The object of this study is the process that generates machine-readable Country-by-Country reports in XML format using large language models. This paper addresses the task related to the current dependence of the process that generates these reports on specialized software, which leads to additional financial costs.

The research and analysis of the effectiveness of publicly available large language models for generating Country-by-Country reports with new data showed high results, provided that an example model of such generation was prompted. Three large language models out of nine studied yielded results close to ideal (obtained by manual preparation or specialized systems), namely 96 points out of 100 according to the devised evaluation methodology. Four other studied models demonstrated slightly lower efficiency, but their level is also sufficient for practical use. At the same time, the resulting average cost of generating one report (US cents 4.2) is significantly lower than in the case of using specialized systems.

Regarding the effectiveness of general-purpose large language models for generating Country-by-Country reports in the absence of a generation example, it is currently insufficient for practical use. In this case, all of the models studied showed results close to 0 points, i.e., completely incorrect reports were obtained. Such results are attributed to the insufficient amount of sample data during training of publicly available models.

Thus, publicly available large language models could in practice replace specialized software systems designed to generate Country-by-Country reports in XML format, at least in the case of generating new reports

Keywords: transfer pricing, transfer pricing documentation, large language models, XML generation

UDC 004.8:004.62

DOI: 10.15587/1729-4061.2025.337405

GENERATION OF MACHINE-READABLE COUNTRY-BY-COUNTRY REPORTS WITH LARGE LANGUAGE MODELS

Yakiv Yusyn
PhD
Department of Computer Systems Software
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"
Beresteiskyi ave., 37, Kyiv, Ukraine, 03056
E-mail: yusyn@pzks.fpm.kpi.ua

Received 29.05.2025
Received in revised form 17.07.2025
Accepted 11.08.2025
Published 29.08.2025

How to Cite: *Yusyn, Y. (2025). Generation of machine-readable country-by-country reports with large language models. Eastern-European Journal of Enterprise Technologies, 4 (2 (136)), 6–13. https://doi.org/10.15587/1729-4061.2025.337405*

1. Introduction

The rapid spread of artificial intelligence (AI) technologies has affected all areas of human professional activity, including traditionally conservative areas such as finance, accounting, and tax accounting. Basic AI technologies such as classification, clustering, and regression analysis have already become widespread in these areas [1]; the emergence of generative artificial intelligence has further accelerated their modernization [2]. This is because, unlike other AI technologies, generative artificial intelligence (primarily large language models (LLMs)) allows a wide range of tasks to be solved using a single software tool. One of the most common tasks is working with text documents – information extraction, transformation, generation, etc. – which is very valuable for the finance, accounting, and tax industries [2].

An example of such a task in practice is the task of preparing transfer pricing documentation. Defined as part of the action plan to combat base erosion and profit shifting (BEPS) [3, 4], this documentation consists of the following layers [5, 6]:

1. Country-by-Country (CbC) report in native and machine-readable (XML) formats [7].

2. Master file in native text format.

3. Local file in native text format.

According to estimates from 2016 (after the implementation of this framework), the costs of preparing documentation in Austria were EUR 165,000 per company, EUR 140,000, and EUR 155,000 per company, respectively, for each level [8]. Taking into account inflation, these estimates in 2025 correspond to costs of EUR 214,000, 182,000, and 200,000. These costs consist of salaries of specialists involved in the preparation of documentation, licenses, and software maintenance costs, etc. Any opportunities to reduce these costs, including through the latest advances in the field of AI development, are of significant practical and, accordingly, scientific interest. At the same time, the need to prepare transfer pricing documentation under this framework is not limited to OECD member countries, as many other countries have also adopted it and adapted it into their legislation. For example, according to the State Tax Service, in Ukraine an average of more than 2400 reports are submitted annually, for an average of UAH 2.4 trillion per year [9].

The use of generative artificial intelligence could significantly reduce the costs of preparing such documentation of any level by automating its preparation and partial generation based on semi-structured input data. In addition, the use of a single software tool (SST) to solve many different tasks could reduce dependence on specialized software tools, respectively, also affecting the reduction of costs. Therefore, research on automating the preparation of transfer pricing documentation using large language model technology is relevant.

Copyright © 2025, Authors. This is an open access article under the Creative Commons CC BY license

2. Literature review and problem statement

In [10], the potential of generative artificial intelligence for improving transfer pricing processes is considered and analyzed. At the same time, generative artificial intelligence is not considered as a universal solution but as one of many tools for eliminating "wastes" – unproductive or excessively manual steps in the process. In this case, critical aspects include analysis and warnings about the shortcomings of generative AI technology, which without human verification and participation could lead to the emergence of new types of errors and even a general deterioration of the situation. Based on analysis of the advantages and disadvantages of generative artificial intelligence, the paper proposes a three-stage approach "Define, Discover, Solve" as a framework for integrating generative AI into transfer pricing workflows. In general, in addition to the proposed approach, the cited work is more overview and valuable precisely by analyzing the advantages and disadvantages of generative artificial intelligence technology in the context of transfer pricing, but without analyzing examples of specific applications. This is most likely due to subjective factors, namely the definition of the target audience of the work as ordinary tax specialists who are just starting their "acquaintance" with generative artificial intelligence technologies.

