Chat2VIS: Fine-Tuned Data Visualisations using Natural Language with Multilingual Capabilities via Large Language Models

Journal:	*Transactions on Visualization and Computer Graphics*
Manuscript ID	TVCG-2023-05-0238
Manuscript Type:	Regular
Keywords:	I.2.1.l Natural language interfaces < I.2.1 Applications and Expert Knowledge-Intensive Systems < I.2 Artificial Intelligence <, H.5 Information Interfaces and Representation (HCI) < H Information Technology and Systems, I.6.9.c Information visualization < I.6.9 Visualization < I.6 Simulation, Modeling, and Visualization < I Computing Methodologie, I.2.7.c Language models < I.2.7 Natural Language Processing < I.2 Artificial Intelligence < I Computing Methodologies

SCHOLARONE™
Manuscripts

Chat2VIS: Fine-Tuned Data Visualisations using Natural Language with Multilingual Capabilities via Large Language Models

Paula Maddigan and Teo Susnjak

Abstract—In our data-saturated world, there is a pressing need to harness technology to derive insights. Yet, traditional tools entail heavy learning overheads to work with complex charting techniques. Such barriers can hinder those who may benefit from harnessing data for informed decision-making. However, generating data visualisations directly from natural language text (NL2VIS) is an emerging field that addresses this issue. This study showcases Chat2VIS, a state-of-the-art NL2VIS solution for conversational chart generation. This work capitalises on the latest in AI technology leveraging large language models (LLMs) such as GPT-3, Codex, and ChatGPT. This work illustrates how Chat2VIS can generate and fine-tune data visualisations conversationally with capabilities that are beyond those demonstrated in previous studies. In addition, this paper presents the novel capability of Chat2VIS to comprehend multilingual natural language requests. Our work is evaluated against two NL2VIS benchmark datasets. In the process, we propose an automated methodology for conducting evaluations which are otherwise performed manually and which are also rare. We contribute findings and recommendations going forward with respect to improving the development of benchmark datasets for NL2VIS in order to enable automated evaluations, and in the process facilitate an acceleration of advancements in this field.

Index Terms—ChatGPT, Codex, GPT-3, end-to-end visualisations from natural language, large language models, natural language interfaces, text-to-visualisation.

1 Introduction

In an era where data has become a valuable commodity, industries are continuing to witness immense growth in its volume. The richness of this data holds the key to transformative insights that can steer decision-making, shape strategic visions, and unlock untapped potential. Data visualisations offer an effective and compelling approach to communicating these insights and facilitating better-informed business decisions. In our quest to democratise access to these data visualisation tools and make them more user-friendly [1], the emerging field of Natural Language Interfaces (NLI) is poised to transform the way we interact with and understand our data.

The ambition of articulating visualisation requests through natural language (NL) text and intuitive interfaces is fast gaining traction [2]. To generate suitable charts without the understanding of chart type nuances is an enticing objective and well-designed NLIs can avoid the need for programming skills and arduous learning curves [3].

Recently, the surge in interest in large language models (LLMs) is driving research to develop NLIs leveraging this technology. By utilising the advanced capabilities of OpenAI's GPT-3, Codex, and ChatGPT models, this study delves into the potential of Large Language Models (LLMs) in advancing data visualisation. These models are trained on a large amount of internet information and code repositories. Consequently, they exhibit a high level of skill in language semantics and code scripting. This unique capability is of significant importance in this work.

One remarkable feature of the conversational capabilities of ChatGPT is its unique ability to build on prior exchanges. Incorporating this feature within NLIs enables the fine-tuning of visualisations. This study showcases how progressive iterations to user prompts can yield more definitive plot elements, transcending the capabilities of prior NL2VIS architectures.

These LLMs were primarily trained English corpora, thus the proficiency of the LLMs in understanding and responding to requests in alternative languages is ultimately dependent on the quantity and quality of training data furnished to them from non-English sources. There is little evidence of NL to visualisation (NL2VIS) studies exploring the capabilities of generating visualisations from queries in languages other than English. A plausible reason for this could be the scarcity of multilingual NL datasets for training contemporary NL2VIS models. The NLI explored in this study demonstrates how the innovative architecture capitalises on the LLMs multilingual skills to interpret requests in mixed languages and render the intended visualisations.

The establishment of benchmarks is imperative to facilitate more rapid advancements in this domain. Nonetheless, evaluating NLIs and their resulting visualisations poses a challenge due to their inherent reliance on subjective human judgement. These manual evaluation approaches are also laborious, requiring teams of assessors and thus present a bottleneck. Aiming for enhanced efficiency and objectivity, this work also introduces a novel methodological framework that automates the benchmarking and evaluation of NLI-produced visualisations. Our study highlights the im-

• *P. Maddigan and T. Susnjak are with the School of Mathematical and Computational Sciences, Massey University, Auckland, New Zealand.*

E-mail: paula.maddigan@gmail.com, t.susnjak@massey.ac.nz

Manuscript received May 12, 2023

portance of ensuring benchmarks can fulfil all the quality characteristics [4] of *reproducibility*, *fairness*, and *verifiability* expected of such baselines. As benchmark datasets begin to emerge in this domain, we also note that it is imperative that they embody a measurement methodology, defining the process to implement the standard, collect measurements, and evaluate the results [4].

1.1 Contribution

This study advances the field of NL2VIS by presenting novel features within the LLM-based NLI Chat2VIS framework. We demonstrate its capacity to comprehend multilingual NL texts to generate data visualisations, establishing it as a truly global tool. We illustrate its remarkable power to incorporate iterative refinements to queries to customise the generated charts and aesthetics. We show with its novel design it is not confined to solely refining a predefined set of chart components as demonstrated by previous NL2VIS approaches.

The current state of benchmarking tools for accurately measuring the capabilities of NL2VIS is still in its infancy. We contribute to existing literature by presenting an innovative automated approach to conducting quantitative analysis of NL2VIS performance. Specifically, we present the results of our evaluation of Chat2VIS against two benchmarks. Our evaluation highlights the challenges of developing robust structured methodologies and measurable baseline standards for NL2VIS.

2 Related Work

Early NL2VIS systems were built on symbolic-based NLP approaches, relying on heuristic algorithms [5], rule-based architectures, and probabilistic grammar-based methods for translating NL queries. Although each technique displayed increasing accuracy, they required more computational resources. Systems such as Articulate [6], DataTone [7], Eviza [8], and Deep-Eye [9] all used varying symbolic NLP methodologies in translating NL to data visualisations. However notable approaches like NL4DV [10] and FlowSense [11] employed NLTK [12], NER, and Stanford CoreNLP [13] semantic parsers to improve accuracy.

Recent advancements in NL2VIS have focused on deep-learning models to achieve greater levels of adaptability, robustness and flexibility compared to that of previous approaches [14]. Systems such as ADVISor [15] are supported by BERT [16], a large language transformer-based model. The rendered visualisation styles are predetermined based on a defined mapping rule.

An alternative transformer-based approach, ncNet [1], is a machine learning model trained using the nvBench [17] dataset. The model accepts an optional chart template in addition to the requested NL query to guide chart styling of the rendered visualisation. The system has recently been expanded to include speech-to-visualisation capabilities [18].

Furthermore, the hybrid approach of RGVisNet [19] initially retrieves the most relevant visualisation query from a large-scale visualisation codebase. It then revises it via a GNN-based deep-learning model, and subsequently generates the visualisation.

