

# Tenjin: A Web based Report System for Foreign Language Learning

Kumiko TANAKA

University of Tokyo

7-3-11 Hongo, Bunkyo-ku, Tokyo, Japan

[kumiko@r.dl.itc.u-tokyo.ac.jp](mailto:kumiko@r.dl.itc.u-tokyo.ac.jp)

## Abstract

*We report on Tenjin, a web-based system supporting foreign language education. It provides an integrated environment enabling students to work on assignments and teachers to view the students' progress. Tenjin automatically evaluates all assignments submitted by students at the upload time using a dynamic programming algorithm. This has the benefit of letting teachers concentrate only on further hands-on additional verification. We report our classroom experiences using Tenjin in the year of 2003 to 2004.*

## 1 Introduction

Automatic feedback is important for language e-learning systems in order to make the system interactive for students and to decrease the teachers' workload. Recent trends in NLP are corpus-based, performed by conducting statistical analysis on a large collection of text data. A bottleneck of this approach is the construction of corpus itself, which can only be manually constructed and maintained. Additionally, one issue of corpus construction is the lack of "negative" examples. Usually, people only publish correct writings and not incorrect ones. This problem is crucial for constructing automatic proofreading and assessment tools for writing, because negative examples are needed when looking for mistake patterns.

E-learning language courses offer a natural framework of collecting such corpora. An online exercise can collect dozens of answers with mistake patterns. These can be mined automatically and fed back into assessment tools. As most students study language at universities, a large volumes of language data may be collected in a short term. Note that the e-learning methodology is essential here, as extraction of mistake patterns can be conducted only on digital data.

In pursuit of this objective, we built a web-based e-learning platform called Tenjin. Tenjin provides an interactive environment enabling students to work on assignments and teachers to evaluate them. Tenjin already includes some automatic assessment procedures that evaluate assignments submitted by students. When students upload their assignments, their results are compared with that of teacher's using dynamic programming algorithm, and the feedback is immediately given to the students. This not only enables the teacher to concentrate on further hands-on evaluation, but also enhances student's self-learning. This procedure is further to be enhanced by the analysis performed on collection of students' data. For those exercise which are difficult to be evaluated automatically, tutors may give hands-on verification which also form a part of resulting corpus. Another feature of Tenjin is that it supports multiple languages, including this evaluation feature. All text is handled on the unicode basis and procedures inside Tenjin are applicable to any language.

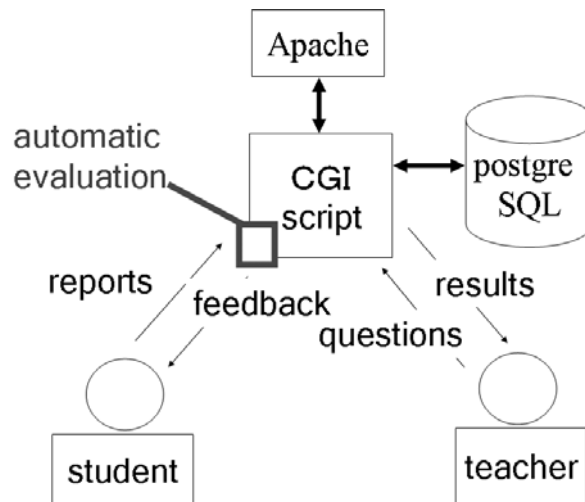


Figure 1: Overall design of Tenjin

There are many previous works ranging from commercial sites [1][3] to CALL systems[5] utilized at universities. One key features of these previous works is the automatic feedback for learners' results. Related to our work on evaluation on writings, ETS [4][2] provides the feedback based on syntactic analysis, discourse cues and topical analysis in English. Compared with these works, our final target is an integrated environment that assists education in multiple languages. In this paper, we report on Tenjin system and our classroom experience using Tenjin.

## 2 Tenjin System

### 2.1 Design

Tenjin is web-based system specially designed for language e-learning tasks. Tenjin's architecture is shown in Figure 1. This architecture was adopted from EMMA system[6][8], especially designed for programming language courses, which was successfully introduced by the author in 2001. Basically, Tenjin is a CGI script written in Ruby running on Linux via Apache server. The CGI script runs as an interface for a database constructed using PostgreSQL. The database consists of questions, student results, grades, teacher comments, and private information about each user.

Tenjin can handle various kinds of text, document formats (such as Word, PDF) and also multi-media files. Links to other URL pages can be shown on the question page, too. With these materials, teachers construct assignments on pronunciation, vocabulary, grammar, listening and writing. In contrast, students' answers are currently limited to typed-in text (no files can be uploaded for the moment).

