

# Richard: Towards a Dialogue System Supporting Automatic Event Identification

Yibin Jiang  
RWTH Aachen  
Templergraben 55  
52062 Aachen, Germany  
yibin.jiang@rwth-aachen.de

Tiansi Dong, Armin B. Cremers  
B-IT University of Bonn  
Dahlmannstraße 2  
53113 Bonn, Germany

Joachim Köhler  
Fraunhofer IAIS  
Schloss Birlinghoven  
53754 St. Augustin, Germany

## Abstract

*A dialogue system, Richard, for free communication on daily issues is under construction. Daily issues are classified into tree-structured event knowledge-base. For each event topic, we retrieve dialogue examples from the Internet. Basic-level Upper Ontologies (BLUO) are proposed to categorize words of these examples. Event identification is realized by comparing the similarity between BLUOs of an input sentence and BLUOs of example sentences in the event knowledge-base. Cosine similarity algorithm is adopted to select the most possible event domain. BLUOs of an input sentence are combined to the sentence ID. A difference distance acts as a metric to determine the most similar ID in the script with input sentence ID. An example of free dialogue within Greeting-Shopping-Restaurant domains is described. On-going work on meaning-based dialogue system and dialogue-based learning is outlined.*

## 1. Introduction

“A dialogue system is a computer program that communicates with a human user in a natural way” Arora et al. (2013). Dialogue systems have many application areas, e.g. call-center service, shopping assistance, traveling service. Research on human-computer dialogue system can be dated back to Eliza by Weizenbaum (1966). The first dialogue system Eliza was based-on word-level pattern matching method, e.g. Weizenbaum (1966), Norvig (1992). Eliza can act as a psychotherapist to soothe a human patient. Although Eliza system also relies on the context, its strategy is not sufficient for carrying out daily dialogues since people often talk from one context to other context.

A recent milestone would be IBM’s Watson, High (2012), which is based-on statistical reasoning supported by a large knowledge-base. Watson is able to answer quiz questions in English. LogAnswer is a German question-answer system based on formal logic reasoning supported

by the German Wikipedia, Furbach et al. (2010), Dong et al. (2011). For human-computer dialogue in everyday life, the dialogue system might not need a very large knowledge base, however, it shall be aware of the dynamic chances of topics, and adjust its language analysis strategies. Another difference from pure question-answer systems is that a dialogue system sometimes should also raise questions, make suggestions. This is, question answering systems are passive, while dialogue systems can be active. We set out two criteria for dialogue systems for the daily-use as follow: (1) they should be able to automatically identify communication domains; (2) they should be able to accept ungrammatical utterances.

We report an approach to identifying dialogue topics and switching among dialogue domains. Section 2 introduces the basic idea for domain identification; Section 3 describes the tree-structured domain knowledge system; Section 4 presents the cosine similarity algorithm for automatic domain identification; Section 5 elaborates the usage of BLUOs for sentence ID identification in the script; Section 6 describes the Richard dialogue system; Section 7 lists two pieces of on-going work: meaning-based domain identification, and dialogue-based learning of domain knowledge.

## 2. The research on “Basic Level Upper Ontologies” in daily dialogues

Rosch et al. (1976) identified “basic level category” as a certain level of taxonomies of objects, which *carry the most information and the highest category cue validity*. Basic level categories are *the first categorizations which people make during perception of the environment*, and also are *the earliest categories named by children*. What we inspired is that when people change environments, e.g. going out of home, walking in subways, looking around in shopping malls, basic-level words used in their communications would be a valid cue to identify the changes of environments. We carried out a computational research to identify basic-level words and their upper layer ontologies in daily

dialogues, which we name them “Basic Level Upper Ontologies”.

We collect dialogues from Internet, which have been classified under functional domains, such as “daily greeting”, “shopping”. For each domain, we manually go through these dialogues and identify basic-level categories, and construct a word-category dictionary. For the basic-level categories, we identify their upper ontologies, which serve as certain mid-level taxonomies between basic-level categories and the domain. When a new utterance is received, the system will first map the words in the utterance into basic level categories, and reasoning on the possible domain.

So, Basic Level Upper Ontologies are used for (1) normalization of input utterances, (2) identification of dialogue domain, e.g. topics, environment, based on the knowledge-base, and (3) identification of sentence ID in the script. For the utterance “I go to Italian restaurant”, we would map “I” into the category EgoPerson, “Italian” into Nationality, and “restaurant” into FoodPlacePublic. The category name is constructed as follows: an upper category of the word and several characteristic features are combined and ordered alphabetically: “I” is a Person with the feature Ego, “restaurant” is a Place with feature of Food and Public.

