Approach to Interchange-Format Based Chinese Generation

Wenjie CAO, Chengqing ZONG, Bo XU

National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences, Beijing, 100080, China
{caowj, cqzong, xubo}@nlpr.ia.ac.cn

Abstract

Interlingua-based machine translation is an important approach to implement multi-lingual speech-to-speech (S2S) translation. The natural language generation (NLG) is one of the key components in the interlingua-based machine translation systems. This paper introduces our approach to Chinese generation based on the Interchange Format (IF) developed by the C-STAR organization. In our approach, the hybrid method of feature-based deep generation method and template-based method are employed. The deep generator ensures that the generation component possesses the merits of flexibility and domain portability. The template-based generator makes the system more efficient. We also introduce another simplified Chinese generator applied in specific domain. The experimented results show that our approach is effective and practical for the natural language generation in the Interchange-Format (IF) based S2S translation system.

1. Introduction

Natural language generation (NLG) is to find out how to transform non natural language semantic expressions into natural language expressions, so as to generate understandable, exact, and fluent natural language sentences. As the rapid development of natural language processing techniques, NLG has been widely used in many aspects. The target language generation in machine translation system is one of the most typical applications.

This paper introduces an Interchange-Format (IF) based Chinese generation in multi-lingua S2S translation system under the C-STAR (C-STAR, Consortium for Speech Translation Advanced Research) framework.

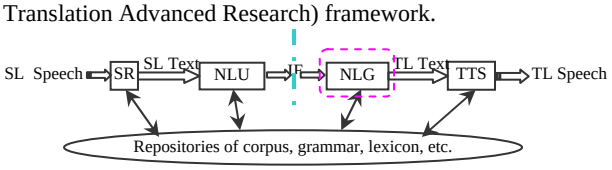


Figure 1: Frame of IF-based S2S translation system.

Figure 1 shows the frame of IF-based S2S translation system. When a piece of source language speech is input into the translation system, it is automatically recognized into the source language text. Then the NLU (Natural language understanding) module parses the meanings of the recognized text, and transforms it into IF expressions. When the target end gets IF, it firstly generates the sentences in target language from the IF expressions, and then uses the TTS module, to generate the corresponding target speech. This paper describes researches on NLG in the target end of the IF-based translation system.

Since the first NLG system was developed in 1960's, there have mainly emerged four kinds of NLG methods: canned-text method, template-based method, phrase-based method and feature-based method. **Canned-text method** is a most simple and most inflexible generation method. It is mainly used in the prompt information systems of software products. **Template-based method** is more flexible than canned-text method in generation. Also it could acquire higher efficiency and nicer results in specific domain than the following methods. But it is not flexible enough and hardly to obtain good performance when applied to other domains. One typical example for template-based method is ANA system advanced by Kukich in 1983. **Phrase-based generation method** is flexible, powerful and robust. But it may encounter inappropriate phrase expansions when the generation structure is beyond a certain size. One typical example is the MUMBLE system (McDonald, 1980; Meteer, McDonald, et al., 1987). **Feature-based method** is very flexible. Any distinction in language can be added to the system as a feature. Yet it is difficult to maintain feature interrelationships and control feature selection. One example was PENMAN system (Mattiessen, 1983), and KPML system (Bateman，Maier et. al., 1991).

Compared with text generation, generation in the S2S translation system needs to be flexible, efficient, and robust. The definition of IF has also brought out new problems to the IF-based translation system, including the generation module, and the understanding module.

