 0.5 | 1 | 1.5 | 2 |
| Albert_Einstein | 1.33 | 12 | 3 | . | 2 | . | 7 |
| Australia | 1.00 | 9 | 4 | . | 1 | . | 4 |
| Beer | 1.50 | 10 | 2 | . | 1 | . | 7 |
| Chicago_Blackhawks | 2.00 | 8 | . | . | . | . | 8 |
| Excalibur | 1.40 | 10 | 2 | . | 2 | . | 6 |
| Greenhouse_gas | 1.71 | 7 | . | . | 2 | . | 5 |
| Ludvig_van_Beethoven | 1.43 | 14 | 4 | . | . | . | 10 |
| Movie | 1.25 | 12 | 4 | . | 1 | . | 7 |
| River | 1.77 | 13 | 1 | . | 1 | . | 11 |
| Roman_Empire | 1.67 | 12 | 2 | . | . | . | 10 |
| Rugby_football | 1.32 | 14 | 4 | 1 | . | . | 9 |
| Scientific_theory | 1.50 | 9 | 1 | 1 | 1 | . | 6 |
| Sword | 1.75 | 12 | 1 | . | 1 | . | 10 |
| United_States | 1.40 | 10 | 2 | . | 2 | . | 6 |

Table 5: Rating of answers by characters trained on the pooled output from the question generation tools

Since the success of a character depends both on the training data and the questions asked, we designed the next experiment to look at a wider array of questions, which would better represent what a conversational character might encounter in a live interaction with people.

## 4. Experiment 2: Questions collected from users

### 4.1 Method

Our second experiment was intended to investigate whether a character knowledge base, created automatically as in the previous experiment, could provide good answers to typical questions asked by users who are not familiar with the system. The experiment involved the following steps.[5]

1. Select texts and create a character knowledge base as in the previous experiment.
2. Collect questions and answers from naive participants to serve as a test set.
3. Present the test questions to the character, and evaluate the quality of resulting responses using the answers in the test set as a reference.

#### 4.1.1 MATERIALS

Since this experiment concentrated on evaluating character responses to collected user questions, we used a small selection of the source texts from the previous experiment. We expected performance to drop due to the more varied questions, so we chose three of the five top-performing source texts:

5. This experiment was reported in abbreviated form in Chen et al. (2011).

Sword, River and Roman_Empire (see Table 2). We only trained the characters based on the pooled output of the two question generation tools, since previously that resulted in the best performance.

#### 4.1.2 TEST SET

Since our method is intended to create a virtual character that can answer questions by human users, we need to test our character knowledge base against typical questions that a person may ask; at the same time, the test set should take into account the fact that the character can only respond with information that is present in the source texts. We therefore collected test questions from participants both before reading the source text and after having read it. The procedure for collecting the test data was as follows.

1. The participant wrote five questions about a topic (swords, rivers, or the Roman Empire), without having read any text materials about it.
2. The participant read the source text about the topic.
3. The participant wrote five additional questions about the topic, based on the source text.
4. The participant provided answers to all of their questions, where each answer was a contiguous segment of text from the source. If the participant felt that the text did not contain an answer, they marked the answer as "N/A".

The data were collected using the Qualtrics on-line survey tool.[6] The procedure was repeated three times, once for each topic, so in total each participant provided 30 questions and answers – 5 questions for each topic before reading the text and 5 after reading. We had 22 participants for a total of 660 collected questions and corresponding answers (220 for each topic). Topics were presented to all the participants in the same order – first swords, then rivers, and finally the Roman Empire.

#### 4.1.3 EVALUATION

We evaluated the character knowledge bases by presenting each of the test questions to NPCEditor and rating the resulting answer against the user-provided answer. Two raters (the second and third authors) rated all of the responses independently on four dimensions – two binary dimensions, and two dimensions on a three-point scale.

**Question quality** 1 Comprehensible, 0 Incomprehensible.

**System response** 2 Full answer, 1 Partial answer, 0 Not an answer.

**Is the system response good enough** 1 Yes, 0 No.

**Is the system response the best in the text** 1 Yes, 0 No, $-1$ There is no good response in the text.

The first dimension, question quality, was rated merely to make sure that the questions created by the participants were understandable; the vast majority of questions (634 of 660) were rated as comprehensible by both coders, so we did not analyze this further. The main dimension for evaluation was the second, answer quality, and it forms the basis for the analysis in the next section; a few examples of this annotation are shown in Table 6. The third and fourth dimensions were