### 3. Knowledge of Events and Domains

In the Richard system, domains are understood as families of related event structures. Dialogues start with greeting and end with separating. We introduce the greeting domain as the first domain, and the good-bye domain as the last domain. Domains can be overlapped. For example, shopping domains and restaurant domains may share payment domain in common. Events are spatial-temporal entities, e.g., Zacks and Tversky (2001). Two events can follow 13 qualitative temporal relations, as described in Allen (1983): for example, a greeting event can be directly followed by a going-to-cafeteria event.

Domains are structured in a lattice. Every domain in the lattice is a node. We use a child of a node to represent the sub-domain of a domain. All nodes or leaves with the same parent have the parallel relation. If two domains have the overlapping part, we extract this part from them and put it as a new domain in the subtrees of both domains.

For Richard system, we introduced restaurant domain and shopping domain. As to restaurant domain, it contains before-restaurant, in-restaurant and after-restaurant domains. Under the in-restaurant domain, there are two parallel domains, order domain and payment domain. After-restaurant domain has positive domain which means positive comments on the restaurant and negative domain which means negative comments. As for the shopping domain, it

contains before-shopping, in-shopping and after-shopping domains. There are look domain and payment domain under the in-shopping domain. The structure of domains in Richard system is shown in Figure 1.

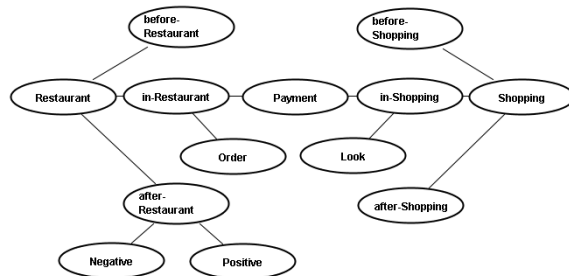


Figure 1. The Sample of Domain Structure

### 4. Automatic Event Identification

The aim of the research is to automatically identify event domain. This is carried out by a deviated cosine similarity function, followed by a decision making process as follows: for each domain, we define a domain ID vector, whose size is the number of all BLUOs within this domain appeared in dialogue corpus, and each vector element is corresponding to a BLUO and set to frequency number of this BLUO in the corpus. In the current dialogue system, the restaurant domain ID vector has the rank of 22, the shopping domain ID vector has the rank of 19. Given an input sentence, we define the input vector as follows: it has the same rank of the domain ID vector; if there is a word in the input sentence belongs to a BLUO in the domain, the corresponding element of the input vector is set to 1, otherwise 0. The input sentence is classified into the domain with highest cosine similarity value. If there are more than two maximum cosine values, among which there is the domain to which the last input sentence belong, we choose the last domain, otherwise, we randomly choose a dialogue domain.

Just an example, suppose that the restaurant domain ID vector has the rank of 3, which corresponds to three BLUOs: Restaurant, Eat, and Hungry, each appears 3, 4, 2 times in the dialogue corpus. The restaurant domain ID vector turns out to be [3, 4, 2]; suppose that the input sentence is “I want to eat”, among those words, “I” belongs to the BLUO of EgoPerson, “want” is neglected, “eat” belongs to Eat. So, the input vector is [0, 1, 0]. The cosine similarity is  $\frac{3*0+4*1+2*0}{\sqrt{3^2+4^2+2^2}\sqrt{0^2+1^2+0^2}} = 0.7428$ . As to the shopping domain, if its ID vector has the rank of 3 as well. The corresponding BLUOs are Shop, Buy, and trolley which appear 3, 3, 1 times respectively in the dialogue

corpus. The shopping domain ID vector is [3, 3, 1]. However, in the above sample sentence, “I want to eat”, none of the shopping domain BLUOs appear so the input vector is [0, 0, 0]. Since in the cosine similarity, the denominator is  $\sqrt{3^2 + 3^2 + 1^2} \sqrt{0^2 + 0^2 + 0^2} = 0$ , the cosine similarity is assigned to 0. As 0.7428 is greater than 0, the sample sentence, “I want to eat”, is regarded as belonging to the restaurant domain. The algorithm for event domain identification is outlined as follows.

```

d = current domain;
while input not empty do
  get BLUOs and their frequencies;
  for number of domains do
    construct input vector based on domain BLUOs;
  end
  calculate correlations by similarity;
  list ← get biggest correlations;
  if d not in list then
    d = random(list);
  else
    end
  end
end

