A Language-Independent System for Generating Feature Structures from Interlingua Representations

Murat Temizsoy Ilyas Cicekli

Department of Computer Engineering and Information Science,
Bilkent University, 06533 Bilkent, Ankara, Turkey,
e-mail: temizsoy@cs.bilkent.edu.tr, ilyas@cs.bilkent.edu.tr

Abstract

Two main problems in natural language generation are lexical selection and syntactic structure determination. In interlingua approach to machine translation, determining sentence structures becomes more difficult, especially when the interlingua does not contain any syntactic information. In this paper, a knowledge-based computational model which handles these two problems in interlingua approach is presented. The developed system takes interlingua representations of individual sentences, performs lexical selection, and produces frame-based syntactic structures. The system takes all the information about the target language from knowledge resources, in other words its architecture is language-independent. The implemented system is tested with Turkish through small-sized resources such that its output can be fed into a previously developed tactical generator to produce the final realizations of Turkish sentences.

1 Introduction

Interlingua approach to machine translation (MT) aims at achieving the translation task by using an intermediate, language-independent meaning representation [Nirenburg et al., 1992]. The use of such an artificial language, interlingua, makes the design of analysis and generation components separate in interlingua-based systems. Analysis is responsible for representing the input source text in interlingua, and generation produces the target text from those previously constructed representations. In other words, the source and the target language are never in direct contact in such systems.

Generation in such systems should at least perform lexical selection, syntactic structure creation, morphological inflection, and word order determination if planning (determination of overall text structure and sentence boundaries) is not considered. One approach to the design of generation module in interlingua-based MT systems is to handle the first two tasks in a separate architecture, get a form of syntactically represented target sentences, and achieve the last two tasks with a tactical generator. In this way, only the interlingua dependent tasks are handled in processing interlingua representations.

The aim of this paper is to present a computational architecture for generation which performs the tasks of lexical selection [Dorr, 1993] and syntactic structure determination [Mitamura and Nyberg, 1992] in interlingua approach. The system is designed to take the interlingua representations of individual sentences and produce their frame-based syntactic representations in which selected lexemes are included [Temizsoy, 1997]. A knowledge-based approach is utilized in the developed architecture such that information about the target language

is taken from knowledge resources. In other words, its architecture is language-independent. The utilized interlingua is mainly based on an *ontology*, a hierarchical world model, to represent propositional content. It also utilizes special frames to represent semantic and pragmatic phenomena encountered in analysis. The architecture uses ontology while processing interlingua representation in addition to *lexicon*, *map-rules* (relation between interlingua and target language syntactic structure), and target language's *syntax representation formalism*. The architecture of the designed system is given in Figure 1.

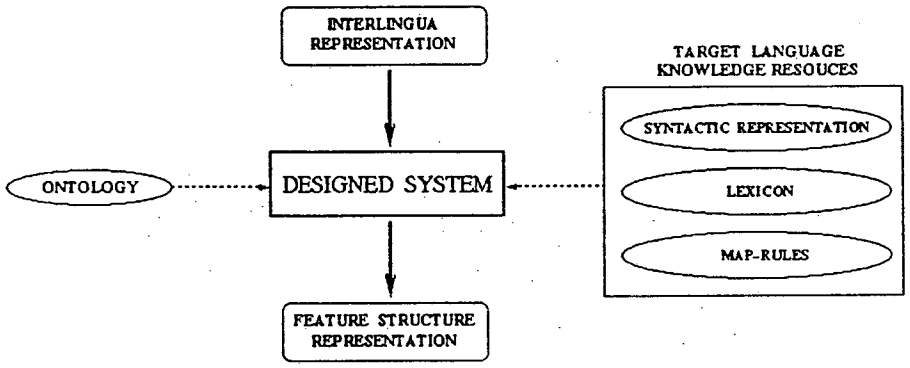

Figure 1: Architecture of the Designed System

The implemented system is used to generate the syntactic structure representations of Turkish sentences from their corresponding interlingua representations. The syntax representation formalism of Turkish is taken from a Turkish tactical generator previously developed by Hakkani [Hakkani, 1996]. The output of the system can be directly fed into this generator to produce the final realizations of Turkish sentences. Although input resources do not provide full coverage of Turkish, special consideration is given to linguistic phenomena encountered in Turkish such as free word-order and narrative tense [Temizsoy, 1997].

