3. Our Approach

Before discussing our approach to language teaching, we briefly discuss the structure of Marathi language; this will also be useful to get a feel of the different aspects to be covered in the teaching process. This is not meant to be an exhaustive tutorial, but only to bring up the major structural elements and associated properties and semantics. We will ignore most of the nuances and special cases.

3.1 About Marathi

Marathi grammar shares similarities with other modern Indo-Aryan languages such as Hindi, Gujarati, Punjabi, etc. The words in the sentences are classified on the basis of their functions. There are eight major parts of speech (POS) in Marathi, as in English: Noun, Pronoun, Adjective, Verb, Adverb, Postposition, Conjunction and Interjection. Marathi has three grammatical genders (liMga), viz., masculine, feminine and neuter; two numbers (vachana), viz., singular and plural; and three persons (purusha), viz., first, second and third. Nouns in Marathi have eight cases, viz., nominative, objective, instrumental, dative, ablative, genitive, locative and vocative. Words are also classified according to whether they would be inflected or uninflected. Inflectional words change their form with the change in gender, number, case and tense whereas the noninflectional words do not undergo any type of change and always remain in their standard form. In Marathi, noun, pronoun, adjective and dhaatoo are generally inflectional words. Adverb, postposition, conjunction and interjection are noninflectional words.

Dhaatoo (root form of verb) is inflected in a sentence, with the change in gender, number and person of ‘dhaaturoopesha’. In active voice, subject of sentence is ‘dhaaturoopesha’, whereas in passive voice, object is ‘dhaaturoopesha’. In Marathi, verbs are classified into two main types, viz., transitive and intransitive. Apart from these two main types, there are two more important categories, viz., compound and auxiliary. Auxiliary verbs are those by the help of which, participle is in a position to complete the meaning of sentence. While helping the participle, these verbs make some changes in the original meaning to express possibility, obligation, permission, ability or capacity, request, regularity, etc. For example, ‘chadhata aahe’, ‘chadhalo aahe’, ‘chadhoona jaa’ are compound verbs. According to [Damale, 1911], participle
+ auxiliary verb = compound verb. In compound verb, participle represents the main action and auxiliary verb supports the participle to complete the meaning of sentence.

You can’t form compound verb by just putting together the combination of any participle with any auxiliary verb; rather you need to keep in mind what and how many meaning the combination of participle and auxiliary verb generates. By combining selected participle of dhaatoo (root form of verb) and selected auxiliary verb, one can get a compound verb. Actually, a compound verb represents a single action and infers one meaning from sentence. For example, ‘chadhata aahe’ is a compound verb. But, ‘vaachoona jhopa’ is not a compound verb, since it represents two separate actions vaacha (read) and jhopa (sleep). We will see the compound verb in detail in subsection 4.2.3.4.

Tense is divided into three main periods (kaaLa): vartamaana (present), bhoota (past) and bhavishya (future). Each of them is further divided into four sub-categories, viz., simple (saamaanya), continuous (apurNa), complete (purNa), and ritee. Marathi has prayoga comparable to voices in English. In a broad sense, prayoga is the relationship of verb with subject or object. In Marathi language, with respect to prayoga, three types of sentence structures are seen.

- Kartaree: verb is inflected as per the gender, number and person (G-N-P) of subject. Similar to active voice in English
- KarmaNee: verb is inflected as per the G-N-P of object. Similar to passive voice in English.
- Bhaaave: verb is not inflected as per G-N-P of subject or object; rather verb is always in third person, neuter gender and singular number.

From this, one can say that prayoga gives the idea that the verb is inflected due to G-N-P of subject or object or independent of both [Gowilkar, 2006].

In case of kartaree prayoga (active voice), transitive and intransitive sentences are possible. Intransitive kartaree prayoga can be made in all tenses. Transitive kartaree prayoga can be made in all tenses except the past tense. Subject is always in nominative case. The object may or may not be present.

In case of karmaNee prayoga (passive voice), object is important because the verb inflects as per G-N-P of object. Object is in nominative case where as subject is
declinated. In this case, object is a must in the sentence. [Damale, 1911] has classified karmaNee prayoga into two main types, namely, ‘Pradhaana Kartruka KarmaNee’ (Subject Oriented Passive) and ‘GauNa Kartruka KarmaNee’ (True Passive Voice). In subject oriented passive, both subject and object are nominal, but the verb agrees with the G-N-P of object and not with the subject. In Marathi, for transitive verb, active voice sentence in past tense cannot be formed. Hence, such sentences are formed in passive voice. Since subject is also important in these sentences, [Damale, 1911] has given the name to this prayoga as Pradhaana Kartruka KarmaNee or Subject Oriented Passive.