```

**Algorithm 1:** Event Domain Identification Algorithm

## 5. Sentence ID Identification

All BLUOs contained in an input sentence can be combined together as a sentence ID used to choose the response. BLUOs in the sentence ID is reordered alphabetically. In the script, a response is categorized according to an ID of a sentence which that response responds to. We assign a difference distance to different sentence IDs. The difference distance between two IDs are the sum of distinct BLUOs in both IDs. Assume *id1* and *id2* are two sentence IDs and *BLUO()* is a BLUO extraction operator. The formula of the difference distance is below.

$$D(id1, id2) = BLUO(id1) \vee BLUO(id2) - BLUO(id1) \wedge BLUO(id2)$$

For example, the difference distance between *FoodOrder* and *FoodQuantity* is 2. If a sentence ID *id1* has all BLUOs contained in another sentence ID *id2*, we call *id2* is the super-set of *id1* and *id1* is the subset of *id2*. *Food* is a subset of *FoodQuantity*.

In Richard system, we search in a bidirectional way to find one smallest super-set called down-to-up search and one biggest subset called up-to-down search of the input

sentence ID in the script. Then, we calculate the difference distance between the input sentence ID and its super-set and between the input sentence ID and its subset. We choose the super-set or subset based on the smaller difference distance. For example, in the restaurant domain, we have two IDs *FoodDirectionLocationState* and *Location*. The id from the input sentence is *DirectionLocation*. The difference distance between *DirectionLocation* and *FoodDirectionLocationState* is

$$\begin{aligned}
 &BLUO(DirectionLocation) \vee BLUO(FoodDirectionLocationState) \\
 &- BLUO(DirectionLocation) \wedge BLUO(FoodDirectionLocationState) \\
 &= 2
 \end{aligned}$$

The difference distance between *DirectionLocation* and *Location* is

$$\begin{aligned}
 &BLUO(DirectionLocation) \vee BLUO(Location) \\
 &- BLUO(DirectionLocation) \wedge BLUO(Location) = 1
 \end{aligned}$$

Since the difference distance between *DirectionLocation* and *Location* is smaller than that between *DirectionLocation* and *FoodDirectionLocationState*, we choose the rules below the ID *Location*.

```

cd = chosen domain;
tag = BLUOsTag(input);
tag1 = up2downSearch(tag, cd);
tag2 = down2upSearch(tag, cd);
if tag1.length < tag2.length then
  return tag1;
else
  return tag2;
end

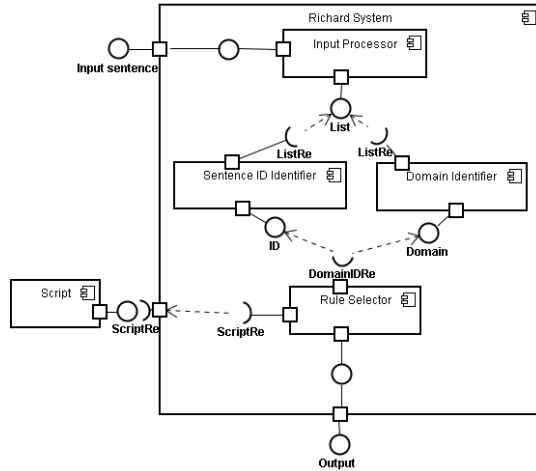
```

**Algorithm 2:** Sentence ID Identification Algorithm

## 6. System Implementation

Sample dialogue sentences are obtained from the Internet. For the presented work, we used sample dialogues at <http://www.englishspeak.com/de/english-lessons.cfm> and <http://www.eslfast.com/robot/>. The model of Richard contains five main components as in Figure 2. Inside Richard, there are input processor, domain identifier,

sentence ID identifier and rule selector. Outside the system, Richard retrieves the rules from a script. The circle handle is a provided interface and the semicircle handle is a required interface.



**Figure 2. The Model of Richard**

We divide the construction of Richard system into two major parts. They are the knowledge-based part and the dialogue part. In the knowledge-based part, we retrieve sample dialogue sentences from the Internet. Based on domains we choose, these sentences are categorized to each of them. Then, we extract the BLUOs from each sentence and construct the domain ID vector for each domain based on the BLUOs of this domain. Finally, we assign each sentence under an ID of a sentence which the sentence assigned by us responds to.