The latest state-of-the-art artefact, Chat2VIS [20], presents the first NL2VIS NLI to generate data visualisations via LLMs. It addresses the next generation of NL2VIS architecture, simplifying the NL2VIS pipeline. The underlying structure provides flexibility and robustness around free-form and complex visualisation requests. Decisions pertaining to suitable chart selection and aesthetics are delegated to the LLMs.

The architecture underpinning Chat2VIS is exceptionally flexible and decidedly diverse enough to further refine charting elements using NL without additional enhancements to the NL2VIS architecture. This is the first study to address this gap evident in earlier systems. In addition, unseen in previous approaches, this work demonstrates the art of fulfilling multilingual requests with ease, omitting the need for additional prompting, further architectural manipulations, or model retraining.

With the sparse existence of NL2VIS benchmarks, we seek to evaluate Chat2VIS against the only two baselines nvBench [17] and the NLV utterance corpus [21] identified in the existing literature. Evaluations [19] [21] against these benchmarks for current NL2VIS approaches provide a degree of comparison for this study. To that end, our analysis contributes to the gap distinctly evident in the literature regarding NL2VIS benchmarking.

3 NL2VIS Architecture

Chat2VIS generates data visualisations using free-form NL text. With an interface utilising OpenAI's state-of-the-art LLMs, it demonstrates unique decision-making skills to autonomously select chart types and plot elements.

3.1 Large Language Models

Chat2VIS is based on the Davinci family of models, currently some of the most capable and advanced model set available within the OpenAI suite. It employs GPT-3 model *"text-davinci-003"*, Codex model *"code-davinci-002"*, and contrasts results with the latest state-of-the-art ChatGPT model *"gpt-3.5-turbo"*.

Using OpenAIs text completion endpoint API to access Codex and GPT-3 models, it retains default parameters with the exception of adjustments to the following:

1) Setting of the temperature parameter to zero to encourage the LLMs to be more consistent with their code generation — causing them to use more-common syntax with less creativity;
2) Evading excessively verbose scripts by setting the max_tokens parameter to 500 — an ample limit for this study; and
3) Requesting a stop parameter of *"plt.show()"*. This will cease generation upon plot rendering syntax - avoiding the LLMs presenting alternative scripts.

With the recent release of the official ChatGPT API, it uses the new chat-completion endpoint[1]. Traditionally, prompts submitted to Codex and GPT-3 models are depicted as a succession of "tokens". In contrast, requests to ChatGPT are submitted as a sequence of messages, and

1. https://openai.com/blog/introducing-chatgpt-and-whisper-apis

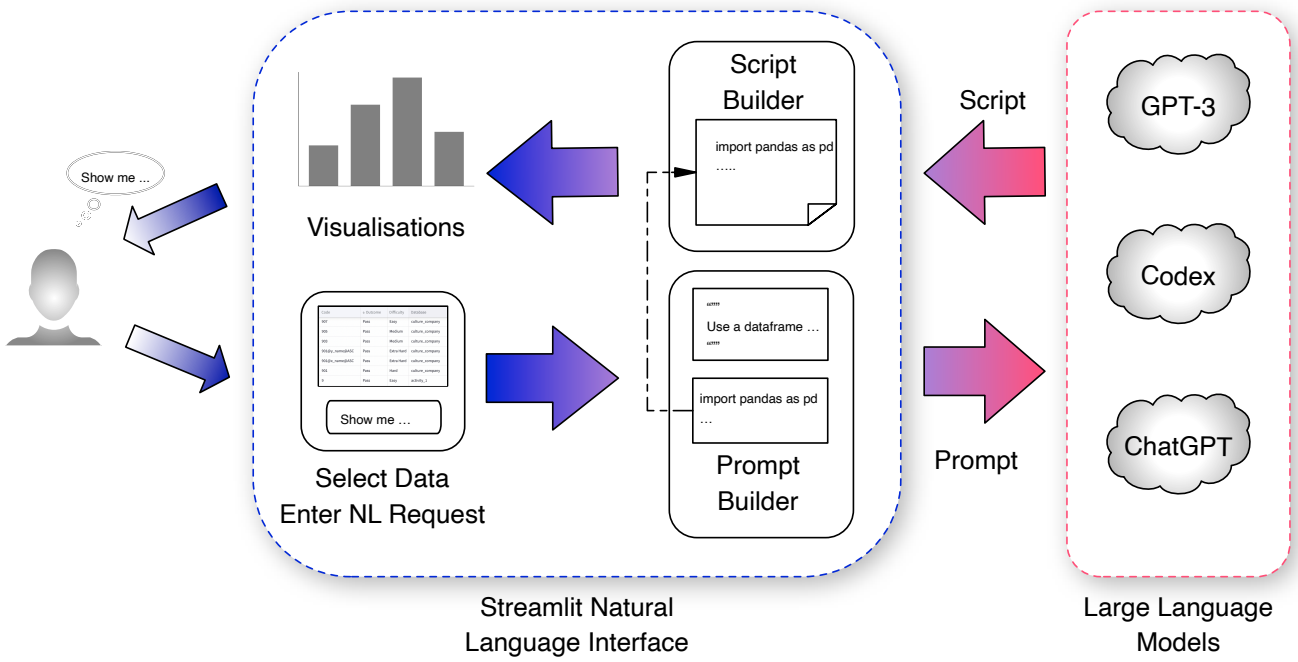

Fig. 1. The Chat2VIS architecture translating NL text into data visualisations via large language models.

subsequently converted to tokens using the new "Chat Markup Language" (ChatML)[2]. Fig. 2 shows the message structure.

```
messages=[
  {"role":"system","content":"Generate Python Code Script."},
  {"role":"user","content":<prompt>}
]
```

Fig. 2. ChatGPT API message structure.

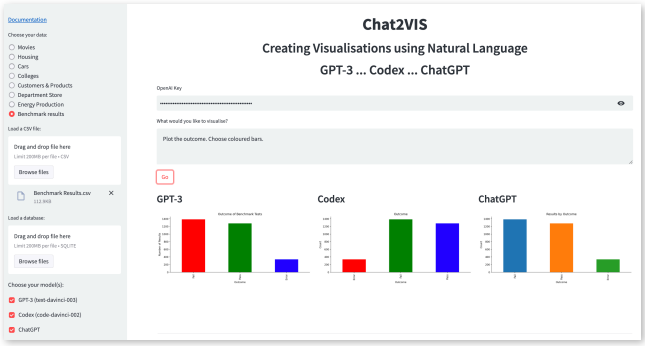

Fig. 3. Chat2VIS Interface

3.2 Chat2VIS

Chat2VIS[3] is built using Streamlit, an open-source Python framework for building web-based applications. The adoption of this technology provides a means to encapsulate the NL2VIS user interface, prompt engineering, LLM connectivity, and rendering of visualisations from generated scripts. Fig. 1 depicts an overview of the architecture of Chat2VIS. Using the interface illustrated in Fig. 3, a user enters a request in the form of NL text in reference to a selected dataset. Chat2VIS engineers the prompt, submits it to the chosen LLMs, formats the returned script, and renders the visualisation.