The rest of this paper is organized as follows. In Section 2, the interlingua formalism utilized in this work and its use of ontology is presented. Then, knowledge resources that provide information about the target language to the developed model are described in Section 3. Computational architecture of the system is presented in Section 4, and some specific examples from Turkish are given to demonstrate the system usage in Section 5. Finally, conclusion and and some possible future works are given in Section 6.

2 Interlingua and Ontology

The work described in this paper is based on interlingua approach to MT. In this approach, the meaning conveyed in the source text is represented using a language-independent, artificial language. The language formalism that is utilized in this paper is developed for MicroCosmos project at New Mexico State University and it is called as text meaning representation (*TMR*) [Mahesh and Nirenburg, 1996, Beale et al.. 1995]. Its formalism is based on two main knowledge resources: speaker's world knowledge about entities, events, and their relationships which are described in ontology, and linguistic information about semantic (aspect, modality, etc.) and pragmatic (speech-act, stylistics, etc.) issues. In this section, first a brief description of the ontology

is given, and then the interlingua formalism is presented with a demonstrative example.

The ontology used in this work is a hierarchical model of the real world [Mahesh, 1996]. It is built upon proposed abstractions, concepts, about the world entities, events, and relations. The concepts in the ontology are not designed to denote word senses in a specific language, instead they are defined to represent our common sense knowledge about the world. Each concept is represented as a frame and the information about its abstraction is described through a set of features with their value domains. For example, the concept *HUMAN* is defined to denote all human-beings in the world and it corresponds to the words '*man*', '*woman*', '*child*', '*John*', etc. in English. The frame given below is the simplified description of *HUMAN*.

$$\begin{bmatrix} concept & HUMAN & \\ definition & \begin{bmatrix} type & common/proper \\ name & human\text{-}names \\ gender & male/female \\ age & \geq 1 \ \& \leq 120 \\ job & teacher/engineer/\ldots \end{bmatrix} \end{bmatrix}$$

Representation of events in the ontology is somehow different from the entities since they are treated as predicates over arguments. So, an event concept provides extra information about its thematic structure such that each thematic role can take a set of entity concepts as its values. All concepts in the ontology are connected to others through a set of relations. The main relation, *is-a*, provides the hierarchical interpretation in the ontology such that child concepts define additional properties and put some constraints on the definition of their parent concepts. So, a *HUMAN* is a *MAMMAL*, which is an *ANIMAL*, etc. There are also other types of relations to provide additional information like a *MONITOR is-part-of* a *COMPUTER*.

The utilized language formalism, *TMR*, does not contain any specific information about the source language like lexemes and syntactic structure. It uses a frame-based notation and it is heavily based on the ontology. The concepts from the ontology are used to denote the propositional content of the input sentences. But since concepts are only abstractions, their features should be instantiated to denote real things when used in *TMR*. Although concept instances provide the information about the propositional content, semantic and pragmatic properties of the sentence should also be described in *TMR*. To facilitate this, *TMR* language provides special frames for representing aspectual properties, temporal relations, speech-acts, stylistic factors, etc. Instead of describing the *TMR* language in full detail, an example representation is given to demonstrate its formalism. The *TMR* of the sentence "*The man gave a book to the child*" is given in Figure 2.

Note that, although English words are used as concepts, they are not denoting English word senses, they are just generic abstractions. Each frame in a *TMR* is indexed to differentiate between frames with the same name. Both of the phrases '*the man*' and '*the child*' are represented with frames of the same concept, *HUMAN*, but their instantiated features are totally different. The given *TMR* simply denotes the event $give(man, child, book)$ with its aspectual properties ($aspect_1$) and its temporal relation with the time of utterance ($temp\text{-}rel_1$). Information about the speech situation is described with $speech\text{-}act_1$ frame. Observe that, there is nothing specific about the English sentence that is represented in the given *TMR*.