Work [11] considers the possibilities and limitations of the application of artificial intelligence technologies in the field of transfer pricing in general; it also includes examples of such specific applications. The work focuses on the capabilities of artificial intelligence to quickly process large volumes of data for the purpose of preliminary analysis of operations, identification of risks and anomalies, filtering databases, etc. In the context of working with documentation, the capabilities of AI are also emphasized both for automatically extracting key data from documents and for creating or filling in standard parts of transfer pricing documentation. It is emphasized that any conclusions of artificial intelligence are based on data, and not on economic logic or legal context, therefore, for example, AI cannot correctly assess the economic essence of transactions. Therefore, it is concluded that artificial intelligence should act as a support tool, not a replacement, including in the preparation of transfer pricing documentation, but no specific examples are provided.

Paper [12] suggests that machine learning, predictive analytics, and natural language processing are key artificial intelligence technologies for the transfer pricing industry; their advantages are discussed. Machine learning is considered a universal tool for data analysis to identify patterns and anomalies, assess risks, and predictive analytics acts as a decision support tool. The paper identifies the potential of using natural language processing tools for working with documentation: data mining, monitoring legislative changes in real time, and automating the preparation of documentation according to OECD requirements. However, unlike the above papers analyzed, this study does not provide an overview of the possible disadvantages of implementing the technologies described, focusing only on the advantages, such as reducing costs, reducing effort, and time. In addition, the cited paper remains at the overview level, not including empirical examples of implementing the described technologies, including generating documentation. This could be explained by the too broad context of the study, which does not focus on specific artificial intelligence technology and its application.

In [13], the comprehensive application of artificial intelligence technologies is considered, expressing the idea of a virtually fully automated system for devising and monitoring the implementation of transfer pricing strategies with the detection of anomalies and risks. In addition, the idea is considered that such a system could automatically adapt to changes in legislation through the use of natural language processing technologies. In the task of preparing transfer pricing documentation, the idea of its automatic generation using language models is also repeated, as well as the idea of their use to improve the style of texts written by a person. At the same time, this issue is considered not only in the context of abstract documentation but with direct mention of reports by country. However, the empirical examples of application given in the work do not mention the use of AI technologies in this area, focusing only on existing applications for devising and optimizing transfer pricing strategies. This is explained by both objective (the author's lack of data on cases of AI use in the context of documentation) and subjective (focus on working with strategies) factors.

In [14], the entire transfer pricing process is represented as a four-stage, twelve-step structure, in order to identify the possibilities of implementing artificial intelligence at each step. These stages include data preparation and integration, analytics and modeling, strategy development and optimization, and implementation and monitoring, each of which is divided into three separate steps. Documentation preparation is also highlighted as a separate, 11th step of the last stage of the developed structure. As recommendations for implementing AI at this step, the following two points are highlighted: automatic generation of most of the documentation, and verification of the finished documentation for compliance with the regulatory requirements of various jurisdictions. However, despite the visualization of the process, these statements remain at the level of abstraction, without technical or methodological details, which is most likely due to the subjective limitations of the research context.

In [15], a simulation model was proposed to assess the impact of AI-based monitoring on profit distribution, audit risks, and compliance with transfer pricing guidelines. The goal of using such a model was to increase the accuracy, transparency, and auditability of transfer pricing systems in multinational companies. The model built represents multinational companies as multi-layered networks (goods, services, intangible assets) with an analysis of metrics such as price deviation, profit distribution disparity, and network centrality. Simulation using the developed model on artificial data showed that firms using AI demonstrate higher accuracy in pricing, especially in the areas of services and intangible assets. It was also shown that the effect of using AI is enhanced when it is applied to structurally central nodes of the network, which reflects the structure of a multinational company. Other tasks of operational transfer pricing, such as preparation of documentation, are not considered in the work, although they are mentioned as already reported in the literature.

In [16], the task of automating the process of preparing comparative studies (benchmark study), which are then used in the preparation of local files to justify the selected prices and transfer pricing methods, is considered. The preparation of such studies is a process that currently requires a lot of human effort due to the need for a stage of manual selection of companies for analysis from available databases. The study proposes the use of artificial intelligence technologies at this stage, reducing the number of companies for manual analysis by automatically parsing company websites and classifying them based on the received texts. According to the reported results, the devised prototype makes it possible to automatically screen out up to 80% of companies that were previously screened out manually, respectively, reducing the overall effort

and time spent on preparing a comparative study. The use of the described artificial intelligence technologies for preparing the other two levels of transfer pricing documentation remained undisclosed in the work, which is likely due to the limitations of the context and size of the study.

In [17], a wide range of applications of various artificial intelligence technologies in the tax domain, which are already implemented in practice or are at the prototype stage, is considered. In the context of transfer pricing, in addition to the already mentioned automation of the preparation of comparative studies, the work also describes the automation of the generation of other text components of the locale file based on the provided input data. To this end, the prototype built uses a large language model with pre-prepared prompts for generating a text description, along with stable rules for extracting information and inserting it into a specific place. In this case, the LLM is used without additional fine tuning and without providing examples of generation. In addition to generating text content, the large language model is also used in the prototype to automatically translate the generated locale file in order to obtain it in English and German. Considering the development of the idea of automating the preparation of comparative studies, it could be noted that the cited work fully covers the task of automating the preparation of local files using AI technologies, but the other two levels are not disclosed. This is due to the argument presented in the work that the preparation of local files is the costliest one (in terms of time and money) of all three levels.

