

Challenges in Foreign Language Teaching in the Age of LLMs: Designing an Electronic Textbook for Chinese

Ľuboš GAJDOŠ

Abstract This article examines the use of an electronic textbook in foreign language teaching (FLT) in the context of natural language processing (NLP) and large language models (LLMs). It focuses on the teaching of Chinese as a foreign language and points to several limitations of traditional textbooks, such as slow updating, limited use of corpus-based data, and insufficient adaptation to smaller language communities, including Slovak learners. The paper presents GouBook, an electronic textbook designed primarily for the Chinese-Slovak language pair. The application offers Chinese word segmentation, *Hanyu pinyin* transcription, part-of-speech (POS) annotation, shared and personal glossaries, tooltip-based lexical information, dictionary import and export, and the export of learning materials. It also allows teachers to distribute texts and enables students to work with them both online and offline. The paper argues that such tools cannot replace systematic language learning, but they can make teaching materials more flexible, context-sensitive, and better adapted to actual language use.

Keywords Foreign language teaching, Large Language Models, Chinese language teaching; electronic textbook, NLP, GouBook, Chinese-Slovak language pair

Introduction

Foreign language teaching (FLT) has recently been undergoing significant changes. Traditional approaches to teaching are, in many respects, being challenged by the use of technologies based on natural language processing (NLP). From this perspective, the emergence of Large Language Models (LLMs) appears to be one

of the decisive developments. What, then, do these technologies bring, and how can they be actively—and perhaps effectively—used in FLT? This paper attempts to address these and related questions from the perspective of a teacher who actively uses these technologies not only in FLT, but also in linguistic research.

The emergence of LLMs initially appeared to be a revolution that would transform many areas of both professional and private life. After the initial enthusiasm, however, it has become clear that these expectations, sometimes excessively emphasized by technology giants, have been fulfilled only partially, and in some areas not yet at all. At the same time, it cannot be denied that there are areas in which the use of LLMs has brought about a minor revolution.¹ One such area is the study of foreign languages, or translation in general, where LLMs demonstrate considerable potential. Let us therefore take a closer look at the “traditional” approach to teaching a foreign language—in this case, Chinese. Due to space limitations, the discussion will focus only on teaching materials in the Chinese-Slovak language pair.

Teaching experience over more than two decades suggests that, despite being regularly updated (e.g. in five-year cycles), teaching materials still tend to follow traditional design principles and rarely incorporate quantitative indicators such as findings from corpus-based research. What is meant by this? One example is the selection of vocabulary and syntax based not on frequency distributions, but rather on selected thematic units. At the same time, it should be acknowledged that such topics need to be pedagogically relevant as well as engaging for students. On the other hand, first-year textbooks very often contain vocabulary that is not frequent in everyday communication, and some syntactic constructions do not reflect the current state of language use. It should also be added that quantitative data, such as frequency distributions, are available from various sources, including large language corpora, frequency dictionaries, and similar resources.

Another question that naturally arises in connection with the electronic textbook is why other available solutions, such as Wenlin and Pleco, are not used

1 Our internal data from the past two to three years also indicate declining student interest in these fields.

instead. There are several reasons for this. First, these tools are not available in Slovak (or in other languages with a relatively small number of speakers). Second, they do not allow sufficient customization to meet the specific needs of teaching in the department. In addition, the collected language data will serve as a basis for further refinement of the teaching process. Finally, this approach enables students to be actively involved in the entire process, for example through data collection in corpus linguistics courses or through the development of an electronic Chinese-Slovak dictionary.