6. http://qualtrics.com

| Question | User answer | System answer | Ratings | |
|---|---|---|---|---|
| What is a sword made of? | Metal | A sword is a hand-held weapon made of metal | 2 | 2 |
| What are swords made out of? | A sword is a hand-held weapon made of metal | Larger swords such as longswords or claymores are used with two hands | 0 | 0 |
| What is a katana sword? | A Katana has one sharp edge and a small guard near the handle. | A sword is a hand-held weapon made of metal | 0 | 0 |
| What is a bayonet? | N/A | A sword is a hand-held weapon made of metal | 0 | 0 |
| How fast is a river? | It sometimes flows very slowly | A wide slow river is called an "old river" | 1 | 0 |

Table 6: Sample rating of system answers by 2 coders

intended to give insight into nuances that are not captured by a single numerical rating for answer quality: whether the answer provided by the system was good enough as a response for a virtual character engaged in dialogue, and whether it was the best possible answer that could be found in the text. The latter question also allowed the raters to judge whether an answer to a question was available, independently of the judgment of the participant who provided the question.

Agreement between the annotators was high for all the rating dimensions: $\alpha = 0.865$ for answer quality, $\alpha = 0.804$ for whether an answer was good enough, and $\alpha = 0.784$ for whether an answer was the best available.[7] While the three questions were intended to capture distinct aspects of an answer's suitability, in practice there was not much difference in the annotators' responses, and the ratings display very high correlations ($r = 0.92$ for one coder and $r = 0.95$ for the other coder between answer quality and good enough, and $r = 0.81$ and $r = 0.82$ between answer quality and best available). We therefore proceed with the analysis using only the results from the answer quality annotation.

## 4.2 Results

### 4.2.1 QUESTION DISTRIBUTION

The questions collected from experiment participants show us what human users think they may want to ask a virtual character (since these questions were asked off-line rather than in conversation, they are less indicative as to what users actually do ask). The participants produced questions under two conditions – before reading the source text and after having read it – and provided their own judgments as to whether an answer was available in the text. Table 7 shows the differences between the two conditions, broken down by question type. The most common question type was *what*,

7. For the answer quality question we used $\alpha$ with the interval metric as explained in footnote 4, whereas for the best available question the judgments are categorical so we used the nominal metric; for binary distinctions like the "good enough" question, the two metrics are equivalent.

| Question type | Total | | Before Reading | | | | After Reading | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | avail | | n/a | | avail | | n/a | |
| | N | % | N | % | N | % | N | % | N | % |
| What | 363 | 55 | 90 | 51 | 66 | 43 | 196 | 63 | 11 | 61 |
| Yes/No | 59 | 9 | 10 | 6 | 16 | 10 | 28 | 9 | 5 | 28 |
| Who | 50 | 8 | 17 | 10 | 10 | 7 | 23 | 7 | 0 | 0 |
| When | 46 | 7 | 24 | 14 | 5 | 3 | 17 | 5 | 0 | 0 |
| Where | 46 | 7 | 12 | 7 | 15 | 10 | 17 | 5 | 2 | 11 |
| How much | 45 | 7 | 13 | 7 | 24 | 16 | 8 | 3 | 0 | 0 |
| How | 39 | 6 | 6 | 3 | 13 | 8 | 20 | 6 | 0 | 0 |
| Why | 9 | 1 | 5 | 3 | 3 | 2 | 1 | 0 | 0 | 0 |
| Other | 3 | 0 | 0 | 0 | 1 | 1 | 2 | 1 | 0 | 0 |
| Total | 660 | 100 | 177 | 100 | 153 | 100 | 312 | 100 | 18 | 100 |

Table 7: Test question types

| Question type | Total | | River | | Roman | | Sword | |
|---|---|---|---|---|---|---|---|---|
| | N | % | N | % | N | % | N | % |
| What | 363 | 55 | 143 | 65 | 99 | 45 | 121 | 55 |
| Yes/No | 59 | 9 | 19 | 9 | 13 | 6 | 27 | 12 |
| Who | 50 | 8 | 0 | 0 | 41 | 19 | 9 | 4 |
| When | 46 | 7 | 0 | 0 | 30 | 14 | 16 | 7 |
| Where | 46 | 7 | 26 | 12 | 6 | 3 | 14 | 6 |
| How much | 45 | 7 | 15 | 7 | 15 | 7 | 15 | 7 |
| How | 39 | 6 | 15 | 7 | 10 | 5 | 14 | 6 |
| Why | 9 | 1 | 1 | 0 | 5 | 2 | 3 | 1 |
| Other | 3 | 0 | 1 | 0 | 1 | 0 | 1 | 0 |
| Total | 660 | 100 | 220 | 100 | 220 | 100 | 220 | 100 |

Table 8: Test questions by topic

constituting 55% of all questions (including *what* and *which* restricted by a noun or preposition phrase). The distribution is different for questions produced before and after reading the source text ($\chi^2(8) = 37$, $p < 0.001$): participants produced more *what* and yes/no questions after reading the source text – they had asked more varied question types before. Of the questions asked before reading the text, 46% had no answers available in the text; of those asked after, only 5% were without answers.