In the dialogue part, when a sentence comes into the input processor, it will be divided into a word list and each word is replaced with its BLUO. This forms a BLUO list sent to both domain and sentence ID identifiers. In the domain identifier, for each domain ID vector, the BLUO list is transformed to corresponding input vector and the cosine similarity between these two vectors are calculated. The domain with the biggest similarity is chosen. In the sentence ID identifier, the BLUO list is formed to a sentence ID. Based on this ID, an up-to-down search finds the biggest subset and a down-to-up search finds the smallest superset. The difference distance calculation is conducted for the most similar ID in the script. At last, a rule selector selects a response under this ID in the chosen domain randomly.

## 7. On-going Work

Based on Richard system, we have two more on-going work. Firstly, our dialogue system is based on the Basic

Level Upper Ontologies defined manually. Human use language to communicate feelings and ideas. A model of ideas can be formed based on the semantic approach Simmons (1973). Therefore, we want to switch the domain in terms of meaning called meaning-based domain identification. The second task is to make the system learn the domain knowledge, which is dialogue-based learning of domain knowledge.

## Acknowledgment

The work reported here was carried out during the “IPEC Winter School 2015 on Speech Technology and Python” from Feb. 2, 2015 to Mar. 31, 2015 at B-IT.

## Sample dialogues with Richard

```
HUMAN>Hi, Richard.
[Domain:GreetingDomain]
RICHARD>Hey
HUMAN>How are you?
[Domain:GreetingDomain]
RICHARD>Fine, how are you doing?
HUMAN>Fine, too.
[Domain:GreetingDomain]
RICHARD>What do you want to do?
HUMAN>I am hungry. I want to eat.
[Domain:BeforeRestaurantDomain]
RICHARD>Let's go to an Italian
          restaurant now.
HUMAN>How about this restaurant?
[Domain:OrderDomain]
RICHARD>Yeah, it is. I come here
          all the time.
HUMAN> What do you want to order?
ANALYSIS> [Domain:OrderDomain]
RICHARD> I'm not sure. I haven't
          decided yet. Can you recommend
          something?
HUMAN> Fish.
ANALYSIS> [Domain:OrderDomain]
RICHARD> Good choice.
HUMAN> Let's pay now.
ANALYSIS> [Domain:PaymentDomain]
RICHARD> Yes, please.
HUMAN>I want to go shopping.
[Domain:BeforeShoppingDomain]
RICHARD>What time are you leaving?
HUMAN>I will leave now.
[Domain:BeforeShoppingDomain]
RICHARD>Will you buy a ham
          sandwich for me at the store?
HUMAN>Ok.
```

[Domain:BeforeShoppingDomain]  
 RICHARD>But I have no money now.  
 HUMAN>You can give me later.  
 [Domain:BeforeShoppingDomain]  
 RICHARD>Thanks.  
 HUMAN> I bought a lot of food.  
 ANALYSIS> [Domain:AfterShoppingDomain]  
 RICHARD> Where did you buy it?  
 HUMAN> In the supermarket.  
 ANALYSIS> [Domain:AfterShoppingDomain]  
 RICHARD> Was it expensive?  
 HUMAN> They are cheap  
 ANALYSIS> [Domain:AfterShoppingDomain]  
 RICHARD> Really?  
 HUMAN>[Abort]  
 Bye! See you next time!

Zacks, J. M. and Tversky, B. (2001). Event structure in perception and conception. *Psychological bulletin*, 127(1):3–21.

## References

- Allen, J. F. (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843.
- Arora, S., Batra, K., and Singh, S. (2013). Dialogue System: A Brief Review. pages 2–5.
- Dong, T., Furbach, U., Glöckner, I., and Pelzer, B. (2011). A Natural Language Question Answering System as a Participant in Human Q&A Portals. In *IJCAI*, pages 2430–2435, Barcelona, Spain.
- Furbach, U., Glöckner, I., and Pelzer, B. (2010). An application of automated reasoning in natural language question answering. *AI Communication*, 23(2-3):241–265.
- High, R. (2012). The Era of Cognitive Systems: An Inside Look at IBM Watson and How it Works. page 14.
- Norvig, P. (1992). Paradigms of Artificial Intelligence Programming: Case Studies in Common Lisp. pages 151–174.
- Rosch, E., Mervis, C. B., Gray, W. D., Johnson, D. M., and Boyes-Braem, P. (1976). Basic objects in natural categories. *Cognitive Psychology*, 8:382–439.
- Simmons, R. F. (1973). Semantic networks : Their computation and use for understanding english sentences. *Computational Models of Thought and Language*, pages 63–113.
- Weizenbaum, J. (1966). ELIZA—a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9:36–45.