Animated Agents Capable of Understanding Natural Language and Performing Actions*

Hozumi Tanaka[1], Takenobu Tokunaga[2], and Shinyama Yusuke[3]

[1] Department of Computer Science, Tokyo Institute of Technology
tanaka@cl.cs.titech.ac.jp
[2] Department of Computer Science, Tokyo Institute of Technology
take@cl.cs.titech.ac.jp
[3] New York University yusuke@cs.nyu.edu

Summary. This chapter describes a system called Kairai and its Natural Language Understanding (NLU) capabilities. It identifies its strength and shortcomings and identifies requirements for future NLU systems. The NLU research environment has changed drastically in the past two decades. Better technologies in speech recognition, natural language processing and computer graphics are now available and make us much easier to develop a life-like animated agent (a software robot) who can understand commands in spoken language and perform actions specified by the commands. Combining these technologies, a life-like animated agent system named Kairai was developed at our laboratory to conduct preliminary research on an NLU system. Although Kairai includes many innovative features, several important problems hindering the building of a better NLU system still remain. After describing several issues the Kairai system can handle, we will conclude by outlining what problems we have to solve in the future. The results obtained from our research should be naturally applicable to hardware robots.

1 Introduction

Historically, the most important Natural Language Understanding (NLU) system was SHRDLU developed by Winograd at MIT in the early 1970's [31]. This system was a kind of a software robot that worked in a toy block world simulated in a virtual space. However, rather than head, feet and hands, the robot was equipped with only a small stick. SHRDLU was not regarded as a life-like animated agent, but it has all the distinctive features. It could understand English dialogue input from keyboards (no speech input) according

* This is a Grant-in-Aid for Creative Scientific Research supported by th Ministry of Education, Culture, Sports, Science and Technology, and Japan Society for the Promotion of Science, and is the extended version of the original manuscript which was published at International Workshop on Life-like Animated Agents in PRICAI-02, pp. 89-94, August, 2002.

to which it carried out very simple tasks such as "Pick up a red block on the table" and "Put it in the green box" by building its action plan and executing it. The system could also answer simple queries about the current state of the toy block world. It could resolve anaphoric ambiguities in the input sentence. SHRDLU demonstrated the promising future of NLU research at that time.

Recently better technologies have been obtained in speech recognition and natural language processing. Furthermore, significant progresses in the field of computer graphics have enabled to generate complex and realistic 3-D animated robots (or software robot/agents) in a virtual space. The authors are now in a good position to go beyond the SHRDLU system. Two typical related works are reviewed briefly.

Badler et al. [4, 5] built 3-D animated agents who could take commands and perform adequate actions in a virtual space. The agent was given commands from which it extracted parameters for its actions. The parameters contains various information such as linguistic, spatio-temporal, manner information that was often expressed as adverbs, and in some cases both applicability and terminating conditions. However, although it is normally very important to handle ellipsis and anaphoric expressions, which often occur in spoken language, they pay little attention to these expression types [3].

Later, Cassell et al. [6] pointed out that conversational skills are not only limited to the ability of language understanding and language usage but also non-verbal behavior such as using facial expressions, hands and tone of the voice to regulate the process of conversation. They developed a system called *Rea system*, which was an embodied conversational agent with social, linguistic, and psychological conversation conventions. *Rea*, an agent with a human-like body, can respond to humans using eye gaze, body posture, hand gestures or facial expressions. While the *Rea* system emphasizes the importance of non-verbal functions in conversations, the system does not handle the problem of vagueness in agent actions.

Sect. 2 will discuss the reason why we choose software robots instead of hardware robots along with considering their advantages and disadvantages. We explain our Kairai system in Sect. 3 together with sample dialogues. Although Kairai operates in a very limited task-oriented domain, it makes proper interpretations for such adverbs as "left" and "right" as well as anaphora resolutions, which is discussed in Sect. 4 where our new method will be introduced. In Sect. 5, we will discuss some problems in the Kairai system, most of which should be solved by any future NLU system, and then we will explain why a one-to-many conversational pattern is important and consider more in the future. According to the empirical study on the *Kairai system*, we will discuss about the next generation NLU system in Sect. 6.

2 Why Software Robots Instead of Hardware Robots?

Before going into the discussion of the Kairai system, this section explains the reason why software robots instead of hardware robots are used.