This paper presents a hybrid approach that combines the template-based and feature-based generation strategies. The reasons that we use such approach can be summed up as following. First, there are fixed expression modes in domain specific spoken language dialogs. According to our preliminary statistics of dialogs: imperative simple sentences using "请" (please) take up 17 percent; those using "有…吗", "有没有"(is/are there any…), and "能不能/可以不可以" (can/may sb. do …) to express query mood take up 44 percent or so; those sentences that comprise time or numeral words take up 22 percent. In such sentences, many fixed expressions are very suitable for template-based translation (for directly translation systems) or target language generation (for Interlingua-based MT systems). The introduction of template method is very helpful for the simplicity and efficiency of the translation procedure. Second, the expressive manner of spoken language is versatile. For those non-fixed expressions, the flexible feature-based deep generation approach is definitely more appropriate. Third, for multi-lingual generators, compared with other approaches, the feature-based method treats the differences of language as features, so that they can be easily dealt with in a uniform procedure framework.

In this paper, section 2 introduces the definition and characteristics of IF; section 3 describes our generation approach; experimental results and analysis are presented in section 4. Finally, we derive our conclusion.

2. Interchange format (IF)

IF is adopted by C-STAR as Interlingua. Currently, the applied domains are travel planning and medical emergencies. An IF expression normally includes four parts: speaker, speech act, concepts and arguments. About the specific meaning of each part, please refer to reference [2].

IF is based on speech act theory (J. Austin, 1962; J. Searle, 1969). Speech act theory considers language is not only used to descript fact, but also assume the intent /purpose of the speakers. The foundation of IF determines IF possesses following characteristics: (1) for multilingual dialog translation system, if the IF can exactly capture speaker's communication intention, a lot of information needed for sentence generation will be deduced, so that the definition of IF will be greatly simplified. (2) Since the definition of IF does not involve syntactic and semantic relations of the main participants of a sentence, IF is more suitable for multilingual translation system than Interlinguas based on Fillmore's Case Grammar, and is helpful for resolving "head switching" mismatches problems[3]. (3) IF is a kind of underspecified semantic representations (USR). So, the generator needs to infer missing information from IF expressions and domain knowledge before generating the surface form of a sentence. (4) There is no predicate-argument frame available in an IF expression. So generator should firstly deduce the predicate-argument frame from the main part of an IF expression, and then map the IF into the semantic representations that are suitable for sentence generation.

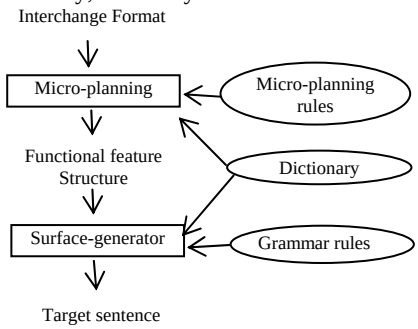
Example: c: request-information +departure +transportation (transportation-spec= (flight,identifiability=yes, destination= tokyo), time=(clock=(hours=2)))

Sentence: Does the flight to Tokyo leave at 2 o'clock? || 飞往东京的航班2点钟离开吗？

In the example above, the intent of the speaker is to inquire information. The topic is about the departure info of vehicles. In the argument list, the IF gives specific transportation information and time information. Relating it with generation, we can see that the mood info (inquiry) could be inferred from the speech act (request-information); predicate frame or sentence type is determined by the domain action (c:request-information +departure +transportation); shallow phrases info can be obtained from the "argument-value" pairs.

3. Hybrid generation approach

As an indispensable part of multilingual S2S translation system, target language generator is required to have the virtues of flexibility, efficiency and robustness.



- *Figure 2*: target language generator

Currently, our generator is mainly applied in hotel reservation domain. It is illustrated in figure 2. The generator comprises micro-planning and surface sentence generating modules. By using micro-planning rules and dictionary, micro-planning module maps an IF expression to a functional feature structure of sentences, which is depicted by complex feature sets, and involves all the information that needs to be expressed to the listener in target end. Surface generator is to generate surface sentence from the feature structure by unifying the features with corresponding grammars. Systemic Functional Grammar is used as our generating grammar, and functional unification algorithm is adopted to generate the surface form of target sentences.

Hybrid method combining template-based generation and feature-based deep generation is the main approach to the generation. Template-based generation is efficient. But it is inflexible and comparatively hard to be applied to other domains. Feature-based deep generation is very flexible, and has better capability in universal usage. But it is inefficient. Combining the two methods, we can gain better performance by compromising merits of efficiency and flexibility.