Moreover, the preparation of a textbook itself often takes several months, more than a year, or even longer. To put it very simply, a textbook may already be outdated by the time it is published. If we also take into account the fact that a relatively large number of textbooks are produced for languages such as English, German, or Korean, while the situation is much more complicated for languages with a smaller number of speakers, such as Slovak, the problem becomes even more apparent. For this reason, the only available solution is often to use textbooks designed for native speakers of English² or to prepare *ad hoc* texts.³ How, then, can these and related problems be addressed? An electronic textbook may therefore represent a partial solution, although certainly not the only one, to these difficulties. It should also be emphasized that even the best teaching material cannot work miracles by itself, and the hard work involved in memorizing, for example, Chinese characters or sentence patterns remains the responsibility of students.

1 Word-Level Processing and Metadata

In order for an electronic textbook to help overcome the problems mentioned above, it should meet certain basic requirements. On the basis of previous experience gained from the development of the CAT-based instructional

² We will not discuss in detail here the problematic nature of such an approach, since it is very often the only available solution.

³ This approach is common especially in courses that respond to current topics or to new vocabulary and syntactic phenomena. Examples include Colloquial Chinese or the reading of journalistic texts.

application Goucat,⁴ the following section provides an overview of the proposed and implemented solutions.⁵

1.1 Tokenization and POS Annotation

In the NLP processing of Chinese, one of the key operations is tokenization, that is, the segmentation of continuous text into individual words.⁶ Subsequent text processing is often based precisely on the results of this operation. The tokenization procedure relied on two tokenizers: one based on open-source HSK data, and a second based on the Jieba library, specifically its Rust implementation, Rjieba.⁷ The same modules were also employed for POS annotation. Experience from translation-oriented courses indicates that students tended to show little interest in this feature in Goucat and used it only rarely. Similarly limited use of this feature in FLT is therefore expected.⁸ The functionality is included essentially because the results are already available during tokenization and are therefore available at no additional computational cost.

⁴ For more details, see Gajdoš, 2024.

⁵ Since the application is written in Python, the solutions discussed here are available for this language. At the same time, most of the open-source libraries mentioned are also available for other programming languages.

⁶ It should be recalled that Chinese texts do not contain spaces between words. For this reason, tokenization is a relatively trivial operation in languages written in the Latin alphabet, but a considerably more complex task in the case of Chinese character-based writing.

⁷ Dictionary-based tokenization tools, such as Jieba, rely primarily on predefined dictionaries. As a result, especially in the case of neologisms, the output may sometimes be unsatisfactory, although it is also possible to use a custom dictionary. Other solutions provide more accurate results, but they are often very large libraries. A future extension of the system will include tokenization using a locally running LLM, for example by processing the text directly at the point of input by the teacher.

⁸ The classification of words into parts of speech is a complex issue, not only linguistically but also didactically. However, due to the different focus of this paper, it will not be discussed here.

1.2 Phonetic Annotation

Among the available libraries for phonetic transcription, Pypinyin was selected on the basis of previous experience with Goucat, primarily due to its simplicity and relatively high accuracy. The input consists of tokens obtained through tokenization, for example with *Rjieba*.

1.3 Adding Comments

This functionality was inherited from Goucat. It provides additional information for a selected word or part of the text. The comment is displayed automatically when the cursor hovers over the given text. Highlighting of comments can be enabled in the graphical user interface. Text with a comment is then underlined, similarly to the way text is underlined in a word processor.

1.4 Collocation Search

The availability of empirical data from the Hanku⁹ corpora made it possible to include collocation search as one of the application's functionalities. For reasons of simplicity and efficiency, the application uses quantitative data, partly produced as outputs from courses in corpus linguistics. These data are downloaded from a web repository and subsequently stored locally. At present, this function is available only for verbs and nouns.

1.5 Tooltip Functionality

A key difference from Goucat is the use of tooltip functionality rather than an original-versus-translation interface. This means that when the user moves the mouse over the text, the application automatically displays the word, its phonetic transcription in *Hanyu pinyin*, POS annotation, the user's own translation if available, the user's own comment if available, and entries from the shared dictionary, hereafter referred to as the glossary. This function can also be disabled, and the results can instead be displayed in the lower part of the graphical user interface. See figure 1.

