Ontological Approach for Generating Natural Language Texts from Knowledge Base

Longwei Qian, Wenzu Li
Department of Intelligent Information Technology
Belarusian State University of Informatics and Radioelectronics
Minsk, Belarus
qianlw1226@gmail.com, wzzggml@gmail.com

***Abstract*—The computer systems are expert in dealing with structured data. However, when computer systems attempt to impart information to the end-users, generating natural language text that expresses structured data is a significant challenge. Currently graphical knowledge representations as a kind of forms to represent structured data are gradually becoming universal in computing. In this paper, we present a unified semantic model for generating fluent, multi-sentence, appropriate natural language text (e.g., Chinese language text) from knowledge base to the end-users. This article describes the development of semantic model for natural language generation, and the optional linguistic ontologies which may be used in the processing of generation. The main novelty is that it is possible to integrate different approaches and linguistic knowledge to generate natural language text from the structured data of the computer systems represented in graph form. For the ordinary end-users it will be an easier access to the information in the computer systems.**

***Keywords*—natural language generation, ontology, knowledge-driven, knowledge base, OSTIS**

I. INTRODUCTION

A. Objective and Relevance of the Work

The goal of this article is to use ontology-based approach as a basis to develop a unified semantic model for generating natural language text from the knowledge base. Generating natural language text is considered as a part of natural language interface that is based on the technology of ontology-based design of intelligent system user interface [1]. The technology of ontology-based design of intelligent system user interface itself is a part of Open Semantic Technology for Intelligent Systems Design (OSTIS Technology). Natural language interface focuses on processing of natural language text, taking into account methods and principles of user interface design. The proposed approach involves the development of natural language interface formal ontology. One of the advantages of this technology is the possibility to introduce various knowledge for generating natural language text.

Natural language processing is considered as a cognitive problem in the field of artificial intelligence. Many researchers have tried to address the problem of natural language understanding, in contrast, there are not many researches on issues of natural language generation (NLG). Although the number of works in NLG is fewer, their content varies due to the difficulty to precisely define NLG. Now everybody agrees that the output of NLG should be natural language text, but the exact form of the input can vary substantially. Generally speaking, the input can be divided into unstructured textual forms, semi-structured textual forms and structured forms (e.g., tabular data, knowledge base). According to this point, the machine translation, text summary, data-text generation, as well as generation from visual input such as image or video entirely are instances of NLG. The difficulties in terms of NLG usually come from following two aspects:

- Analysis and processing of the input information forms;
- Analysis of features of the target natural language text that was generated by intelligent system.

Nowadays the application of intelligent system is ubiquitous. One of the most important research directions for designing intelligent system is the development of knowledge-based system, in particular after the concept of knowledge graph was proposed by Google [2]. In order that this kind of intelligent system can interact with users more conveniently, it's necessary for the system to have the capacity to understand the knowledge base and generate grammatically reasonable natural language text for the users. It had become possible with the increase of computing power and the proposal of the novel models. However, due to the various graphical structure and the optional multilinguality for the output, the generation of natural language text is still an open challenge.

B. Problems, Need to be Solved

Firstly, our goal is the development of natural language interface. Secondly, there are lots of models, methods and means to generate natural language text from structured data. They are successfully applied as components of natural language interface of intelligent system. Nevertheless, the following problems still need to be considered:

- Absence of unification of development process of natural language interface, thus development process of natural language interface cannot be achieved in parallel, as well as we cannot reuse the already developed component;
- The difficulties in analysis and processing of knowledge representation structure. The knowledge is not stored discretely in the intelligent system; it is interconnected to form a structured data in the system;
- There is no ability to use unified tools to represent various kinds of knowledge (e.g., the linguistic knowledge), which is necessary to provide useful information for NLG;
- Developed problem solvers are not flexible at each stage of NLG, e.g., for generating text in a new language it's difficult to change the already existing components or to extend NLG component to adapt.

The relevance of these problems will be explained in detail by analyzing related work to natural language generation.

C. Analysis of related work for natural language generation

In this paper, we focus on the NLG part of natural language interface. The NLG part can be considered as a separated system, or as a component of user interface. But it's difficult to achieve the development of separated NLG system that can be reused into the component of user interface due to absence of unified principle for user interface design. The NLG part in this paper tends to generate natural language text from structured form, in particular, from knowledge base.

Early in the application domain, the successful NLG system includes the weather reporting, “robo-journalist” and so on, which convert the tabular data or data of information box into reasonable natural language text by filling placeholders in a predefined template text [3]. In this situation, the text generated on the base of these predefined template is very simple and inflexible. The rule-based approaches convert data into resulted text by a serious of grammar and heuristic rules. The factors influencing these approaches to generate natural language text are the linguistic rules. For analysis of natural language, linguists proposed many linguistic theories that focus on interpreting linguistic formalism, for example, the dependency grammar is used for the interpretation of the syntactic structure. However these approaches lack supporting of unified basis for representing various linguistic knowledge. Moreover, traditional rule-based approaches focus not on the semantic of natural language text, but rather on the syntax.

There is a classic pipeline architecture [4] for NLG. Based on this pipeline, generally, the six tasks are frequently found in NLG system (Fig. 1). In fact, the classic pipeline architecture can be considered as the modular approach to solve the NLG problem. Different modules in the pipeline incorporate different subsets of the tasks described above. However, the biggest problem for the ordering of the modular approach is the generation gap [5] that refers to the mistakes of early tasks in the pipeline passed further downstream.