Question types also differed by topic as shown in Table 8, and again the difference was significant ($\chi^2(16) = 115$, $p < 0.001$). The topic of rivers received no *who* or *when* questions, which together constituted almost a third of the questions about the Roman Empire.

| Authoring time | Answer | Mean | N | Distribution | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | 0 | 0.5 | 1 | 1.5 | 2 |
| Before reading | Available | 0.53 | 177 | 103 | 19 | 21 | 8 | 26 |
| | N/A | 0.12 | 153 | 131 | 12 | 7 | 1 | 2 |
| After reading | Available | 0.99 | 312 | 127 | 26 | 21 | 4 | 134 |
| | N/A | 0.36 | 18 | 13 | 2 | 0 | 1 | 2 |
| Total | | 0.65 | 660 | 374 | 59 | 49 | 14 | 164 |

Table 9: Quality rating of character responses (0–2 scale)

#### 4.2.2 ANSWER QUALITY

The quality of the answers provided by the character to the user questions varied depending on the question authoring time (before or after reading the text) and the availability of an answer in the source text. We used the mean of the scores given by the two raters, so each answer received a score between 0 and 2 in half-point increments. Table 9 shows the mean rating in each group as well as the distribution of scores; recall that the availability of an answer was judged by the participants whereas the quality of an answer was judged by the raters, which explains why a small number of answers are considered good even though the original participant thought there was no good answer in the text. The vast majority of good answers come from questions that were authored after the participant had read the source text, whereas for questions written before having read the text, NPCEditor has a much harder time finding an appropriate answer even when one is available. A likely explanation for this is that questions written after having read the text are more likely to use vocabulary and constructions found in the text – that is, the texts cause some form of lexical priming (Levelt and Kelter, 1982) or syntactic priming (Bock, 1986). Looking only at user questions with an available answer, questions asked after having read the text have an out-of-vocabulary word token rate of 42%, compared to 52% for questions asked before having read the text. NPCEditor is built on cross-language information retrieval techniques (Leuski and Traum, 2010) and thus it does not require the questions and answers to share a vocabulary, but it does require that the test questions be reasonably similar to the training questions. Since the questions in the training data are derived from the source text, a better alignment of user questions with the source text should make it easier to map the questions to appropriate answers. Finally, we note that answers provided by the characters to questions that do not have an answer in the source text are typically very poor (note however that NPCEditor is able to tell when its confidence in an answer is low, see section 5.2 below).

The quality of the answer is also affected by the question type. Table 10 gives the mean ratings for each question type, broken down by authoring time, both for all questions of the type as well as just those questions that have an answer available in the text. We see substantial differences between the question types – *who* questions do particularly well, whereas yes/no questions do rather poorly. This may be a byproduct of the question generation tools, which are able to identify some types of information better than others. For instance, the named entity recognizers have been trained on annotated data to recognize people names, and thus *who* questions are usually linked to a correct target answer in the training data. In contrast, *why* questions are generated from lexical matching of causal clue words, such as *the reason* or *due to*; this matching is not discriminate enough in

| Question type | All Questions | Before Reading | | After Reading | |
|---|---|---|---|---|---|
| | | all | avail | all | avail |
| What | 0.70 | 0.31 | 0.46 | 1.00 | 1.02 |
| Yes/No | 0.27 | 0.15 | 0.25 | 0.36 | 0.41 |
| Who | 1.15 | 0.67 | 0.94 | 1.72 | 1.72 |
| When | 0.79 | 0.62 | 0.67 | 1.09 | 1.09 |
| Where | 0.54 | 0.22 | 0.50 | 1.00 | 1.12 |
| How much | 0.53 | 0.39 | 0.65 | 1.19 | 1.19 |
| How | 0.22 | 0.16 | 0.50 | 0.28 | 0.28 |
| Why | 0.11 | 0.12 | 0.20 | 0.00 | 0.00 |
| Other | 1.33 | 0.00 | — | 2.00 | 2.00 |

Table 10: Mean ratings by question type (0–2 scale)

finding out the reason and result of events, which may be a reason that generated *why* questions are generally of low quality.