In summary, to date, the best represented in the literature on the three levels of transfer pricing documentation is the generation of local files using LLM, while the country-by-country report and master file remain poorly studied. However, the approaches used to generate local files could also be applied to master files, but the task of generating a country-by-country report has its own specificity that differs from other levels. This especially applies to creating its machine-readable representation in XML format, which is still an extraordinary task for ordinary tax specialists. All this leads to the fact that in practice the task of generating country-by-country reports in XML format is solved using specialized software tools, such as "Aibidia TXM" [18], "PwC CbC2Go" [19], "WTS CbCR-2-XML" [20] or "tpcbc" [21], and many others. All these specialized software tools solve the task of generation in the conventional way: by transforming the collected data into the XML structure of the report using clear rules defined by the developers based on the XSD schema [7]. At the same time, these tools either include this functionality as a component of the preparation of the entire set of transfer pricing documentation ("Aibidia TXM") or specialize only in generating this part of the documentation. However, the use of such systems leads to dependence on another specialized software tool, and, accordingly, to additional costs for its licensing and/or maintenance, which increases the total costs of preparing the documentation. Therefore, the use of large general-purpose language models as a replacement for such specialized, conventional systems looks attractive from an economic point of view.

All this allows me to argue that it is advisable to conduct a study aimed at generating machine-readable CbC reports in XML format according to OECD requirements using publicly available large language models.

3. The aim and objectives of the study

The purpose of this study is to determine the conditions under which it is possible to use large general-purpose language models to generate machine-readable country-by-country reports in XML format. This would allow for the improvement of the processes of digital preparation of transfer pricing documentation, reducing dependence on specialized software solutions, which could have a positive economic effect in the form of a reduction in financial costs for the preparation of documentation.

To achieve this aim, the following objectives were accomplished:

– to analyze the effectiveness of publicly available large language models in generating CbC XML reports in the absence of an example of the expected result for the model;

– to analyze the effectiveness of publicly available large language models in generating CbC XML reports in the presence of an example of the expected result for the model.

4. The study materials and methods

4. 1. The object and hypothesis of the study

The object of this study is the process of generating machine-readable Country-by-Country reports in XML format using large language models.

The principal hypothesis of the study is built on the assumption that publicly available large language models could cope with the task of generating an XML report based on semi-structured input of entity data, without the need for additional training.

The following additional assumptions were also accepted:

– the principal hypothesis of the study could be corroborated at least when providing an example model of the expected result;

– the results for large language models with and without reasoning should be approximately the same.

The main simplification that was adopted in the study is to consider only the task of generating a Country-by-Country report, which contains only new data and is transmitted to the tax authorities for the first time. The XML report standard [7], in addition to such a case, also describes cases of correction and deletion of previously transmitted reports by generating a new report with a reference to the old one. The generation of such reports was not considered in this work, since, in addition to information about entities, the process of their creation also requires consideration and analysis of a previous report in XML format.

4. 2. Large language models under study

Based on ChatBot Arena [22], the state-of-the-art large general-purpose language models at the time of the study are shown in Fig. 1.

Based on the data analysis illustrated in Fig. 1, the following large language models from various vendors were selected for this study:

– GPT-4o by OpenAI (gpt-4o-2024-11-20);

– GPT-4.1 by OpenAI (gpt-4.1-2025-04-14);

– o3 by OpenAI (o3-2025-04-16);

– DeepSeek-V3 by DeepSeek (deepseek-v3-0324);

– DeepSeek-R1 by DeepSeek (deepseek-r1-0528);

– Claude Sonnet 4 by Anthropic (claude-sonnet-4-20250514);

– Claude Opus 4 by Anthropic (claude-opus-4-20250514);

– Gemini 2.5 Flash Preview by Google (gemini-2.5-flash-preview-05-20);

– Gemini 2.5 Pro Preview by Google (gemini-2.5-pro-preview-06-05).