Fig. 4 illustrates the inner workings of Chat2VIS. The architecture is discussed by way of an example dataset created from the results of our benchmarking evaluation in Section 5.4. The process is described as follows:

1) Fig. 4(a) shows a sample of the dataset together with the query *"Plot the outcome."*.
2) The engineered prompt in Fig. 4(b) comprises of two parts: (1) a Python docstring description, encapsulated with triple double quotes (green); (2) a Python code section providing a starting point for the requested script (blue). The **bolded** type highlights variable substitution values specific to this example.
3) Upon submission of the prompt to the selected LLMs, Fig. 4(c) details the returned script (red) — a continuation of the code section within the engineered prompt.
4) In Fig. 4(d), the prompt code section (blue) is inserted at the beginning of the generated script (red).
5) The newly-created script is executed to render the requested visualisation, as pictured in Fig. 4(e).

2. https://github.com/openai/openai-python/blob/main/chatml.md
3. https://chat2vis.streamlit.app/

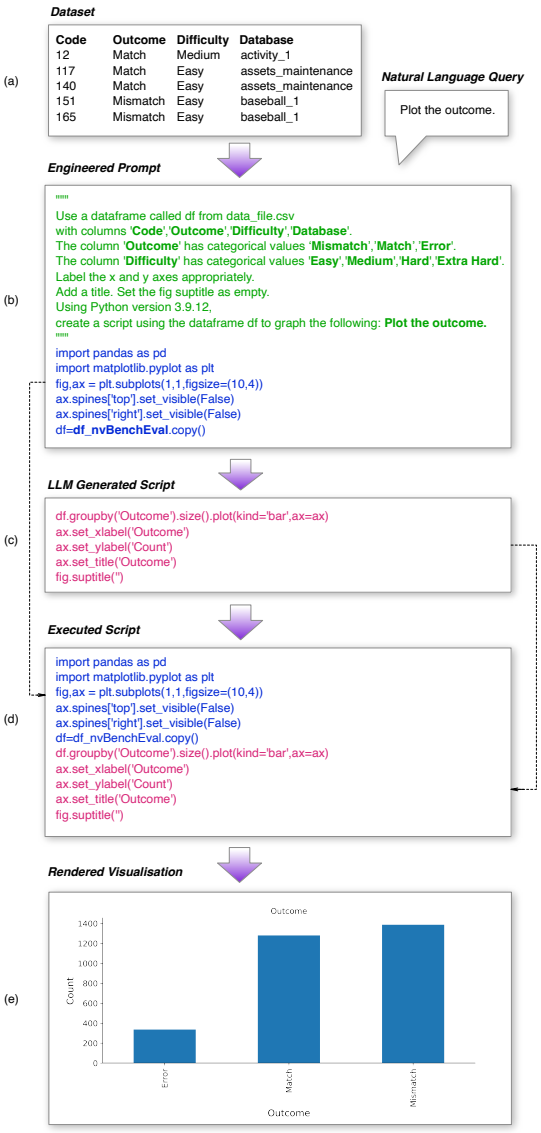

Fig. 4. Illustrated example of the process to convert a NL query into a data visualisation.

4 Methodology

This study explored (1) the refinement of plot aesthetics using NL queries, (2) the flexibility of Chat2VIS in comprehending multilingual requests, and (3) a quantitative evaluation of Chat2VIS against two benchmark datasets developed in previous studies. All figures in this paper are rendered through Chat2VIS to showcase its abilities.

4.1 Chart Refinements and Multilingual Requests

Previous work [20] confirmed the unique decision-making skills of the LLMs to autonomously select suitable chart types and plot aesthetics. Here, we demonstrate the system's efficacy with iterative refinements to the input query for nominating a specific chart type, enhancing plot elements, and changing styles.

We further illustrate the proficiency of the LLMs to interpret multilingual (French, Croatian, Te Reo Māori and German) requests to generate data visualisations and adjust chart labeling.

Three case studies are presented exemplifying these refinements using Chat2VIS. The results are assessed visually for accuracy.

4.2 Quantitative Evaluation

We conduct a more comprehensive quantitative analysis using the nvBench benchmark[4]. Encompassing 153 databases, 7,274 visualisations, and 7 chart types, it is considered the first public large-scale NL2VIS benchmark [17]. Given the size of the dataset, we propose an automated evaluation strategy. Each example instance comprises of a NL-to-visualisation pair, denoted (NL, VIS). Attributes are stored inside a JSON specification, permitting Vega-Lite chart rendering. Examples are further classified into four categories to denote the difficulty of the query — easy, medium, hard, and extra hard.

In addition, we perform a second evaluation using the NLV Utterance dataset[5] [21], referred to in our study as nlvUtterance. The benchmark covered three databases[6]: movies, cars, and superstore. This benchmark comprises 814 NL queries, with 10 visualisations for each database. Queries were generated from the results of an online study using 102 participants suggesting utterances for the display of each respective chart. Here we use a manual evaluation approach.

4.2.1 Model Selection

We select the Codex *"code-davinci-002"* model to measure results against nvBench. Codex, evolved from GPT-3, was trained on an immense amount of publicly-available GitHub code. It is skilled in more than a dozen programming languages, most notably Python, the underlying programming language of Chat2VIS. Codex is available in Davinci or Cushman models. Among the OpenAI suite of models, the Davinci family is the most capable, and can often perform all tasks of other models using fewer instructions. Cushman, although faster, is less competent than Davinci. In prioritising accuracy over speed, the Davinci model was regarded as the most appropriate choice for this task. Therefore, we deemed Codex *"code-davinci-002"* well-suited for this evaluation.

4.2.2 nvBench Benchmark Evaluation

Determining how to assess the equality between a Chat2VIS chart and its nvBench counterpart is challenging. The benchmark specification omits guidance of any evaluation methodology determining what constitutes a match or mismatch. Our attempts to employ image comparison tools proved unreliable due to the complexity involved. Hence we devised a strategy that constructs vectors of the x and y coordinates for each plot, and uses these as a basis for comparative analysis.

Since Chat2VIS is designed to generate charts from a tabular dataset, we removed nvBench instances querying multiple SQL database tables to achieve interoperability. This methodology is consistent with the one used for ncNet

4. https://sites.google.com/view/nvbench
5. https://nlvcorpus.github.io/
6. https://github.com/TsinghuaDatabaseGroup/nvBench/databases.zip

[1]. The exclusion criterion relied on identifying the SQL `JOIN` operator inside the VQL mark within the nvBench JSON specification. In addition, we excluded examples containing SQL subqueries within the `WHERE` clause referencing tables distinct from the principal `SELECT` clause, again, in order to preserve compatibility.

Due to the difficulty in automating accurate comparisons across all chart types, we confined our benchmark testing to bar charts. nvBench generally includes several variants of a query for a given visualisation, and in our methodology, we chose the first option. Fig. 5 illustrates an example JSON specification[7] for the (VIS, NL) pair *"474@x_name@DESC"*, highlighting areas of interest within the specification discussed in this evaluation approach. The final benchmark test set across 138 databases comprised 3,003 instances, with 812 considered *easy*, 1572 *medium*, 386 *hard*, and 233 *extra hard*.