9 For more details, see Gajdoš et al., 2016.

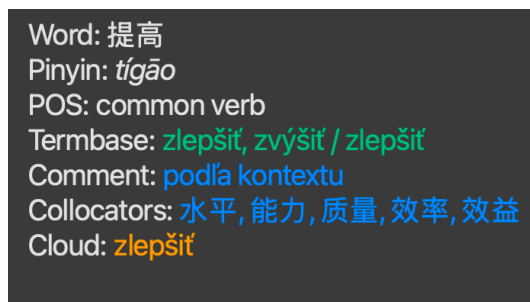


Figure 1
Tooltip overview.

1.6 Color Highlighting in the Text

Student feedback from the use of Goucat led to the inclusion of color highlighting as a text-annotation functionality. This feature essentially simulates the highlighting of text in a printed publication.

1.7 Translation

The application also provides online translation of selected text passages. This function can be disabled by the teacher for individual texts. Although Goucat included automatic online word translation, the results proved inadequate, and the function was therefore used only minimally. For this reason, automatic online word translation was not incorporated into the application.

1.8 Text-to-Speech

A simple aid for students is reading the text aloud, using the built-in text-to-speech functionality of the operating system.

2 *Application Functionality*

2.1 Sharing of Language Data

During instruction and use of the application, some language data are shared in a private cloud.¹⁰ These language data are not accessible to third parties and are used only for didactic and linguistic research conducted at our academic institution. The shared data include texts and selected vocabulary.

2.2 Import and Export of a Personal Dictionary

Since no electronic Chinese-Slovak dictionary was available as of spring 2026, students are encouraged from the beginning of their studies to build their own electronic dictionary, for example in a spreadsheet application. For this reason, the option to import a personal dictionary is available as a basic functionality. The application then allows users to add new vocabulary to their personal dictionary and subsequently export it in a spreadsheet format, typically ODS or XLSX. It is also possible to use a dictionary from the previously mentioned Goucat tool.

2.3 Export and Printing

Although most students prefer to create notes in electronic form, the option to export language material into a printable format remains important. For this reason, the application supports the export of texts and selected vocabulary into PDF, HTML, ODT, or DOCX format. The exported material may include a frequency-ranked vocabulary list, with the most frequent lexical items appearing first. Slovak equivalents are provided for words included in the personal dictionary. In addition, vocabulary can be categorized according to HSK levels.

¹⁰ The application allows users to disable this option.

Text

加入世界贸易组织之后

近日，中国经济景气监测中心与中央电视台《中国财经报道》互相协助，就中国加入世贸组织五年内生活质量、思想观念、就业水平等方面的问题，对600余位居民进行了问卷调查。调查中，近六成居民预测在未来五年内，生活质量将比现在有明显提，从多数居民对中国加入世贸组织后的五年的生活预期良好中，看出人们对中国经济在此时全面参与世界抱有信心，并认为国家经济水平上升，必然对国民有益与此同时，38.5%的居民预测自己或家庭的生活质量与现在基本相当，认为更大的发展在后面取有2%预测生活会不太令人满意。对于社会服务水平，91.5%的居民预测未来五年社会总体服务水平将会提高。逐步兴起的现代服务行业的竞争局面使绝大多数人为之欢喜，看到了希望，人们期待着服务方面有更多更好的选择，期待着国际标准服务随处可见。

我国已于2001年12月正式加入世贸组织，这意味着我国将在更大范围参与世界竞争与合作，得到好的发展机遇，享受多边协定的成果，获得进入其他国家市场的机会。从长期来看，它将对我国经济增长和对外贸易产生积极的影响，给我国传统的劳动密集型产品和技术成熟且已具有一定规模的轻工产品出口带来大量机会。据世界银行估计，加入世贸组织后五年内，我国贸易量在世界上所占的比重将从目前的4.5%提高到6.5%。加入世贸组织以后，我国企业将面临外国企业的致竞争，我国经济发展的状况将主要取决于我国企业在国际竞争中的地位与实力，而我国现行的经济体制还存在着不少不足，使本来就因为生产力发展水平仰万而缺乏国际竞争力的我国不少企业更难以应付国际竞争的挑战。

