

Knowledge-based Question Answering for DIYers

Doo Soon Kim* and Zhe Feng and Lin Zhao

Bosch Research, Sunnyvale, CA 94085, USA

Abstract. DIY (Do-It-Yourself) requires extensive knowledge such as the usage of tools, properties of materials, and the procedure of activities. Most DIYers use online search to find information but it is usually time-consuming and challenging for novice DIYers to understand the retrieved results and later apply them to their individual DIY tasks. In the work, we present a Question Answering (QA) system which can address the DIYers' specific needs. The core component is a knowledge base (KB) which contains a vast amount of domain knowledge encoded in a knowledge graph. The system is able to explain how the answers are derived with reasoning process. Our user study shows that the QA system addresses DIYers' needs more effectively than the web search.

1 Problem

Our goal is to build a QA system for the home improvement DIY projects available at Bosch-Do-It ¹. Table 1 shows the common DIY question types, which were collected through a user study. Given a question, the QA system should provide not only an answer but also an explanation on how the answer is derived. The explanation capability is particularly important for the DIY questions which are generally a non-factoid question.

2 The DIY QA System

Fig. 1 shows the overall architecture of our system. The key component is a KB which is graphically represented using RDF and stored in Stardog ², a semantic web platform. The ontology defines a taxonomy of about 300 entity concepts and 150 action concepts. The entity concepts include the common DIY objects such as tools (e.g., JIGSAW), accessories (e.g., DRILL-BIT) and materials (e.g., WOOD-SCREW). The action concepts include the DIY actions such as SAWING and GLUING, and also include the tool-related actions such as REPLACING-BLADE. Each concept is associated with domain knowledge such as the definition and other attributes. Each DIY project is then represented using the concepts. Specifically, the project representation consists of the required entities and the action structure. The required entities indicate the entities needed in the project along with their specification information, while the action structure describes a hierarchical structure of the sub-actions to represent the decomposition of the project steps. Our KB also contains the product knowledge which are used to recommend the tools or accessories suitable for a project. Different types of knowledge are inter-connected to one another and therefore can be combined to answer more complex questions.

* The first author is now affiliated with Adobe Research (contact: dkim@adobe.com).

¹ <http://www.bosch-do-it.com/>

² <http://stardog.com>

Question Type	Sub-question type	Example
Project Question	required entities / properties	<i>What power tools do I need in the project?</i> <i>What is the length of the drill bit needed in the project?</i>
	alternative entities	<i>Can I use a circular saw instead of a jigsaw in step 2?</i>
	time / difficulty / cost	<i>How long does the project take?</i>
	explanation of actions	<i>Can you explain the sawing step in more detail?</i>
	specific location of actions	<i>Where should it be screwed?</i>
Domain Question	alternative actions	<i>Are there other options instead of pre-drilling?</i>
	definition of tool / accessory / material	<i>What is jigsaw?</i>
	related action	<i>What can I do with a jigsaw?</i>
	tips	<i>Is there any safety tip for using a jigsaw?</i>
	structural info	<i>What does a jigsaw look like?</i>
	comparison	<i>How does a jigsaw differ from a circular saw?</i>

Table 1: Selected types of the DIY questions and their examples

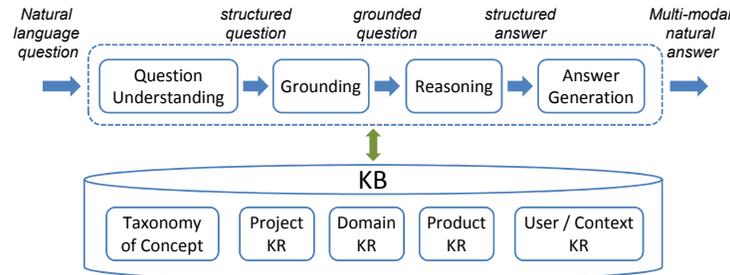


Fig. 1: Architecture of Our QA System

The KB is then used by the reasoner to derive an answer. For most question types, our system converts the question into a SPARQL query using in-house NLP solutions and execute it against the knowledge graph. For some complex question types (e.g., 'alternative entities' in Table. 1), we use advanced AI reasoning techniques³.

3 Evaluation and Discussion

In the pilot study, we compared our system against online search, a common method for information seeking. Specifically, we conducted a user study where 20 users were given a sample DIY project along with 5 questions and were asked to find an answer using online search and our QA system in separate sessions. With online search, the average time of finding an answer was 3.8 min. while our QA system can instantly provide an answer. The users' satisfaction rate with our system was also found to be significantly better than that with online search. In the talk, we also want to share the lessons we learned from this project: pipelined architecture vs. end-to-end architecture, importance of explainability, and the knowledge acquisition bottleneck.

³ See Wang, Y., Lee., J., Kim, D.S.: A logic based approach to answering questions about alternatives in DIY domains, *Innovative Application of Artificial Intelligence (IAAI)*, 2017.