Firstly, even though hardware robots have made a rapid progress recently, the actions they can perform are still too limited due to mechanical limitations. Compared with hardware robots, software robots have capabilities to carry out complex movements including non-verbal actions such as laughing, crying, nodding, and so on. As it is desirable to issue rather complex natural language commands to software robots, they are more suitable for the NLU research in conjunction with action performing tasks.

Secondly, we do not want to deal with the vision problem, which is indeed one very important but difficult problems for hardware robots besides moving and performing actions in the real world. We are going to concentrate on the NLU problem not bothered with the vision problem. With respect to software robots, it is not necessary for us to be worried about such a problem, since knowing everything in a virtual space/world is possible without sensory devices.

Thirdly, it is easy to create a multi-agent environment by simply making copies of many software robots in a virtual space. Therefore, software robots let us study multi-agent systems easily. On the contrary, it is not only difficult but also expensive to create a multi-agent environment with hardware robots.

Finally, hardware robots often have difficulties due to mechanical problems. Hardware robots have to be kept in a good condition through frequent hardware maintenance. The higher the number of hardware robots, the more frequently such mechanical troubles occur. This contrasts sharply with software robots for which no worry about such troubles is necessary.

The four reasons listed above are all benefits of adopting software robots instead of hardware robots. However, there are several drawbacks in using software robots. Each software robot has to simulate the Newtonian physical world in order to move in the virtual space/world. To solve the Newtonian physical equations in generating robot's movements is cumbersome and computationally intensive enough to justify the use of stereotyped motion patterns accumulated from motion capturing devices.

Another difficult problem for software robots is the so-called "frame problem" [20, 23], which each autonomous software robot has to solve (See Sect. 4). However, this is also a problem for hardware robots building a task plan before carrying out their actions.

3 Kairai System

For the feasibility study on the next generation NLU system tightly combining speech recognition, NLU and computer graphics, authors have developed a prototype NLU system called Kairai [24, 25, 27, 28].

3.1 Architecture of Kairai System

The Kairai system incorporates several 3-D software robots with which a user can converse. It accepts voice input (spoken input), interprets them and performs the tasks in the virtual space. The dialogues become task-oriented ones [7, 12, 31].

The task executions are visible on a display screen as an animation. There are four software robots in the Kairai system. In addition to three visible software robots, a horse, a chicken, and a snowman. A cameraman is also a software robot controlling his camera to give a different perspective of the virtual space. The cameraman and his camera are invisible on the display screen and the camera handling is specified through commands such as "Go near the horse." In consequence, the figure of the horse is enlarged.

Kairai understands what we say in natural language, especially the words such as "left", "right", "in front of" and "behind" that indicate relative location in a virtual space. Typical actions performed by the software robots are "push", "go", and "turn." It is interesting to observe that interpretations of "left" and "right" are determined by considering both the position and orientation of a software robot and objects in the virtual space in addition to a view of the human who issued the command.

Fig. 1 shows the outline of the Kairai system whose architecture is not new and is divided into three parts: speech recognition module, NLU module, and animation generation module. The speech recognition module transforms speech input into a sequence of words that become input to the NLU module. The NLU module then analyzes the input by using both a grammar and a dictionary, and extracts a meaning structure called a frame structure along with anaphora resolution and ellipsis handling. The latter two form discourse processes that refer to the context of past history of the dialogues between the human (user) and Kairai. After a task plan is created by the NLU module, it is forwarded to the animation generation module to yield an animation on the display. The animation is generated by Alice[4] which interprets programs written in Python, a scripting language. To visualize agents' actions we also have to solve the problem of vagueness, for instance how far the robot ought to move. The current version of the Kairai system solves this problem by simply taking a default value.

Fig. 2 is a snapshot of an animation generated by the Kairai system. Readers can see three software robots in a virtual space. According to the commands provided by the human, software robots including a cameraman can move and perform appropriate actions in a common space.