但对老百姓来说，则是有益无害。就业总量增加，就业结构发生改变俄国的消费者将面对更多的消费选择和更便宜的商品。

加入世贸组织之后，我国的国有企业和国有银行都将面临的一个突出问题这就是专业人才的缺乏和流失。所以，我国高等教育的主要任务之一，就是要尽快地培养各种专业人才，尤其是建立社会主义市场与对外开放所急需的法律人才。（字数统计：824）

Vocabulary

Word / Phrase	Transcription	Translation	Note
我国	wǒguó	Čína	
加入	jiārù	vstúpit	
经济	jīngjì	ekonomika	
发展	fāzhǎn	rozvoj	
之后	zhīhòu	potom	
世界	shìjiè	svet	
国家	guójiā	krajina	

Figure 2

Example of export to HTML format.

2.4 Controls and User Interface

The current generation of students has grown up with smartphones and often carries these usage habits over to desktop applications. In other words, they often prefer simplicity over a greater number of functions. They also tend to prefer interfaces similar to those used on smartphones.

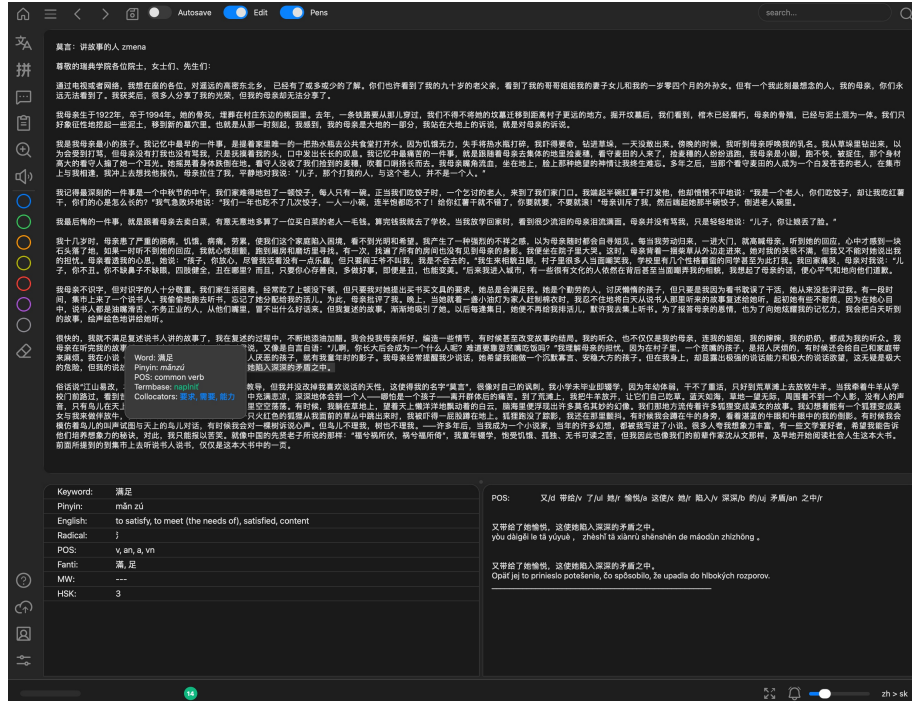


Figure 3
Application interface overview.