GauNa Kartruka KarmaNee is the real or true karmaNee prayoga. In this type of sentences, object rules the verb. The subject is seen in different cases other than nominative. GauNa Kartruka KarmaNee is further divided into four types, namely, PuraaNa (old), Naveena (new), Shakya (can), and Samaapana.

According to [Joshi, 1993], there are three different historical sources in Sanskrit for dative subject in Marathi. In case of dative subject, verb is inflected as per the G-N-P of object.

Having looked at the various major aspects of Marathi language, we now discuss our approach to teaching Marathi language.

3.2 Our Approach to Language Teaching

As mentioned before, our approach to language teaching is inspired by the way children learn their first language. Every child learns the language of his parents. The parents usually help the child to learn the first language, but they do not teach in the traditional sense. The teaching learning process starts with words without telling the child what type of word it is. The child begins to imitate what he has heard from his parents as best as he can. The parents in subtle ways correct the child if he makes any error and reward him for correct attempts by responding positively.

When a child starts putting words together, he begins to understand the basic rules of syntax and how to apply them correctly. Parents, then, change carefully the constituents of the sentence and child repeats the same. In this process, child starts making association and also learns new words. Children absorb what they hear, i.e., words, sentences, phrases, etc. They look for and abstract patterns from them and, in
the process, construct their own grammatical rules which they then apply to create new phrases and sentences. Our approach is based on these observations of first language learning where children learn the language through continuous trials and appropriate interventions from the parent.

We know that the most critical component of any language is sentence; it is also the most challenging aspect to teach correctly and effectively. To teach sentence construction, the learner needs to be made familiar with various language constituents, their positional choices and how these language constituents get inflected (if any) with change in the other constituents. Our approach to teach sentence construction is based on showing carefully selected variants of various types of sentences. For example, consider the sentence “mee sakaaLee mumbailaa jaato” (I go to Mumbai in the morning). Here, to make the learner familiar with the positional choices and inflections for the various constituents, we show variants of this sentence by changing the components one at a time. For example, change ‘mee’ to ‘tee’ or ‘te’, change ‘mumbailaa’ to ‘puNyalaa’ or ‘londonlaa’, change ‘sakaaLee’ to ‘dupaaree’. When we generate the variants of a sentence, by changing the component one at a time, the learner can see the structural and inflectional roles played by various constituents without having to name and describe the various constituents and their roles. The sentence teaching process along this approach requires generation of syntactically correct and meaningful sentences as per the grammar of the language, generation of selected variants of a given sentence and feedback mechanism to detect student’s mistake if any. We discuss in detail these aspects in subsequent sections.

3.2.1 Teaching Learning Process

In TLP, pedagogy, i.e., the way teacher plans and deliver his lessons, plays an important role. While designing any LTS, one has to keep the balance between the three pedagogical goals, viz., fluency, accuracy and complexity [Skehan, 1998]. Keeping in mind these goals, we used the most fundamental pedagogical principles of language learning [William and Kinshuk, 2002], viz., practice, learner orientation, changing role of a teacher, situation based learning, and written versus spoken language learning in our TLP. The pedagogy model we considered is the basic model
for interaction/dialogue and one which gives emphasis on sentence teaching by generating variants of sentences.

A number of pedagogical elements have been used to define structures or units of education material. This could be a lesson, an assignment, a multiple choice question, a quiz, a case study, etc. Our TLP starts with basic lesson on alphabets and vocabulary building. Then there are lessons on word inflections and finally lessons on construction of various types of sentences, which is the core part of the model.

As mentioned earlier, our focus in this thesis is on teaching sentences. But, before teaching sentence construction, it is necessary to make learner familiar with basic learning unit, i.e., alphabets and basic vocabulary. The implemented system includes these functionalities also, and in the teaching sequence these are included. These learning units will be associated with audio and a pronunciation key so as to enable learner to get well-versed with the pronunciation also. An image will also be displayed to represent a language constituent in a situated context.