Interestingly enough, we did not find an effect of source text on the quality of the responses. We conducted a 4-way ANOVA looking at source text, availability of a response, question authoring time, and question type. The three factors discussed above came out as highly significant main effects: response availability ($F(1,592) = 106$, $p < 0.001$), authoring time ($F(1,592) = 53$, $p < 0.001$) and question type ($F(8,592) = 6.5$, $p < 0.001$); there was also a significant interaction between source text and question type ($F(14,592) = 3.6$, $p < 0.001$). But the main effect of source text was not significant ($F(2,592) = 2.8$, $p = 0.06$), and there was only one additional marginally significant interaction, between answer availability and question type ($F(7,592) = 2.1$, $p = 0.04$).

## 5. Experiment 3: character augmentation

### 5.1 Method

The first two experiments showed that question generation can be used for creating question answering virtual characters from a text. Our third experiment set out to investigate how the addition of an automatically generated question-answer knowledge base affects the performance of an existing, hand-authored character.[8] Ideally, such an augmented character would be able to answer the same questions as the original character without a substantial performance loss, and also be able to answer some questions covered by the added knowledge base. We chose to experiment with an existing character for which we already had an extensive test set of questions with known correct responses; to this character we added successive knowledge bases generated by the question-answering tools.

#### 5.1.1 MATERIALS

The base character for the experiment was the twins Ada and Grace, a pair of virtual characters situated in the Museum of Science in Boston where they serve as virtual guides (Swartout et al., 2010). The Twins answer questions from visitors about exhibits in the museum and about science

8. This experiment was reported in Nouri et al. (2011).

| Source text | Questions | Answers | Q-A Pairs |
|---|---|---|---|
| Twins | 406 | 148 | 483 |
| Twins + Australia | 652 | 342 | 999 |
| Twins + Beer | 559 | 268 | 807 |
| Twins + Beethoven | 849 | 421 | 1412 |
| Twins + Australia + Beer | 804 | 462 | 1323 |
| Twins + Australia + Beer + Beethoven | 1245 | 735 | 2252 |

Table 11: Training data for the augmented characters

in general; these topics will be referred to as the *original topics*, because these are what the original knowledge base was designed for. All the training data for the Twins were authored by hand.

The base character was successively augmented by adding three of the automatically generated knowledge bases from Experiment 1 (the choice of knowledge bases was arbitrary). These will be referred to as the *new topics*. We trained a total of five augmented characters in addition to the baseline; Table 11 shows the number of questions, answers and links in each of the sets of training data.

### 5.1.2 TEST SET

**Original topics.** To test performance of the augmented characters on questions from the Twins' original topics we use an extensive test set collected during the initial stages of the Twins' deployment at the Museum of Science, when visitor interaction was done primarily through trained handlers (relying on handlers allowed us to deploy the characters prior to collecting the required amount of visitor speech, mostly from children, necessary to train acoustic models for speech recognition). The handlers relay the visitors' questions through a microphone to be processed by a speech recognizer; they also tend to reformulate user questions to better match the questions in the Twins' knowledge base, and many of their utterances are a precise word for word match of utterances in the Twins' training data. Such utterances are a good test case for the classifier because the intended correct responses are known, but actual performance varies due to speech recognition errors; they thus test the ability of the classifier to overcome a noisy input.

The same test set was used in Wang et al. (2011) to compare various methods of handling speech recognizer output; here we use it to compare different character knowledge bases. The speech recognizer output remains constant in the different test runs – all characters are tested on exactly the same utterance texts. To a classifier for the original topics, question-answer pairs from the new topics can be considered as training noise; what the different characters test, then, is how the addition of knowledge bases for the new topics affects the performance of the original, hand-authored part of the character.

The test set consists of 7690 utterances. These utterances were collected on 56 individual days so they represent several hundred visitors; the majority of the utterances (almost 6000) come from two handlers. Each utterance contains the original speech recognizer output retrieved from the system logs (speech recognition was performed using the SONIC toolkit, Pellom and Hacıoğlu, 2001/2005). Some of the utterances are identical – there is a total of 2264 utterance types (speech recognizer output), corresponding to 265 transcribed utterance types (transcriptions were performed manually). The median word error rate for the utterances is 20% (mean 29%, standard deviation

36%). This level of word error rate is acceptable for this application – as we will see below, the original character fails to understand only 10% of the input utterances, and this error rate declines rapidly when the character is allowed to identify its own non-understanding (Figure 1).

**New topics.** We also tested performance of the augmented characters on questions relating to the new topics. Since we do not have an extensive set of spoken utterances as for the Twins' original topics, we used the same test sets constructed for Experiment 1.

#### 5.1.3 EVALUATION

**Original topics.** To evaluate performance on questions from the original topics, we ran our test set through each of the characters in Table 11. For each utterance we sent the text of the speech recognizer output to NPCEditor, and compared the response to the answers linked to the corresponding manual transcription. A response was scored as correct if it matched one of the linked answers, otherwise it was scored as incorrect. We also collected the confidence scores reported by NPCEditor in order to enable the analysis in Figure 1 below (the confidence score is the inverse of the Kullback-Leibler divergence between the language models of the ideal response and the actual response; see Leuski and Traum, 2010).