3.2 Sample Dialogue with Kairai

A typical dialogue that Kairai can understand is shown below. As stated above when discussing the virtual space, there are four software robots (animated

[4] http://www.alice.org/

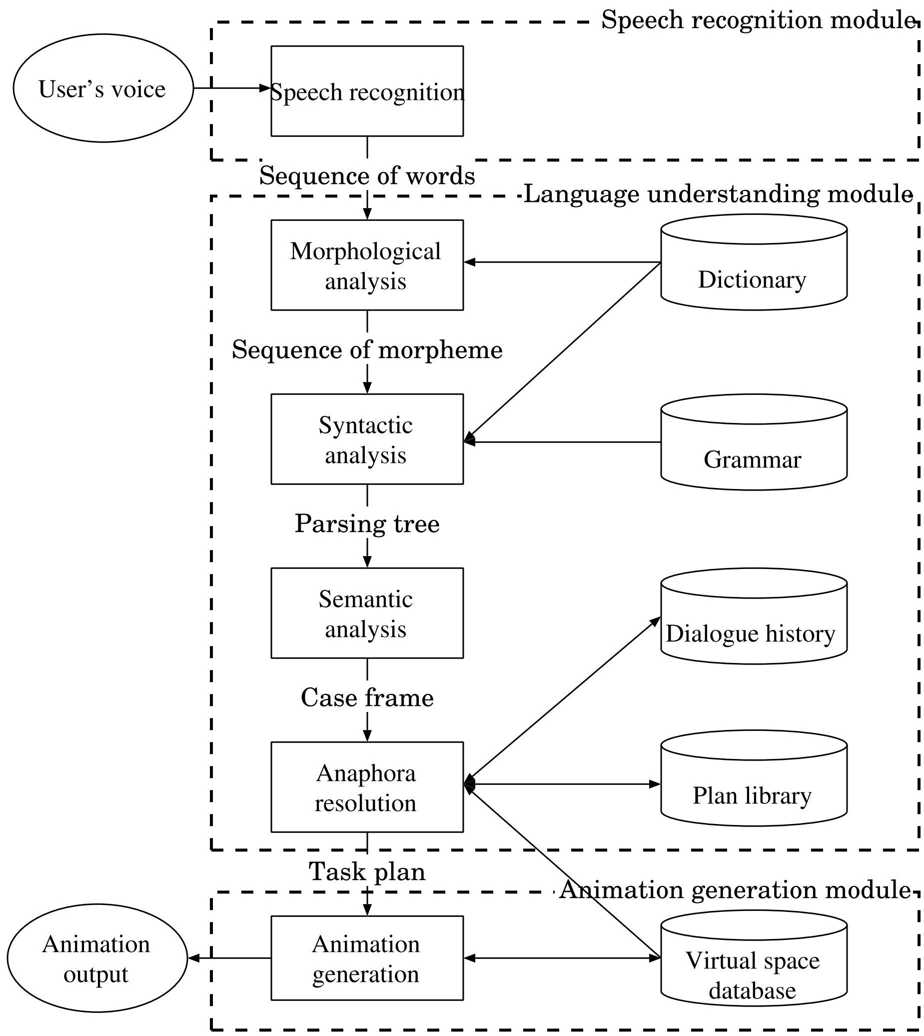

Fig. 1. Outline of Kairai System

agents), a horse, a chicken, a snowman and a cameraman, the last of which is invisible but manipulates his camera to take the view of the current virtual space. In addition to them, we purposely put two red spheres and two blue spheres bringing about an interesting problem named *object identification problem* of a *deictic expression*. Consider the command "Push a red sphere." In order to perform the action, each software robot has to decide which "red sphere" is meant by the command. A reasonable answer will be the red sphere

Fig. 2. A Snapshot of Kairai (*Horse, push the blue sphere to the front of Chicken.*)

which is visible and nearest to the robot. Through voice input provided by the user, Kairai accepts imperative sentences one by one.

1. Human: *Horse, push the blue sphere to the front of Chicken.*
   > The phrase "*the blue sphere*" is ambiguous, since there are two blue spheres in the virtual space. *Horse* has to decide which blue sphere the command actually indicates through solving the *object identification problem* by making refernce to the current state of the virtual space. Suppose he picks up "*the blue sphere*" nearest to him and does the push action.
2. Human: *Push the red sphere, too.*
   > Considering *Horse*'s view point, Kairai decides which red sphere *Horse* should push and then the *Horse* performs the action. Note that *the red sphere* is an example of a *deictic expression.*
3. Human: *Chicken, push it, too.*
   > Kairai resolves the anaphoric ambiguity given by *it* using a history of context, namely the preceding commands. Furthermore, he/she has to solve the *object identification problem* again, before carrying out his/she action. In this case, *it* indicates the red sphere, which *Horse* previously pushed. *Chicken* does the action.
4. Human: *Further.*
   > Although there is no subject, no object and no verb, Kairai augments these elliptical words by considering the context accumulated through

the dialogue between the user and Kairai. Kairai forces the *Chicken* to push the red sphere further. Due to the visualization, it is necessary for Kairai to determine how far the *Chicken* should push the sphere. This problem is called *vagueness* by linguists. Kairai successfully carries out anaphora resolution and ellipsis augmentation in this case. Kairai simply uses a default value to handle the vagueness problem but it is certainly an unsatisfactory solution.