$GIVE_1$		$HUMAN_1$	
agent	$HUMAN_1$	*type*	*common*
destination	$HUMAN_2$	*gender*	*male*
theme	$BOOK_1$	*age*	≥ 18
polarity	*positive*	*reference*	*definite*
aspect	$aspect_1$		
time	$time_1$	$HUMAN_2$	
		type	*common*
$aspect_1$		*age*	≤ 12
phase	*perfect*	*reference*	*definite*
duration	*momentary*		
iteration	*single*	$BOOK_1$	
telicity	*false*	*reference*	*indefinite*
$speech\text{-}act_1$		$temp\text{-}rel_1$	
type	*declarative*	*type*	*after*
scope	$GIVE_1$	arg_1	$time_2$
time	$time_2$	arg_2	$time_1$

Figure 2: TMR Representation of "The man gave a book to the child"

3 Knowledge Resources

The developed architecture is language-independent, it takes the information about the target language from three knowledge resources: lexicon, map-rules, and syntactic structure representation formalism of the target language. Lexicon, besides its other usages, provides information about the relationship between concept instances and word senses of the target language [Dorr, 1993]. Map-rules define how the content of a *TMR* is related to the syntactic structure of the target language [Mitamura and Nyberg, 1992]. The last knowledge resource provides the information about the structure of the syntactic representation formalism.

The interface between concept instances in *TMR* (denoting events and entities) and word senses of the target language is established using semantic and pragmatic properties of lexemes that are defined in the lexicon. Since nouns denote entities and verbs denote events in a language, each word that belongs to one of these categories is also defined as a concept instance in the lexicon. So, for every *TMR* frame that is a concept instance, there is a set of candidate lexicon entries that are defined using the same concept. For example, if the previous example is considered, there are at least two candidates for an instantiated *HUMAN*, that are '*man*' and '*child*'.

The meaning of every noun and verb is defined in the lexicon by constraining the abstraction provided by the parent concept. For example, one sense of '*man*' can be defined as '*a male HUMAN whose age is greater than 17*'. Such definitions are the major source of information used in lexical selection. In addition to meaning definitions, pragmatic properties of word senses can also be defined in the lexicon. For example, the preference of '*guy*' over '*man*' in informal situations to express a negative attitude can be encoded by attaching the necessary stylistic and attitude requirements to the definition of '*guy*'. Note that, words belonging to adjective and adverb categories are not defined as concept instances. Instead, they are represented in *TMR*s as features of events and entities, and their realizations are achieved through map-rules in generation.

The syntactic structure formalism of the target language is represented using a frame-based notation, like feature structures. The developed system uses the syntax formalism through its corresponding tree structures defined in the knowledge resource. The relation between syntactic structure and *TMR* is described using map-rules. Each map-rule is related with either a concept from the ontology or a special frame type used in the *TMR* language to encode certain semantic or pragmatic issues such as aspect, modality and speech-act. Map-rules are utilized to relate thematic roles to grammatical counterparts, to create specific syntactic features such as tense, voice, and modifiers, and to determine the syntactic connection between events. Map-rules defined for concepts follow the inheritance mechanism in the ontology and general syntactic properties are determined in parent concepts.

Each map-rule mainly provides two types of information: *content conditions* and *update operations*. Content conditions should be satisfied by the input *TMR* before update operations are applied. Since map-rules should be *TMR* independent, making references to arbitrary frames in the input *TMR* is not allowed in the definitions of content conditions. In fact, only three frames can be referenced in conditions: current active frame, current event frame, and current speech-act frame. Content conditions are defined to check the existence of certain features and/or their values in these frames. Update operations change the constructed syntactic structure of the sentence when they are applied.There are three types of update operations: *feature addition* such as $add(tense, past)$, *frame addition* such as $add(subject)$, and *frame-to-frame mapping* such as $map(agent, subject)$.

4 Computational Model

The computational model is designed to process the *TMR* of a sentence as input and to construct the syntactic structure of that sentence selecting lexical items for the constituents of that sentence. To achieve these tasks, the model makes use of ontology and knowledge resources developed for the target language. Although lexical selection and syntactic structure construction can work in parallel during *TMR* processing, they can also be handled in two independent submodules. Lexical selection is activated whenever the *TMR* frame is a concept instance, and it is based on the semantic and the pragmatic properties of the candidate lexemes. Each *TMR* frame activates its attached map-rules to update the constructed syntactic structure. Besides these tasks, the model should determine the process order of *TMR* frames So, the main module decides on the processing order and activates the lexical selection and the map-rule application submodules whenever necessary. The architecture is described in Figure 3.