**New topics.** Performance on questions relating to the added knowledge bases was evaluated as in the previous experiments, by sending the text of the question to NPCEditor and manually comparing the response to the predetermined answer key. Since Experiments 1 and 2 have already established that this procedure is highly reliable, the rating was performed by just one person (the fourth author).

### 5.2 Results

#### 5.2.1 PERFORMANCE ON THE ORIGINAL TOPICS

Just counting the correct and incorrect responses is not sufficient for evaluating character performance, because NPCEditor employs dialogue management logic designed to avoid the worst outputs. During training, NPCEditor calculates a response threshold based on the classifier's confidence in the appropriateness of selected responses: this threshold finds an optimal balance between false positives (inappropriate responses above threshold) and false negatives (appropriate responses below threshold) on the training data. At runtime, if the confidence for a selected response falls below the predetermined threshold, that response is replaced with an "off-topic" utterance that asks the user to repeat the question or takes initiative and changes the topic (Leuski et al., 2006b); such failure to return a response (also called non-understanding, Bohus and Rudnicky, 2005) is usually preferred over returning an inappropriate one (misunderstanding).

The capability to not return a response is crucial in keeping conversational characters coherent, but it is not captured by standard classifier evaluation methods such as accuracy, recall (proportion of correct responses that were retrieved), or precision (proportion of retrieved responses that are correct). We cannot use the default threshold calculated by NPCEditor during training, because these default thresholds yield different return rates for different characters. We therefore use a visual evaluation method that looks at the full trade-off between return levels and error rates (Artstein, 2011).

For each test utterance we logged the top-ranked response together with its confidence score, and then we plotted the rate of off-topics against errors at each possible threshold; this was done

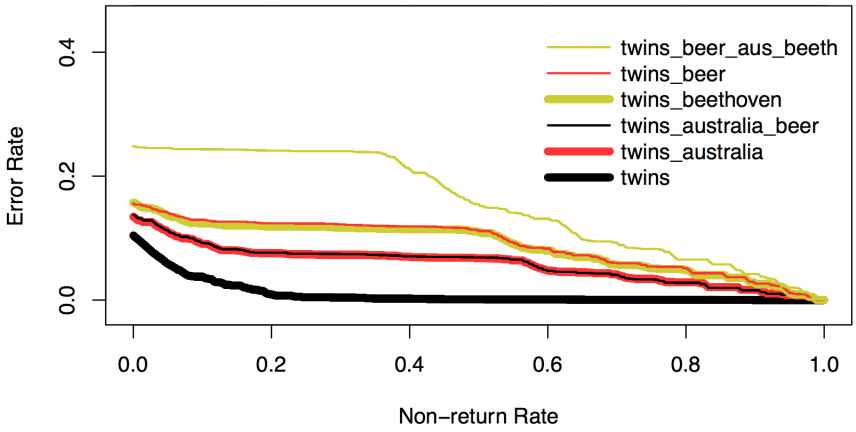


Figure 1: Trade-off between errors and non-returns on the original topics

separately for each character (since confidence scores are based on parameters learned during training, they are not comparable across characters). Figure 1 shows the curves for the baseline character and the five augmented characters: non-returns are plotted on the horizontal axis and corresponding error rates on the vertical axis; at the extreme right, where no responses are returned, error rates are necessarily zero for all characters. Lower curves indicate better performance.

The best performer on the test set for the original topics is the original Twins character, with a 10% error rate when all responses are returned, and virtually no errors with a non-return rate above 20%. Performance degrades somewhat with the successive addition of automatically generated questions from the new topics, though the degradation is mitigated to some extent when higher non-return rates are acceptable. In exchange for an increased error rate on questions from the original topics, the augmented characters can now answer questions pertaining to the new topics.

### 5.2.2 PERFORMANCE ON THE NEW TOPICS

The added knowledge bases are identical to those in Experiment 1, so that represents the ceiling we can expect for performance of the augmented characters on the new topics. We found that performance was only slightly degraded. We tested each question set on those characters that included the relevant knowledge base. The number of correct or partially correct answers is shown in Table 12; in each case, the correct answers are a subset of the correct answers from Experiment 1. We also tested the question sets on the original Twins character – as expected, none of the returned responses was a correct answer.