Tenjin has two important features: being interactive and applicable to any language. The interactive feature is realized especially by the function of automatic evaluation, which will be described in §4. Including this process, Tenjin can handle any language. Currently, Tenjin is used in 4 different languages (detailed in §5). This feature is realized by all tasks inside Tenjin (including automatic evaluation) being language independent, and unicode base language handling. All actions made into Tenjin systems are transformed into Unicode. Thus, it enables mixture of languages: some courses use two languages, the mother tongue and the target language.

## 分類 Writing

日本語に合うように空欄を埋めなさい。

- 問題
- 1) 身体の技法という観念は難しいとはいえ、これらの事象すべてをわれわれが理解する助けになるだろう。
  - 2) 物理的であれ心理的であれ、ただひとつの視点では十分ではなかった。
  - 3) あまりやさしくはないこのテキストに彼が関心をもつように彼女は大いに努力した。
  - 4) その若い女性が修道院育ちだとわかったのは彼だけだ。
  - 5) 今日あるような社会の批判を最初に出版するのは彼女だろう。

## 関連

リンク None

ク

- 回答
- 1) Bien que la  , elle  faits.
  - 2) Qu'il soit physique  ou qu'il soit psychologique  , un seul point de vue  .
  - 3) Elle  pour  .
  - 4) Il  à  a été  .
  - 5) Elle  à  telles  .

自動採点結果

notion de ~~technique~~\_du corps soit difficile | Correct! nous aidera à comprendre tous ces | Correct! il soit physique | Correct! qu'il soit psychologique | Correct! n'était pas suffisant | Correct! a fait beaucoup d'efforts | qu'il ~~s'intéresse~~\_à ce texte peu facile | Correct! est le seul | Correct! avoir reconnu que cette jeune fille | Correct! élevée au couvent | Correct! sera la première | Correct! publier une critique des sociétés | Correct! qu'elles existent aujourd'hui

Figure 2: An automatically evaluated student's result

### 3 Examples on Use of Tenjin

#### 3.1 Student's Pages

Tenjin displays the homework when a student logs in. The student reaches his assignments by clicking on classes and links. For example, one question would look like Figure 2, where he sees features such as assignment category, question, link and forms that he has to fill in. Here, students were asked to fill in several blank boxes in French translated from given Japanese. The student solves the problems by referring to other course texts. When he completes solving the assignment, he submits the result by using GUI interface provided by Tenjin.

Immediately after, the automatic evaluation is displayed as is shown at the lower part of the page Figure 2. This student has made mistakes in his first and the third question. Incorrect results are crossed out suggesting some other words to be placed instead (under-bars indicate there is something missing.) The student may look at this automatic evaluation and correct his result and submit until he gets a "perfect" score, when no mistakes are found.

Students may solve questions as is described here, while other questions are to be evaluated manually. For these questions, the student will be able to see any further hands-on evaluation the next time he logs in.

#### 3.2 Teacher's Page

Meanwhile, the teacher uses Tenjin for three purposes: preparing problems, viewing results, and administrating classes and users. On preparation of problems, the teacher describes questions and the format for the answer by using CGI forms. For example, the question shown in the previous section can be created with a form shown in the left part of Figure 3. The teacher puts features such as grouping, method of evaluation, level and makes the question and the format. The format for answering, which is based on creation of fill-in boxes, is created using '#' marks (that indicate holes) and HTML . For example, 5 sentence boxes seen in Figure 2 can be created by using the format together with HTML tags:

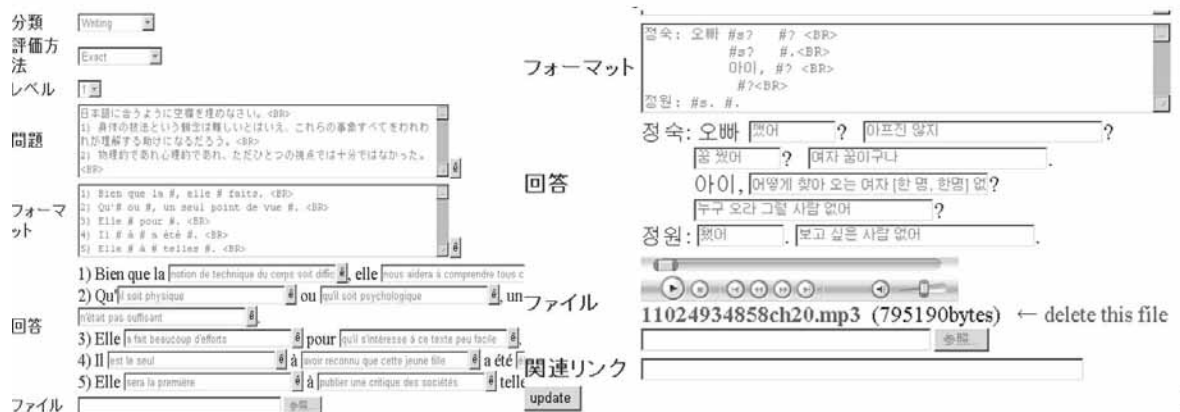


Figure 3: Teacher's questions construction pages



Figure 4: Teacher's evaluation pages

- 1) Bien que la # , elle # faits. <BR>
- 2) Qu' # ou # , un seul point de vue #. <BR>
- 3) Elle # pour # . <BR>
- 4) Il # à # a été. <BR>
- 5) Elle # à # telles #. <BR>

Other than ordinary text based questions, the teacher may create questions on media. The right part of Figure 3 shows the dictation question in Korean, where control panel is indicated. When the teacher uploads audio file, the system automatically prepares these control panels according to its file type. The teacher may put the answers in these boxes, which are utilized for automatic evaluation.

At another page, teachers view results of all students through the table of students and questions. The leftmost figure of Figure 4 shows a table of students on the vertical axis and two questions on the horizontal axis. 'O' and 'X' marks indicate whether the question was solved by the student or not. By clicking either a student or a question, the teacher will reach individual pages where he may see a student's progress more in detail. For example, the middle figure of Figure 4 shows how a student made progress in a question in Chinese. Looking from the

bottom towards the top, the student improves gradually (which can be seen by the decrease of text chunk crossed-out). The right-most figure shows the integral results of the whole language class. Teachers may make analysis by clicking on buttons at the first row, that functions for sorting students by that feature.

When the teacher decides to manually evaluate a student's result, Teachers are led to a page where student results are shown. On this same page, the teacher may make hands-on corrections and evaluation. The result is aligned with the student's original result and the differences of the two are fed back to the user.

### 3.3 Other Functionalities

There are other functions attached to Tenjin as follows:

- Bulletin board
- Publicizing good student results
- Sharing questions by teachers
- Archiving questions and answers

Bulletin board function for enhancing the communication in between the teacher and the student. The second feature is utilized for announcing good students' results and for motivating students to present better model results.

The latter two features are used to decrease the teachers' workload of question construction. Teachers of the same language may share all questions and reuse the other's results. All questions can be archived and reused.

## 4 Automatic Writing Evaluation

### 4.1 Types of Student Results

Currently on Tenjin, student results are limited to text and automatic evaluation is the procedure of evaluation on text. Students' results in text can be varied according to the constraint put to the content: from exact match to free text. This freedom can be classified into a Chomsky-like hierarchy:

**Level 0:** Text free of a theme

**Level 1:** Text with an assigned theme

**Level 2:** Text which is linearly the same

**Level 3:** Text that can be verified by exact match

Readers might think of employing conventional language analysis methods based on CFG, proposed in NLP. However, we have several requirements for the automatic evaluation process in classrooms:

- The evaluation precision should be very near to 100%. If not, students will be at a loss whether the system evaluation is inappropriate or his submission is incorrect.

- The evaluation should be simple, so that students will understand how it is evaluated. Even if the system evaluation is slightly incorrect, the student should be able to judge whether his writing is wrong, or the result of automatic evaluation is wrong.
- The method should be applicable to any languages, because several languages are scheduled to be taught on Tenjin, including non-segmented languages.

Our final choice was to give up automatic evaluation of Level 0, but evaluate the rest.

## 4.2 Evaluation for Level 3

Level 3 target at evaluation on expressions with variety of regular grammar, though current implementation remains to be almost exact match with some word selection feature.

Our evaluation is based on dynamic programming. There are two inputs into the evaluation function: the teacher's text and the student's text. These two are compared using longest common subsequence (LCS in the followings). For example, two texts match as follows:

Student's result: This is pens.

Correct answer: This is a pen.