4.1 Lexical Selection Module

Lexical selection is performed for every *TMR* frame which is a concept instance. Since there are generally more than one candidate lexeme for such a frame, the module should select the most near-perfect word sense that carries the meaning residing in the *TMR* frame into the target sentence. So, lexical selection in this work is mainly based on the meaning distance between the frame being processed and the candidate lexemes [Temizsoy, 1997]. The distance calculation is done through assigning penalties to features that are not matched in the two definitions. After calculating the proximities between the meaning in the *TMR* frame and the candidate lexemes, the module returns the closest one as the selected word sense. Although proximity of meaning is the major criterion.

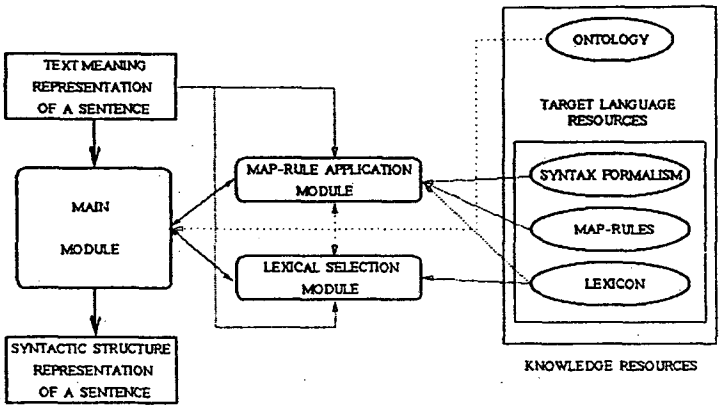

Figure 3: Computational Model

there are cases in which there are still ambiguity between candidates. In such cases, in addition to the semantic constraints of lexical items, their pragmatic properties are also taken into account.

Lexical selection is achieved in three successive steps: first the candidates whose subcategorization constraints are not satisfied in the *TMR* frame are removed from the list (*context-dependent* selection), then a distance is assigned to the remaining candidates by comparing the meaning residing in the *TMR* frame with their definitions in the lexicon (*context-independent* selection), and if it is still impossible to make a selection on those calculated distances, the stylistics and pragmatic properties of candidates are utilized . The architecture of lexical selection module is described as in Figure 4.

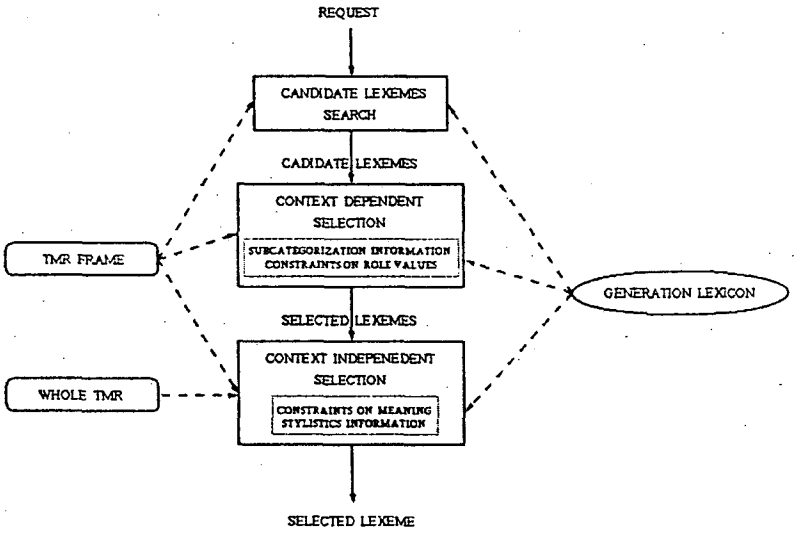

Figure 4: Lexical Selection Module

There are some heuristics that are utilized in calculating the distance between a *TMR* frame and a lexical item definition, and they can be summarized as follows:

- A penalty value is assigned to a feature that is in the lexeme definition, but not in the *TMR* frame, to minimize extraneous meaning introduction.
- Another penalty value is assigned to a feature that is in the *TMR* frame definition, but not in the lexeme definition, to reduce uncoverage of meaning.
- Match between two values from the same domain is proportional to the distance in ordered values and the intersection sizes in ranges.
- The calculated match is normalized by the domain size of the feature to minimize distances in larger domains.
- The final distance is rated by its importance on the overall meaning such that mismatches in less relevant features have smaller influence over the final proximity.