## 6. Discussion

The experiments demonstrate that our approach is viable – using question generation tools to populate character knowledge bases in question-answer format results in virtual characters that can give appropriate answers to user questions at least some of the time. Some types of questions do better than others, and *who* questions do particularly well. The differences between the question types probably have to do with the question generation tools and the kinds of question-answer pairs they

| Character | Test Set | | |
|---|---|---|---|
| | Australia | Beer | Beethoven |
| N = | 9 | 10 | 14 |
| Experiment 1 | 5 | 8 | 9[a] |
| Twins + Australia | 5 | | |
| Twins + Beer | | 7 | |
| Twins + Beethoven | | | 9 |
| Twins + Australia + Beer | 5 | 7 | |
| Twins + Australia + Beer + Beethoven | 5 | 6 | 9 |
| Twins | 0 | 0 | 0 |

[a]This number differs from that in Table 5 because one answer which was marked as correct in Experiment 1 was marked as incorrect in Experiment 3.

Table 12: Correct answers from the augmented characters

extract. It is no surprise that characters rarely give an appropriate answer to questions without an answer in the source text; the solution to this problem is twofold – find source texts that contain the information users want to ask about, and enable a mechanism for the character to recognize questions that cannot be answered, together with strategies for appropriate responses that are not answers (Patel et al., 2006; Artstein et al., 2009). However, there remain many user questions with an answer in the source text that the character is not able to find, and this is where there is substantial room for improvement.

A key factor for any question answering character is getting a good match between actual questions the users want to ask and the answers the character is able to provide. A study of user questions can guide the creators of a character towards appropriate texts that contain answers to the common questions. The questions collected in our study show that people ask different kinds of questions for the various topics presented to them; this can serve as the beginning of a systematic study of question patterns that depend on the topic. There is also a need to bridge the gap between the vocabulary of user questions and that of questions extracted from the source texts, through improvements to the question generation process and the use of lexical resources.

The current work suggests several directions for future research. Our experiments always rated the response that got the highest ranking from NPCEditor. However, NPCEditor is more nuanced than that, and it can use the confidence scores that rank the responses to tell to some degree whether the chosen response is likely to be correct or whether it is more likely that an appropriate response is not available. This functionality may allow the character itself to judge the quality of its answers.

Additionally, our main test set of questions and answers was collected ahead of time in a questionnaire format. It constitutes a broad test set that can be used to compare different question generation or classification mechanisms. Ultimately, however, the purpose of this research is to create conversational virtual characters, so it would be appropriate to also test the characters in conversation.

## Acknowledgments

We wish to thank Michael Heilman, author of Question Transducer, for giving us access to his code and allowing us to use it in our experiments.

The project or effort described here has been sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

## References

Ron Artstein. Error return plots. In *Proceedings of the SIGDIAL 2011 Conference*, pages 319–324, Portland, Oregon, June 2011. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W11/W11-2037`.

Ron Artstein, Sudeep Gandhe, Jillian Gerten, Anton Leuski, and David Traum. Semi-formal evaluation of conversational characters. In Orna Grumberg, Michael Kaminski, Shmuel Katz, and Shuly Wintner, editors, *Languages: From Formal to Natural. Essays Dedicated to Nissim Francez on the Occasion of His 65th Birthday*, volume 5533 of *Lecture Notes in Computer Science*, pages 22–35. Springer, Heidelberg, May 2009.

J. Kathryn Bock. Syntactic persistence in language production. *Cognitive Psychology*, 18(3):355–387, 1986.

Dan Bohus and Alexander I. Rudnicky. Sorry, I didn't catch that! – An investigation of non-understanding errors and recovery strategies. In *Proceedings of the 6th SIGdial Workshop on Discourse and Dialogue*, pages 128–143, Lisbon, Portugal, September 2005.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19 (2):263–311, 1993. URL `http://www.aclweb.org/anthology/J/J93/J93-2003.pdf`.

Grace Chen, Emma Tosch, Ron Artstein, Anton Leuski, and David Traum. Evaluating conversational characters created through question generation. In *Proceedings of the Twenty-Fourth International Florida Artificial Intelligence Research Society Conference*, pages 343–344, Palm Beach, Florida, May 2011. AAAI Press.

Abdessamad Echihabi and Daniel Marcu. A noisy-channel approach to question answering. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 16–23, Sapporo, Japan, July 2003. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P03/P03-1003.pdf`.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by Gibbs sampling. In *ACL '05: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370, Morristown, NJ, USA, 2005. doi: 10.3115/1219840.1219885.

Sudeep Gandhe and David Traum. Evaluation understudy for dialogue coherence models. In *9th SIGdial Workshop on Discourse and Dialogue*, Columbus, Ohio, 2008.

Sudeep Gandhe and David Traum. I've said it before, and I'll say it again: An empirical investigation of the upper bound of the selection approach to dialogue. In *Proceedings of the SIGDIAL 2010 Conference*, Tokyo, 2010.