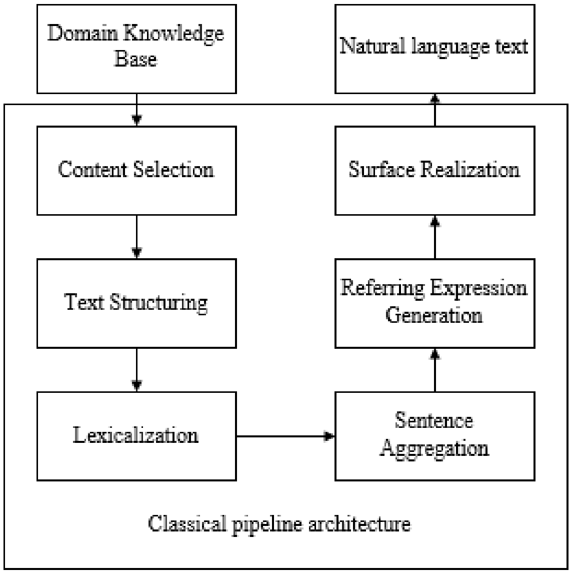

Figure 1: Classic pipeline of natural language generation

The pipeline continues influence on alternative architectures that were proposed in recent years for NLG. The proposed approaches often end up blurring the boundaries between modules. These approaches tend to emphasize statistical methods that is data-driven. From the pipeline perspective, there are three simplified steps:

- content selection;
- content planning;
- surface realization.

In the subtask of SemEval AMR-to-English generation, the abstract AMRs to be converted into syntactic structures by a symbolic generator, then the syntactic structures are linearized with an off-the-shelf tools (e.g. statistical linearizers) [6].

With the advent of deep learning, the most influential architecture for NLG is the Encoder-Decoder [7]. The encoder encodes various kinds of input (e.g., natural language text, structured data, image, video and knowledge base) into a low dimension vector representing the semantic of the input. The decoder generates natural language text from the vector embedding. In practice, the knowledge representation languages like RDF and others are widely used as a kind of input of various modern neural generation models for knowledge-based system constructed by the W3C standards. Currently the

W3C standards are widespread used for development of knowledge-based system. The standard to specify knowledge is RDF, a formal language based on semantic networks. The WebNLG [8] is a project oriented on developing technologies that give humans easy access to machine-readable data of Web, usually this data is in form of RDF. The WebNLG challenge tends to develop neural generator by deep learning models. The training data of WebNLG challenge consists of Data/Text pairs where the data is a set of triples extracted from DBpedia [9] and the text is a verbalization of these triples. However the neural generators usually ignore the distinction between text planning and realization that causes difficulties in controlling over the generated text structure [10]. Another problem is dataset acquiring, because the high quality aligned training data is the core for neural generator. Currently the WebNLG supports the English and Russian datasets. There are also high quality Chinese knowledge bases, such as CN-DBpedia [11], zhishi.me [12] and others, but the aligned Data/Text pairs datasets are difficult to obtain.

Nowadays the mainstream methods consist in applying the neural network model to achieve NLG with the help of the high quality dataset, recently rare works on researching the ruled-based method for NLG. The system NaturalOWL [13] proposed to construct generation resource to verbalize OWL. The system in fact is a symbolic generator; manually constructed resources of this system determine the quality of the generated text. The single model cannot solve the efficiency and quality of generated texts simultaneously, especially the quality dataset is absent for certain languages. The integration of various methods and the execution efficiency of various methods is the main problems for NLG.

As can be noted, the above-mentioned approaches can solve partial problems for development of NLG part of natural language interface. The effective integration of different approaches is still unsolved. Through the discussion of related work, for developing compound NLG part of natural language interface, the integrated approaches will be needed to solve the various problems of NLG.

D. *The Proposed Approach*

We propose to use ontological approach to develop the unified representation for linguistic ontologies and to design problem solvers having ability to integrate various approaches for NLG solution within OSTIS Technology framework [14]. In this paper, natural language generation is developed as a component of natural language interface that is a part of user interface of the ostis-system. The ostis-system is a hybrid intelligent system being developed on OSTIS Technology. The OSTIS Technology is aimed at knowledge processing and various knowledge presentation for intelligent systems, it's focused on the development of knowledge-driven computer systems. Each ostis-system consists of sc-model of knowledge base, sc-model of problem solver and sc-model of user interface. The sc-model of knowledge base is based on several basic principles (e.g. the hierarchical system of subject domain and ontologies) that provides the ability to represent knowledge of various types in the knowledge base [15]. The sc-model of problem solver is based on the principle that a hierarchical system of agents react to situation and events in sc-memory and interact with other corresponding agents in the sc-memory [16]. The sc-model of user interface is based on some principles to resolve specific interface tasks [1].

As a basis for knowledge representation in the framework of OSTIS Technology, a unified version of coding any kind of information named SC-code is used. The SC-code is a semantic network language with set-theoretic interpretation. Several universal variants of visualization of SC-code [14], such as SCg-code (graphic variant), SCn-code (nonlinear hypertext variant), SCs-code (linear string variant) will be shown below.