Truly speaking, vocabulary learning is followed in child’s language learning also; but alphabet is usually not introduced. This is often done long after the child acquires good familiarity with spoken language, primarily for reading and writing material in the language. However, since we are using text based interaction\(^1\) on a computer, we will need alphabet familiarity on the part of learner. Hence, alphabet familiarity is necessary from our implementation point of view. We emphasise more on visual recognition rather than speech recognition. Our focus is not on alphabet learning.

After introductory lessons on alphabets and vocabulary building, lessons on word inflection are being covered. These lessons will cover noun and verb inflection. Knowledge of inflectional operation will help learner while learning sentence construction, which is the next stage. In sentence construction, the learner will be taught to form various types of sentences, viz., simple declarative, simple declarative with multiple subjects and multiple objects, sentences in passive voice, interrogative sentences, exclamatory sentences, and compound sentences. The learner is advised to take the lessons in above mentioned order. But, this is not imposed on the learner.

\(^1\) Automatic speech recognition and synthesis, particularly for Indian languages, is still quite immature to consider speech based interaction for language teaching.
During first language learning, parents do not tell the child about the type of sentence he is learning. Also, no particular sequence is followed while teaching various types of sentences. But, most of the time the sentence teaching starts with building simple sentences. We followed same strategy in our TLP. Also, our strategy is flexible enough that it allows learner to take lessons on passive voice, exclamatory sentences, and interrogative sentences in any order. But, before learning simple compound sentences, the learner is required to take lessons on simple sentences. Since the simple compound sentences are built using two or more simple sentences. Hence, it is necessary to have an idea about simple sentences, before moving on compound sentences. Thus when the dependency exists, we have enforced it, leaving the other choices open to the learner.

3.2.2 Teaching Sentence Construction

Teaching sentences start with construction of simple sentences. Here, by simple sentence, we mean a sentence which consists of subject, object and verb, optionally including an adjective and adverb. To teach sentence construction, we show carefully selected variants of sentence. We change the constituent of sentence one at a time so that learner can get the idea of positional choices and inflectional role played by various constituents. To understand this approach, consider Table 1 which shows a set of sentences.

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Person</th>
<th>Gender</th>
<th>Constituents of Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>kartaa</td>
</tr>
<tr>
<td>1</td>
<td>Third</td>
<td>M</td>
<td>to</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>M</td>
<td>to</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>M</td>
<td>to</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>F</td>
<td>tee</td>
</tr>
</tbody>
</table>

In Table 1, sentence no. 1 is the base sentence and sentence no. 2, 3 and 4 are some of its variants. Consider the second sentence in Table 1. Here we changed the adverb ‘sakaaLee’ of base sentence to ‘dupaaree’. With this change in adverb, as per grammar rules, all other constituents remain the same, i.e., no change occurs in other constituents. Now, consider the third sentence in which we changed the object ‘chitra’
of base sentence to ‘raaMgoLee’. With this change in object also, there is no change in other constituents. Consider the fourth sentence where the masculine subject ‘to’ is replaced by the feminine one ‘tee’. With this change, there is a change in the verb. Instead of ‘kaaDhato’, a verb form ‘kaaDhate’ is used. Looking at a number of such examples, learner will come to know that if there is a change in adverb or object, then the other constituents remain same, but, if there is a change in subject then a change may occur in verb, i.e. the verb gets inflected differently. Here, the verb inflects due to the change in gender of the subject. In this way, by generating system driven variants of sentence, the learner can see and understand the positional choices as well as inflectional role played by various constituents without explicitly discussing them.

We also need to allow the learner to check out variants of his/her own choice. For example, if he is getting doubts as to how different subjects are affecting the verbs, he may want to experiment with his own choice of subjects in a given sentence. We need to allow the learner to control the generation of variants of sentence. Here, the system generates a base sentence, and the learner can change constituent of sentence one at a time which results in a revised sentence. This scheme helps the learner in clearing any particular/specific doubt in positional choices or inflectional role played by constituent. Learner can practice this lesson for any number of iterations for different sentences. In this way, controlled repetition helps learner to get mastery over particular topic, and this forms the core of our approach.

3.2.3 Interaction and Assessment

Interaction serves a variety of functions in the educational transaction. The primary one is intervention by the teacher in the form of feedback or hint provided when student makes mistake while solving a problem [Sims, 1999]. [Dewey, 1916] referred to interaction as the defining component of the TLP that occurs when student transforms the inert information passed to them from another, and constructs it into knowledge with personal application and value.