A. Graesser, J. Otero, A. Corbett, D. Flickinger, A. Joshi, and L. Vanderwende. Guidelines for Question Generation Shared Task and Evaluation Campaigns. In V. Rus and A. Graesser, editors, *The Question Generation Shared Task and Evaluation Challenge Workshop Report*. The University of Memphis, 2009.

Arno Hartholt, Jonathan Gratch, Lori Weiss, and The Gunslinger Team. At the virtual Frontier: Introducing Gunslinger, a multi-character, mixed-reality, story-driven experience. In Zsófia Ruttkay, Michael Kipp, Anton Nijholt, and Hannes Högni Vilhjálmsson, editors, *Intelligent Virtual Agents: 9th International Conference, IVA 2009, Amsterdam, The Netherlands, September 14–16, 2009 Proceedings*, volume 5773 of *Lecture Notes in Artificial Intelligence*, pages 500–501, Heidelberg, September 2009. Springer. doi: 0.1007/978-3-642-04380-2_62.

Michael Heilman and Noah A. Smith. Question generation via overgenerating transformations and ranking. Technical Report CMU-LTI-09-013, Carnegie Mellon University Language Technologies Institute, 2009.

Michael Heilman and Noah A. Smith. Good Question! Statistical Ranking for Question Generation. In *Proc. of NAACL/HLT*, 2010.

Hugo Hernault, Paul Piwek, Helmut Prendinger, and Mitsuru Ishizuka. Generating dialogues for virtual agents using nested textual coherence relations. In Helmut Prendinger, James Lester, and Mitsuru Ishizuka, editors, *Intelligent Virtual Agents: 8th International Conference, IVA 2008, Tokyo, Japan, September 1–3, 2008 Proceedings*, volume 5208 of *Lecture Notes in Artificial Intelligence*, pages 139–145, Heidelberg, September 2008. Springer. doi: 10.1007/978-3-540-85483-8_14.

Boris Katz. Using English for indexing and retrieving. In *Proceedings of the 1st RIAO Conference on User-Oriented Content-Based Text and Image Handling (RIAO '88)*, 1988.

Klaus Krippendorff. *Content Analysis: An Introduction to Its Methodology*, chapter 12, pages 129–154. Sage, Beverly Hills, California, 1980.

Anton Leuski and David Traum. Practical language processing for virtual humans. In *Twenty-Second Annual Conference on Innovative Applications of Artificial Intelligence (IAAI-10)*, 2010.

Anton Leuski, Brandon Kennedy, Ronakkumar Patel, and David Traum. Asking questions to limited domain virtual characters: how good does speech recognition have to be? In *Proceedings of the 25th Army Science Conference*, 2006a.

Anton Leuski, Ronakkumar Patel, David Traum, and Brandon Kennedy. Building effective question answering characters. In *Proceedings of the 7th SIGdial Workshop on Discourse and Dialogue*, Sydney, Australia, July 2006b.

Willem J. M. Levelt and Stephanie Kelter. Surface form and memory in question answering. *Cognitive Psychology*, 14(1):78–106, 1982.

Manish Mehta and Andrea Corradini. Handling out of domain topics by a conversational character. In *Proceedings of the 3rd international conference on Digital Interactive Media in Entertainment and Arts*, DIMEA '08, pages 273–280, New York, NY, USA, 2008. ACM. doi: 10.1145/1413634.1413686.

Jack Mostow and Wei Chen. Generating instruction automatically for the reading strategy of self-questioning. In Vania Dimitrova, Riichiro Mizoguchi, Benedict du Boulay, and Art Graesser, editors, *Artificial Intelligence in Education – Building Learning Systems that Care: From Knowledge Representation to Affective Modelling*, volume 200 of *Frontiers in Artificial Intelligence and Applications*, pages 465–472, Amsterdam, 2009. IOS Press.

Elnaz Nouri, Ron Artstein, Anton Leuski, and David Traum. Augmenting conversational characters with generated question-answer pairs. In *Question Generation: Papers from the AAAI Fall Symposium*, pages 49–52, Arlington, Virginia, November 2011. AAAI Press.

Ronakkumar Patel, Anton Leuski, and David Traum. Dealing with out of domain questions in virtual characters. In Jonathan Gratch, Michael Young, Ruth Aylett, Daniel Ballin, and Patrick Olivier, editors, *Intelligent Virtual Agents: 6th International Conference, IVA 2006, Marina Del Rey, CA, USA, August 21–23, 2006 Proceedings*, volume 4133 of *Lecture Notes in Artificial Intelligence*, pages 121–131, Heidelberg, August 2006. Springer. doi: 10.1007/11821830_10.

