Generating Grammar Exercises

Laura Perez-Beltrachini
Université de Lorraine
LORIA, UMR 7503
Vandoeuvre-lès-Nancy
F-54500, France
laura.perez@loria.fr

Claire Gardent
CNRS, LORIA, UMR 7503
Vandoeuvre-lès-Nancy
F-54500, France
claire.gardent@loria.fr

German Kruszewski
Inria, LORIA, UMR 7503
Villers-lès-Nancy
F-54600, France
german.kruszewski@inria.fr

Abstract

Grammar exercices for language learning fall into two distinct classes: those that are based on "real life sentences" extracted from existing documents or from the web; and those that seek to facilitate language acquisition by presenting the learner with exercices whose syntax is as simple as possible and whose vocabulary is restricted to that contained in the textbook being used. In this paper, we introduce a framework (called *GramEx*) which permits generating the second type of grammar exercises. Using generation techniques, we show that a grammar can be used to semi-automatically generate grammar exercices which target a specific learning goal; are made of short, simple sentences; and whose vocabulary is restricted to that used in a given textbook. We evaluate the approach on several dimensions using quantitative and qualitative metrics as well as a small scale user-based evaluation. And we show that the *GramEx* framework permits producing exercises for a given pedagogical goal that are linguistically and pedagogically varied.

1 Introduction

Textbooks for language learning generally include grammar exercises. *Tex's French Grammar* [1] for instance, includes at the end of each lecture, a set of grammar exercises which target a specific pedagogical goal such as *learning the plural form of nouns* or *learning the placement of adjectives*. Figure 1 shows the exercises provided by this book at the end of the lecture on the plural formation of nouns. As exemplified in this figure, these exercices markedly differ from more advanced learning activities which seek to familiarise the learner with "real world sentences". To support *in situ* learning, this latter type of activity presents the learner with sentences drawn from the Web or from existing documents thereby exposing her to a potentially complex syntax and to a diverse vocabulary. In contrast, textbook grammar exercices usually aim to facilitate the acquisition of a specific grammar point by presenting the learner with exercices made up of short sentences involving a restricted vocabulary. In other words, while certain learning activities target the *in situ* acquisition of a specific linguistic point, many textbooks additionally include grammar exercises that are made up of short sentences whose vocabulary is restricted to the vocabulary taught so far by this textbook; and whose syntax is as simple as possible. In this way, the learner can focus on the specific construct being taught without being distracted by unknown words or complex constructions.

As shall be discussed in the next section, most existing work on the generation of grammar exercices has concentrated on the automatic creation of the first type of exercices i.e., exercices whose source sentences are extracted from an existing corpus. In

[1] *Tex's French Grammar* http://www.laits.utexas.edu/tex/ is an online pedagogical reference grammar that combines explanations with surreal dialogues and cartoon images. Tex's French Grammar is arranged like many other traditional reference grammars with the parts of speech (nouns, verbs, etc.) used to categorize specific grammar items (gender of nouns, irregular verbs). Individual grammar items are carefully explained in English, then exemplified in a dialogue, and finally tested in self-correcting, fill-in-the-blank exercises.

```
Give the plural form of the noun indicated  in parentheses. Pay attention to
both the article and the noun.
1.      Bette aime _____ . (le bijoux)
2.      Fiona aime ______ . (le cheval)
3.      Joe–Bob aime ______ américaines. (la bière)
4.      Tex n'aime pas ______ . (le choix)
5.      Joe–Bob n'aime pas ______ difficiles. (le cours)
6.      Tammy n'aime pas ______ . (l'hôpital)
7.      Eduard aime ______. (le tableau)
8.      Bette aime ______ de Tex. (l'oeil)
9.      Tex aime ______ français. (le poëte)
10.     Corey aime ______ fraîches. (la boisson)
11.     Tammy aime ______ américains. (le campus)
12.     Corey n'aime pas ______ . (l'examen)
```

Figure 1: Grammar exercises from the *Tex's French Grammar* Textbook

this paper, we present a framework (called *GramEx*) which addresses the generation of the second type of grammar exercices used for language learning i.e., grammar exercises whose syntax and lexicon are strongly controlled. Our approach uses generation techniques to produce these exercises from an existing grammar describing both the syntax and the semantics of natural language sentences. Given a pedagogical goal for which exercises must be produced, the *GramEx* framework permits producing *Fill in the blank* (FIB, the learner must fill a blank with an appropriate form or phrase) and *Shuffle* (given a set of lemmas or forms, the learner must use these to produce a phrase) exercises that target that specific goal and use a simple syntax and vocabulary similar to that used in the *Tex's French Grammar* textbook.

We evaluate the approach on several dimensions using quantitative and qualitative metrics as well as a small scale user-based evaluation. And we show that the *GramEx* framework permits producing exercises for a given pedagogical goal that are linguistically and pedagogically varied.