5. Human: *Cameraman, move close to the red sphere.*
   > Kairai makes the camera move close to the red sphere mentioned before. As the result, it zooms in on the red sphere and changes the view of the virtual space.

4 Plan-based anaphora resolution

The importance of situation dependent NLP (SDNLP) should be emphasized to construct an NLU system. Anaphora resolution is one of such problems. In the domain of task-oriented conversation such as that seen in Kairai, a user issues a sequence of commands to indicate a goal that a system has to achieve. As each command in the sequence usually states a subgoal, constructing a sequence of subtask plans becomes very important. According to the task plan execution, agents are trying to satisfy the goal step by step by achieving its subgoals. As mentioned in [7, 8, 15, 16], the plan-based approach was empirically recognized as very useful for relating task plan execution and understanding task-oriented dialogues.

Most of these efforts were focused on analyzing speaker's intentions through plan recognition. However, Cohen [7] discussed the referent identification with the assumption that speakers give their commands to listeners who could easily identify corresponding referents. There was no serious attempt to deal with plan-based anaphora resolution in [8, 19].

Consider a fragment of a dialogue to explain our plan-based anaphora resolution method.

(1) *Agent X, pick up the red ball.*
(2) *Move to the front of the blue ball.*
(3) *Drop it.*

The pronoun *it* in (3) refers to the *red ball* and not the *blue ball* in the preceding command (2). After executing plans specified by (1) and (2), some of the *Effects* will be expressed roughly as,

Effect: above(red-ball,ground) from (1),
Effect: adjacent-to(agent-X,blue-ball) from (2).

The above two *Effect*s show the situation change after executing two consecutive actions mentioned in (1) and (2). These *Effect*s become a part of

*Precondition*s for constructing the next “drop” plan, since we would like to keep coherence between adjacent plans. Note that the *Effect: above(red-ball,ground)* must hold even after executing the “move” action specified by (2), which is called the *frame problem* in the field of Artificial Intelligence. The “move” action does not change the *above(red-ball,ground)* relation. After the “drop” action specified by (3) is finished, the relation does not hold any more. In the virtual world, software robots/agents have to take account of what changed and what remains unchanged after executing some action.

Anyway, before executing the plan specified by (3), software robots have to check its *Precondition*s, that is whether or not *above*(*it*, *ground*) holds. One of the *Precondition*s will be satisfied if and only if *it=red-ball* which software robots realize by looking for the *Effect*s previously accumulated through dialogues. Consequently, the robots can solve the anaphora resolution as *it = red-ball*. Interestingly, this kind of anaphora resolution is not possible when we adopt the centering theory advocated by [13, 14, 26], since the theory focuses mainly on surface level linguistic cues. Our plan-based approach adopted in our Kairai system utilizes deeper NLU results like *Effect*s representing the situation or state change after the execution of plans and actions.

In case of handling a huge volume of texts as fast as possible, it prefers centering theory to our plan-based anaphora resolution even though the former is one of anaphora resolution method using only superficial linguistic information. However, in the task-oriented domain like Kairai, investigating the plan-based anaphora resolution method based on deeper NLU understanding seems to be promising and we are going to do more research on the resolution method.

5 Experiences with Kairai system

The experiences with respect to the Kairai system which was developed as a prototype system, made us not only realize many further problems but also extract important research themes for any future NLU system.

5.1 Problems

.

The first and the most important problem is that Kairai was not really a multi-agent system composed of autonomous agents [10, 17, 30].

As each software robot (agent) in the Kairai seems to carry out its action independently, the Kairai system, at a glance, seems to be a multi-agent system. However, the Kairai system is not an actual multi-agent system but a one-to-one communication/conversation system. In addition to four software robots/agents mentioned before, there is another special agent who knows everything about the virtual space, receives and processes a sequence of words sent by the speech recognition module. This is illustrated in Fig. 3.