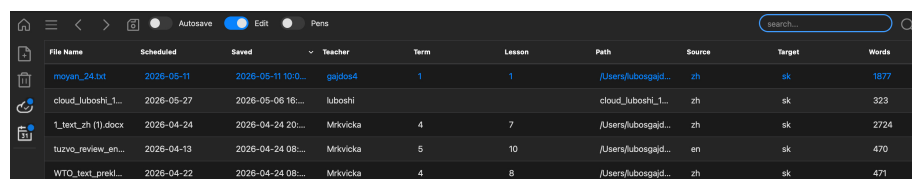
For this reason, PySide6 was selected as the basis of the interface,¹¹ since it offers greater flexibility than, for example, the simpler and visually less modern Tkinter.¹² In addition, the interface is designed not to rely on a GUI layout that separates the original text from its translation in parallel columns, nor on

- 11 Working with PySide is considerably more complex than working with Tkinter. In this respect, the alternating use of GPT-5.4 and Gemini 3.1 was of great help, although it must also be noted that this was far from what is often described as *vibe programming/coding*.
- 12 Tkinter is Python's standard library for creating simple graphical user interfaces, such as windows, buttons, and forms. It enables quick development of basic desktop applications without external tools. <https://docs.python.org/3/library/tkinter.html>

sentence-level segmentation. Compared with a CAT tool, this approach places greater emphasis on context, since the text often takes the form of a dialogue or another continuous structure.¹³

3 *Workflow from the Teacher's Perspective*

The workflow is fairly simple. The teacher uploads a text and assigns a teaching date to it. This process may be automated, for example by sending the text as an email attachment or, more simply, by uploading it directly to the cloud. After launching the application, the file is then downloaded automatically for the student,¹⁴ or the student can add it manually by opening it directly from the local drive of their computer. When adding the file manually, the student then enters the metadata manually: teacher, teaching date, semester, and lesson number. These data can subsequently be used to filter the display of the history of taught texts.



File Name	Scheduled	Saved	Teacher	Term	Lesson	Path	Source	Target	Words
moyan_24.txt	2026-05-11	2026-05-11 10:0...	gajdos4	1	1	/Users/lubosgajd...	zh	sk	1877
cloud_luboshi_1...	2026-05-27	2026-05-06 16:...	luboshi			cloud_luboshi_1...	zh	sk	323
1_text_zh (1).docx	2026-04-24	2026-04-24 20:...	Mirkvicka	4	7	/Users/lubosgajd...	zh	sk	2724
tuzvo_review_en...	2026-04-13	2026-04-24 08:...	Mirkvicka	5	10	/Users/lubosgajd...	en	sk	470
WTO_text_prekl...	2026-04-22	2026-04-24 08:...	Mirkvicka	4	8	/Users/lubosgajd...	zh	sk	471

Figure 4
Overview of files and translation tasks with key metadata.

¹³ The menu includes an option to display each sentence on a separate line.

¹⁴ The metadata for the text are also added automatically, for example the teacher, teaching date, group, course, and so forth.

4 *Overview*

The GouBook tool enables the teacher to distribute the text to be covered in class with little effort. Teachers can also set various parameters for the text, for example whether the text will be editable, that is, whether students will be able to make notes directly in the text, whether an online translator will be available, and so forth.

Students, on the other hand, receive a tool in which they can read the text, display their own or a shared dictionary, attach notes to words or parts of the text, view phonetic annotation, and so on. They may also use color highlighting in the text. The application also works in offline mode, although some functions require an online connection.

GouBook is primarily designed for the Chinese-Slovak language pair, but it is also intended to support other European languages, such as Czech, Hungarian, German, and English. With minor modifications, it can also be used for other language pairs.

The application is written almost entirely in Python and uses available open-source solutions. GouBook is designed as an application for Windows 10 and 11.¹⁵ JSON is used as the basic format for data storage.¹⁶

5 *Future Functionalities*

The original project envisioned the use of a local LLM at the academic institution. This model would serve as an aid for the teacher in designing exercises focused, for example, on practicing the syntactic specifics of vocabulary from a given text. At the same time, it would also serve as an aid for students, for example in the linguistic analysis of texts, including the previously mentioned tokenization, explanation of sentence structure, and similar functions. Given the currently high hardware costs associated with prompting, the implementation of this functionality has been deferred to future development.