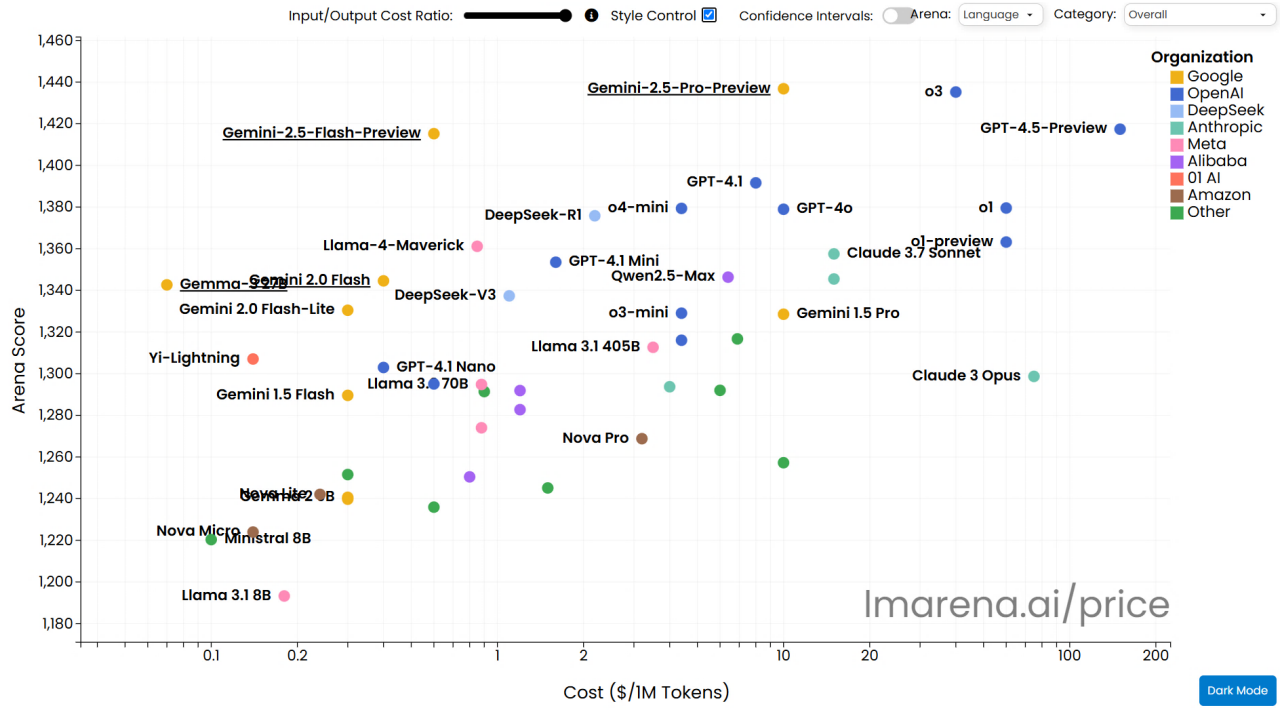

Fig. 1. Price-performance plot for large general-purpose language models [22]

4. 3. Experimental design

To solve the tasks set in my study, a single experimental design was developed (Fig. 2), which is common to both tasks.

Fig. 2 shows that the experiment consists of the following stages:

1. The initialization stage, during which the system prompt and input data for the large language model are generated.

2. The system prompt and input data are passed to the large language model, receiving an XML file with the generated report at the output.

3. The stage of technical validation of the received XML file using the XSD schema [7]. Any errors detected at this stage, if the stopping criterion has not been reached, are returned to the large language model together with a prompt asking them to correct them and re-generate the XML report. If the stopping criterion has already been reached, then the execution of the experiment is stopped.

4. In the case of successful completion of the technical validation stage, the generated XML report is additionally checked manually by an expert whose role was assigned to me.

As a stopping criterion for the technical validation stage, a simple counter was used, the maximum value of which was set to 10. Thus, each large language model was given 10 attempts to correct all the errors found and successfully complete this stage.

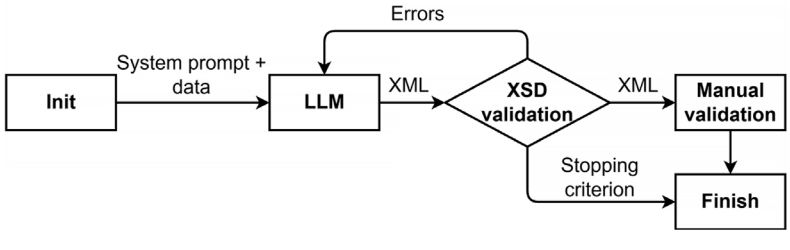

Fig. 2. Schematic showing the experimental design

The experiments performed under this statement, for the two tasks, differed only in the system prompt generated at the initialization stage, namely the presence or absence of an example for the model. This example, if present, contained an instance of the input data (covering most of the capabilities of the report standard) and a manually generated CbC XML report corresponding to them.

The following test cases were selected as input data for the experiments (from simplest to most complex):

1. One entity, without additional information.

2. One entity, with additional information.

3. Several entities from one country, without additional information.

4. Several entities from one country, with additional information.

5. One entity from several countries, without additional information.

6. One entity from several countries, with additional information.

7. Several entities from several countries, without additional information.

8. Several entities from several countries, with additional information.

The input data corresponding to the test cases were fed to the input of the language model in a semi-structured format, namely in the form of a simple text description of each entity and information about it. If additional information was available, it was added to the input after the entity description, also in the form of a textual semi-structured description.

4. 4. Developed software

In order to automate the execution of a fairly large number of experiments during this study, software was developed, which was termed "CBC XML AI". That allowed me to significantly reduce the human factor and the

amount of human effort, leaving human intervention in the course of experiments only at the stage of manual validation of the received report.