Practically, there exist errors in the speech recognition and the NLU module, also there may be errors and missing of information in IF expressions. To fulfill the robustness requirement, we loose constrains on participants of micro-planning and grammar rules, and allow the generator to generate non-integrate sentences. Besides, we set up default values under the direction of domain knowledge.

3.1. Micro-planning

The micro-planning procedure has following functions: Determining sentence type according to IF expression and obtaining the predicate-argument frame of sentence; Transforming domain concepts into lexicon and getting corresponding lexical information from the dictionary; Changing domain relations into grammar relations; Obtaining information of mood, tense, modal, etc..

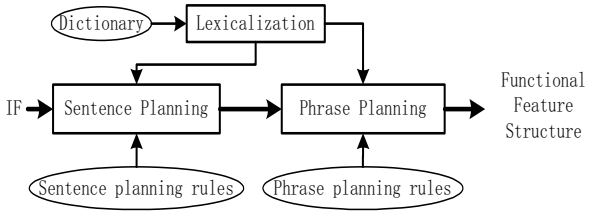


Figure 3: micro-planning

As figure 3 shows, micro-planning module is divided into two layers: sentence planning and phrase planning. The function of sentence planning is to deduce the top information of a sentence, such as main verb, tense, mood, etc., and obtain the predicate-argument frame according to the main verb. The role of phrase planning is to map the arguments and some concepts of IF expression to the shallow structures of a sentence. Through micro-planning, IF expression is transformed into semantic and syntactic feature sets of a sentence, and sent directly to the surface generating module to generate the surface form of the sentence. Repositories involved in the micro-planning module are sentence and phrase planning rules, domain knowledge and dictionary. Among the repositories, domain knowledge does not exist as an independent entity. Instead, it is reflected in the designing of planning rules. That is, make different maps from IF to

syntactic and semantic features in the light of different domain knowledge.

3.1.1. Sentence planning

Sentence planning rule is describes by a triple (P, C, A). "P" represents "Pattern", which refers to the pattern of IF expression (including speaker and domain action). "C" represents "Constraints", which could be null set, or restriction set that constrains the status of IF's concepts and arguments. "A" represents "Action". When the input IF expression satisfies the restrictions of "P" and "C", the module began to perform what "Action" specified, and obtain the semantics of main verb as well as the semantic and syntactic features of sentence. If there are several candidates in dictionary for the main verb, the lexicalization process then start to work, and select the most felicitous word.

Template approach is also implemented in this layer. Sentence templates are designed for sentences whose expressions are comparatively fixed. Variables are introduced into the templates that are corresponding with specific domains, and semantic classes are defined to constrain the properties of each variable in templates. The definition of semantic classes is consistent with value classes defined by IF. By introducing variables into templates, some sentences that originally need deep generation can now be generated through template approach. The generator maintains flexibility on one hand, and improves efficiency on the other hand.

Template rules adopt the same rule format as sentence planning rules. This consideration makes the generator could process IF expressions under the same framework. During the course of generating, if the input IF is consistent with a template rule, generator will directly enter surface generation, replace the variables in the template, and get the surface form of the sentence.

3.1.2. Phrase planning

In our current phrase planning process, "argument-value" pairs in IF are corresponding to the shallow phrase structures. IF has pointed out the head word in "argument-value" pairs. Modifiers and corresponding grammar info are given and encapsulated together with the head word. This form is already very suitable for phrase generation. Any further mapping could bring nothing but complexity and lower efficiency. So before generating phrases, phrase planning module doesn't transform the pairs into other deep semantic representations. Instead, it directly uses the pairs to represent the semantic info of phrases. In case domain correlative semantic info would destroy the universal merits of surface generator, phrase realization doesn't suspend into surface generating phase. So, maybe it is more suitable to call this phase "phrase generation".