We propose the approach for generating natural language text based on OSTIS Technology. The following main advantages of this technology are considered:

- Use of unified semantical models and tools for structuring and managing the knowledge base. When faced on the new language the unified tools to design new linguistic ontologies allow to decrease laboriousness and time;
- In the user interface design using OSTIS Technology, the syntax and semantics of external natural language are described using SC-code with the appropriate ontology. Hence, the translation mechanisms between the intelligent system and the external natural language are not depend on external language. In the processing of interaction between human beings and system, for new specialized language the specification of the syntax and semantics of the language is only need;
- For the development of different components of the problem solvers in the user interface of ostis-system, the focus is on the features of different components, making any changes to the ostis-system is unified;
- The technology provides designed component oriented on natural language generation modifiability, i.e. the ability to extend its functionality or to improve its performance.

E. *Tasks to be Solved for Proposed Approach Implementation*

The clarification of the generation of natural language text from knowledge base is, in fact, the clarification of constructing natural language generation component using the principle of ontology-based user interface design.

Taking into account the features of natural language generation component and the design principle of ontology-based user interface, the following tasks should be solved for proposed approach:

- To develop the sc-model of natural language interface knowledge base;
- To develop the sc-model of natural language interface problem solvers used for natural language generation. According to design principle of problem solvers within OSTIS Technology, problem solvers are presented as a hierarchical system of agents.

II. The General Structure for Designing the Natural Language Interface

One of the principles of ontological method to design user interface is to treat the user interface as a specialized ostis-system oriented on the interface tasks solution. From the perspective of OSTIS Technology, as a part of user interface, natural language interface is also designed based on general principle. The design of natural language interface involves the development of the ontological model of knowledge base of natural language interface and the ontological model of problem solvers oriented on natural language interface. As a part of natural language interface, NLG component generating natural language text from knowledge base of specific ostis-system is designed underlying the principles of designing natural language interface.

The knowledge base of natural language interface provides the necessary linguistic knowledge for generating natural language text. The knowledge base provides different levels of linguistic knowledge from basic word to syntactic and semantic structure of natural language text. Some specific words that can serve as predicates have predicate-argument structures according to specific language. In the OSTIS Technology, these predicate-argument structures can be encoded into representations of knowledge base through SC-code. When the sc-structure that needs to be converted into natural language text is determined, the syntactic structure of the resulted text can be determined through the argument structure of the predicates in the sc-structure. Further, the sc-link of each sc-element in the sc-structure can be appropriately filled to generate the text, which satisfy syntax of certain language. For certain specific sc-structure, the representation form doesn’t have much flexibility, the template-based method is the best choice. Templates for generating natural language text corresponding to the syntactic and semantic structure can be constructed as a logical statement in the logical ontology.

The development of problem solver of natural language interface is based on the classical pipeline architecture for realizing NLG. Based on OSTIS Technology, a group of sc-agents need to be developed to achieve the function of each part in the pipeline architecture with the help of linguistic knowledge. According to the principles of ontology-based problem solver design, the realization of the function of each part can adopt various suitable approaches. In addition, due to the complexity of sc-structure, there are a series of prepossessing of sc-structure to convert sc-structure into message triples, which are easier to express as natural language text. The multiple message triples can be transferred into a sentence. The message triples are not randomly combined to generate a sentence. Even different orderings of each element in a message triple will correspond different syntactic structures. In this situation, aggregation between these message triples need to be considered. When multiple triples are aggregated into a compound sentence, the use of referring expression can make the sentence more natural and fluent. For generating resulted text, the inflectional forms of the word stored in sc-link have to be provided to achieve the function of surface realization.

III. The Structure of the Knowledge Base of the Natural Language Interface

The knowledge base of the natural language interface has two parts (Fig. 2). Moreover, the knowledge base involves own problem solvers oriented on natural language generation from the specific subject domain based on linguistic ontologies. The language part is the ontological model of knowledge base of specific linguistics, e.g. sc-model of knowledge base for Chinese language. The subject part is sc-model of knowledge base for specific domain (e.g. History, Movies).

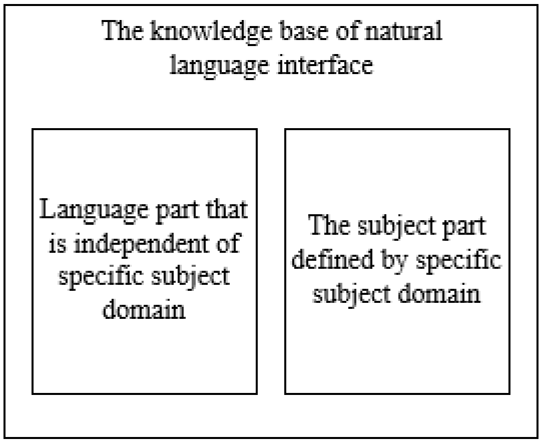
The knowledge base of natural language interface
Language part that is independent of specific subject domain
The subject part defined by specific subject domain

Figure 2: The knowledge base of the natural language interface

Within OSTIS Technology, the knowledge base structure of any ostis-system is described by a hierarchy of subject domains (SD) and the corresponding ontologies [15]. Subject domain is one of the key concepts in determining the structure of knowledge base. Each subject domain is the description of connections about the relevant class of research objects. The structure of knowledge base is an interconnection of various subject domains that allows to consider objects of research on different levels of detail. Each ontology is a specification of subject domain, i.e. specification of a system of concepts used in this subject domain. Each ontology provides a conceptualization of a knowledge domain by defining the classes and subclasses of the domain's individuals (entities), the types of possible relations between them etc. According to the properties of concept researched in the subject domain, every subject domain is represented by the following distinguished ontologies: structural specification, set-theoretical ontology, logical ontology and so on [14].