The software was developed using the .NET 8 platform [23] and the C# 12 programming language [24]. "CBC XML AI" has a console interface with the transfer of all input data in the form of launch parameters, for the implementation of which the "System.CommandLine" library was used [25]. As a result, the software provides the following functionalities, which are regulated by the corresponding parameters:

– reading input data from a text file (parameter --in);

– generating a system prompt for the LLM, without or with a report example (parameter --add-sample);

– connection to the selected large language model using the API with the passed key (parameters --model and --api-key);

– automatic validation of the generated XML report according to the XSD schema, using the built-in APIs of the .NET platform, with re-generation in case of errors (parameters --xsd-max-steps and --xsd-delay);

– logging of the generation process to the console (received XML, XSD validation errors), with the ability to turn it off (parameter --silent);

– in the case of successful completion of the technical validation stage, saving the received XML at the specified address (parameter --out) and setting a successful exit code (exit code);

– saving the entire history of communication with the LLM (parameter --dump);

– setting an incorrect exit code if no valid XML was received for the specified number of XSD validation steps.

To work with the API of large language models, the developed software uses the "LlmTornado" library [26]. This library is a gateway that provides a single, harmonized API for working with models from different vendors, including those selected for the study, which accelerated the development of the application.

4. 5. Principles of evaluating the results of experiments

The results of the experiments performed according to the scheme described above were evaluated in this work according to the following principles:

1. The range of scores is from 0 to 100.

2. Each unsuccessful attempt to pass the report validation according to the XSD schema reduces the maximum score by 10.

3. If actual discrepancies between the input and the data obtained in the XML report are detected at the manual validation stage, 0 is accepted as the final score.

4. If other errors are detected at the manual validation stage, the final score is reduced by the value determined by the expert depending on the criticality of the error, with an indication in the protocol.

Thus, perfect generation results receive a score of 100, obviously incorrect (according to XSD schema or manual validation) generation results receive a score of 0. Any other scores (in the range from 0 to 100) correspond to generation results that successfully passed technical validation not on the first attempt, and/or contain errors of a non-factual nature.

5. Results of the study on generating CbC XML reports using large language models

5. 1. Analyzing the effectiveness of publicly available LLMs in the absence of an example of the expected result

Analysis of the effectiveness of large general-purpose language models in the absence of an example of the generation result was carried out using the developed software "CBC XML AI" (parameter --add-sample = false), for each of the formulated test cases. The results obtained, evaluated in accordance with the evaluation principles described above, are given in Table 1.

Table 1

Results of analyzing the effectiveness of selected large language models in the absence of an example

Model \ Case	1	2	3	4	5	6	7	8
gpt-4o-2024-11-20	0	0	0	0	0	0	0	0
gpt-4.1-2025-04-14	0	0	0	0	0	0	0	0
o3-2025-04-16	0	0	0	0	0	0	0	0
deepseek-v3-0324	0	0	0	0	0	0	0	0
deepseek-r1-0528	0	0	0	0	0	0	0	0
claude-sonnet-4-20250514	0	0	0	0	0	0	0	0
claude-opus-4-20250514	72	0	0	0	12	0	0	0
gemini-2.5-flash-preview-05-20	0	0	0	0	0	0	0	0
gemini-2.5-pro-preview-06-05	32	0	0	0	0	0	0	0

Note: light orange corresponds to a failed technical validation stage; light gold corresponds to a failed manual validation stage; green corresponds to a successful completion of both validation stages

Table 1 demonstrates that none of the studied large language models was able to achieve a stable positive result on all test cases:

– seven models out of nine were unable to generate a correct report for any of the cases: neither technical (light orange color) nor manual (light gold color) validation was passed;

– the Claude Opus 4 model was able to generate a correct report for two test cases out of eight (1st and 5th), passing technical validation on the third and ninth attempts, respectively;

– the Gemini 2.5 Pro model was able to generate a correct report only for a single, 1st test case, spending six unsuccessful attempts at technical validation on this.

At the same time, in the case of three correct reports, the manual check revealed the following non-critical errors, which also affected the final score:

– incorrect timestamp of report generation in the Timestamp element (–4 points) – all three reports;

– lack of role designation (Role element) for the entity selected as the reporting entity (–4 points) – Claude Opus 4/5th case and Gemini 2.5 Pro/1st case;

– lack of postal code (PostCode element) in the entity address (–4 points) – Claude Opus 4/1st case.

5. 2. Analyzing the effectiveness of publicly available LLMs in the presence of an example of the expected result

Analysis of the effectiveness of publicly available large language models in the presence of an example of the expected result was carried out using the developed software "CBC XML AI" (parameter --add-sample = true), for each of the formulated test cases. The results obtained, evaluated in accordance with the evaluation principles described above, are given in Table 2.

Table 2 demonstrates that most of the studied large language models successfully generate a CbC report for all test

cases (green color) if there is a generation example. The only exception is Google's LLM: Gemini 2.5 Pro was able to generate a correct report only for the first test case, and the results of Gemini 2.5 Flash never failed to pass technical validation. The best results were shown by the OpenAI o3 model and both models from Anthropic – these three models scored the highest among those obtained for all eight test cases. At the same time, two of these three models are models with reasoning support (o3 and Claude Opus 4).