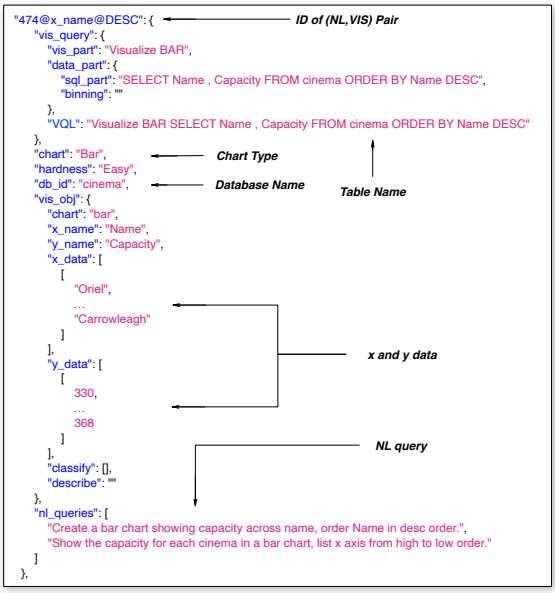

Fig. 5. Example JSON specification from nvBench.

Fig. 6 summarises our proposed automated testing methodology. The steps outlining the methodology are described below:

1) Select a JSON test specification from nvBench.
2) Extract the database name from the db_id field, and the table name specified in the VQL field following the FROM keyword.
3) Import the SQLite database table into a Python Pandas DataFrame structure.
4) Extract the first query from the nl_queries field.
5) Submit query and dataset to Chat2VIS opting for Codex, and render visualisation.
6) Document the outcome as an Error should the code fail to execute.
7) Construct x and y data coordinate vectors by extracting the Chat2VIS chart elements.
8) Construct x and y data coordinate vectors using the x_data and y_data fields in the nvBench JSON specification.
9) Apply adjustments to the vectors to address complications impacting a successful comparison:
   - Ensure naming consistency of calendar units, such as recasting Tue, Tues, and tuesday as Tuesday; Sept, Sep september as September.
   - Sort by ascending y values if keywords "sort" and "order" are not specified.
   - Cast integer values to floats, and round all floats to 5dp.
10) Compare Chat2VIS and nvBench vectors. A precise match marks the outcome as a "Match", else it is marked as a "Mismatch", while, an "Error" indicates that a visualisation failed to be rendered most-frequently attributed to a Python code error.

It should be noted that a "Mismatch" does not necessarily mean that the generated visualisation is incorrect, and may in some instances even be a more effective and appropriate visualisation (see Figure 11 in Appendix A).

4.2.3 nlvUtterance Benchmark Evaluation

A manual visual inspection of the results on the nlvUtterance dataset was used due to its smaller size. All chart types within nlvUtterance were used, namely, bar charts, histograms, line charts, and scatter plots, together with their variations. As outlined in the benchmark description [21]: histograms and single attribute bar charts are used to visualise one categorical or quantitative attribute; bar charts, scatter plots, and line charts for two attributes; and grouped bar charts, stacked bar charts, multi-line charts, coloured scatter, and faceted scatter charts for visualising three or more attributes. 755 of the 814 queries are considered *"singleton"* utterance sets, consisting of a single query request. The remaining 59 instances are considered *"sequential"* utterance sets, containing multiple utterances. After removing erroneous plots without data, the final dataset consisted of 758 queries for testing.

Chat2VIS renders charts for up to 3 models. We employed a three-stage testing methodology. Firstly, the queries are submitted to Codex and the corresponding performance metrics are presented. Secondly, unsuccessful queries are submitted to GPT-3, with the corresponding performance again measured. Finally, any remaining mismatched queries are submitted to ChatGPT. The overall performance statistic for successful matches provides insight into the likelihood that a benchmark result will be generated by at least one LLM.

It is not the intention of this work to compare LLMs *inter se*, but instead contrast the use of LLMs with alternative approaches. Therefore, we do not present benchmark metrics comparing the performance accuracy of Codex, GPT-3 and ChatGPT relative to each other.

5 Results

We demonstrate the conversational ability to fine-tune charts with subsequent requests on the first two case studies,

7. https://github.com/TsinghuaDatabaseGroup/nvBench/blob/main/NVBench.json

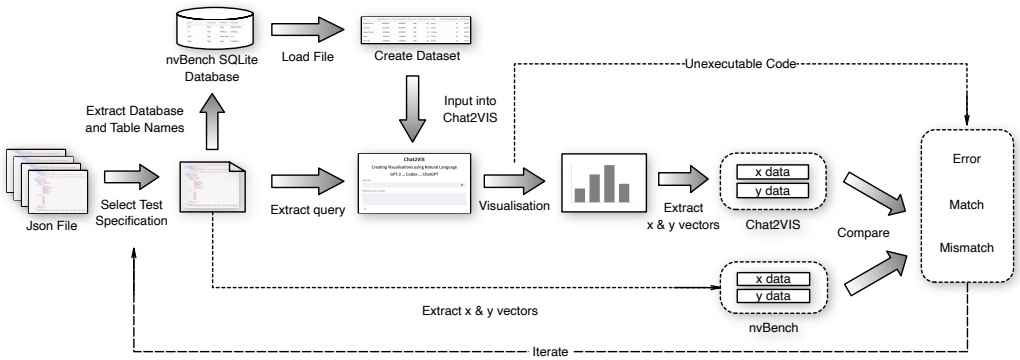

Fig. 6. Overview of nvBench Benchmark Testing.

followed by the third case study illustrates multilingual requests. The dataset used for the illustration are results from the nvBench evaluations. Finally, we analyse Chat2VIS' performance against the two benchmarks using the illustrations generated by Chat2VIS.

5.1 Case Study 1: Conversational Fine-tuning

Fig. 7 demonstrates the first conversational refinement of Fig. 4(e) *"Plot the outcome."* to *"Plot the outcome by difficulty."*. The subsequent fine-tuning requests that the chart be rendered *" as a stacked bar plot. Use red for error, light green for match, blue for mismatch."*, followed by a request to refine it with the instruction to *Increase the font size of the axis labels and numbers. Make the title 'Evaluation Results by Difficulty Level' with very large font.*. The figures demonstrate that all LLMs interpreted the requests reliably while also demonstrating differences in how they interpreted vague font size requests.

5.2 Case Study 2: Conversational Fine-tuning

Again, the base query *"Plot the outcome."* from the dataset example in Fig. 4(e) serves as a starting point. Illustrations in Fig. 8 demonstrate refining the query with the subsequent instruction *...using a pie chart. Hide axis labels. Use pastel colours.*. Results show a high accuracy across all LLMs. Exceptions are to the instruction to use pastel colours which was followed by all LLMs except GPT-3, while the instruction to remove the axis label *"outcome"* was adhered to by both GPT-3 and ChatGPT, but ignored by Codex.

5.3 Case Study 3: Multilingual Requests

The capability of the LLMs to comprehend mixed multilingual requests for conversational chart fine-tuning is demonstrated here and depicted in Fig. 9, building on the initial prompt *"Plot the outcome."* from Fig. 4(e) to plot results categorised by difficulty. The fine-tuning requests that the *outcome* be plotted along the x-axis using French *"Regroupez la difficulté par résultat sous forme de graphique à barres. l'axe des x est le résultat."*, translated[8] as *"Group difficulty by outcome as a bar chart. The x axis is the result."*. Subsequently, the plot is refined by asking for the title "Benchmark Results" to be altered using Croatian *"Promijenite naslov u 'Rezultati benchmarka'."*. Each LLM correctly rendered the refinements, while retaining both legend and bar labels in English. Meanwhile, GPT-3 and Codex translated both axes' labels into Croatian, while ChatGPT preserved the English labelling.