The paper is structured as follows. We start by discussing related work (Section 2). In Section 3, we present the framework we developed to generate grammar exercises. Section 4 describes the experimental setup we used to generate exercise items. Section 5 reports on an evaluation of the exercise items produced and on the results obtained. Section 6 concludes.

2 Related Work

A prominent strand of research in Computer Aided Language Learning (CALL) addresses the automation of exercise specifications relying on Natural Language Processing (NLP) techniques.

Much of this work targets the automatic generation of so-called objective test items (i.e., test items whose answer is strongly constrained and can therefore be predicted and checked with high accuracy) and of their solutions. These include multiple choice questions, fill in the blank and cloze exercise items (Mitkov et al., 2006; Heilman and Eskenazi, 2007; Karamanis et al., 2006; Chao-Lin et al., 2005; Coniam, 1997; Sumita et al., 2005; Simon Smith, 2010; Lin et al., 2007; Lee and Seneff, 2007). Mostly, these approaches use large corpora and machine learning techniques for finding stems, keys and distractors i.e., exercise sentences, correct and incorrect answers respectively.

In many of these approaches, the exercise items produced address vocabulary learning, mostly focusing on open-class words (except (Lee and Seneff, 2007) who worked on prepositions) and ignoring grammatical learning.

Nonetheless, there are also proposals for semi-automatically producing grammatical exercices. Thus, (Chen et al., 2006) describes a system called FAST which supports the semi-automatic generation of multiple-choice and error detection exercises while (Aldabe et al., 2006) presents the ArikiTurri automatic question generator for constructing Fill-in-the-Blank, Word Formation, Multiple Choice and

Error Detection exercices. These approaches are similar to the approach we propose. First, a bank of sentences is built which are automatically annotated with syntactic and morpho-syntactic information. Second, sentences are retrieved from this bank based on their annotation and on the linguistic phenomena the exercise is meant to illustrate. Third, the exercise question is constructed from the retrieved sentences.

There are important differences however.

First, in these approaches, the source sentences used for building the test items are selected from corpora and can therefore be very complex so that most of the generated test items are targeted for intermediate or advanced learners. In contrast, we use generation to produce these sentences. In that way, we can control both the syntax and the vocabulary present in the exercices produced. Indeed, (Aldabe et al., 2006) found that some of the linguistic phenomena included in the language schools curricula were not, or insufficiently, present in their source corpus. By using generation and a large scale grammar, we can ensure that the desired linguistic coverage is present in the sentence bank.

Second, while, in these approaches, the syntactic and morpho-syntactic annotations associated with the bank sentences are obtained using part-of-speech tagging and chunking, in our approach, these are obtained by a grammar-based generation process. As we shall see below, the information thus associated with sentences is richer than that obtained by chunking. In particular, it contains detailed linguistic information about the syntactic constructs (e.g., cleft subject) contained in the bank sentences. This permits a larger coverage of the linguistic phenomena that can be handled. For instance, we can retrieve sentences which contain a relativised cleft object (e.g., *This is the man whom Mary likes who sleeps*) by simply stipulating that the retrieved sentences must be associated with the information *Cleft Object*).

In the area of Natural Language Generation, (Harbusch et al., 2007; Karin Harbusch and Kühner, 2007) presents a tool for "sentence combining" exercises which make use of generation techniques both to guide the student during essay writing and to provide her with feedback based on the generator expectations while (Zamorano-Mansilla, 2004) describes an approach, also based on an NLG system, which aims at detecting the source of learner errors and providing her with relevant feedback. Neither of these approaches support the generation of grammar exercices.

To sum up, our approach differs from most existing work on the automatic generation of grammar exercices in that it targets the production of syntactically and lexically controlled grammar exercices. Because we use a grammar and a lexicon to generate the exercices, we can control both the syntactic and the lexical coverage of the exercices produced. In other words, while previous work has focused on producing grammar exercices based on sentences extracted from an existing corpus, our approach permits generating textbook style grammar exercices of the type illustrated in Figure 1.

3 Generating Exercises

Given a pedagogical goal (e.g., learning adjective morphology), *GramEx* produces a set of exercise items for practicing that goal. The item can be either a FIB or a shuffle item; and *GramEx* produces both the exercise question and the expected solution.

To generate exercise items, *GramEx* proceeds in three main steps as follows. First, a generation bank is constructed using surface realisation techniques. This generation bank stores sentences that have been generated together with the detailed linguistic information associated by the generation algorithm with each of these sentences. Next, sentences that permit exercising the given pedagogical goal are retrieved from the generation bank using a constraint language that permits defining pedagogical goals in terms of the linguistic properties associated by the generator with the generated sentences. Finally, exercises are constructed from the retrieved sentences using each retrieved sentence to define FIB and Shuffle exercises; and the sentence itself as the solution to the exercise.

We now discuss each of these steps in more detail.