4.2 Map-Rule Application Module

This module collects all the map-rules associated with the *TMR* frame being processed and updates the constructed syntactic structure for map-rules whose content conditions are satisfied. The map-rules developed for ontology concepts follow the inheritance mechanism provided in the ontology. So, while processing a *TMR* frame which is an concept instance, this module should traverse the ontology in a bottom-up fashion to apply map-rules that are associated with the ancestor concepts of the concept instance. Note that, since a lexical item can require some updates on the syntactic structure, this module also applies the map-rules associated with the selected lexical item. If the processed *TMR* frame is not a concept instance, the map-rules associated with its frame type are applied to update the constructed syntactic structure.

As mentioned, the syntax formalism of the target language is represented as tree structures in which frames are the internal nodes and the features are the leaves. Since frames and features in such a representation are used to describe distinct syntactic phenomena, unique names should be given to them. This uniqueness property is utilized to find the place of a feature or a frame directly in the tree structure without traversing. So, feature or frame addition to the constructed tree is achieved by just finding its place, forming a partial tree through traversing the defined tree structure in a bottom-up fashion, and merging that partial tree to the previous constructed syntactic structure. Note that, these operations can be done in logarithmic time [Temizsoy, 1997].

Some syntactic constructs have the same form although their syntactic realizations are different, like noun phrases. So, generally their structure is defined under a common frame which can be the value of various features in the overall structure. For example, noun phrases are the fillers of grammatical roles subject, direct-object, etc. To utilize such a form, the representation formalism is allowed to have more than one tree in its definition (one for verbal phrases, another for noun phrases, etc.). The tree representing verbal phrase is taken to be the main one, all constructed children trees should be attached to it. The information about the attachment place of a child tree (noun phrase is the subject, place, etc.) is obtained from previous *frame-to-frame mapping* rules such as $map(agent, subject)$.

4.3 Main Module

The main module is responsible for determining the processing order of the *TMR* frames in the input. In this work, a depth-first strategy is used in ordering which is utilized in processing *TMR*s that have more than one event. Since verbal phrases are represented with the main tree in the syntax formalism, trees constructed for supplementary events should be attached to the tree built for the main event. Since depth-first processing guarantees that all children frames together with their parent frame are processed before processing the other *TMR* frames, the algorithm can safely constructs the syntactic structures of supplementary events and connects them to the main tree.

So, the main module first constructs a processing stack which contains the main event (scope of the speech-act), relations or special frames (casual, temporal, textual relations, speech-acts, etc.), and other events in the given order [Temizsoy, 1997]. After creating the syntactic tree of a supplementary event, the algorithm finds the syntactic relation of that event to the main one. This determines the attachment place of the child tree in the main tree. There are three cases in which events are related to the main one:

- Another event is used to describe a thematic role of the main event, like in "*I want to read a book*". In this example, the phrase '*read a book*' is processed individually by the algorithm, and its corresponding constructed tree is attached as the direct-object of the sentence (assuming that $map(theme, direct\text{-}object)$ is previously applied).
- The connection between two events is a relation (casual relations, conjunctions, etc.), like in "*Since John did not study enough, he could not pass the exam*". In this example, first the main event, *PASS*, is processed, then the frame which defines the relation is taken from the processing stack. Since one of its arguments is not processed yet (the event *STUDY* in this example), the algorithm first constructs the tree structure of *STUDY*, and then apply the syntactic realization of the relation to the constructed trees of *PASS* and *STUDY*.
- Another event is introduced to give some additional information about the main event or one of its components, like in "*John, who came to your birthday party last month, went to Istanbul*". In this example, the algorithm first constructs the corresponding tree of *GO*, then it processes the event *COME*, and finally finds its relation to *GO* (definition of subject) and merges its constructed tree to the main one.

5 Implementation

The implementation of the presented architecture is done in Prolog. Currently, the implemented system is tested with Turkish. Turkish syntax formalism is taken from a previously developed Turkish tactical generator [Hakkani, 1996] such that the successive execution of the two systems produces real Turkish sentences from interlingua representations. For example, when the *TMR* example given in Figure 2 is fed into the developed system, the feature structure representation, which is shown in Figure 5, of the Turkish sentence "Adam kadına bir kitap verdi" is produced. Then, this feature structure is fed into the tactical generator to produce the surface form of the sentence.