The subsequent fine tuning can be seen in the next plot in Fig. 9 where an instruction is issued in English *...Write the plot labels in German. Enlarge label size to 20. Enlarge axis to 18. Make title 24 font.*. Next, a change in colour ordering of plot bars is issued in a (New Zealand's Te Reo Māori) with the phrase *"Whakamahia nga tae whero, karaka, kākāriki, kikorangi."*, meaning that the colours red, orange, green and blue be used. All 3 LLMs changed font sizes of the requested elements, with ChatGPT also increasing tick label sizes. Both GPT-3 and ChatGPT provided German labels for the title and axes as requested, however, Codex reverted to English labelling. GPT-3 and Codex correctly interpreted the colouring request, with ChatGPT retaining its original colour ordering scheme. Overall, the example demonstrated high-fidelity results with some variability across different LLMs.

5.4 Evaluation against nvBench

When evaluated against nvBench, Chat2VIS demonstrates significant potential. Out of the 3,003 queries tested, 1,280 charts showed an exact match. This 43% "Match" rate, as illustrated in Fig. 4(e), is a notable achievement considering both the narrow and strict conditions used to define a match and the overall complexities involved in data visualisation tasks. Fig. 7 presents summary statistics of the results by difficulty level, further emphasising the effectiveness of the system under various scenarios. Overall only 336 (11%) instances represented charts which could not be rendered, primarily due to erroneous code generated by the LLMs. A detailed examination of both the nature of successful and mismatched instances where our analysis has enabled the identification of certain conditions under which the system performs particularly well is given next.

5.4.1 nvBench Benchmark Matches

Chat2VIS excels in producing exact matches to the nvBench benchmark in a number of specific scenarios, as outlined below:

8. via Google translate https://translate.google.com

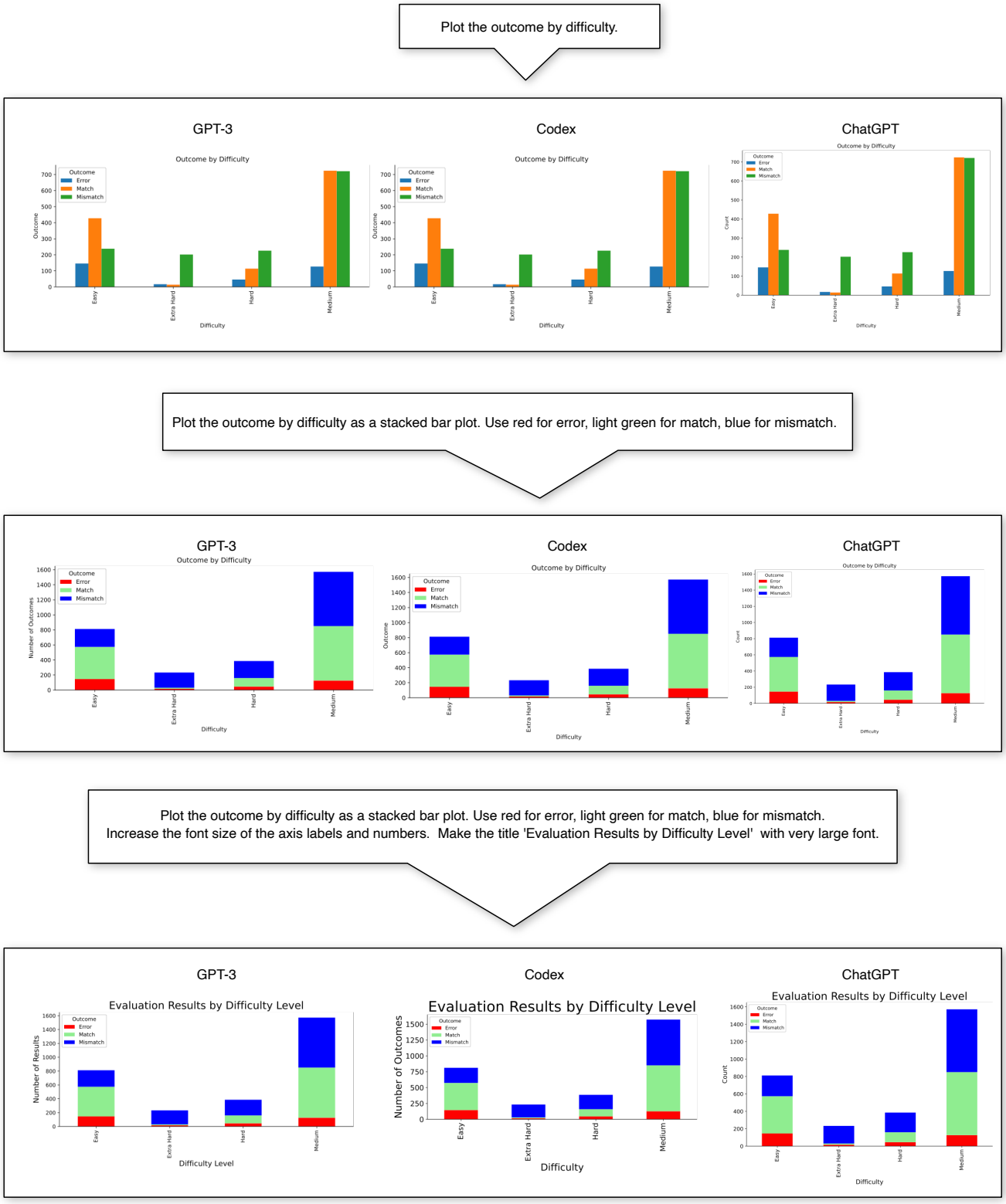

Fig. 7. Case Study 1: Conversational Visualisation Refinements using the nvBench Evaluation Results by Difficulty

- When the plot type is explicitly specified in the query, the system excels in delivering accurate visualisations.
- Chat2VIS demonstrates a high degree of accuracy when grouping information, such as 'by Weekday', 'by Month', etc., is explicitly provided in the query.
- If the query is unambiguous and specific, it greatly increases the chances of Chat2VIS providing the cor-