3.1 Constructing a Generation bank

The generation bank is a database associating sentences with a representation of their semantic content and a detailed description of their syntactic and morphosyntactic properties. In other words, a gen-

Sentence realisation:
"Tammy a une voix douce"

Lemma-features pairs:
{"lemma": "Tammy",
"lemma-features": {anim:+,num:sg,det: +,wh:-,cat:n,
func:suj,xp: +, gen:f},
"trace": {propername}},
{"lemma": "avoir",
"lemma-features": {aux-refl:-,inv:-,cat:v,pers:3,pron:-,
num:sg,mode:ind, aspect:indet,tense:pres,stemchange:-,
flexion:irreg},
"trace": {CanonicalObject,CanonicalSubject,n0Vn1}},
{"lemma": "un",
"lemma-features": {wh:-,num:sg,mass:-,cat:d,
gen:f,def:+},
"trace": {determiner}},
{"lemma": "voix",
"lemma-features": {bar:0,wh:-,cat:n,num:sg,
mass:-,gen:f,flexion:irreg,
"trace": {noun}},
{"lemma": "doux",
"lemma-features": {num:sg,gen:f,flexion:irreg,cat:adj},
"trace": {Epith,EpithPost}}

Figure 2: Morphosyntactic information associated by *GraDe* with the sentence *Tammy a un voix douce*

eration bank is a set of (S_i, L_i, σ_i) tuples where S_i is a sentence, L_i is a set of linguistic properties true of that sentence and σ_i is its semantic representation.

To produce these tuples, we use the *GraDe* grammar traversal algorithm described in (Gardent and Kruszewski, 2012). Given a grammar and a set of user-defined constraints, this algorithm generates sentences licensed by this grammar. The user-defined constraints are either parameters designed to constrain the search space and guarantee termination (e.g., upper-bound on the number and type of recursive rules used or upper-bound on the depth of the tree build by *GraDe*); or linguistic parameters which permit constraining the output (e.g., by specifying a core semantics the output must verbalise or by requiring the main verb to be of a certain type). Here we use *GraDe* both to generate from manually specified semantic input; and from a grammar (in this case an existing grammar is used and no manual input need be specified). As explained in (Gardent and Kruszewski, 2012), when generating from a semantic representation, the output sentences are constrained to verbalise that semantics but the input semantics may be underspecified thereby allowing for morpho-syntactic, syntactic and temporal variants to be produced from a single semantics. For instance, given the input semantics *L1:named(J bette_n) A:le_d(C RH SH) B:bijou_n(C) G:aimer_v(E J C)*, *GraDe* will output among others the following variants:

> Bette aime le bijou (*Bette likes the jewel*), Bette aime les bijoux (*Bette likes the jewels*), C'est Bette qui aime le bijou (*It is Bette who likes the jewel*), C'est Bette qui aime les bijoux (*It is Bette who likes the jewel*), Bette aimait le bijou (*Bette liked the jewel*), Bette aimait les bijoux (*Bette liked the jewels*), ...

When generating from the grammar, the output is even less constrained since all derivations compatible with the user-defined constraints will be produced irrespective of semantic content. For instance, when setting *GraDe* with constraints restricting the grammar traversal to only derive basic clauses containing an intransitive verb, the output sentences include among others the following sentences:

> Elle chante (*She sings*), La tatou chante-t'elle? (*Does the armadillo sing?*), La tatou chante (*The armadillo sings*), Chacun chante -t'il (*Does everyone sing?*), Chacun chante (*Everyone sings*), Quand chante chacun? (*When does everyone sing?*), Quand chante la tatou? (*When does the armadillo sing?*) Quand chante quel tatou? (*When does which armadillo sing?*), Quand chante Tammy? (*When does Tammy sing?*), Chante-t'elle? (*When does she sing?*) Chante -t'il? (*Does he sing?*), Chante! (*Sing!*), Quel tatou chante ? (*Which armadillo sings?*), Tammy chante-t'elle? (*Does Tammy sing?*), Tammy chante (*Tammy sings*), une tatou qui chante chante (*An armadillo which sings sings*), C'est une tatou qui chante (*It is an armadillo which sings*), ...

Figure 2 shows the linguistic properties associated by *GraDe* with the sentence *Tammy a une voix douce* (Tammy has a soft voice) by *GraDe* . To generate exercises, *GramEx* makes use of the morpho-syntactic information associated with each lemma i.e., the feature-value pairs occurring as values of the lemma-features fields; and of their linguistic properties i.e., the items occurring as values of the trace fields.

3.2 Retrieving Appropriate Sentences

To enable the retrieval of sentences that are appropriate for a given pedagogical goal, we define a query language on the linguistic properties assigned by *GraDe* to sentences. We then express each pedagogical goal as a query in that language; and we use these queries to retrieve from the generation bank appropriate source sentences. For instance, to retrieve a sentence for building a FIB exercise where the blank is a relative pronoun, we query the generation bank with the constraint $RelativePronoun$. This will return all sentences in the generation bank whose trace field contains the $RelativePronoun$ item i.e., all sentences containing a relative pronoun. We then use this sentence to build both the exercise question and its solution.