Any ostis-system created on OSTIS Technology is considered as a child system of **IMS Metasystem** [17]. IMS Metasystem is an ostis-system about ostis-system design automation. The knowledge base of IMS Metasystem includes a system of top-level formal ontologies used for formal description of a sense knowledge representation SC-code, such as subject domain of entities, subject domain of connections and relations and so on. They ensure quality of internal language for knowledge representation of intelligent system. Hence, from perspective of semantic network that is formal basis for SC-code, elements in SC-code is named sc-elements, such as nodes – sc-nodes, connections – sc-connectors (sc-arcs, sc-edges).

The *Subject Domain of linguistics* represents the language part. To provide formal representation for the various kinds of linguistic knowledge, a number of subject domains and their ontologies need be developed. The general structure of *Subject Domain of Linguistics* is presented in SCn-language.

Subject domain of Linguistics
⇒ *private subject domain**:
- ***SD of Chinese language texts***
- ***SD of English language texts***
- ***SD of Russian language texts***

In order to design universal translation mechanisms from the internal to the external natural language and back, specification of the syntax and semantics of the language is the core need. The syntax and semantics of specific language texts become the main object of research in *Subject Domain of linguistic*. Look at a general structure of *SD of Chinese language texts* represented in SCn.

SD of Chinese language texts
⇒ *particular SD**:
- ***SD of Chinese language syntax***
- ***SD of Chinese language semantics***

The *SD of Chinese language syntax* and *SD of Chinese language semantics* describe specification of a system of concepts from the syntactic aspect and the semantic aspect of the Chinese language, respectively [18].

In the Fig. 3, a logical statement in the logical ontology represented in SCg indicates a simple heuristic rule used for generating natural language text. The heuristic rule can be used to generate a simple declarative sentence based on template. When the role of each element of a triple in the sentence is determined, e.g., identifier of a element is "binomial theorem" served as the subject of sentence, identifier of another element is "observation" served as the object. The relation between them is membership. Based on the template the resulted sentence "The binomial theorem is a kind of observation" is generated.

Subject Domain of specific domain represents the subject part, which is dependent from the ostis-system. For example, in the knowledge base of OSTIS intelligent tutoring system for Discrete Math, the subject part contains various information about the discrete math domain, such as the type of theorem, inclusion relation of graph theory and so on.

In the knowledge base of ostis-system, name of each sc-element is an arbitrary unique string stored in the sc-link. There are three main commonly used options for naming sc-elements: system identifier, main identifier, and identifier. Among them, it is recommended to use the system identifier as the name for sc-elements. The system identifier is unique for a sc-element within the entire knowledge base of a given ostis-system. We follow the convention that how to construct a system identifier of sc-elements with specific label. However, there are obvious distinguish between the system identifier and the name of the sc-elements used in natural language texts. For words corresponding to system identifiers that the systems wish to use in the resulted texts, the lexical unit (lexicon entry) has to be provided, which specifies properties of that word (e.g., inflectional forms of word in English), as well as other information.

Lexical unit is a fragment of linguistic ontologies, it represents the words that need to be used in this subject domain. Strictly speaking, it belongs to **SD of natural language word**. The lexical unit consists of closed-class words and opened-class words. The lexical units of closed-class words, like determiners and prepositions, are domain-independent. The lexical units of opened-class words usually are basic for constructing system identifier of sc-elements in specific subject domain.

Fig. 4 shows the linguistic knowledge for lexical unit (in Chinese). The lexical units for adjectives and others are similar.

The sc-structure specifies that a lexical unit whose system identifier is "I_ch_to_be". The role relations "arg0" and "arg1" indicate predicate-argument relation in the linguistic theory FrameNet. Notice that in addition to describing the inflectional forms of the lexical unit (when it's for English or Russian language), there are also other