After accomplishing NLU tasks, the special agent decides which software robot should perform what kind of actions and then activates an appropriate software robot. This is the reason why Kairai is not really a multi-agent system. The problem discussed here brings about another problem.

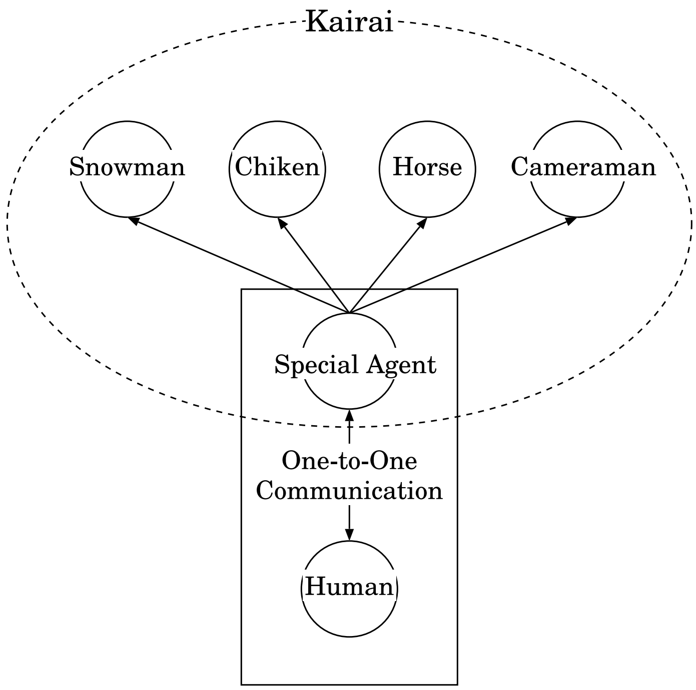

Fig. 3. One-to-One Communication in Kairai

Since current software robots in Kairai are not autonomous, it is very difficult to conduct *cooperative* actions including several robots. Consider "gazing" [22], a simple cooperative action. In the current Kairai system, even though a software robot is conducting a task in the virtual space, the other robots are not paying any attention to his action. In human society, it is natural for a person to gaze at another one working near him/her. These were already implemented in [6], but due to the absence of autonomous robots in the current version of Kairai, it is impossible for any robot to directly communicate with another. Problems of gazing as well as the other cooperative actions can be solved naturally by introducing autonomous robots and one-to-many communication/conversation mode in the virtual space[5]. Making software robot autonomous is actually not a difficult problem when describ-

[5] See also the chapter of M. Mateas & A. Stern in this book.

ing it in a concurrent programming language. The difficulty really resides in making each software robot perform cooperative works in a virtual space by using shared plans etc. [16].

Currently, Kairai does not deal with non-verbal expressions including intonation in speech, gazing, facial expressions, and body actions as well as hand gestures. Facial expressions are related to emotional behaviors. It is well-known that non-verbal expressions play an important role in communication [6]. We would like to account for such non-verbal and para-linguistic expressions but it remains a challenging research topic for the future Kairai system as well as the other similar systems. Fortunately, compared to hardware robots, software robots can emulate para-linguistic and emotional behavior much easier since they do not have any mechanical limitations.

Kairai is not equipped with the ability to reason about speaker's intentions as well as speaker's non-verbal expressions that several researchers were trying to solve along with plan recognition [6, 7, 8, 15]. More research is needed before it is incorporated in any system like Kairai.

Finally, software robots in the Kairai system cannot communicate with each other. Each software robot performs actions in a virtual space following commands issued by a user. It is desirable to endow robots with the ability to answer back or ask a question when needing to resolve ambiguities which cannot be solved directly. Such ability will be very helpful for agents required to perform cooperative works in a virtual space, too. We would like to leave this problem as one of future research themes to improve the current Kairai system.