3.2.1 GramEx Query Language

We now define the query language used to retrieve sentences that are appropriate to build an exercise for a given pedagogical goal. Let B be a generation bank and let (S_i, L_i, σ_i) be the tuples stored in B. Then, a *GramEx* query q permits retrieving from B the set of sentences $S_i \in (S_i, L_i, \sigma_i)$ such that L_i satisfies q. In other words, *GramEx* queries permit retrieving from the generation bank all sentences whose linguistic properties satisfy those queries.

The syntax of the *GramEx* query language is as follows:

BoolExpr → BoolTerm
BoolTerm → BoolFactor | BoolTerm ∨ BoolFactor
BoolFactor → BoolUnary | BoolFactor ∧ BoolUnary
BoolUnary → BoolPrimary | ¬ BoolPrimary
BoolPrimary → PrimitiveCond | (BoolExpr) | [BoolExpr]
PrimitiveCond → $traceItem$ |
$feature = value$

In words: the *GramEx* query language permits defining queries that are arbitrary boolean constraints on the linguistic properties associated by *GraDe* with each generated sentence. In addition, complex constraints can be named and reused (macros); and expressions can be required to hold on a single lexical item ([BoolExpr] indicates that BoolExpr should be satisfied by the linguistic properties of a single lexical item).

Grammatical Properties (traceItem)	
Argument Realisation	Cleft, CleftSUbj, CleftOBJ, ..., InvertedSubj Questioned, QuSubj, ... Relativised, RelSubj ... Pronominalised, ProSubj, ...
Voice	Active, Passive, Reflexive
Aux	tse, modal, causal
Adjective	Predicative,Pre/Post nominal
Adverb	Sentential, Verbal
Morpho-Syntactic Properties (feature=value)	
Tense	present,future,past
Number	mass, count, plural, singular
Inflexion	reg,irreg

Table 1: Some grammatical and morpho-syntactic properties that can be used to specify pedagogical goals.

The signature of the language is the set of grammatical ($traceItem$) and morpho-syntactic properties ($feature = value$) associated by *GraDe* with each generated sentence where $traceItem$ is any item occurring in the value of a `trace` field and $feature = value$ any feature/value pair occurring in the value of a `lemma-features` field (cf. Figure 2). The Table below (Table 1) shows some of the constraints that can be used to express pedagogical goals in the *GramEx* query language.

3.2.2 Query Examples

The *GramEx* query language allows for very specific constraints to be expressed thereby providing fine-grained control over the type of sentences and therefore over the types of exercises that can be produced. The following example queries illustrate this.

(1) a. EpithAnte
Tex pense que Tammy est une jolie tatou (Tex thinks that Tammy is a pretty armadillo)

b. [Epith ∧ flexion: irreg]
Tex et Tammy ont une voix douce (Tex and Tammy have a soft voice)

c. POBJinf ∧ CLAUSE
POBJinf ≡ (DE-OBJinf ∨ A-OBJinf)
CLAUSE ≡ Vfin∧¬Mod ∧ ¬CCoord∧ ¬Sub
Tammy refuse de chanter (Tammy refuses to sing)

Query (1a) shows a query for retrieving sentences containing prenominal adjectives which uses the grammatical (*traceItem*) property EpithAnte associated with preposed adjectives.

In contrast, Query (1b) uses both grammatical and morpho-syntactic properties to retrieve sentences containing a postnominal adjective with irregular flexion. The square brackets in the query force the conjunctive constraint to be satisfied by a single lexical unit. That is, the query will be satisfied by sentences containing a lexical item that is both a postnominal adjective and has irregular flexion. This excludes sentences including e.g., a postnominal adjective and a verb with irregular flexion.

Finally, Query (1c) shows a more complex case where the pedagogical goal is defined in terms of predefined macros themselves defined as *GramEx* query expressions. The pedagogical goal is defined as a query which retrieves basic clauses (CLAUSE) containing a prepositional infinitival object (POBJinf). A sentence containing a prepositional infinitival object is in turn defined (second line) as a prepositional object introduced either by the *de* or the *à* preposition. And a basic clause (3rd line) is defined as a sentence containing a finite verb and excluding modifiers, clausal or verb phrase coordination (CCORD) and subordinated clauses[2]

3.3 Building Exercise Items

In the previous section, we saw the mechanism used for selecting an appropriate sentence for a given pedagogical goal. *GramEx* uses such selected sentences as source or stem sentences to build exercise items. The exercise *question* is automatically generated from the selected sentence based on its associated linguistic properties. Currently, *GramEx* includes two main types of exercises namely, Fill in the blank and Shuffle exercises.