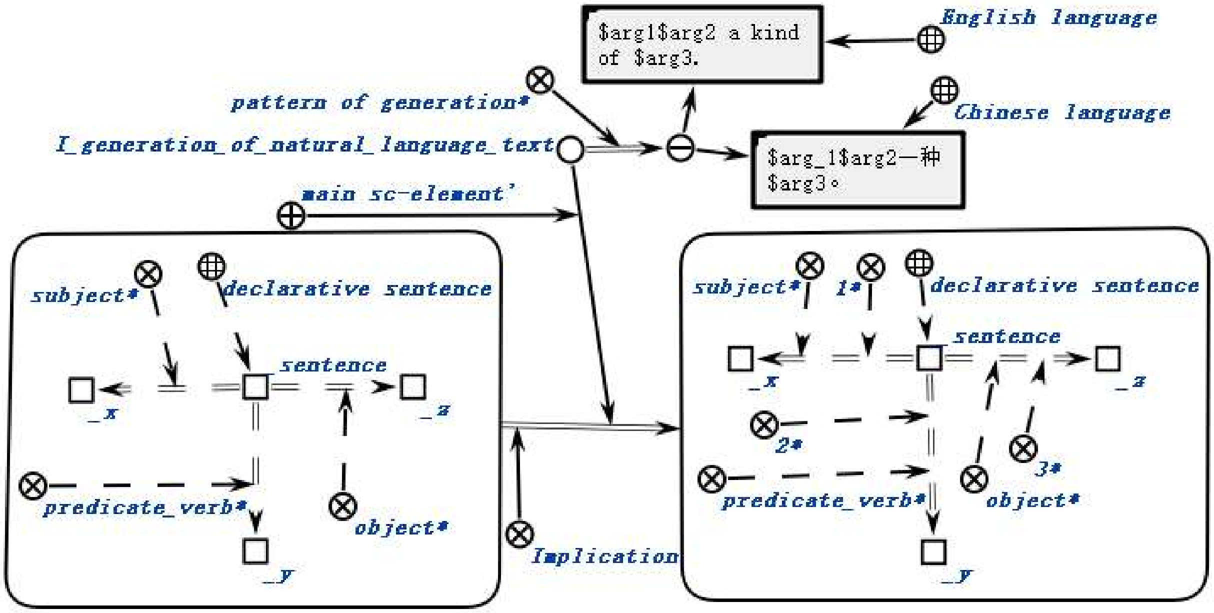

Figure 3: Logical statement about generating natural language text

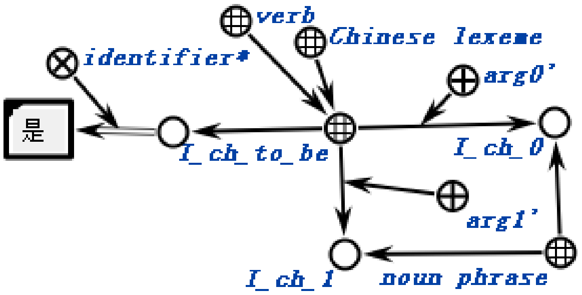

Figure 4: linguistic information for lexical unit "to be"

information about syntactic level and semantic level of the lexical unit to be specified, for example, the part-of-speech of the lexical unit. These information allows to generate most appropriate words in the resulted generated texts.

The information about semantic level of lexical units is essential for constructing syntactic structure. We provided the information about semantic level in *the subject domain of natural language semantic*. Among them, the most important core part is the semantic role frame of predicate. We note that there are many useful linguistic resources for constructing ontologies of natural language. For construction of ontologies of lexical units, some general-purpose lexical unit would be to exploit, such as Chinese WordNet [19], Chinese part of ConceptNet [20] and so on. In *the Subject domain of Chinese language semantic*, with the help of the CPB, Mandarin VerbNet [21], and other linguistic resource, we will consider the concepts, such as semantic frame of predicate, semantic role and so on. The Subject domain of Chinese language semantic provides the semantic information for certain specific lexical units. The general-purpose lexical units, however, often do not cover the highly technical concepts of domain knowledge base. Sometimes it is necessary to tailor or design lexical units from general-purpose lexical units that are suitable for development of a specific domain knowledge base.

IV. The Problem Solver Structure of the Natural Language Interface

Within the OSTIS Technology, the development of problem solvers actually comes down to development of its knowledge base. The constructed knowledge base includes the own problem solvers having program agents. The multi-agent approach is used as a basis for problem solvers design. The interaction of agents will be performed exclusively in the semantic memory (sc-memory), which stores the SC-code constructions. Such approach provides the flexibility and modularity of developed problem solvers, as well as provides the ability to integrate various methods to problem solutions. In the term of implementation, agent programs can implement logical reasoning based on a hierarchy of statements comprised in the logical ontology, as well as the data-driven learning algorithms using various programming language.

Abstract sc-agent generating external texts from the knowledge base is a group of sc-agents that implement the mechanisms of natural language generation from knowledge base, i.e. given a sc-structure containing a set of sc-sentences, sc-agents generate a corresponded fluent natural language text. One of the possible approaches for implementation of this abstract sc-agent is to build a collection of simpler abstract sc-agents by the relation abstract sc-agent decomposition*. The following is general structure for problem solver of natural language interface represented in SCn-language.

Abstract sc-agent of natural language interface
⇐ *decomposition of an abstract sc-agent**:
{
- *Abstract sc-agent translating external texts into the knowledge base*
- *Abstract sc-agent generating external texts from the knowledge base*
  ⇐ *decomposition of an abstract sc-agent*:*
  {
  - *Abstract sc-agent for content selection*
  - *Abstract sc-agent text planning*
  - *Abstract sc-agent for micro-planning*
  - *Abstract sc-agent for surface realization*
  }

}

In the technology framework of OSTIS, a sc-agent is some entity that can perform actions in the sc-memory. The notions of sc-agent and related concepts are specified in the subject domain of abstract sc-agent and the corresponding ontologies. The abstract sc-agents is a certain class of functionally equivalent sc-agents, various items of which can be implemented in different ways to specific tasks.

Implementing content selection from knowledge base includes a sc-agents of the following types:

Abstract sc-agent for content selection
⇐ *decomposition of an abstract sc-agent**:
{
- *Abstract sc-agent determining sc-structure*
- *Abstract sc-agent dividing determined sc-structure into basic sc-structure*
- *Abstract sc-agent transferring basic sc-structure into message triple*
- *Abstract sc-agent determining the candidate sc-structures*
- *Abstract sc-agent transferring candidate sc-structures into message triples*

}

Abstract sc-agent for determining sc-structure - the groups of agents that provide the retrieval from the domain knowledge base sc-structures containing sc-sentences from which we will transform to natural language texts. This stage is to determine what we want to talk.

Abstract sc-agent dividing determined sc-structures into basic sc-structures – the agents that implement the mechanisms of decomposition of retrieved sc-structures into basic structures, which can be transferred into message triples. Sometime several message triples transformed from sc-structure are redundancy, so the system finally selects among the basic sc-structures the ones to be transferred.