5.2 One-to-Many Communication

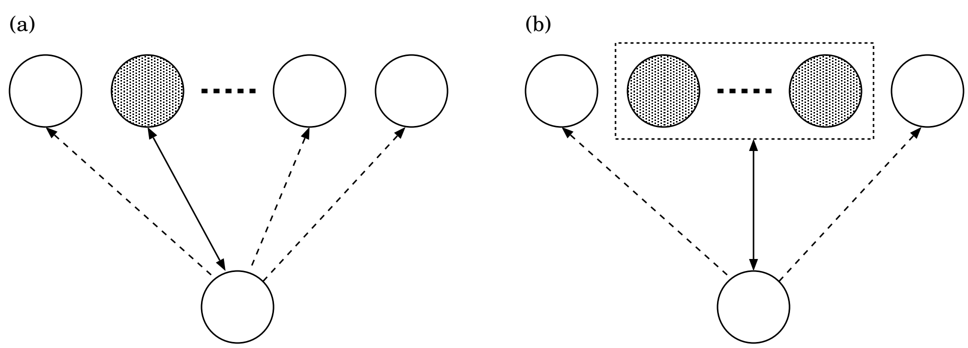

Fig. 4. One-to-Many Communication

NLU systems should deal with a one-to-many communication/conversation together with one-to-one conversation. One-to-many conversation makes sense in a multi-agent environment [10, 30]. In one-to-one conversation it is easy to decide who is the intended listener. On the contrary, in the one-to-many conversation as shown in Fig. 4, due to many potential listeners, it is difficult to decide to whom a command issued by a speaker is intended. Usually, the listener is mentioned explicitly in the first utterances of the dialogue, but he/she is not mentioned in the rest of the dialogue. Confusion can happen between agents if each agent is unable to recognize who is the addressed agent required to perform some tasks according to the command given by a speaker. The problem also occurs when a subject or an object does not appear in the command due to ellipses.

To understand the above situation clearly, consider the following conversation in a multi-agent environment.

1. Agent X: *Hey, Agent A, I will throw a red ball.*
   > *Agent A* looks at *Agent X*. The other agents might also gaze at *Agent X*.
2. Agent X: *Catch it.*
   > Even though there is no subject in this command, *Agent A* begins the action to catch the red ball. Normally, the other agents just gaze at *Agent A* or *Agent X*.

Note that the second command issued by *Agent X* lacks a subject, but *Agent A* has to perform the action *Catch* along with resolving the anaphora expression *it* that should be correctly identified as the red ball. Normally, the other agents should not perform the action *Catch* even though they hear the *Catch it* command. If the above conversation occurs in an American football game, both allies and enemies are going to catch the same ball. However, the meaning of actions taken by enemies is completely different from that of allies' actions. That is, enemies are going to intercept the red ball. In the multi-agent environment, such an interesting phenomenon will often happen. It seems obvious that each software robot should be autonomous in the multi-agent environment and have the ability to control his behavior by himself.

6 Next Generation NLU System

As discussed in the preceding section, many problems were extracted from the Kairai system and most of them are also important research themes for any future NLU system. We would like to summarize them in this section.

At first, consider a fragment of a one-to-one dialogue as shown below:

1. Human: *Open the curtain covering the window a little.*
   > An agent goes to the curtain, and grasps it by his/her hand and opens it.
2. Human: *A little bit more.*
   > The agent opens the curtain a little bit more.

3. Human: *Too much.*
   > The Agent closes the curtain a little.
4. Human: *The Air in the room is polluted.*
   > The agent opens the window.

The first command issued by the Human makes an agent create a plan to go to the curtain, grasp and open it. Such a plan is called a macro level plan. There are many ways to grasp and open the curtain and we have to solve the so-called problem of *vagueness* by linguists. The agent has to select one of the possibilities to generate a micro level plan in order to carry out his/her actions. We can conclude whether the agent understands natural language by observing the agent's actions corresponding to what the Human says. In other words, an agent's actions, which are visualized in a virtual space as an animation, verify the NLU ability of the agent. That is, the visualized actions provide us with a more precise NLU evaluation method than that of the Turing test, since the latter does not take the visualized behavior of AI systems into account [5].

The second and third commands lack a verb in addition to a subject and an object. Agents have to augment these elliptical words by taking account of the context of the current dialogue and environment in which the robot resides. With respect to the second command, the agent has to infer "open" as an appropriate elliptical verb, and then carries out the action "open." On the other hand, in the third command, "open" is also the correct elliptical verb, but the agent has to perform an opposite action "close" in this case. In other words, the agent has to extract the intended actions from indirect speech act commands [2, 9]. The second and third commands are also related to the problem of the *vagueness*, which was often overlooked in past NLU research.

The fourth command includes a typical indirect speech act that should be understood correctly for an agent to perform the "open the door" action. Extracting speaker's intentions in an indirect speech act is one of the most difficult but interesting problems discussed by many researchers in the past [1, 9, 8, 15].