Sentence construction process discussed in previous subsection will give learner a fair idea about the various types of sentences, positional choices and inflectional roles played by various constituents. To see, whether learner understands a particular
concept properly or not, assessment is essential. Assessment also helps in identifying the remedial solutions for misconception in any topic.

There are a number of assessment models being employed in practice, including multiple choice questions, short answer questions, long answer questions, subjective questions, practical tests, viva-voce, presentations, etc. Different types of assessments provide different kinds of feedback about the learner’s understanding. Rather than looking at the response to theoretical questions, more effective assessment is to observe the learner in practice. This is not often possible due to various reasons, for all subjects. In our case of language learning, we could ask the learner to write sentences and we can check them for correctness. But, this is difficult to implement in practice.

Consider the learner’s response in the form of a sentence. Normally, one can analyse any natural language input for syntactical well-formedness by parsing it. Parsing process identifies sentence structure for syntactically correct sentence only. Since, the student is a learner, his response may not always be syntactically correct. In such a situation, parser is not in a position to generate sentence structure called as parse tree. Since the sentence structure is not available, it is difficult to analyse student’s mistake and hence not possible to provide any feedback to the learner to correct his mistake. Errors produced by typical parsers are useful only for those who are fairly knowledgeable about the language. On top of this, general language parsers are very difficult to design and normally not available for Indian language. We, therefore, adopt a model of scaffolding, ensuring restricted direct user input so that validation is easier.

3.2.4 Gap Filling Exercise – A Scaffold

Students progress most effectively, when they have been well guided during TLP. Scaffold is a form of assistance provided to a student by a teacher that helps the student to perform a task and become confident in a subject of new knowledge through interactions or dialogue [Vygotsky, 1967; McLoughlin and Marshall, 2000]. Scaffolding engages the students actively at their current level of understanding. Scaffolding is also often used in order to support the problem solving approach.
Teaching strategies discussed in previous subsections give the learner a fair idea about sentences. Also, he gets the idea of inflectional rules and the role played by the language constructs. Now, to see if any gap is left in learner's understanding about previous lessons, problem solving approach is used. Ideally, a tutor can ask learner to write various type of sentences and check for its correctness. But, there are various issues in implementing this mode as discussed in subsection 3.2.3.

[Bruner, 1978] said that for learning to take place, appropriate social interaction frameworks must be provided, as in case of children learning their first language. In first language learning, the instructional component consists of a supporter or helper (often mother) who provides structure or framework that allow the child to learn a language. In our LTS, we provided the temporary structure or framework like Gap Filling Exercise. The learner can use this support for constructing the pieces which are left out and bridge the gap in his understating. From TLP point of view, it is important to identify the content that needs scaffolding [Lajoie, 2005].

Most of the time, the inflectional changes in sentence occur only in subject and verb, and most of the language problems occur in this regard. Hence, we devise two scaffolds, viz., for constructing the right form of a verb and for constructing the right form of a pronoun, both focusing on the inflectional aspects of sentence construction. In first case, a sentence is generated without verb whereas in second case, the sentence is generated without subject. The learner is supposed to fill the left out constituent and complete the sentence. To complete this task, the LTS provides clues whenever required, carries out diagnosis of student’s mistake, and provides appropriate feedback and motivation. We, now, discuss these two scaffolds in detail.

3.2.4.1 Constructing the Right Form of a Verb

In this mode, where learner is asked to construct right form of a verb, a sentence is generated and shown without the verb. However, the dhaatoo of the verb is shown. The learner is asked to complete the sentence by filling in the right form of a verb using specified ‘dhaatoo’ with appropriate gender, number and tense. The learner’s response is compared with the correct one in generated version. If the learner’s answer does not match with the expected output, the appropriate rule is shown to him along with his mistake. But, if multiple rules satisfy his answer, the system tries to
interact with the learner in order to understand where he has made a mistake. This interaction will eventually lead the learner to identify correct answer.

Consider the following example where the learner is asked to complete the sentence with verb using dhaatoo ‘khaa’ for feminine gender and present tense.

**tee sakaalLee aaMbaa _____.** (she ______ mango in the morning.)

Here, the person and number are not provided to the learner. The learner is supposed to infer this information from the subject. Using the person and number thus inferred and the gender and tense provided, learner is supposed to identify the right form of a verb.² The learner response is compared with the system’s answer. If it is right, he can proceed for next sentence, else, the system checks his misconception about subject regarding the person or number using incremental feedback and right time interventions.