¹⁵ Since the application is written in Python, compilation for Linux or macOS is also possible.

¹⁶ JSON was selected as the data format because it is easily readable by humans and because linguistic data can be extracted from this format relatively easily.

In the future, word search in the Hanku corpora is planned as an online extension complementing the existing basic collocation search functionality.

Conclusion

The most important question remains whether this type of textbook can support the acquisition of Chinese, or foreign-language acquisition more generally. The answer, as is often the case, is not entirely straightforward. Teaching experience over the past two decades suggests a gradual shift in priorities in foreign-language learning, with real-time dialogic communication increasingly coming to the foreground at the expense of monologic forms of communication. This shift may be attributed to several factors, and the development of LLMs is likely to further accelerate this tendency.

Indeed, LLM-based translation between Chinese and other languages has reached a relatively high level of quality, gradually repositioning the student or translator in the role of a linguistic proofreader rather than a primary producer of the target text. It may therefore be concluded that learners now have access to an unprecedented amount of linguistic information; the remaining challenge lies in critically evaluating and applying it effectively.

Table 1
Overview of Key Libraries Used

Functionality	Library / Module Used
GUI framework (widgets, signals, graphics, printing, networking)	PySide6 (The Qt Company, 2026)
Font Awesome icons	Qtawesome (Spyder Project Contributors, 2026)
HTTP requests / web APIs	Requests (Reitz & Requests Contributors, 2026)
Working with tabular data	Pandas (The pandas development team, 2020)
Word documents	Python-docx (python-openxml contributors, 2026)
Language detection	Simplemma (Barbaresi, 2024)
Chinese – word segmentation	Rjieba (messense, 2026), custom tokenizer based on open-source HSK data
Chinese – Hanyu pinyin	Pypinyin (mozillazg, 2025)
Text translation	Deep_translator (deep-translator contributors, 2023)

References

- Barbaresi, Adrien. 2024. *Simplemma* (Version 1.0.0). [Computer software]. <https://zenodo.org/records/11401403>
- deep-translator contributors. (2023). *Deep-translator* (Version 1.11.4). [Computer software]. <https://pypi.org/project/deep-translator/>
- Gajdoš, Ľuboš. 2024. “Goucat: Tackling Translation Challenges with Innovative CAT Tool Features.” *Studia Orientalia Slovaca*, 23(1), 1–20.
- Gajdoš, Ľuboš, Garabík, Radovan, & Benická, Jana. 2016. “The New Chinese Webcorpus Hanku—Origin, Parameters, Usage.” *Studia Orientalia Slovaca*, 15(2), 21–33.
- messense. 2026. *rjieba: jieba-rs Python binding* (Version 0.2.1). [Computer software]. <https://pypi.org/project/rjieba/>

- mozillazg. 2025. *Pypinyin* (Version 0.55.0). [Computer software].
<https://pypi.org/project/pypinyin/>
- python-openxml contributors. 2026. "Python-docx documentation.! Read the Docs.
Accessed May 6, 2026. <https://python-docx.readthedocs.io/>
- Reitz, Kenneth, & Requests Contributors. 2026. "Requests: HTTP for Humans." Read
the Docs. Accessed May 6, 2026. <https://docs.python-requests.org/en/master/>
- Spyder Project Contributors. 2026. "QtAwesome: Iconic fonts for PyQt." Read the Docs.
Accessed May 6, 2026. <https://qtawesome.readthedocs.io/en/latest/>
- The pandas development team. 2020. *pandas-dev/pandas: Pandas*. [Computer software].
Accessed May 6, 2026. <https://doi.org/10.5281/zenodo.3509134>
- The Qt Company. 2026. "Qt for Python documentation." Qt Documentation. Accessed
May 6, 2026. <https://doc.qt.io/qtforpython-6/>

Department of East Asian Studies, Comenius University, Bratislava, Slovakia