Table 2

Results of analyzing the effectiveness of selected large language models with an example

Model \ Case	1	2	3	4	5	6	7	8
gpt-4o-2024-11-20	92	92	92	92	92	92	92	87
gpt-4.1-2025-04-14	96	96	96	96	92	96	96	96
o3-2025-04-16	96	96	96	96	96	96	96	96
deepseek-v3-0324	92	92	96	96	96	96	96	96
deepseek-r1-0528	96	96	86	96	92	96	96	96
claude-sonnet-4-20250514	96	96	96	96	96	96	96	96
claude-opus-4-20250514	96	96	96	96	96	96	96	96
gemini-2.5-flash-preview-05-20	0	0	0	0	0	0	0	0
gemini-2.5-pro-preview-06-05	52	0	0	0	0	0	0	0

Note: light orange color corresponds to a failed technical validation stage; light gold color corresponds to a failed manual validation stage; green color corresponds to a successful completion of both validation stages (the darker the color, the better)

The main errors detected by manual verification are:

– incorrect report generation time stamp in the Timestamp element (-4 points) – this error is inherent in all reports that were generated;

– lack of role designation (Role element) for the entity selected as the reporting entity (–4 points) – GPT-4o for all cases, GPT-4.1/5th case, DeepSeek-V3/1st and 2nd cases, DeepSeek-R1/5th case, Gemini 2.5 Pro/1st case.

For GPT-4o, a specific error was also received in the eighth test case (–5 points): inclusion of correct country information in additional information that was not directly available in the input data.

Most of the correct reports were successfully generated on the first attempt, the only exceptions being DeepSeek-R1/3rd case (on the second attempt) and Gemini 2.5 Pro/1st case (on the fifth attempt).

6. Discussion of results related to the study on generating CbC XML reports using large language models

The results obtained show the possibility of using publicly available large language models to generate CbC XML reports, but only if they are provided with an example of the expected result.

The results of analyzing the effectiveness of the state-of-the-art models when generating reports in the absence of an example (Table 1) demonstrate that none of these models is able to cope with this task. This is explained by the fact that the task of generating CbC XML reports is quite niche, so publicly available large language models did not have enough examples related to this task during their training. Therefore, the reports generated in this way, with a general "understanding" of the task, contain a large number of structural errors, including "fictional" XML elements. As an example of such "fictional" elements, I can note various variants of the root element obtained during the research – CountryByCountryReport, CbCReport, CBCR_OECD – with models from different vendors being prone to different variants. Even with feedback from the XSD validation system, currently available LLMs were unable to correct all structural errors within a certain number of attempts, or as a result of the corrections they introduced factual errors.

The easiest way to significantly improve the efficiency of publicly available large language models in the task of generating CbC XML reports is to provide an example of input data and the generation result, showing the results obtained (Table 2). This is explained by the fact that in this case the generation task is reduced to the task of converting input data from one format to another – a task in which large language models conventionally show high efficiency.

It should be noted that the results shown in Tables 1, 2 are influenced by the way the experiments were performed, namely, access to the models using the API, and not the chat user interface. This fact explains the lack of a correct generation time stamp in all reports generated by the studied models, since this information was absent from the system prompt provided to them by the developed software. When accessing language models through a user chat window, most providers automatically add a system prompt containing the current date and time, which makes it impossible for this error to occur with such access. Therefore, it could be assumed that with this use case, all the "green" results from Tables 1 and 2 would have a score that is 4 points higher than the one obtained in this study. In this case, three of the nine models studied would receive perfect, 100-point results on all test cases, which should fully correspond to the results of existing conventional systems, which confirms the main hypothesis of the study.

The results reported in this study fit into the twelve-step structure of the transfer pricing process described in [14] and could also be integrated into the automated system given in [13]. However, unlike existing studies [16, 17], which determine the conditions and effectiveness of AI in the preparation of local files, this study determines the conditions and effectiveness of LLM in the preparation of the report by country.

In this case, the total cost of the study for generating reports was USD 39.33, which was enough to obtain 935 reports, i.e., the average cost of generating one report was USD 0.042. The total number of reports includes not only obtaining the results reported in this work but also additional generation during prototyping and debugging of the developed software. It could be argued with a high degree of probability that these economic indicators significantly exceed the indicators that could be achieved using existing conventional systems for generating CbC XML reports, with the same performance indicators. In the case of such systems, the cost of generating a report is defined as the distribution of total licensing/maintenance costs between individual generated reports. For systems that specialize only at the country level (such as the aforementioned "PwC CbC2Go" [19], "WTS CbCR-2-XML" [20], "tpcbc" [21]), these costs could amount to hundreds or thousands of US dollars, and for systems that

cover all three levels (such as "Aibidia TXM" [18]) – tens of thousands. The use of existing conventional systems could still be economically justified in more advanced scenarios that include not only generation but also, for example, joint collection and preparation of data for it. But in the baseline scenarios tested in the study, the use of large general-purpose language models that could be used to solve many other tasks is more cost-effective than existing conventional systems. It also makes it possible to reduce the dependence of the transfer pricing documentation preparation process on these systems or completely replace them with a large language model if the coverage of the baseline scenarios is reasonable. The only condition for such use without loss of efficiency, as this study has shown, is to provide an initial prompt model with an example of input data and the resulting XML report.