If the answer of the learner is wrong, then the system tries to find out what misconception about the subject leads to the wrong answer provided by the learner. There are two broad possibilities. One, he has applied a transformation under wrong condition. Two, he has not inferred the conditions correctly. If learner has added the endings corresponding to wrong person, then there is a misconception about the person of the subject. Hence, system asks the student to tell the person of the subject. If his answer is correct, then system informs it to the learner and also asks about the number of the subject. If number is also correct, then it means that the learner has applied an invalid rule. If the number is wrong, then system lets the learner correct the number, and asks again to complete the same sentence with correct verb. If it is correct, then it is accepted and learner is prompted for next sentences, else, the system shows the relevant rule and correct form of verb.

In this way, the learner will be informed about the mistake he has made, be it with the person, the number, the rule applied or the application of incorrect rule. In this whole process, system does not show directly the correct answer to the learner; instead he will be guided to arrive at the answer with right time interventions and incremental feedback.

² The reason of providing the gender is that the same subject name (pronoun) is seen in more than one gender. For example the subject name ‘tee’ has G-N-P (Feminine-Singular-Third) and (Neutral-Plural-Third) and it takes different endings. So it is necessary to provide the gender.
3.2.4.2 Constructing the Right Form of a Pronoun

In the test for construction of right form of a pronoun, a sentence is generated with the subject in the standard form and learner is asked to replace it with appropriate variant (see example below). Here, gender and number of subject is provided along with tense to complete the sentence. To get the correct pronoun, the learner tries to analyze the person of the pronoun and accordingly gives his answer. The system generates its own answer and compares with the learner’s answer and informs the outcome to the learner. If system finds that the learner’s answer is wrong, then it tries to interact with the learner in order to understand where he went wrong, in a process similar to that adopted for verb form.

Consider the following sentence, where the learner is asked to complete the sentence with appropriate pronoun having plural number, feminine gender and present tense.

________ saMdhyaakaaLee baageta basato.

To get the correct subject form, one needs gender, number and person. Here, person is not provided. But, it can be derived using gender, number, tense and pratayaya (endings) that is added to dhaatoo ‘basa’. If the learner’s response is correct, then he continues to the next sentence. Otherwise, system tries to indentify the learner’s misconception about the person of subject. So, it asks about the person of subject. If the learner’s response is correct, it means that the learner has inferred the correct person from the verb form, but applied an invalid rule. In this way, the system guides the learner to the correct subject form.

In this way, adequate scaffolding, right time intervention, and incremental feedback help the learner in understanding a particular topic in a better way, which results in successful learning. But, to execute these teaching strategies like generation of sentence and its variants (system as well as user driven), recognition of student response and analysis of it, giving him appropriate feedback, etc, system requires to ‘know’ the full grammar of the language. Keeping in mind these aspects, IELIL is structured as an ITS around this core notion. Now, we discuss the overall framework of IELIL.
3.3 Framework of IELIL

The first aspect to take into consideration is the design of underlying structure, for a constructivist web-based LTS, modelled around the architecture of an ITS. An important reason for selecting this framework is the separation of concerns among components it offers. This section describes in detail the overall framework of IELIL (see Figure 2).

Figure 2: Framework of IELIL

Figure 2 shows that the learner interacts with Pedagogic Manager (PM) through the graphical user interface (GUI). The PM is interconnected with every other component. The PM consults with other components such as the Learner Model, Domain Knowledge Base (KB) and Pedagogic Knowledge Base (KB) and accordingly plans the teaching strategies and content. These components together capture the various types of knowledge required to effectively implement the different behaviours mentioned in the previous section.

It was mentioned previously that the learner can take session on any topic in any order. To get an idea of how our system works, consider that the learner has selected the lesson on simple sentences. The simple sentences are taught in two ways. First, by generating system driven variants and, secondly, by generating user driven variants.

If the learner is taking the lesson on simple sentences first time, then, it is suggested to him to go through the system driven simple sentences. The system driven simple sentences are the set of sentences where the various constituents such as subject, adverb and object are changed one at a time in each sentence.

To generate system driven simple sentences, PM interacts with pedagogic KB and domain KB. These components, the nature of knowledge maintained by each and their role and requirement in the teaching learning process will be explained in later
sub-section. The purpose here is to give a feel of the overall flow of system and the broad role of the components.