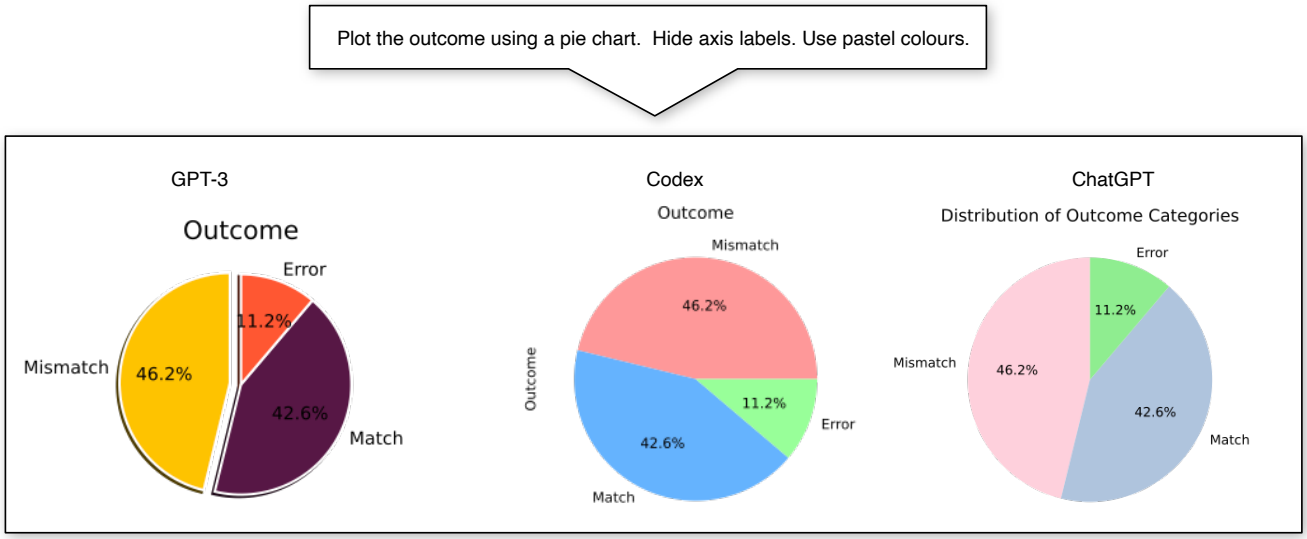

Fig. 8. Case Study 2: Conversational Visualisation Refinements using the nvBench Evaluation Results by Outcome

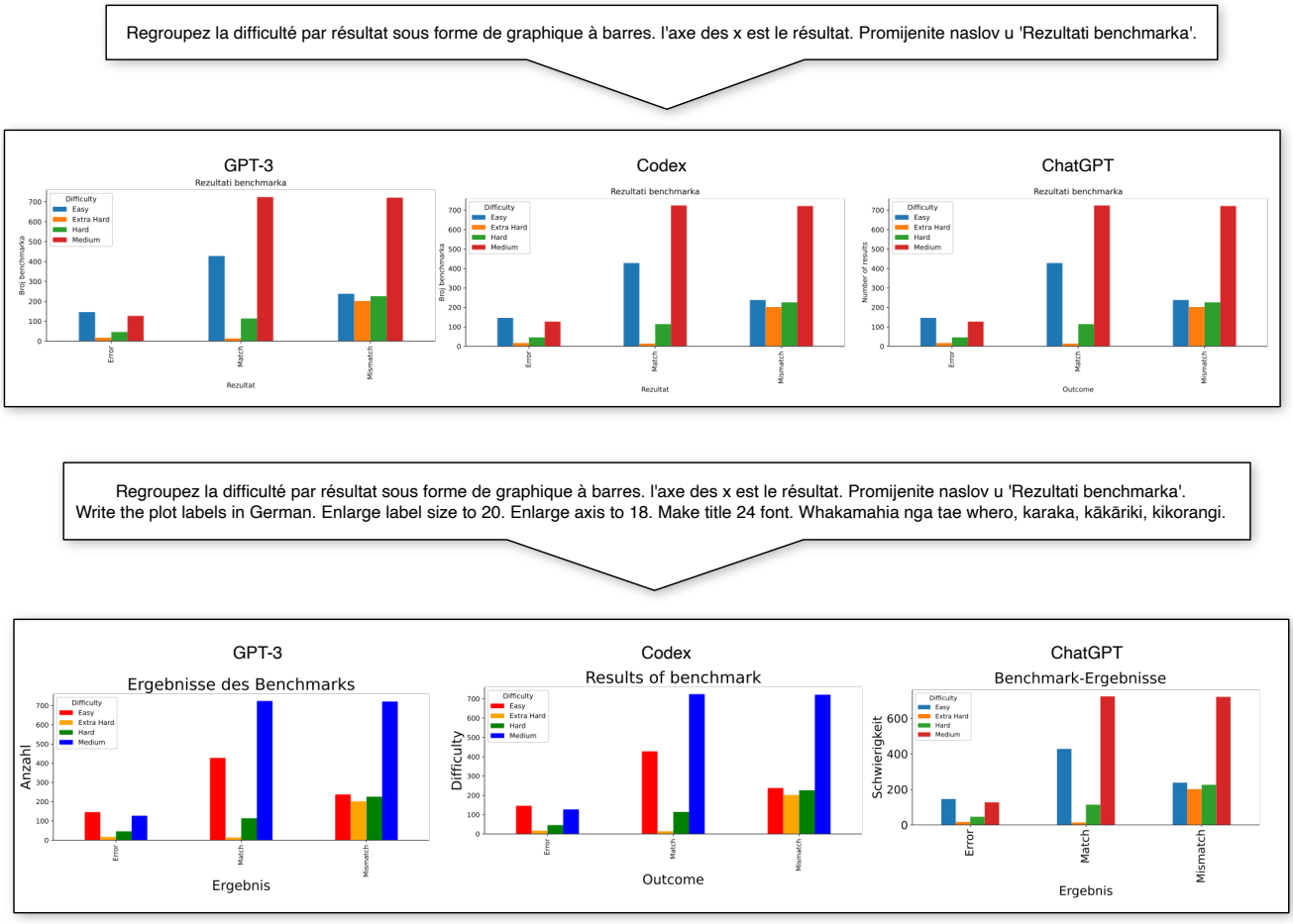

Fig. 9. Case Study 3: Conversational Visualisation Refinements using the nvBench Results Data via Multilingual Requests.

rect visualisation.

- When there is a request for ordering or sorting within the query, Chat2VIS successfully handles this, provided that the request is explicit, and all categories

have different numerical values, ensuring matching sorting.

- The system performs well when the benchmark plot also accurately represents the information in the database and correctly interprets the query request.
- When there was the presence of non-zero values in the plotted data, this tended to positively impact Chat2VIS' performance.
- When the database structure itself correctly represents the data types it holds, and null values are represented in line with database standards, the system can more accurately produce the desired visualisations.
- Lastly, Chat2VIS system performs optimally when the plot type specified in the query aligns with the nature of the requested data.

5.4.2 Benchmark Mismatches

Here we summarise broad categories under which mismatches were detected with the benchmark. Many are a result of applying too narrow a definition of a match when devising an automated system which does not have subjective capabilities, while a large percentage of the mismatches can be attributed to deficiencies within the benchmark. We observed instances where the Chat2VIS chart was 'correct' in its rendering, however, errors within the nvBench specification and the inconsistency within the database information resulted in the mismatch classification. We illustrate in Fig. 11(a) a discrepancy in categorical values with variant spellings of *"Andreo"* and *"Lo"* in the nvBench specification contrasted with *"Andreou"* and *"Lou"* persisted in the database. We noted the presence of at least five nvBench specifications omitting the first query. On passing an empty string to Chat2VIS, the LLM demonstrates its effective decision-making to render a visualisation of interest, as shown in 11(b).

In a subset of examples, we found inconsistencies between the query request and the SQL denoted in the nvBench specification. Our findings unearthed instances where the SQL included an `ORDER BY` clause but the NL query omitted instructions to *"order"* or *"sort"* results. Consequently, the Chat2VIS visualisation accurately depicted the data points but was classified as mismatched with that of nvBench's sorted chart. These examples illustrate repeatedly that a mismatch in a strict sense when using automated approaches is not necessarily an incorrect output.

Our methodology utilises only the query component of the specification, disregarding other instructions contained within. Consequently an assortment of examples stored requests for grouping results inside the "binning" mark of the specification, therefore omitting to incorporate the request as part of the NL query. We illustrate in Fig. 11(c) a common mistake where the Chat2VIS chart summarised by date, and the equivalent nvBench specification grouped by weekday, an instruction solely provided within the "binning" mark.