One of the prominent features of Turkish is its free word-order structure. Changes in the default word-order generally serve to introduce pragmatic differences. For example, the constituent

Feature Structure produced by the developed system:

```
[[s-form,finite], [clause-type,predicative], [speech-act,declarative], [voice,active],
 [verb, [[sense,positive], [mode,past], [root,'ver'], [category,verb]]],
 [arguments,
    [[subject, [[referent, [[arg, [[root,'adam'], [category,noun]]],
                           [agr, [[person,third], [number,singular]]]]],
               [specifier, [[quan,[[definite,positive]]]]]]],
     [goal, [[referent, [[arg, [[root,'çocuk'], [category,noun]]],
                        [agr, [[person,third], [number,singular]]]],
             [specifier, [[quan, [[definite,positive]]]]]],
     [dir-object, [[referent, [[arg, [[root,'kitap'], [category,noun]]],
                              [agr, [[person,third], [number,singular]]]]]]]]]]
```

Surface Form produced by the tactical generator:

"Adam kadına bir kitap verdi"

Figure 5: The Results of Generation

which is placed right before the verb is the focused element in the sentence. So, representing the sentence "Camı Ali kırdı" ("it was Ali who broke the window") is achieved by attaching a *saliency* (importance the speaker attribute to) attitude such that its value is greater than a predefined value. To process this information in *TMR*, a map-rule is associated to the *ENTITY* concept which checks the existence of such an attitude and perfoms the introduction of *topic*, *focus*, and *background* information into the feature structure representation. *Topic* indicates the sentence initial position, *focus* is the preverbal position, and *background* indicates the postverbal positions.

6 Conclusion

Lexical selection and syntactic structure construction are two important tasks to be handled in interlingua-based generation. This paper presents a computational model which is designed to achieve these tasks in interlingua approach. It takes individual sentences represented in a specific interlingua formalism and produces frame-based syntactic structures of the target sentences. It utilizes a knowledge-based approach to this generation task to make its architecture language-independent. It takes all the information about the target language from three knowledge resources: lexicon, map-rules, and target language syntax formalism.

The implemented system is used to produce feature structure representations of Turkish sentences. The feature structure formalism is taken from a tactical generator previously developed for Turkish such that the output of our system can be fed into this generator to produce the final realizations of Turkish sentences. By using these two systems, generation of Turkish sentences is achieved from the specific interlingua formalism.

References

[Beale et al., 1995] Beale, S., Nirenburg, S., and Mahesh, K. 1995. Semantic analysis in the mikrokosmos machine translation project. In *Proceedings of the 2nd Symposium on Natural Language Processing (SNLP-95)*, Bangkok, Thailand.

[Dorr, 1993] Dorr, B. J. 1993. The use of lexical semantics in interlingua machine translation. *Machine Translation*, 4:3:135–193.

[Hakkani, 1996] Hakkani, D. Z. 1996. Design and implementation of a tactical generator for turkish, a free constituent order language. Master's thesis, Bilkent University, Ankara Turkey.

[Mahesh, 1996] Mahesh, K. 1996. Ontology development for machine translation: Ideology and methodology. In *Memoranda in Computer and Cognitive Science MCCS-96-292*, Las Crues, New Mexico State University.

[Mahesh and Nirenburg, 1996] Mahesh, K. and Nirenburg, S. 1996. Meaning representation for knowledge sharing in practical machine translation. In *Proceedings of the FLAIRS-96. Track on Information Interchange, Florida AI Research Symposium*, Key West, Florida.

[Mitamura and Nyberg, 1992] Mitamura, T. and Nyberg, E. 1992. Hierarchical lexical structure and interpretive mapping in machine translation. In *Proceedings of COLING-92*, Nantes, France.

[Nirenburg et al., 1992] Nirenburg, S., Carbonell, J., Tomita, M., and Goodman, K. 1992. *Machine Translation: A Knowledge-Based Approach*. Morgan Kaufmann, San Mateo, California.

[Temizsoy, 1997] Temizsoy, M. 1997. Design and implementation of a system for mapping text meaning representations to f-structures of turkish sentences. Master's thesis, Bilkent University, Ankara Turkey.