PM selects randomly a sentence structure for simple sentence from pedagogic KB. As per the subject type given in sentence structure, PM extracts the subject from domain KB. Then, PM looks for the next constituent of the sentence, i.e., adverb of time. It sends a request to domain KB, which picks up randomly one of the action of time and returns it to PM. The third constituent of the sentence is object. As per the type of object mentioned in sentence structure, PM extracts an object from domain KB. The last constituent of sentence is the verb. To get the verb form, verb category, dhaatoo type, and tense mentioned (if any) in the sentence structure are used. As per the dhaatoo type, PM extracts dhaatoo from domain KB. The verb inflects as per the G-N-P of controlling word. The verb category and tense decides whether the controlling word is subject or object. To get the G-N-P details of controlling word, PM again looks into the domain KB. Using G-N-P details of controlling word and selected tense, it searches for a verb inflection rule in domain KB. After getting a rule, it extracts the ending and adds it in dhaatoo, which results in desired verb form. Now using all these four constituents, it compiles the full sentence and displays it on the GUI for learner.

To understand the concerns and role of these components in language TLP, we discuss in brief each component in subsequent subsections.

3.3.1 The Pedagogic Manager

The goal of Pedagogic Manager (PM) is to promote student’s learning outcomes by balancing the goal of instruction with the goal of maintaining student engagement and motivation. To balance these two goals, the PM performs following jobs: (1) lesson planning, (2) incorporating teaching strategy, (3) providing appropriate feedback, and (4) assessment. The task performed in (1) is sequencing and selecting an appropriate pedagogic element of the language to be taught. The task involved in (2) is also selection, e.g., how a teaching strategy is selected based on learner’s current knowledge state and the learning goal. Teaching strategies basically guide how to present the teaching material to the learner. These strategies are based on the topics to be taken up next, the content of the lesson being presented, the problems to be solved,
and the relevant pedagogical rules. PM does all these jobs to provide the various functionalities discussed in section 3.2.

3.3.2 Domain KB

As mentioned earlier, for language teaching we are following non-grammar based approach. In this approach, learner is not exposed to grammar rules directly. This does not mean that system does not require knowledge of grammar rules. The important requirement of sentence generation process is to generate only syntactically correct as well as meaningful sentences. To do this, system needs full knowledge of Marathi grammar and associated semantics. In our model, we will be storing the grammar rules in syntactic part of domain KB where as semantic aspects of the language are stored in semantic part of domain KB.

To generate, say a simple sentence, system needs subject, object and verb. The verb inflects as per the G-N-P of subject (or object, for some types of sentences). So, to generate correct form of the verb, system needs dhaatoo (the root form of the verb) along with G-N-P of the controlling word (dhaaturoopeshai) and tense of the sentence. Inflection is usually addition of something, i.e., ending, called as ‘pratyaya’ in Marathi. Hence, the verb inflection rules are structured in the following format: ‘rule_id:person:gender:number:tense:ending’.

For example, consider the intransitive type of dhaatoo say ‘basa’ and tense as ‘past’. In this case, verb inflects as per the G-N-P of subject. Let the subject be ‘to’ (i.e., he). Now, to generate the appropriate form of verb for dhaatoo ‘basa’ as per subject ‘to’, system needs the G-N-P details of subject. Here, the G-N-P of ‘to’ is Masculine-Singular-Third. If you see the structure of verb inflection rule, the subject is not explicitly specified; only the G-N-P properties are mentioned. We need to infer G-N-P from the subject for all possible subject words. For this, another table is maintained, where words and their G-N-P details are stored as domain knowledge in the format: ‘word:person:gender:number’.

Now, if a subject is given, then one can get its G-N-P details from this domain KB. Using these G-N-P details and tense, an appropriate rule for verb inflection can be identified. For the subject ‘to’, the selected rule is ‘38:Third:M:Ekavachana:Bhoota:laa’. This rule states that if the subject is of third
person, masculine in gender, and singular in number then add ending ‘lāa’ in dhaatoo to get the appropriate form of verb in past tense. Addition of ending ‘lāa’ in dhaatoo ‘basa’, results in the verb form ‘basalaa’.