Abstract sc-agent determining the candidate sc-structures – the agents that implement the mechanism of determination of appropriate candidate sc-structures that satisfy specific end-users. ***Abstract sc-agent transferring candidate sc-structures into message triples*** – the agents that implement the mechanism of conversion of candidate sc-structures to message triples.

Abstract sc-agent for text planning - the agents that implement the function that message triples are ordered, in effect the ordering of sentences in the resulting text. The following structure of this abstract sc-agent is considered:

Abstract sc-agent for text planning
⇐ *decomposition of an abstract sc-agent**:
{
- *Abstract sc-agent ordering message triples*
- *Abstract sc-agent ordering entities of a message triple*

}

Abstract sc-agent for miro-planning - the agents that implement the mechanism of transferring message triples to abstract sentence specifications that are varied accorss NLG system, for example, the simple text templates with slots, syntactic structures and so on. The structure of abstract sc-agent for micro-planning:

Abstract sc-agent for micro-planning
⇐ *decomposition of an abstract sc-agent**:
{
- *Abstract sc-agent constructing sentence plan*
- *Abstract sc-agent for sentence aggregation*
- *Abstract sc-agent generating the referring expression*

}

Abstract sc-agent for surface realization - the group sc-agents that implement the mechanisms of concatenating the sc-links to generate natural language text, i.e. retrieving resulting forms of lexical units according to rules, as well as filling and concatenating the sc-links.

The above listed problem solvers are not fixed. It is possible to adjust and make extensions for the already developed abstract sc-agents, i.e. the structure of problem solver is flexible and changeable. This kinds of features for designing problem solver are due to the advantages of multi-agent approaches by OSTIS Technology: adding a new agent or removing (deactivation) of one or more existing agents usually does not lead to changes in other agents; agents often work in parallel and independently from each other and so on.

V. Implementation of Chinese Language Generation

With the help of previously constructed linguistic ontologies and problem solvers in the ostis-system, the following example shows the processing stage for implementing Chinese language generation.

Several constrains are defined:

- The input of system is completed and has sense;
- The input given sc-structure formally represented in the knowledge base of specific SD (Discrete Math);
- The output of system is a simple narrative Chinese sentence;
- The knowledge base includes entity signs with Chinese identifier, which will be used in the resulted sentence

For generating natural language text from ostis-system specific subject, we propose that the realization of natural language generation is roughly divided into two steps: rule-based symbolic generator converting structure of knowledge base to message triples; rule-based approached or statistical generator (when high quality aligned datasets is accessible.) translating message triples to natural language text. Unfortunately, the high quality aligned datasets is relatively difficult to access. The example in this article is just to generate a simple narrative Chinese sentence. The rule-based approach is used to illustrate the process.

Step 1 We are given a sc-structure formally represented in SCg (Fig. 5).

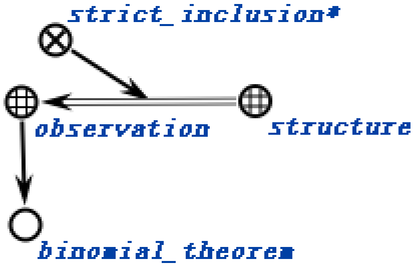

Figure 5: Sc-structure of knowledge base of Discrete Math

Step 2 The given sc-structure is transferred to a set of message triples, the identifier of each sc-element of message triples corresponds to a certain lexical unit of specific language. Description of lexical units that is stored in the linguistic knowledge base had been mentioned above. In the example, we just consider one of the set of message triples. Moreover, the predicate "to be" will be used for the independent property "instanceOf" (Fig. 6).

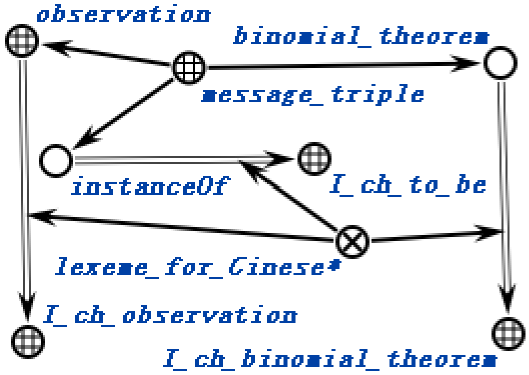

Figure 6: Message triple for part of sc-structure

Step 3 For each property of message triples, a suitable predicate-argument structure from the linguistic ontologies is matched. For this example, the specific template shown in the Fig. 3 will be used.

Step 4 The agent fills the sc-links of corresponded sc-elements of message triples, i.e. a certain sc-link needs to be filled with the result of a certain inflection form of a lexical unit. For Chinese language, there is not a certain inflection form for lexeme, e.g., the result of a lexical unit "binomial theorem" in Chinese (Fig. 7). Therefore, the processing of this step is relative simpler.

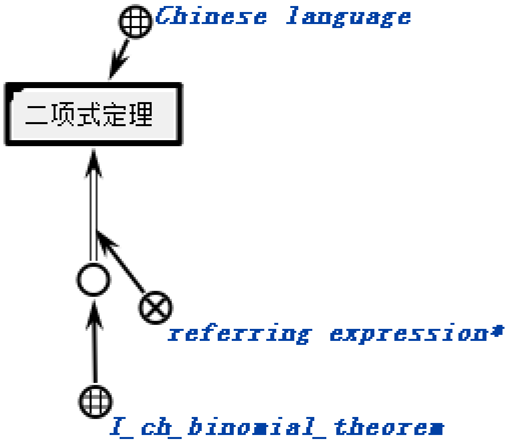

Figure 7: The sc-link filled by the Chinese lexical unit "binomial theorem"

Step 5 Previous sc-links are concatenated to generate the resulted Chinese sentence according to valid ordering (Fig. 8).