LCS matches for "This is " and ".". From this result, the system finds out that "a" and "pen" are missing and "pens" is unnecessary. Resulting example can be seen in the Figure 2: missing characters are displayed underlined, and unnecessary characters are crossed out. LCS is performed at character level for non-segmented languages, and at word level for segmented languages. The LCS is made faster by limiting the range of the search field. Also, the evaluation function was devised so that the number of matched chunks will be the minimum.

Additionally, two more functions are supported in the automatic evaluation process so as to add some flexibility of the evaluation process. Firstly, regular expression is allowed inside the correct answer. For example, a correct answer such as "[He, She] said that the flower is beautiful." is provided by the teacher, meaning both "He" and "She" are allowed as subject.

The number of matches are transformed into grading from 1 to 5, and is shown with the detailed result (as is shown in Figure 2). When an answer perfectly matches with the correct answer, the system prints out "Perfect!".

## 4.3 Evaluation for Level 1 and 2

For assignments of free writing style, the automatic evaluation is very limited even with current NLP techniques. As is explained in §4.1, the requirement on e-learning systems is that the precision of the automatic evaluation should be very close to 100%. According to our preliminary evaluation process that utilize Japanese text segmentation process (which had the precision of around 95%), students tended to stop using Tenjin, once having found out that they could not judge whether the system evaluation is incorrect or their results are incorrect.

Therefore, we chose semi-automatic evaluation as follows. The teacher may give keywords that should be found and utilized in student's writings. For instance, in the case of self-introduction, the teacher may provide "live, hobby, job" and students are required to utilize these words on submission time. The more detailed evaluation is currently left to teacher's manual work. We are currently working on better evaluation of free texts using web-based technologies as proposed in [7].

Table 1: Course results for three different classes

language	year, semester	number of students	students' domain	course content	number of assignments
French	2003, winter	48	science	General (beginner)	55
French	2004 summer	18	literature	Composition	6
French	2004	42	law	General (beginner)	105
English	2004 winter	26	literature	Listening	1745
French	2004 winter	53	literature	General (advanced)	400
French	2004 winter	45	literature	General (beginner)	45
French	2004 winter	19	literature	General (advanced)	55
Chinese	2004 winter	53	science	General(beginner)	90
Chinese	2004 winter	45	literature	General(beginner)	90
Chinese	2004 winter	57	medicine	General(beginner)	90
Korean	2004 winter	9	literature	General (advanced)	35
Korean	2004 winter	32	science	General (beginner)	35
Korean	2004 winter	43	science	Presentation	5

#### 4.4 Manual Evaluation

Manual evaluation is performed when teacher wants to evaluate fully in detail. In this case, teachers modify student results. On submission, the difference in teacher's modification and student's results are automatically calculated and fed back to the student. Here again, LCS algorithm is used for alignment of the two results. The aligned result will help students immediately locate the modification made on his writings.

## 5 Classroom experience

### 5.1 Overview of 2003-2004

We tested Tenjin by using it in classes over a year period shown in Table 1. All classes were taught by tutors specialized in language teaching. The level of classes ranges from introductory to advanced. One class is 90 mins long per week held for one semester (half a year, 10 to 13 lessons). Those classes of 2004 winter semester are still on-going classes.

In every class, questions related to the main topic of the class are assigned to students as homework. Basic questions consist of training of verb conjugations, questions in grammar, translation and dictation of conversation, whereas advanced questions consist of summarizing newspaper articles or writing impressions about an URL site.

Students solve these assignments outside of class. Around 10 % of the students did not own computers at home, and they had to do assignments on terminals at the university computer center. All other students worked on assignments at home.

### 5.2 Qualitative Evaluation of 2003

As classes of 2004 are all still ongoing, we present here the evaluations obtained for the French class conducted in 2003. The submission rate was high, with class average of 85.9% if all students are included, and 96.8% excluding 5 students who subscribed but did not appear even once to the class. Thus, many students submitted all of the assignments. So far, we cannot conclude anything from this percentage, as we have not compared the effect with other classes that did not utilize Tenjin. However, looking at students' answers for our questionnaire, it seems that the motivation of students was successfully maintained using Tenjin, which resulted in this high