Also, given a form of verb and G-N details of the subject, one can get the subject assuming a pronoun subject. For example, consider a verb ‘jhopalaa’ and G-N of subject as masculine and singular. Here, we need dhaatoo also. Using dhaatoo (jhopa), one can get the endings from verb. Here, it is ‘lāa’. Now, using G-N details and endings, one can get verb inflection rule from syntactic part of domain KB. Here, the rule is: ‘38:Third:M:Ekavachana:Bhoota:lāa’. The second component in the rule signifies that the subject is of third person. Now, using G-N-P details, one can get the subject from domain KB and it is ‘to’.

If the discussion in this subsection is observed, we can see that the grammar rules are stored explicitly in external repository, i.e., Domain KB as far as possible. The reason is that the feedback and dialogue mechanism requires these rules when student makes mistake or needs help. If these rules are hard coded, i.e., embedded in the code itself, then one needs to scan the whole code at run time, to identify and explain the mistakes, which is nearly impossible. Explanation in terms of program code won’t make sense to learners! Also, addition and modification of existing rules will be difficult if they are scattered in the actual program code. Hence, they are stored in explicit form in domain KB. In addition to these tables, the syntactic KB will contain grammar rules including verb inflection rules, noun number inflection rules, saamaanyaroopa rules, noun case inflection rules, etc, all clearly separated from program code, for the same reason.

We saw how grammar rules and domain knowledge are stored in syntactic KB and how it is used in generation of sentences as per the grammar of the language. Now, let us see the role of semantic KB in generation of meaningful sentences.

Consider the set of actions say eat, drink, swim, etc. To perform these actions, you need a legitimate set of subjects and objects. For example, for eating, you need

---

3 Here, multiple combination of G-N-P may give the same subject for same tense of the sentence. As we are providing the gender and number only, to overcome this issue we show all the subjects (answers). If the learner’s answer matches with one of the correct answer (subject) then system also shows other correct ones.
eatables, for drinking, you need drinkables, etc. If this information is not provided to sentence generation process then it results in generation of garbage sentences. E.g., ‘I eat table’. ‘Mohan drinks paper’, etc. To avoid generation of such meaningless sentences, all subjects and objects are stored along with some semantic characteristics. These details are stored in the format ‘noun_type:noun_name’. E.g. ‘personified: raama’, ‘eatable:aaMbaa’, ‘drinkable:chahaa’, ‘swimmable:nadee’, etc. Capturing full semantic properties of words is a massive task and involves addressing a number of theoretical analyses and modelling, as can be seen in the case of the CYC project [Lenat, 1995]. We do not need such a clear and elaborate model, given our limited concern of avoiding silly sentences. Here we adopted a simple mechanism.

This categorisation of objects is helpful in retrieving and using the correct noun for different verbs for generation of meaningful sentences. E.g. ‘raama aaMbaa khaato’ (‘ram eats mango’). When verb is eatable, a noun from the class eatable is selected. Note that for this, somewhere we need to specify that nouns to be used with the verb ‘eat’ must be of ‘eatable’ type. We specified it in the sentence template, a structure proposed for representing Marathi sentence. In sentence template, we mark the verb as ‘eat’ and at the same time we define object type as ‘eatable’. We discuss various aspects related to this template in next subsection.

Similarly, the semantic KB will contain information on nouns, pronouns, dhaatoo, postpositions, adverb of time, adjectives, tense, quantifiers, person, number, gender, conjunction, etc.

### 3.3.3 Pedagogic KB

Sentence generation and error checking demand attention on two aspects. The system’s domain model should understand the sentence structure and also the process of synthesising and analyzing inflection. The second aspect, analyzing inflection, is covered in subsection 3.2.3. We now discuss a representation of sentence structure, keeping all these concerns in mind and the issue of ensuring that silly sentences are not generated. This is an important part of our research work and is central to our teaching learning process.

The most common structure of Marathi sentence is subject + object + verb. We use an ad-hoc template to represent this structure of Marathi sentence. The reasons leading to
the formulation of the template structure are as follows, beginning with the concerns expressed earlier in this chapter.

Ideally, one can generate all types of sentences using just the grammar rules. But, there is no guarantee that these generated sentences are meaningful. Because, if the verb is ‘drink’, object from ‘drinkable’ class and subject from ‘animal’ class are required. Similarly, if the verb is ‘swim’, then, object must be from ‘swimmable’ class and subject from ‘personified’ class are required. From this, we can say that, certain activities can be performed with particular set of subjects and objects only. Also all verbs do not support all kinds of adverbs and a particular adverb is used to modify the meaning of a specific verb. Similar kind of dependency is also seen between verb and other parts of speech. So, we can say that mostly the verb decides what other constituents (parts of speech) go along with it in the sentence.