Further inspection of mismatches showed periodically, without additional stipulations inside the specification, nvBench incorrectly renders only a partial result set, while Chat2VIS visualises the complete set of data. An example is depicted in Fig. 11(d). The reasoning for such behaviour within nvBench is unknown. Additionally, we noted for some instances executing the SQL query from the nvBench specification directly against the database yields conflicting figures to those visualised by nvBench, as shown in Fig. 11(e). Furthermore, we encountered truncating of numeric float types to integer values without additional stipulations inside the specification as shown in Fig 11(f). Consequently causing additional mismatches between nvBench and Chat2VIS results.

5.4.3 Methodology Limitations

Our approach to automated benchmarking against nvBench, in the absence of a standardised methodology, presents some limitations. Occasionally, Chat2VIS generates visually accurate charts but the proposed comparison methodology does not always yield expected results. Differences in the treatment of categories devoid of data (Chat2VIS omits, nvBench assigns zero) lead to mismatches, despite similar visual outcomes, as shown in Fig. 11(g).

The nvBench queries often specify the ordering or sorting of results based on a nominated axis, either in ascending or descending order. As we do not rearrange the x and y vectors if sorting is requested, an exact match is necessary for the charts to be equivalent. However, this approach presents a challenge when multiple x values have identical y values, rendering a correctly ordered, but not identical, chart. We illustrate this assorted ordering in Fig. 11(h) with the majority of bars having value 1.

Occasionally, LLM's unconventional plotting arising from Python's diverse charting techniques, may lead to unexpected outcomes for x and y vectors, resulting in null values at some points of the chart resulting in mismatches while not necessarily being incorrect.

We noted that numeric fields in the nvBench database occasionally contain missing values represented as empty strings instead of NULL values, causing issues with Python's data import and query execution and mismatches. Enhancing the Chat2VIS interface could mitigate this.

Similarly, columns holding numeric data but defined as character type could result in incorrect mathematical computations. Sometimes, older versions of Python syntax generated by the LLM and verbose scripts exceeding token restrictions also led to execution errors.

5.4.4 Ambiguity

Query text ambiguity led to different interpretations and charts by the LLM and nvBench making genuine accuracy assessment challenging. Queries like *"...sort bars in desc order"* or *"...order by the bars from high to low"* were interpreted by Chat2VIS as sorting by y-axis values, whereas nvBench sorted x-axis labels alphabetically, as shown in Fig. 11(i).

Even though bar charts are typically used for comparing group values, we observed instances where they were used to represent two text columns. Fig. 11(j) shows nvBench's approach to this situation, while Chat2VIS inferred a different but more appropriate visualisation, albeit aesthetically unappealing, providing category counts.

5.4.5 Query Misinterpretation

Occasionally, the LLM misinterpreted queries, causing Chat2VIS to generate incorrect visualisations. Date and time

values were challenging. While nvBench correctly used only the date component for grouping, Chat2VIS sometimes considered both date and time, leading to individual groupings of date-times.

The nvBench construction methodology often generated similar benchmark examples. Filter requests such as *"...commission is not null or department number does not equal to 40..."* were present in over 70 instances. Chat2VIS incorrectly used the *"and"* operator instead of *"or"* when interpreting this filter, contributing to a disproportionate amount of mismatched results.

We also noted instances where Chat2VIS erroneously self-imposed a limit on the number of returned results and also misunderstood a Chinese language request. These incidents highlight the LLM's occasional misinterpretations, despite its general proficiency in generating Python code from natural language.

5.5 Evaluation against nlvUtterance

Chat2VIS demonstrates robust results against the nlvUtterance benchmark with respect to the strict methodology used. 50% match rate was observed over all chart types when using Codex. This rate improved to 63% when either Codex or GPT-3 produced a matching chart. Meanwhile, the matching rate rose to 72% when at least one of Codex, GPT-3, or ChatGPT generated a matching chart. Fig. 10 presents the evaluation results, categorized by chart type for each stage of testing. Again, it is worth noting that not all mismatches between Chat2VIS and nlvUtterance generated charts imply inaccuracies. The ambiguity within the query and lack of charting specifications often led to alternative yet 'correct' visualisations, which, due to the lack of defined evaluation guidelines and parameters within the benchmark, were sometimes deemed as mismatches in our objective evaluation.

5.5.1 nlvUtterance Benchmark Matches

Chat2VIS exhibited a high degree of matches under several conditions. Firstly, when single-attribute bar plots and scatter plots were utilised, the system generated matches. These types of plots had the highest representation in the dataset, and the system managed to match them at a high rate. Evaluations with Codex, shown in Fig. 10(a), demonstrated that coloured scatter plots and faceted scatter plots had significantly lower matching rates. However, when generated via GPT-3 (Fig. 10(b)) or ChatGPT (Fig. 10(c)), the results were much more favourable. Secondly, the system showed high matching rates with bar charts when generated via the three LLMs.

5.5.2 nlvUtterance Benchmark Mismatches

Nonetheless, our evaluation also revealed instances where the system struggled, particularly with grouped and stacked bar charts, histograms, coloured and faceted scatter charts, and single and multi-line charts. Fig. 12 presents a sample of generated visualisations for the ten chart types based on the movies dataset, enabling comparisons with those presented in prior work [21]. Specific chart types are detailed below with respect to the generation of mismatches.

Bar Charts: A substantial number of grouped bar charts were classified as a mismatch notwithstanding correct charting of the requested data. Considering the query *"average production budget by creative type and content rating"*, the benchmark arranged results grouped by content rating using colour coding to represent the creative type. However, periodically the Chat2VIS counterpart inversely grouped results by creative type with colours representing content rating. Similar issues were observed with mismatches between stacked bar charts. Furthermore, our findings showed instances where benchmark stacked bar charts were presented by Chat2VIS as grouped bar charts, accurately conveying the information, but not in accordance to the benchmark standard. Had the evaluation methodology deemed these visualisations as matches, as indeed they were despite deviating from the benchmark, the accuracy (match) rate would increase significantly. Nonetheless, the absence of methodological instructions within the benchmark failed to provide guidance in such circumstances.

Histograms: We refrained from providing explicit instructions to the LLM on which chart type to render. Consequently a significant number of benchmark histograms were instead plotted as bar charts. Queries such as *"How many orders were placed for each order quantity?"* and *"show me a bar chart of count by order quantity"* did not imply the data should be represented as a histogram, and hence the LLM decided the most appropriate representation of the data was in the form of a bar chart. Furthermore, queries neglected to provide information pertaining to binning size, and consequently the LLM's decision often conflicted with benchmark visualisations.

Scatter Charts: Colour scatter charts use varied colours to represent categories in a single plot. Codex often overlooked this colour coding leading to single-coloured charts. GPT-3 did not share this limitation, but both exhibited a high percentage of mismatches due to incorrect syntax, often incorrectly setting the *"c"* colour parameter value in the Python plotting function. This misstep resulted in erroneous script execution. ChatGPT was more successful, correctly assigning this function parameter. Faceted scatter charts separate categories from coloured scatter plots into distinct charts. Some queries lacked clear instructions for a faceted chart, prompting us to accept single scatter plots categorised by colour. As in the case of coloured scatter charts, Codex often failed to use colour coding to distinguish categories, while GPT-3 and ChatGPT did not have this limitation. However, due to the lack of a benchmark methodology, we accepted alternative charts, which could otherwise have affected the success rate.