FIB questions. FIB questions are built by removing a word from the target sentence and replacing it with either: a blank (FIBBLNK), a lemma (FIBLEM) or a set of features used to help the learner guess the solution (FIBHINT). For instance, in an exercise on pronouns, *GramEx* will use the gender, number and person features associated with the pronoun by the generation process and display them to specify which pronominal form the learner is expected to provide. The syntactic representation (cf. 2) associated by *GraDe* with the sentence is used to search for the appropriate key word to be removed. For instance, if the pedagogical goal is *Learn Subject Pronouns* and the sentence retrieved from the generation bank is that given in (2a), *GramEx* will produce the FIBHINT question in (2b) by searching for a lemma with category *cl* (clitic) and feature *func=subj* and using its gender value to provide the learner with a hint constraining the set of possible solutions.

(2) a. Elle adore les petits tatous
(*She loves small armadillos*)
b. ... adore les petits tatous (gender=fem)

Shuffle questions. Similarly to FIB questions, shuffle exercise items are produced by inspecting and using the target derivational information. More specifically, lemmas are retrieved from the list of lemma-feature pairs. Function words are (optionally) deleted. And the remaining lemmas are "shuffled" (MSHUF). For instance, given the source sentence (2a), the MSHUF question (2b) can be produced.

(3) a. Tammy adore la petite tatou
a. tatou / adorer / petit / Tammy

Note that in this case, there are several possible solutions depending on which tense and number is used by the learner. For such cases, we can either use hints as shown above to reduce the set of possible solutions to one; or compare the learner's answer to the set of output produced by *GraDe* for the semantics the sentence was produced from.

4 Experimental Setup

We carried out an experiment designed to assess the exercises produced by *GramEx* . In what follows, we describe the parameters of this experiment namely, the grammar and lexicons used; the input and the user-defined parameters constraining sentence generation; and the pedagogical goals being tested.

4.1 Grammar and Lexicon

The grammar used is a Feature-Based Lexicalised Tree Adjoining Grammar for French augmented with a unification-based compositional semantics. This grammar contains around 1300 elementary trees and covers auxiliaries, copula, raising and

[2]The expressions CCoord and Sub are themselves defined rather than primitive expressions.

small clause constructions, relative clauses, infinitives, gerunds, passives, adjuncts, wh-clefts, PRO constructions, imperatives and 15 distinct subcategorisation frames.

The syntactic and morpho-syntactic lexicons used for generating were derived from various existing lexicons, converted to fit the format expected by *GraDe* and tailored to cover basic vocabulary as defined by the lexicon used in *Tex's French Grammar* . The syntactic lexicon contains 690 lemmas and the morphological lexicon 5294 forms.

4.2 Pedagogical Goals

We evaluate the approach on 16 pedagogical goals taken from the *Tex's French Grammar* book. For each of these goals, we define the corresponding linguistic characterization in the form of a *GramEx* query. We then evaluate the exercises produced by the system for each of these queries. The pedagogical goals tested are the following (we indicate in brackets the types of learning activity produced for each teaching goal by the system):

- Adjectives: Adjective Order (MSHUF), Adjective Agreement (FIBLEM), Prenominal adjectives (FIBLEM), Present and Past Participial used as adjectives (FIBLEM), Regular and Irregular Inflexion (FIBLEM), Predicative adjectives (MSHUF)
- Prepositions: Prepositional Infinitival Object (FIBBLNK), Modifier and Complement Prepositional Phrases (FIBBLNK)
- Noun: Gender (FIBLEM), Plural form (FIBLEM), Subject Pronoun (FIBHINT).
- Verbs: Pronominals (FIBLEM), -ir Verbs in the present tense (FIBLEM), Simple past (FIBLEM), Simple future (FIBLEM), Subjunctive Mode (FIBLEM).

4.3 GraDe's Input and User-Defined Parameters

GraDe's configuration As mentioned in Section 3, we run *GraDe* using two main configurations. In the first configuration, *GraDe* search is constrained by an input core semantics which guides the grammar traversal and forces the output sentence to verbalise this core semantics. In this configuration, *GraDe* will only produce the temporal variations supported by the lexicon (the generated sentences may be in any simple tense i.e., present, future, simple past and imperfect) and the syntactic variations supported by the grammar for the same MRSs (e.g., active/passive voice alternation and cleft arguments).

Greater productivity (i.e., a larger output/input ratio) can be achieved by providing *GraDe* with less constrained input. Thus, in the second configuration, we run *GraDe* not on core semantics but on the full grammar. To constrain the search, we specify a root constraint which requires that the main verb of all output sentences is an intransitive verb. We also set the constraints on recursive rules so as to exclude the inclusion of modifiers. In sum, we ask *GraDe* to produce all clauses (i) licensed by the grammar and the lexicon; (ii) whose verb is intransitive; and (iii) which do not include modifiers. Since the number of sentences that can be produced under this configuration is very large, we restrict the experiment by using a lexicon containing a single intransitive verb (*chanter/To sing*), a single common noun and a single proper name. In this way, syntactically structurally equivalent but lexically distinct variants are excluded.