In the knowledge base there are different identifiers for each lexical unit. In this example just the simple declaration sentence is considered, the referring expression of the lexical unit "binomial theorem" is itself in Chinese language. Based on the template, expression referring

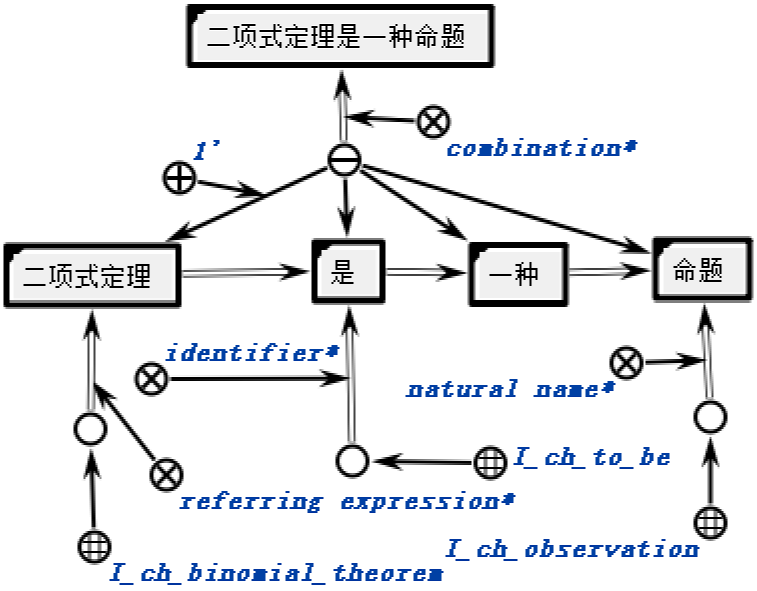

Figure 8: Generation of Chinese sentence corresponding to sc-structure

to the lexical unit "binomial theorem" is served as the subject. The lexical unit "observation" is served as object in Chinese language. The template is predefined with fixed phrase, in the (Fig. 8) the sc-link without corresponding lexical unit is the fixed part.

When these steps are implemented, a fragment sc-structure of knowledge base of Discrete Math domain can be transferred into Chinese sentence with specific sense. This natural language text is easier to access to ordinary end-users.

Nowadays there are two types of methods for evaluating generated natural language text: automatic metrics and manual evaluation. The BLEU [22] and ROUGE [23] and METEOR [24] score are widely used as the automatic metrics. We begin by comparing the natural language text, which is generated by the system, with the natural language text described by human using the common automatic metrics BLEU. Notice that we couldn't compare the performance of system with other state-of-the-art system due to the absence of the system for processing complex semantic structure in knowledge base.

Generally speaking, when describing the performance of a text generation system, the cumulative score from BLEU-1 to BLEU-4 is usually reported. Table. I shows that performance of different approaches for generating text. The templates are usually predefined by human, therefore, the text generated according to templates can get high scores. The BLEU-4 score "0.00" indicates the predefined templates is not flexible. However, for multiple sc-sentences, these approaches couldn't obtain satisfactory scores. The system still has certain deficiencies when generating longer natural language text.

Table I: Evaluation of Approaches for Text Generation

	Template-based for one basic sc-sentence	Rule-based for one basic sc-sentence	Template-based/rule-based multiple basic sc-sentences
BLEU-1	1.00	0.78	0.64
BLEU-2	1.00	0.62	0.53
BLEU-3	1.00	0.53	0.41
BLEU-4	0.00	0.32	0.23

VI. Conclusion

This article has proposed an ontological approach to design a unified semantic model for natural language generation from knowledge base. The use of this approach offers the following advantages:

- The model for natural language generation is applied to generate fluent, coherent, and multi-sentence natural language texts appropriating for end-users;
- The semantic structure that processed by this model is more complex than simple tabular or triples structure;
- as a part of the model, the Chinese linguistic knowledge base is designed to generate grammatically similarity texts. The knowledge base make the model more explanatory than statistical model.

We discussed the structure of knowledge base of natural language generation component of natural language interface, the processing stage for Chinese language generation, the optional linguistic knowledge base used in each stage. Relying on the component to automatically produce texts from knowledge base makes the information of intelligent system easily accessible not only to computer programs, but also to end-users.

We are currently working towards implementing the model for Chinese language generation in certain trial, in order to verify the practicality of the model and to evaluate the quality of generated text. It would also be particularly interesting to explore the model's possibility to support multiple language generation as long as the corresponding linguistic ontology is added.

References

[1] Sadouski M.E., Boriskin A.S., Koronchik D.N., Zhukau I.I., Khusainov A.F.: Ontology-Based Design of Intelligent Systems User Interface. In: 7th International Scientific and Technical Conference "Open Semantic Technologies for Intelligent Systems", pp. 95-106. Belarus Minsk (2017)

[2] Introducing the Knowledge Graph: Things, Not Strings. Available at: https://blog.google/products/search/introducing-knowledge-graph-things-not/

[3] Plachouras V., Smiley C., Bretz H., Taylor O., Leidner J. L., Song D., Schilder F.: Interacting with financial data using natural language. In Proc. SIGIR'16, pp. 1121-1124. Italy Pisa (2016)

[4] Reiter E.: Pipelines and Size Constraints. Computational Linguistics, 2000, vol.26 No 02, 251-259.