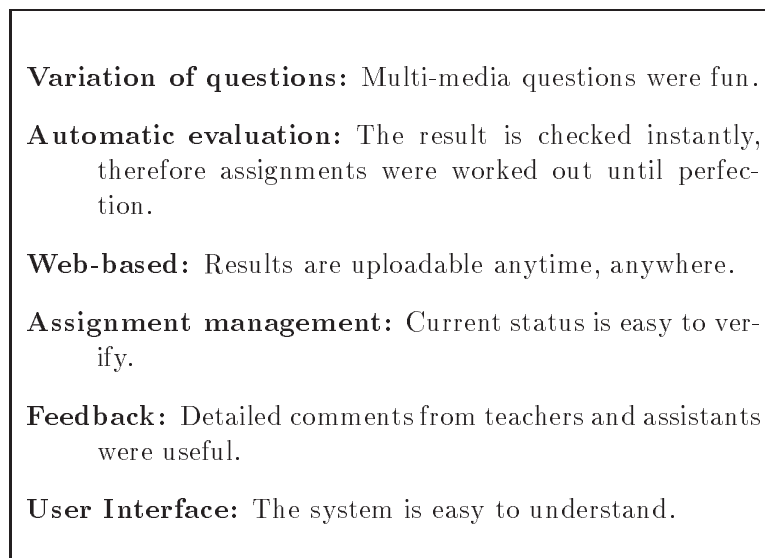


Figure 5: Student comments on “good features” of Tenjin

submission rate. At the same time, we collected more than 150 thousand words from students in this one class.

We asked the students to freely comment on the Tenjin system’s good features and required improvements. The good features are summarized in Figure 5.

Most positive comments were on multi-media assignments. Even for grammar questions, students liked questions with audio files. Questions with songs and interesting URL sites were very much appreciated by students, and it clearly raised the motivation to continue learning in French. Other positive comments included impressions on basic features of Tenjin, such as it being web-based and its user interface.

In contrast, negative comments were on entry of alphabetic characters with accents. This is due to the fact that in the year of 2003, the Tenjin prototype did not support direct entry using French. After solving this problem in March 2004, the user interface has not become an issue.

There were both positive and negative comments on automatic evaluation process. Above all, there are 48 students multiplied by 55 questions which sums up to 2500 assignment results to be scored. Therefore, without any automatic evaluation process, this intensive volume of assignment were not possible. Additionally, most students were positive about the immediate feedback. Many tried to work on assignments as if they were doing video games, targeting at obtaining the “Perfect!” from the Tenjin system.

The negative impression appeared especially when Level 1 texts were automatically evaluated. At this level, students expect their results to be read by the teacher. Still, we are now seeking to integrate more sophisticated evaluation.

Overall, students felt positive about using Tenjin. From the next winter semester, the we intend to involve Tenjin with classes of other languages, as Spanish, German, Chinese and Japanese in addition to current classes.



## 6 Conclusion

We reported on Tenjin, a web-based system supporting foreign language education. It provides an integrated environment enabling students to work on assignments and teachers to evaluate them. Tenjin automatically evaluates all assignments submitted by students at upload time using a dynamic programming algorithm and regular expressions.

Through our classroom experiences using Tenjin in the year of 2003 to 2004, we found that automatic evaluation is indispensable and students were positive about the current way of evaluation, although the procedure is simple. At the same time, we collected corpora of negative examples together with the corrections through using Tenjin in the class.

## References

- [1] BBC. Home page of bbc language learning, 2003. <http://www.bbc.co.uk/language/>.
- [2] J. Burstein and M. Chodorow. Automated essay scoring for nonnative english speakers. In *Proceedings of the ACL99 Workshop on Computer-Mediated language Assessment and Evaluation of Natural Language Processing*, 1999.
- [3] ChineseOn.Net. Internet home page of Chineseon.net., 2003. <http://www.chineseon.net/>.
- [4] M. Chodorow and J Burstein. Beyond essay length: Evaluating e-rater's performance on toefl essays, 2004. TOEFL Research Report 73, ETS PR04-04, Princeton, NJ: Education Testing Service.
- [5] S. Jager, J. Nerbonne, and A. van Essen. *Language Teaching and Language Technology*. Swets & Seitlinger, 2000.
- [6] K. TANAKA-Ishii, K. Kakehi, and M. Takeichi. Emma: A web based report system for programming language course. *Computer & Education*, 2004. in Japanese, accepted, to appear, in Spring 2005.
- [7] K. Tanaka-Ishii and H. Nakagawa. A multilingual usage consultation tool based on internet searching —more than a search engine, less than qa—. In *the 14th International Word Wide Web Conference*, 2005. to appear in May.
- [8] Kumiko Tanaka-Ishii, Kazuhiko Kakehi, and Masato Takeichi. A web-based report system for programming course —automated verification and enhanced feedback—. In *The 9th Annual Conference on Innovation and Technology in Computer Science Education*, page pp.218, 2004. Tips and Techniques.