Input Semantics We use two different sets of input semantics for the semantically guided configuration: one designed to test the pedagogical coverage of the system (Given a set of pedagogical goals, can *GramEx* generate exercises that appropriately targets that goal?); and the other to illustrate linguistic coverage (How much syntactic variety can the system provide for a given pedagogical goal?).

The first set (D1) of semantic representations contains 9 items representing the meaning of example sentences taken from the *Tex's French Grammar* textbook. For instance, for the first item in Figure 1, we use the semantic representation *L1:named(J bette_n) A:le_d(C RH SH) B:bijou_n(C) G:aimer_v(E J C)*. With this first set of input semantics, we test whether *GramEx* correctly produces the exercises proposed in the *Tex's French Grammar* book. Each of the 10 input semantics corresponds to a distinct pedagogical goal.

The second set (D2) of semantic representations contains 22 semantics, each of them illustrating distinct syntactic configurations namely, intransitive, transitive and ditransitive verbs; raising and control; prepositional complements and modifiers; sentential and prepositional subject and object complements; pronominal verbs; predicative, attributive and participial adjectives. With this set of semantics, we introduce linguistically distinct material thereby increasing the variability of the exercises i.e., making it possible to have several distinct syntactic configurations for the same pedagogical goal.

5 Evaluation, Results and Discussion

Using the experimental setup described in the previous section, we evaluate *GramEx* on the following points:

- Correctness: Are the exercises produced by the generator grammatical, meaningful and appropriate for the pedagogical goal they are associated with?
- Variability: Are the exercises produced linguistically varied and extensive? That is, do the exercises for a given pedagogical goal instantiate a large number of distinct syntactic patterns?
- Productivity: How much does *GramEx* support the production, from a restricted number of semantic input, of a large number of exercises?

Correctness To assess correctness, we randomly selected 10 (pedagogical goal, exercise) pairs for each pedagogical goal in Section 4.2 and asked two evaluators to say for each pair whether the exercise text and solutions were grammatical, meaningful (i.e., semantically correct) and whether the exercise was adequate for the pedagogical goal. The results are shown in Table 2 and show that the system although not perfect is reliable. Most sources of grammatical errors are cases where a missing word in the lexicon fails to be inflected by the generator. Cases where the exercise is not judged meaningful are generally cases where a given syntactic construction seems odd for a given semantics content. For instance, the sentence *C'est Bette qui aime les bijoux* (It is Bette who likes jewels) is fine but *C'est Bette qui aime des bijoux* although not ungrammatical sounds odd. Finally, cases judged inappropriate are generally due to an incorrect feature being assigned to a lemma. For instance, *avoir* (To have) is marked as an -ir verb in the lexicon which is incorrect.

Grammatical	Meaningful	Appropriate
91%	96%	92%

Table 2: Exercise Correctness tested on 10 randomly selected (pedagogical goal, exercise pairs)

We also asked a language teacher to examine 70 exercises (randomly selected in equal number across the different pedagogical goals) and give her judgment on the following three questions:

- A. Do you think that the source sentence selected for the exercise is appropriate to practice the topic of the exercise? Score from 0 to 3 according to the degree (0 inappropriate - 3 perfectly appropriate)
- B. The grammar topic at hand together with the complexity of the source sentence make the item appropriate for which language level? A1,A2,B1,B2,C1[3]
- C. Utility of the exercise item: ambiguous (not enough context information to solve it) / correct

For Question 1, the teacher graded 35 exercises as 3, 20 as 2 and 14 as 1 pointing to similar problems as was independently noted by the annotators above. For question B, she marked 29 exercises as A1/A2, 24 as A2, 14 as A2/B1 and 3 as A1 suggesting that the exercises produced are non trivial. Finally, she found that 5 out of the 70 exercises lacked context and were ambiguously phrased.

Variability For any given pedagogical goal, there usually are many syntactic patterns supporting learning. For instance, learning the gender of common

[3]A1,A2,B1,B2,C1 are reference levels stablished by the Common European Framework of Reference for Languages: Learning, Teaching, Assessment (`http://en.wikipedia.org/wiki/Common_European_Framework_of_Reference_for_Languages`) for grading an individual's language proficiency.