[5] Meteer M. W.: Bridging the generation gap between text planning and linguistic realization. Computational Intelligence, 1991, vol.07 No 04, 296-304.

[6] Simon M., Roberto C., Alicia B., Leo W.: FORGe at SemEval-2017 Task 9: Deep sentence generation based on a sequence of graph transducers. In: Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017), pp. 920-923. Canada Vancouver (2017)

[7] Sutskever I., Vinyals O., Le Q. V.: Sequence to sequence learning with neural networks. In Proc. NIPS'14, pp. 3104-3112. Canada Montreal (2014)

[8] Natural Language Generation for the Semantic Web. Available at: https://webnlg.loria.fr/pages/index.html/

[9] DBpedia. Available at: https://www.dbpedia.org/

[10] Amit M., Yoav G., Ido D.: Step-by-Step: Separating Planning from Realization in Neural Data-to-Text Generation. In: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 2267-2277. USA Minneapolis (2019)

[11] Xu B. Xiao Y.H. CN-DBpedia: A Never-Ending Chinese Knowledge Extraction System. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, pp. 428-438. Springer, Cham, 2017.

[12] Niu X., Sun X., Wang H., Rong S., Qi G., Yu Y.: Zhishi.me - Weaving Chinese Linking Open Data. In: The Semantic Web – ISWC 2011. ISWC 2011. Lecture Notes in Computer Science, vol 7032. Springer, Berlin, Heidelberg, 2011.

[13] Androutsopoulos I., Lampouras G., Galanis D.: Generating Natural Language Descriptions from OWL Ontologies: the NaturalOWL System. Journal of Artificial Intelligence Research, 2013, vol.48 No 01, pp. 671-715.

[14] Golenkov V.V. Gulyakina N.A. Proekt otkrytoi semanticheskoi tekhnologii komponentnogo proektirovaniya intellektual'nykh sistem. Chast' 1 Printsipy sozdaniya [Project of open semantic technology of component designing of intelligent systems. Part 1 Principles of creation]. Ontologiya proektirovaniya [Ontology of designing], 2014, no 1, pp. 42-64. (in Russian)

[15] Davydenko I.T.: Ontology-based knowledge base design. In: 7th International Scientific and Technical Conference "Open Semantic Technologies for Intelligent Systems", pp. 57–72. Belarus Minsk (2017)

[16] Shunkevich D.V.: Ontology-based design of knowledge processing machines. In: 7th International Scientific and Technical Conference "Open Semantic Technologies for Intelligent Systems", pp. 73–94. Belarus Minsk (2017)

[17] (2021, May.) The IMS.OSTIS website. [Online]. Available: http://www.ims.ostis.net/

[18] Qian L., Sadouski M., Li W. Ontological Approach for Chinese Language Interface Design. In: Golenkov V., Krasnoproshin V., Golovko V., Azarov E. (eds) Open Semantic Technologies for Intelligent System. OSTIS 2020. Communications in Computer and Information Science, vol 1282. Springer, Cham, 2020.

[19] (2021, May.) The Chinese WordNet. [Online]. Available: https://lope.linguistics.ntu.edu.tw//cwn/

[20] (2021, May.) Chinese part of ConceptNet. [Online]. Available: http://conceptnet.io/c/en/chinese

[21] (2021, May.) Mandarin VerbNet. [Online]. Available: http://verbnet.lt.cityu.edu.hk/#/

[22] K. Papineni, S. Roukos, T. Ward, W.J. Zhu. BLEU: a method for automatic evaluation of machine translation, Proceedings of the 40th annual meeting on association for computational linguistics (2002), pp. 311-318.

[23] C.Y. Lin. Rouge: A package for automatic evaluation of summaries, Text Summarization Branches Out (2004).

[24] S. Banerjee, A. Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments, Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization (2005), pp. 65-72.

Онтологический подход к генерации естественного языка из базы знаний

Цянь Лунвэй, Ли Вэньцзу

В статье рассматриваются существующие методы разработки естественно-языкового интерфейса, а также методы к реализации генераций естественного языка из базы знаний как компантент естественно-языкового интерфейса. Был проведен анализ проблем, возникающих при генерации естественного языка из структурированных данных (в частности база знаний) в настоящее время.

На основании различных рассмотренных методов был предложен онтологический подход к генерации естественного языка, который позволяет интегрировать разные типы методов генерировать тексты естественного языка. Предложенный метод направлен на разработку семантической модели знаний о линквистике. Этапы реализации подхода были созданы лингвистические онтологии и решатели для генерации естественного языка. Лнигвистические онтологии выражает синтаксические и семантические знания конкретного языка, которые можно использовать решателями для генерации естественного языка.

Таким образом, для дальнейшей обработки и реализации части естественно-языкового пользовательского интерфейса в работе предлагается модель генерации естественного языка из базы знаний конкретного домена, основанная на знаниях. Более того, в качестве китайского языка, функции каждых этапов генерации и назначение лингвистических онтологий в процессе генерации проиллюстрированы, чтобы проверять практичность модели.

Received 14.05.2021