When comparing existing specialized systems and large language models, one should also remember the limitations of this study, namely that analysis of the ability of the LLM to generate only new reports (OECD1) was carried out. Editing (OECD2) and deletion (OECD3) of transmitted data is supported by most existing systems, and analysis of the effectiveness of large language models in handling these cases is a promising area for further research.

A disadvantage of this study is the general variability of the field of large general-purpose language models, in which new models from different vendors appear every few months. At the same time, as the latest available data already show, new models are not always more effective than previous ones, for example, they may increase the level of hallucinations. All this may lead to the fact that the results obtained using new models may differ from the results reported in the study, which operates on the state-of-the-art ones available at the time of writing. At the same time, deviations are possible both in the greater and lesser direction, but to a greater extent this may concern the efficiency of large language models in the absence of an example of the expected generation result. The results obtained in the presence of an example of the expected generation result should be more resistant to model updates due to the nature of the task of converting data from one format to another.

Another promising area to build on the current research is to improve the feedback system between the XSD validation system and large language models. As analysis of report generation logs in the absence of an example reveals, the development of a special format for schema error messages could improve the results obtained to the level of obtaining correct reports for all test cases. For example, the OECD XSD schema requires a strict order of elements, and if it is violated, large language models remove "extra" elements instead of changing the order due to incorrect validation errors of the type "invalid element A". When replacing such errors with correct messages of the type "invalid order of elements A and B", one should expect a significant improvement in the correctness of reports in the absence of a generation example.

One more area of further studies is to determine the ability of publicly available large language models to generate CbC XML reports taking into account the specific, more stringent requirements of individual countries, since the OECD requirements are only a mandatory minimum. In practice, most countries develop their own requirements based on them, which are more stringent regarding the content of individual elements. In addition, in the case of some countries (for example, Germany [27]), separate APIs are available for transferring CbC reports to the relevant authorities, which require the OECD report to be embedded in their own schema. However, in this case, with a high degree of probability, one could expect results that will be close to the results reported in this study that are given in Table 1 and Table 2.

7. Conclusions

1. Analysis of the effectiveness of publicly available large language models for generating CbC XML reports in the absence of an example of the expected result has revealed their insufficient effectiveness in such an application. None of the selected modern models from different vendors was able to show stable positive results in all test cases: out of 72 generation results, only 3 turned out to be positive. Such results significantly lose in efficiency to existing conventional specialized systems and cannot be applied in practice. This is explained by the narrow specialization of the task of generating CbC XML reports, which led to a lack of examples of results when training publicly available language models.

2. Analysis of the effectiveness of publicly available large language models for generating CbC XML reports in the presence of an example of the expected result has revealed their high effectiveness in such a use scenario. Two models with reasoning support and one without it were able to achieve a result close to ideal in all test cases, which corresponds to the efficiency of existing traditional systems based on hard mapping of data into XSD schema. Four more models studied have shown slightly lower efficiency, but its level is also sufficient for practical application. Such a high level of efficiency of LLM in this task is explained by the fact that if there is an example of generating a CbC XML report, the generation is reduced to converting data from one format to another. At the same time, the average cost of generating one report (USD 0.042) is significantly lower than in the case of existing specialized systems, for which the cost of generation consists of distributing licensing/maintenance costs among generated reports.

Conflicts of interest

The author declares that he has no conflicts of interest in relation to the current study, including financial, personal, authorship, or any other, that could affect the study, as well as the results reported in this paper.

Funding

The study was conducted without financial support.

Data availability

The manuscript has associated data in the data warehouse: https://doi.org/10.5281/zenodo.15687636

Use of artificial intelligence

The author confirms that he did not use artificial intelligence technologies when creating the current work.