#SP	1	2	3	4	5	6	7	8	9	12	14	21
(S,G)	2	29	12	13	3	6	5	1	5	1	2	6

Table 3: Variability: Distribution of the number of distinct sentential patterns that can be produced for a given pedagogical goal from a given input semantics

nouns can be practiced in almost any syntactic configuration containing a common noun. We assess the variability of the exercises produced for a given pedagogical goal by computing the number of distinct morpho-syntactic configurations produced from a given input semantics for a given pedagogical goal. We count as distinct all exercise questions that are derived from the same semantics but differ either in syntax (e.g., passive/active distinction) or in morphosyntax (determiner, number, etc.). Both types of differences need to be learned and therefore producing exercises which, for a given pedagogical goal, expose the learner to different syntactic and morpho-syntactic patterns (all involving the construct to be learned) is effective in supporting learning. However we did not take into account tense differences as the impact of tense on the number of exercises produced is shown by the experiment where we generate by traversing the grammar rather than from a semantics.

Table 3 shows for each (input semantics, teaching goal) pair the number of distinct patterns observed. The number ranges from 1 to 21 distinct patterns with very few pairs (2) producing a single pattern, many (29) producing two patterns and a fair number producing either 14 or 21 patterns.

Nb. PG	1	2	3	4	5	6
Nb. sent	222	29	9	15	10	6

Table 6: Pedagogical Productivity: Number of Teaching Goals the source sentence produced from a given semantics can be used for

Productivity When used to generate from semantic representations (as in Exp-1), *GramEx* only partially automates the production of grammar exercises. Semantic representations must be manually input to the system for the exercises to be generated. Therefore the issue arises of how much *GramEx* helps automating exercise creation. Table 4 shows the breakdown of the exercises produced per teaching goal and activity type. In total, *GramEx* produced 448 exercises out of 28 core semantics exercises yielding an output/input ratio of 16 (448/28). Further, Table 5 and 6 show the distribution of the ratio between (i) the number of exercises produced and the number of input semantics and (ii) the number of teaching goals the source sentences produced from input semantics i can be used for. Table 6 (pedagogical productivity) shows that, in this first experiment, a given input semantics can provide material for exercises targeting up to 6 different pedagogical goals while Table 5 (exercise productivity) shows that most of the input semantics produce between 2 and 12 exercises.

When generating by grammar traversal, under the constraints described in Section 4, 90 exercises are generated targeting 4 different pedagogical goals. In other words, one input yields 90 exercices which can be used to illustrate four distinct linguistic phenomena.

6 Conclusion

We presented a framework (called *GramEx*) for generating grammar exercises which are similar to those often used in textbooks for second language learning. These exercises target a specific learning goal; and, they involve short sentences that make it easier for the learner to concentrate on the grammatical point to be learned.

We showed that *GramEx* permits semi-automatising the exercise generation process in that it allows for the production of many exercises out of few inputs (Productivity). We further showed that *GramEx* permits generating multiple morpho-syntactic patterns from the same input and for the same learning goal (variability). That is, *GramEx* permits exposing the learner to different sentence patterns all involving the same pedagogical goal and the same semantic content. Finally, a detailed examination of the output showed that the exercises produced are mostly correct both linguistically and pedagogically (Correctness).

One distinguishing feature of the approach is the

Pedagogical Goal	FIBLEM	FIBBLNK	Shuf	FIBHINT
Preposition	—	28	—	—
Prepositions with infinitives	—	8	—	—
Subject pronouns–il	—	—	—	3
Noun number	13	—	—	—
Noun gender	—	54	—	—
Adjective order	—	—	30	—
Adjective morphology	30	—	—	—
Adjectives that precede the noun	24	—	—	—
Attributive Adjectives	—	—	28	—
Irregular adjectives	4	—	—	—
Participles as adjectives	8	—	—	—
Simple past	78	—	—	—
Simple future	93	—	—	—
-ir verbs in present	18	—	—	—
Subjunctive mode	15	—	—	—
Pronominal verbs	14	—	—	—
Total	264	82	30	3

Table 4: Number and Types of Exercises Produced from the 28 input semantics

Nb. Ex.	1	2	3	4	5	6	12	17	18	21	23	26	28	31	37	138
Nb. Sem	1	3	1	4	1	5	4	1	1	1	1	1	1	1	1	1

Table 5: Exercise Productivity: Number of exercises produced per input semantics

rich linguistic information associated by the generator with the source sentences used to construct grammar exercices. Although space restriction prevented us from showing it here, this information includes in addition to the morphosyntactic information and the grammatical properties illustrated in Figure 2 and Table 1 respectively, a semantic representation, a derivation tree showing how the parse tree of each sentence was obtained and optionally, an underspecified semantics capturing the core predicate/argument and modifier/modifiee relationships expressed by each sentence. We are currently exploring how this information could be used to extend the approach to other types of learning activity such as in particular, transformation exercises (e.g., passive/active) where the relation between exercise question and exercise solution is more complex than in FIB exercises. In brief, the idea is to define transformation patterns on derivation trees; and to use these patterns to retrieve from the generation bank, these pairs of sentences which (i) share the same underspecified semantics and (ii) instantiate a tranformation pattern. A preliminary investigation of different types of transformations commonly used in grammar exercices (e.g., Full NP/Pronoun, Assertion/Question, Assertion/Negated Assertion) shows that the association of sentences with a semantic representation and a set of linguistic properties greatly facilitates the idenfication of sentences pairs standing in a transformation relation.