In addition to those caused by erroneous output from automatic speech recognition systems, there are many ill-formed sentences, which include fillers, additions, repairs and repetitions in spoken language. It might be possible to cope with these problems by using constraints obtained from a language processing module rather than from an acoustic processing module [11, 18, 21, 29].

From the empirical study based on the Kairai system, we can summarize what the next generation NLU system must be equipped with.

1. Situation dependent natural language processing (SDNLP) that includes:
   a) Resolution of anaphoric expressions by combining the centering theory and our plan-based approach.
   b) Identification of objects specified by a deictic expression.

   c) Augmentation of ellipsis.
   d) Extraction of indirect speech acts as well as speaker's intention from commands.
   e) Dealing with *vagueness* which is especially critical for the hardware robots that move around and perform actions in the real world.
2. To handle ill-formed sentences that include fillers, additions, repairs and repetitions, which are frequently included in spoken language.
3. To perform cooperative actions as well as one-to-many conversations in a multi-agent environment.
4. To take into account non-verbal and para-linguistic expressions as well as emotional expressions in order to enhance flexible communication through natural language dialogues.

As explained in Sect. 2, our Kairai System will provide us with a platform for the research on multi-agent system.

7 Conclusions

After reviewing the NLU systems in the past, we pointed out several important issues concerning the creation of a next generation NLU system. Kairai, a system which was developed at Tokyo Institute of Technology, played a key role in exemplifying the issues mentioned in the preceding sections.

Even though the system under consideration consists of a set of software robots, the research results are also applicable to hardware robots. We have also discussed the importance of para-linguistic expressions in addition to emphasizing better algorithms for anaphora resolution and ellipsis handling. Furthermore, we would like to emphasize the importance of the *object identification problem* as mentioned in Sect. 3.2, since without identifying the target object, robots cannot perform any action with the object in both real and virtual space. This problem seems to be overlooked in the previous research. Additionally, processing ill-formed sentences which frequently occur in spoken language is also an important issue requiring attention.

The next generation NLU system ought to be a multi-agent system, with a wide array of application areas such as:

1. Entertainment,
2. Helper robots (medical and in-home use),
3. Tutoring systems,
4. Sign language systems,
5. Virtual space navigation systems,
6. Electrical appliances, etc.

Readers easily understand the application areas of entertainment and helper robots, although the latter will need better and more reliable technologies in the future. With respect to tutoring systems, instead of reading a manual, we can understand easier for a software robot (or a pedagogical agent) can instruct us on how to manipulate devices by showing its 3-D models [22].

Navigation guided by voice commands in natural language can possibly apply to any kind of virtual space. Typical examples are 3-D models of internal organs, molecular structure, DNA structure, and geographic space, etc. The recent CT (Computer Tomography) and MRI (Magnetic Resonance Imaging) medical technology enable us to construct 3-D images of internal organs where a virtual camera can enter and move close to any part by receiving navigation commands in spoken language. The authors expect such technologies will change the current method of medical diagnosis in the near future.

Finally, the future electrical appliances will be equipped with "ears" for listening to user's commands and will process these commands to execute them similar to current software robots. In such circumstances, the research on both multi-agent system and one-to-many conversation system will become increasingly important.

Acknowledgments

Thanks to Dr. Helmut Prendinger, Dr. Taiichi Hashimoto and Slaven Bilac for their help to prepare this manuscript.

References

1. J. Allen and C.R. Perrault. Analyzing intention in utterances. *Artificial Intelligence*, (15):143–178, 1980.
2. J.L. Austin. *How to Do Things with Words*. Oxford University Press, New York, 1962.
3. N. Badler. A parameterized action representation for virtual human agents. In *Proceedings of the 1st Workshop on Embodied Conversational Characters*, pages 1–8, 1998.
4. N.I. Badler, M.S. Palmer, and R. Bindinganavale. Animation control for real-time visual humans. *Comm. of the ACM*, pages 65–73, 1999.
5. N.I. Badler, C.B. Phillips, and B.L. Webber. *Simulating Humans - Computer Graphics Animation and Control*. Oxford University Press, 1993.
6. J. Cassell, T. Bickmore, L. Billinghurst, L. Campbell, K. Chang, H. Vilhjalmsson, and H. Yan. Embodiment in conversational interfaces: Rea. In *Proceedings of CHI'99 Conference*, pages 520–527, 1999.
7. P.R. Cohen. The pragmatics referring and the modality of communication. *Computational Linguistics*, 1984.
8. P.R. Cohen and H.J. Lavesque. Persistence, intention and commitment. In *Intentions in Communication*. The MIT Press, 1990.
9. P.R. Cohen, J. Morgan, and M.E. Pollack, editors. *Intentious in Communication*. The MIT Press, 1990.
10. J. Febler. *Multi-Agent Systems - An Introduction to Distributed Artificial Intelligence*. Addison-Wesley Longman, 1999.