Phrase planning rules are also depicted by a triple (Name, Constraints, Rule). "Name" is the name of the argument to be planned. "Constraints" is a set which constrains the status of value and sub-arguments. "Rule" is the corresponding Chinese rule when the input argument meets the constraints. The "Rule" part gives out the category and the form (using Concept Dependency Theory for reference, a list that enumerates sub-arguments and head word) of the phrase. The planning process is top-down and recursive. When a satisfied rule has been found, the module plans every non-terminal sub-argument until all the sub-arguments have been planned and lexicalized. No non-terminal items exist in the planning result.

3.1.3. Lexicalization

The import of lexicalization process has three main objectives. First, find out the most suitable main verb and some other substantives of the sentence to be generated. Second, minimize ambiguity problems. Third, find out the most suitable empty words (article words, auxiliaries, pronouns, prepositions, etc.).

Approach to lexicalization is the combination of *Structure Mapping Systems* and *Discrimination Nets* [8] [9]. For those that have different meanings and need to by the assistants of neighboring substantives, we use Discrimination Nets method. For the semantic items that don't have different meanings, we use the method of *Structure Mapping Systems*.

3.2. Surface generation

The task of surface generation is to use the output of micro-planning and grammar rules of the target language to generate correct and fluent sentence.

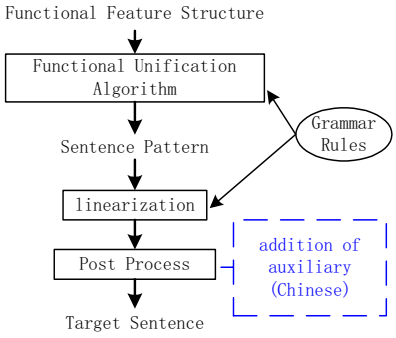


Figure 3: micro-planning

The surface generating module adopts Systemic Functional Grammar (SFG) as our generation grammar, and functional unification algorithm as our generation algorithm.

SFG was advanced by Halliday in 1985. It deems language has a hierarchic structure. Each hierarchy has its own systems. The systems are composed by functional feature items through disjunct or conjunct relations. When steps of related systems in the network are accomplished one by one, language structure will be produced. So the process of SFG is from function to structure, which is accordant with that of NLG. Therefore it is very suitable for NLG applications. Halliday considered every language has three "meta functions" (Ideational Semantics, Interpersonal Semantics, and Textual Semantics). Our functional networks were built based on these. For further information, please refer to [7].

In dealing with complex feature set, common approach is functional unification algorithm. Please refer to [7] and [6] for more information.

The function of linearization is to get the linear order of sentence's participants in the surface form of the sentence according to grammars of target language. Then, in post-processing procedure, we adjust the result according to the characteristics of target languages.

3.3. Experimental result

100 sentences were selected and tagged from BTEC corpus organized by C-STAR.

Table 1: Experimental result

Evaluate	Correct	Understandable	Wrong or no result
Number	73	14	13
Ratio	73%	14%	13%

Formula for computing accuracy is:

$$(Correct + Understand able)/Total \times 100\% \quad (1)$$

From table 1, we can see that our generator has acquired satisfying performance in hotel service domain. Yet there is still a long way to go in the whole travel domain. Went behind the reason, we found that the coverage of micro-planning rules and lexicon are not enough. This will be the emphasis of next phase's job. Besides, how to evaluate the generation quality is also a matter worth to draw our attention.

3.4. A simplified Chinese generator in specific domain

Above has introduced our hybrid approach in Chinese generation. To do more helpful attempts, and to better utilize the characteristics of Chinese, another simplified Chinese generator aiming at hotel reservation domain has been built.

Compared with most western languages, Chinese has its own peculiarities. Chinese doesn't have morphologic changes. Normally, the order of a Chinese sentence's components does not change due to different tenses or voices. Chinese has many functional words, and their positions are generally fixed according to their semantics. With these characteristics, the complex feature sets needn't be retained till surface generating to determine which word should take what form and be placed in which position according to different tenses and moods, etc..