References

1. Yi, Z., Cao, X., Chen, Z., Li, S. (2023). Artificial Intelligence in Accounting and Finance: Challenges and Opportunities. IEEE Access, 11, 129100–129123. https://doi.org/10.1109/access.2023.3333389
2. Dubey, S. S., Astvansh, V., Kopalle, P. K. (2025). Generative AI Solutions to Empower Financial Firms. Journal of Public Policy & Marketing, 44 (3), 411–435. https://doi.org/10.1177/07439156241311300
3. Action Plan on Base Erosion and Profit Shifting (2013). OECD. https://doi.org/10.1787/9789264202719-en
4. Dharmapala, D. (2014). What Do We Know about Base Erosion and Profit Shifting? A Review of the Empirical Literature. Fiscal Studies, 35 (4), 421–448. https://doi.org/10.1111/j.1475-5890.2014.12037.x
5. Transfer Pricing Documentation and Country-by-Country Reporting, Action 13 - 2015 Final Report. In OECD/G20 Base Erosion and Profit Shifting Project (2015). OECD. https://doi.org/10.1787/9789264241480-en
6. Ouelhadj, A., Bouchetara, M. (2021). Contributions of the Base Erosion and Profit Shifting BEPS Project on Transfer Pricing and Tax Avoidance. Financial Markets, Institutions and Risks, 5 (3). https://doi.org/10.21272/fmir.5(3).59-70.2021
7. Country-by-Country Reporting XML Schema: User Guide for Tax Administrations. Version 2.0 (2019). Paris: OECD Publishing. Available at: http://www.oecd.org/tax/beps/country-by-country-reporting-xml-schema-user-guide-for-tax-administrations-june-2019.pdf
8. Bergmann, S. (2016). Neue Verrechnungspreisdokumentationspflichten für multinationale Unternehmensgruppen. Zeitschrift für Gesellschaftsrecht und angrenzendes Steuerrecht, 148.
9. Rezultaty roboty DPS shchodo podatkovoho kontroliu za transfertnym tsinoutvorenniam (2025). Kyiv. Available at: https://tax.gov.ua/data/material/000/780/912318/Dodatok_1.pdf
10. Carey, A., Tanguay, B. H. (2025). How Can GenAI Improve My Transfer Pricing Process? Tax Management International Journal. Available at: https://kpmg.com/kpmg-us/content/dam/kpmg/taxnewsflash/pdf/2025/03/KPMG_GenAI_tmij_March2025_final.pdf
11. Dinev, D., Wojewoda, A. (2024). Opportunities and limitations of AI in transfer pricing. International Tax Review. Available at: https://www.internationaltaxreview.com/article/2dxro1nggp5h8t2flrtog/sponsored/opportunities-and-limitations-of-ai-in-transfer-pricing
12. Khalil, M. (2024). The Role of AI in Enhancing Transfer Pricing Accuracy and Efficiency. Advances in Information Technology, 7 (1), 1–11. Available at: https://acadexpinnara.com/index.php/acs/article/view/350
13. Basharat, A. (2024). The Role of AI in Transfer Pricing: Transforming Global Taxation Processes. Aitoz Multidisciplinary Review, 3 (1), 254–260. Available at: https://aitozresearch.com/index.php/amr/article/view/55
14. Puttaraju, K. H. (2024). Leveraging AI for Transfer Pricing Strategy Development and Execution: A Practical Approach. Interantional Journal Of Scientific Research In Engineering And Management, 08 (11), 1–6. https://doi.org/10.55041/ijsrem32711
15. Moro Visconti, R. (2025). Artificial Intelligence And Transfer Pricing: A Multilayer Network Model for Compliance and Risk Mitigation. https://doi.org/10.2139/ssrn.5209028
16. Beuther, A., Fettke, P., Just, V., Riedl, A. (2020). KI-Einsatz für Effizienzgewinne bei Benchmarkstudien im Bereich Transfer Pricing. beck.digitax, 5, 316–323. Available at: https://wts.com/wts.de/publications/fachbeitraege/2020/2020_05_beck_digitax_316_Beuthe_Fettke_Just_Riedl.pdf
17. Beuther, A., Rombach, A., Stephan, S., Fettke, P., Köppe-Karkutsch, J., Dönnebrink, M. (2024). Künstliche Intelligenz im Steuerbereich: Innovationsstudie zum Potenzial und zur technologischen Entwicklung. KI Studie. Available at: https://wts.de/wts.de/KI%20Studie/KI-Folgestudie%202024_20240429.pdf
18. Aibidia TXM: Verrechnungspreis-Management. TAXPUNK. Available at: https://taxpunk.de/tools/328/aibidia-txm/
19. PwC CbC2Go: Workflow-basiertes CbC-Reporting. TAXPUNK. Available at: https://taxpunk.de/tools/65/pwc-cbc2go/
20. WTS CbCR-2-XML: Umsetzung der XML-Struktur im Rahmen des CbC-Reportings. TAXPUNK. Available at: https://taxpunk.de/tools/85/wts-cbcr-2-xml/
21. TPCBC: OECD konformes Country-by-Country Reporting. TAXPUNK. Available at: https://taxpunk.de/tools/318/tpcbc/
22. Chiang, W., Zheng, L., Sheng, Y., Angelopoulos, A. N., Li, T., Li, D. et al. (2024). Chatbot arena: an open platform for evaluating LLMs by human preference. Proceedings of the 41st International Conference on Machine Learning, 8359–8388. Available at: https://dl.acm.org/doi/10.5555/3692070.3692401
23. What's new in .NET 8 (2024). Microsoft. Available at: https://learn.microsoft.com/en-us/dotnet/core/whats-new/dotnet-8/overview
24. What's new in C# 12 (2024). Microsoft. Available at: https://learn.microsoft.com/en-us/dotnet/csharp/whats-new/csharp-12
25. dotnet/command-line-api at 2.0.0-beta4.22272.1. GitHub. Available at: https://github.com/dotnet/command-line-api/tree/2.0.0-beta4.22272.1
26. lofcz/LlmTornado at v3.5.18. GitHub. Available at: https://github.com/lofcz/LlmTornado/tree/v3.5.18
27. Communication Manual DIP Standard 2.1 BZSt. Available at: https://www.bzst.de/SharedDocs/Downloads/EN/dip_elma/Communication_Manual_DIP_Standard_2.pdf