11. K. Funakoshi, T. Tokunaga, and H. Tanaka. Processing Japanese self-correction in speech dialog system. In *Proceedings of the COLING 2002*, pages 287–293, 2002.
12. B.J. Grosz. Discourse knowledge. In D.E. Walker, editor, *Understanding Spoken Language*, chapter 4, pages 229–344. North-Holland, 1978.
13. B.J. Grosz, A.K. Joshi, and S. Weinstein. Providing a unified account of definite noun phrases in discourse. In *Proceedings of the 21th Annual Meeting of the Associaton for Computational Linguistics*, pages 44–50, 1983.
14. B.J. Grosz, A.K. Joshi, and S. Weinstein. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–226, 1995.
15. B.J. Grosz and C.L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12(3), 1986.
16. B.J. Grosz and C.L. Sidner. Plans for discourse. In *Intentions and Communication*. The MIT Press, 1990.
17. M.N. Huhns and M.P. Singh, editors. *Readings in AGENTS*. Morgan Kaufmann, 1998.
18. J.-C. Junque and G. van Noord, editors. *Robustness in Language and Speech Technology*. Kluwer Academic Publishers, 2001.
19. D.J. Litman and J.F. Allen. Discourse processing and commonsense plans. In *Intentions and Communications*. The MIT Press, 1990.
20. G.F. Luger and W.A. Stubblefield. *Artificial Intelligence*. Addison-Wesley, 3rd. edition, 1998.
21. C.S. Mellish. Some chart-based techniques for parsing ill-formed input. In *Proceedings of 27th ACL*, pages 102–109, 1989.
22. J. Rickel, Ruth Aylett, and Daniel Ballin, editors. *Intelligent Virtual Agents for Education and Training: Opportunities and Challenges*. Springer, 2001.
23. S. Russell and P. Norvig. *Artificial Intelligence*. Prentice-Hall, 2nd edition edition, 1995.
24. Y. Shinyama, T. Tokunaga, and H. Tanaka. "kairai" - software robots understanding natural language. In *Proceedings of the 3rd Workshop on Human Computer Conversation*, pages 158–163, 2000.
25. Y. Shinyama, T. Tokunaga, and H. Tanaka. A software robot kairai that understands natural language (in Japanese). *Transactions of IPSJ*, 42(6):1359–1367, 2001.
26. M. Strube and U. Hahn. Functioning centering–grounding referencial coherence in information strucure. *Computational Linguistics*, 25(3):309–344, 1999.
27. H. Tanaka. *Language Understanding and Action Control (in Japanese)*. Ministry of Education and Science, 2000.
28. H. Tanaka. *Language Understanding and Action Control (in Japanese)*. Ministry of Education and Science, 2001.
29. G. van Noord. Robust parsing of word graphs. In J.-C. Junque and G. van Noord, editors, *Robustness in Language and Speech Technologys*, pages 205–238. Kluwer Academic Publishes, 2001.
30. G. Weiss, editor. *Multiagent Systems*. The MIT Press, 1999.
31. T. Winograd. *Understanding Natural Language*. Academic Press, 1972.

Index

anaphora resolution, 7
autonomous agent, 8

centering theory, 8

deictic expression, 5, 6

ellipsis, 7

frame problem, 8

hardware robot, 3

ill-formed sentence, 12
indirect speech act, 12

Kairai system, 2, 4, 8, 10, 12

multi-agent system, 3, 8

Natural Language Understanding (NLU), 1
next generation NLU system, 12
non-verbal expression, 10

object identification problem, 5, 6
one-to-many communication/conversation, 9, 10
one-to-one communication/conversation, 8

plan-based anaphora resolution, 7

Rea system, 2

SHRDLU, 1
situation dependent natural language processing (SDNLP), 12
software robot, 3
spoken language, 12

vagueness, 7, 12