Bryan Pellom and Kadri Hacıoğlu. SONIC: The University of Colorado continuous speech recognizer. Technical Report TR-CSLR-2001-01, University of Colorado, Boulder, 2001/2005. URL `http://www.bltek.com/images/research/virtual-teachers/sonic/pellom-tr-cslr-2001-01.pdf`.

Paul Piwek, Hugo Hernault, Helmut Prendinger, and Mitsuru Ishizuka. T2D: Generating dialogues between virtual agents automatically from text. In Catherine Pelachaud, Jean-Claude Martin, Elisabeth André, Gérard Chollet, Kostas Karpouzis, and Danielle Pelé, editors, *Intelligent Virtual Agents: 7th International Conference, IVA 2007, Paris, France, September 17–19, 2007 Proceedings*, volume 4722 of *Lecture Notes in Artificial Intelligence*, pages 161–174, Heidelberg, September 2007. Springer. doi: 10.1007/978-3-540-74997-4_16.

Silvia Quarteroni and Suresh Manandhar. A chatbot-based interactive question answering system. In Ron Artstein and Laure Vieu, editors, *Decalog 2007: Proceedings of the 11th Workshop on the Semantics and Pragmatics of Dialogue*, pages 83–90, Rovereto, Italy, May 2007.

Antonio Roque and David Traum. A model of compliance and emotion for potentially adversarial dialogue agents. In *Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue*, pages 35–38, Antwerp, Belgium, September 2007.

Nico Schlaefer, Petra Gieselman, and Guido Sautter. The Ephyra QA system at TREC 2006. In *The Fifteenth Text Retrieval Conference Proceedings*, Gaithersburg, MD, November 2006.

Lee Schwartz, Takako Aikawa, and Michel Pahud. Dynamic Language Learning Tools. In *Proceedings of the 2004 InSTIL/ICALL Symposium*, 2004.

Sidney Siegel and N. John Castellan, Jr. *Nonparametric Statistics for the Behavioral Sciences*, chapter 9.8, pages 284–291. McGraw-Hill, New York, second edition, 1988.

David A. Smith and Jason Eisner. Quasi-synchronous grammars: Alignment by soft projection of syntactic dependencies. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 23–30, New York, June 2006. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W06/W06-3104.pdf.

William Swartout, David Traum, Ron Artstein, Dan Noren, Paul Debevec, Kerry Bronnenkant, Josh Williams, Anton Leuski, Shrikanth Narayanan, Diane Piepol, Chad Lane, Jacquelyn Morie, Priti Aggarwal, Matt Liewer, Jen-Yuan Chiang, Jillian Gerten, Selina Chu, and Kyle White. Ada and Grace: Toward realistic and engaging virtual museum guides. In Jan Allbeck, Norman Badler, Timothy Bickmore, and Alla Pelachaud, Catherine Safonova, editors, *Intelligent Virtual Agents: 10th International Conference, IVA 2010, Philadelphia, PA, USA, September 20–22, 2010 Proceedings*, volume 6356 of *Lecture Notes in Artificial Intelligence*, pages 286–300, Heidelberg, September 2010. Springer.

David Traum, Antonio Roque, Anton Leuski, Panayiotis Georgiou, Jillian Gerten, Bilyana Martinovski, Shrikanth Narayanan, Susan Robinson, and Ashish Vaswani. Hassan: A virtual human for tactical questioning. In *Proceedings of the 8th SIGdial Workshop on Discourse and Dialogue*, pages 71–74, Antwerp, Belgium, September 2007.

Ellen M. Voorhees. Overview of the TREC 2003 question answering track. In *Proceedings of The Twelfth Text Retrieval Conference (TREC 2003)*, pages 54–68, 2004. URL http://trec.nist.gov/pubs/trec12/papers/QA.OVERVIEW.pdf.

Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. What is the Jeopardy model? A quasi-synchronous grammar for QA. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 22–32, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/D/D07/D07-1003.pdf.

William Yang Wang, Ron Artstein, Anton Leuski, and David Traum. Improving spoken dialogue understanding using phonetic mixture models. In *Proceedings of the Twenty-Fourth International Florida Artificial Intelligence Research Society Conference*, pages 329–334, Palm Beach, Florida, May 2011. AAAI Press.

Brendan Wyse and Paul Piwek. Generating questions from OpenLearn study units. In Scotty D. Craig and Darina Dicheva, editors, *AIED 2009: 14th International Conference on Artificial Intelligence in Education, Workshops Proceedings: Volume 1, The 2nd Workshop on Question Generation*, pages 66–73, Brighton, UK, July 2009.