In architecture, the generator is no longer divided into micro-planning and surface generating modules. Instead, we use an integrative structure. The generator is divided into two layers: sentence layer and phrase layer. Sentence layer and phrase layer together with the lexical selection module are the main modules of the generator.

In generating strategies, the hybrid approach is no longer used. Instead, template-based approach is utilized in both layers. Yet templates here are different from the original ones. Here the templates contain not only variables that need referring to the lexicon, but also functions that need further processing with phrase rules. Lexical selection places a crucial role in the determination of main verbs and additions of functional words. The testing result is shown in table 2.

Table 2: Experimental result for the Chinese generator

Total	Correct	Understandable	Wrong or no result	Accuracy
176	150	15	11	93.7%

The simplified generator has got improved performance. Another advantage that cannot be seen from the experimental result is that this generator is much easier for maintenance.

4. Conclusions

This paper introduces our approaches towards target language generation in S2S translation system. Considering the requirement of translation system, hybrid approach that combining template and feature based generation method has been adopted to fulfill the efficiency and flexibility requirement. Regarding the characteristics of IF, our micro-planning strategies has been properly designed, and sentence planning, phrase planning and lexicalization sub-modules has been built. We introduce thought of dependency in phrase planning, and reduce the inter taches to further improve the efficiency and maintainability. In lexicalization sub-module, different words are treated differently to gain better performance. In surface generation module, SFG is used for building up our Chinese generation grammar and functional unification algorithm for our generation algorithm. At the end, a simplified Chinese generator is built according to the characteristics of Chinese.

5. Acknowledgements

The research work described in this paper is supported by the National Natural Science Foundation of China under the grant No.60175012, 60375018 and 60121302, and also supported by the High- Tech Program (the 863 Program) under the grant 2002AA117010, the outstanding overseas Chinese Scholars Fund of Chinese Academy of Sciences under grant No. 2003-1-1, and also PRA project under grant No. SI02-05.

6. References

[1] Lori Levin, et al., "Balancing Expressiveness and Simplicity in an Interlingua for Task Based Dialogue", *Proceedings of Speech-to-Speech Translation Workshop at the 40th Annual Meeting of the Association of Computational Linguistics (ACL-02), Philadelphia, PA, July 2002, Pages 53-60.*

[2] Lori Levin, et al., " The NESPOLE! Interchange Format", *from "http://penance.is.cs.cmu.edu/nespole/ db/current/", Feb. 24th, 2003.*

[3] C-STAR Consortium, " Dialogue Act Annotation", *from "http://www.c-star.org/main/_RESTRICT/IF/htdocs/ comp/", October 5, 1999.*

[4] Hua Wu, Taiyi Huang, Chengqing Zong and Bo Xu, "Chinese Generation in a Spoken Dialogue Translation System", *Proceedings of COLING 2000, Pages 1141-1145.*

[5] Ehud Reiter, "NLG vs. Templates", *Proc of the Fifth European Workshop on Natural-Language Generation (ENLGW-1995), Leiden, the Netherlands, Pages 95-105.*

[6] Hua Wu, "Research on Theory and Methods of Chinese Language Generation and System Realization", *A Dissertation for the Requirement of the Doctor's Degree of Engineering, Institute of Automation, Chinese Academy of Science, Mar. 2000.*

[7] Wenjie Cao, Chengqing Zong and Bo Xu, "Research on IF Based Chinese Generation Approach", *Journal of Chinese Language and Computing, 2004.*

[8] Ehud Reiter, "A New Model for Lexical Choice for Open-Class Words", *Proc. Of the Fifth International Workshop on NLG, 1990.*

[9] James Pustejovsky, Sergei Nirenburg, "Lexical Selection in the Process of Language Generation", *25th Annual Meeting of the Association for Computational Linguistics, 1987, P87-1028.*