Another interesting question which needs further investigation is how to deal with exercise items that have multiple solutions such as example (3) above. Here we plan to use the fact that underspecified semantics in *GraDe* permits associating many variants with a given semantics. We are examining whether an appropriate underspecified semantics can be computed for exercise questions with multiple solutions, which in effect would generate all the possible solutions of those queries.

Finally, it would be interesting to extend the approach so as to provide each exercise with some context as is frequently done in textbooks. For instance, preceding the exercise question with an introductory sentence providing for a more natural setting and minimizing the risk of ambiguity (as noted by our teacher evaluator).

Acknowledgments

The research presented in this paper was partially supported by the European Fund for Regional Development within the framework of the INTERREG IV A Allegro Project (`http://talc.loria.fr/-ALLEGRO-Nancy-.html` and `http://www.allegro-project.eu/`).

References

Itziar Aldabe, Maddalen Lopez de Lacalle, Montse Maritxalar, Edurne Martinez, and Larraitz Uria. 2006. Arikiturri: an automatic question generator based on corpora and nlp techniques. In *Proceedings of the 8th international conference on Intelligent Tutoring Systems*, ITS'06, pages 584–594, Berlin, Heidelberg. Springer-Verlag.

Liu Chao-Lin, Wang Chun-Hung, Gao Zhao-Ming, and Huang Shang-Ming. 2005. Applications of lexical information for algorithmically composing multiple-choice cloze items. In *Proceedings of the second workshop on Building Educational Applications Using NLP*, EdAppsNLP 05, pages 1–8, Stroudsburg, PA, USA. Association for Computational Linguistics.

Chia-Yin Chen, Hsien-Chin Liou, and Jason S. Chang. 2006. Fast: an automatic generation system for grammar tests. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, COLING-ACL '06, pages 1–4, Stroudsburg, PA, USA. Association for Computational Linguistics.

David Coniam. 1997. A preliminary inquiry into using corpus word frequency data in the automatic generation of english language cloze tests. *CALICO Journal*, 14:15–33.

Claire Gardent and German Kruszewski. 2012. Generation for grammar engineering. In *11th International Conference on Natural Language Generation (ENLG)*.

Karin Harbusch, Camiel Van Breugel, Ulrich Koch, and Gerard Kempen. 2007. Interactive sentence combining and paraphrasing in support of integrated writing and grammar instruction: a new application area for natural language sentence generators. In *Proceedings of the Eleventh European Workshop on Natural Language Generation*, ENLG '07, pages 65–68.

Michael Heilman and Maxine Eskenazi. 2007. Application of automatic thesaurus extraction for computer generation of vocabulary questions. In *Proceedings of Speech and Language Technology in Education (SLaTE2007)*, pages 65–68.

Nikiforos Karamanis, Le An Ha, and Ruslan Mitkov. 2006. Generating multiple-choice test items from medical text: A pilot study. In *Proceedings of the Fourth International Natural Language Generation Conference*, pages 111–113, Sydney, Australia.

Ulrich Koch Karin Harbusch, Gergana Itsova and Christine Kühner. 2007. The Sentence Fairy: A natural-language generation system to support children's essay writing. In *Interfaces of Intelligent Computer-Assisted Language Learning*, iicall 2007.

John Lee and Stephanie Seneff. 2007. Automatic generation of cloze items for prepositions. *Proceedings of Interspeech*, pages 2173–2176.

Yi-Chien Lin, Li-Chun Sung, and Meng Chang Chen. 2007. An Automatic Multiple-Choice Question Generation Scheme for English Adjective Understandings. In *Workshop on Modeling, Management and Generation of Problems/Questions in eLearning, the 15th International Conference on Computers in Education (ICCE 2007)*, pages pages 137–142.

Ruslan Mitkov, Le An Ha, and Nikiforos Karamanis. 2006. A computer-aided environment for generating multiple-choice test items. *Natural Language Engineering*, 12(2):177–194.

Adam Kilgarriff Simon Smith, P.V.S Avinesh. 2010. Gap-fill Tests for Language Learners: Corpus-Driven Item Generation. In *Proceedings of ICON-2010: 8th International Conference on Natural Language Processing*.

Eiichiro Sumita, Fumiaki Sugaya, and Seiichi Yamamoto. 2005. Measuring non-native speakers' proficiency of english by using a test with automatically-generated fill-in-the-blank questions. In *Proceedings of the second workshop on Building Educational Applications Using NLP*, EdAppsNLP 05, pages 61–68, Stroudsburg, PA, USA. Association for Computational Linguistics.

Juan Rafael Zamorano-Mansilla. 2004. Text generators, error analysis and feedback. In *InSTIL/ICALL 2004 Symposium on Computer Assisted Learning*.