So, we propose a template structure in which the verb is usually hard coded i.e. fixed. Since the verb is known, we can decide what type of subject and object should be included in the template structure. For example, if the verb is ‘eat’, then ‘eatable’ type of objects and ‘animal’ type of subjects are legal and can be specified in template structure. In this way, template structure provides useful direction in ensuring that the generated sentences are largely meaningful.

Our approach also demands the generation of many variants of a given sentence. To generate multiple sentences in similar context with tightly controlled changes, template structure is the best option. These template structures are easy to build and extend. As per the verb, one can add one or more constituent (adjective, adverb, etc.) and get a new template structure which can be used for generation of similar sentences. Template structure can be reused i.e. one can use it as one of the components in the design of template structure for other type of sentences – compound, interrogative, etc.

Having looked at the rationale for using template structure for representation of Marathi sentences, we discuss formally the template structure, starting with the template structure for simple sentence; an example is shown in Figure 3. We mentioned previously about hard coding of verb in the template. If you look at the template structure in Figure 3, dhaatoo, i.e., the root form of verb is marked as ‘eat’. It
means that using this template structure one can generate sentences about the eating activity. But, for eating, you need eatables as objects and this activity is performed by animal subjects only.

![Figure 3: Template Structure of Simple Sentence]

To ensure that the subject must be animal and object must be eatable when the verb is ‘eat’, we introduce a ‘type’ attribute for noun subject as well as noun object. In case of noun subject, the type has value ‘personified’ and for noun object, it is ‘eatable’. Thus only nouns with this property qualify to occupy this slot. This avoids generation of garbage sentence like ‘khurchee aM baa khaate’ (chair eats a mango), ‘Mohan khurchee khaato’ (mohan eats a chair), etc.

Another important aspect in the sentence generation is the verb formation. The verb is inflected as per the G-N-P of subject or object. So, the sentence generation process requires information about verb dependency, i.e., whether verb formation depends on subject or object. All intransitive verbs inflect as per the G-N-P of subject in all tenses. Transitive verbs inflect as per the G-N-P of subject, when the tense is ‘present’ or ‘future’. If the tense is ‘past’, then transitive verb inflects as per the G-N-P of object.

To indicate the verb formation dependency, we defined ‘category’ attribute for verb. This attribute indicates if the verb is Transitive (T) or Intransitive (I). This template shows that the verb ‘eat’ is having category ‘T’, i.e., it is of transitive type. To generate verb form correctly, we also need the tense of the sentence. This is indicated by the ‘tense’ attribute of verb. Here, ‘tense’ is having value ‘any’, indicating that the template is useful for any tense. If the selected tense is ‘past’, then the verb ‘eat’ inflects as per the G-N-P of noun object. In this way, through template structure we ensure adequate information, so, that the generated sentences are syntactically correct as well as meaningful also. In the next chapter, we explain how this information is
used in the TLP, and also details of the template structure for different types of sentences.

3.4 Summary

As the discussion shows the key elements of our approach are repetition, right time intervention and appropriate feedback. The pedagogic model is built keeping in mind these three elements. Sentences are the core element in a language, and sentence teaching is a core part of our system amongst the various functionalities supported by it.

For teaching sentences, it was decided to show selected variants of sentences, so that the learner can see the structural and inflectional role played by the each constituent of the sentence. We also came up with two scaffolds for strengthening the TLP, by ensuring a manageable but adequately powerful assessment and feedback mechanism. To implement our approach, we found that ITS kind of framework is more suitable so we devised our system’s architecture on the same line. Pedagogic Manager implements all functionalities in coordination with Domain KB and Pedagogic KB. The pedagogic KB contains the templates of various types of Marathi sentences. It also defines the order of teaching a particular lesson, and hence the name. Domain KB contains vital knowledge of Marathi language which is required in generation of correct (syntactically and semantically) sentences. Domain KB is to be used for providing feedback whenever students make mistake. An externalized representation of domain knowledge is used to facilitate this requirement.

In this way, adequate scaffolding, right time intervention and incremental feedback help the students in learning. The implementation of these elements will be discussed in the next chapter.