Line Charts: The least-represented chart types in the dataset are single and multi-line plots. The most common cause of mismatch was the LLM selecting to render the information as a bar chart. However, although it still accurately presented the requested information when a line chart was not explicitly requested, it was not in accordance with benchmark specifications. In addition, Chat2VIS multi-line plots on occasion inversely rendered the x-axis and line colour categories compared to that of nlvUtterance, hence unsuccessful in meeting benchmark standards. Once more, these decisions of determining if benchmark standards are met significantly impact culminating a successful outcome.

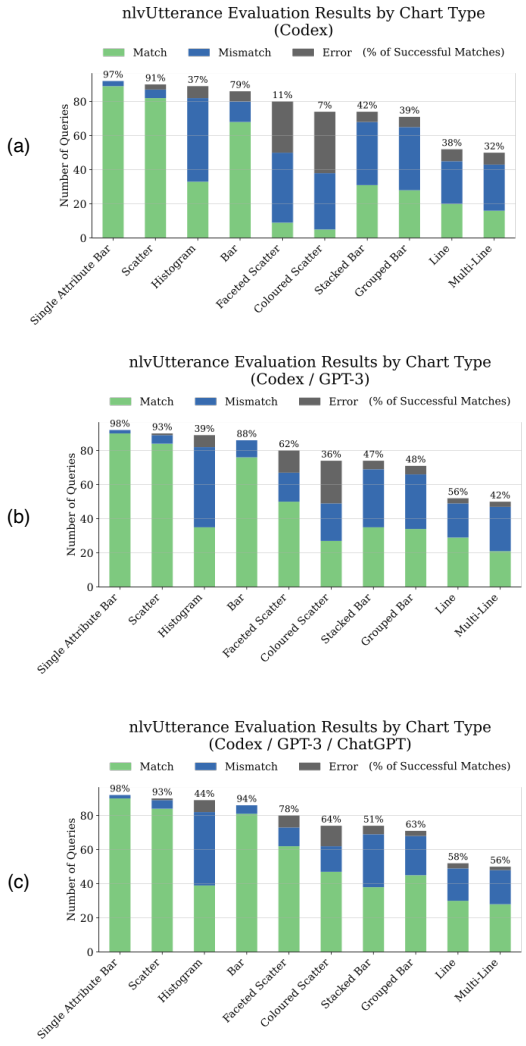

Fig. 10. nlvUtterance Evaluation Results.

5.6 Comparison With Previous Work

Finally, Chat2VIS is contrasted with results from prior studies' on the nvBench benchmark. The comparisons are suggestive but are not exact since our nvBench sample set differs from test set used in previous studies. The overall accuracy of state-of-the-art NL2VIS systems are shown in Table 1 as reported by [19] and contrasted with Chat2VIS, indicating a highly competitive performance of the proposed system.

TABLE 1
nvBench Performance Comparison.

System	Accuracy
Seq2Vis [22]	2%
Transformer [23]	3%
ncNet [1]	26%
RGVisNet [19]	45%
Chat2VIS	**43%**

Table 2 summarises our findings using the nlvUtterance benchmark, with those from NL4DV evaluated on the benchmark in previous studies [21]. It should be noted the NL4DV results are based on a 755 instance dataset pertaining to singleton query sets only. In our study we included all query sets, with the exclusion of those pertaining to the two omitted line charts.

TABLE 2
nlvUtterance Performance Comparison.

System	Accuracy
Chat2VIS (Codex)	50%
Chat2VIS (Codex/GPT-3)	63%
Chat2VIS (Codex/GPT-3/ChatGPT)	72%
NL4DV	64%

6 Discussion

The experiments confirm the advanced capability of Chat2VIS to provide a state-of-the-art solution to the NL2VIS problem, exemplifying the conversational ability to fine-tune chart aesthetics and to do so with multilingual instructions.

Furthermore, we provided results of Chat2VIS against two benchmarking datasets and confirmed the state-of-the-art properties of the proposed system. Given the time-consuming and subjective nature of performing these necessary evaluations, we made a contribution towards the development of automated approaches to help researchers and accelerate advancements in this field. Our proposed automation methodology is a first step in realising this goal and has yielded findings that are helpful in advancing the refinement of existing benchmarks and the development of new ones.

While we gratefully applaud the enormous efforts invested by researchers in developing the existing benchmark datasets, we find that there is room for improvement in fulfilling all the necessary quality characteristics like *reproducibility*, *fairness*, and *verifiability* as defined by [4]. The large number of diverse visualisation elements, aesthetics, and chart styles, from a variety of available programming language libraries raises difficulties in generating *reproducible* and measurable outcomes from NL queries. The predominant use of Vega-lite specifications in current benchmarking studies [21] [17] periodically separates important visualisation information from the NL query, limiting the effectiveness of alternative NL2VIS architectures and reducing *fairness*. Inconsistencies exist between test case data and visualisation outcomes thus compromising *verifiability*.

In future, to establish robust benchmarks for NL2VIS, we foresee definitions of a collection of valid visualisations for each NL query accompanied by a comprehensive methodology for chart comparison and evaluation. The evaluation would be independent of charting frameworks.

7 Conclusion

This seminal study presents the novel features of Chat2VIS for converting natural language into data visualisations in a conversational manner with the ability to fine-tune charts in multiple languages, addressing a previously unsolved research problem.

We demonstrated the capabilities of our system against two benchmarks and have proposed a novel approach for

automating the evaluation and comparison of generated visualisations thus contributing towards an additional gap in literature. We explored the challenges in accomplishing this and have identified areas for improvement in the development of benchmark datasets in the field of NL2VIS in order to accelerate the development of future advancements.

REFERENCES

[1] Y. Luo, N. Tang, G. Li, J. Tang, C. Chai, and X. Qin, "Natural language to visualization by neural machine translation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 1, pp. 217–226, 2021.
[2] L. Shen, E. Shen, Y. Luo, X. Yang, X. Hu, X. Zhang, Z. Tai, and J. Wang, "Towards natural language interfaces for data visualization: A survey," *arXiv preprint arXiv:2109.03506*, 2021.
[3] Y. Wang, Z. Hou, L. Shen, T. Wu, J. Wang, H. Huang, H. Zhang, and D. Zhang, "Towards natural language-based visualization authoring," *IEEE Transactions on Visualization and Computer Graphics*, vol. 29, no. 1, pp. 1222–1232, 2022.
[4] S. Kounev, K.-D. Lange, and J. von Kistowski, *Systems Benchmarking: For Scientists and Engineers*. Springer, 2020.
[5] G. Liu, X. Li, J. Wang, M. Sun, and P. Li, "Extracting knowledge from web text with monte carlo tree search," in *Proceedings of The Web Conference 2020*, 2020, pp. 2585–2591.
[6] Y. Sun, J. Leigh, A. Johnson, and S. Lee, "Articulate: A semi-automated model for translating natural language queries into meaningful visualizations," in *Smart Graphics: 10th International Symposium on Smart Graphics, Banff, Canada, June 24-26, 2010 Proceedings 10*. Springer, 2010, pp. 184–195.
[7] T. Gao, M. Dontcheva, E. Adar, Z. Liu, and K. G. Karahalios, "Datatone: Managing ambiguity in natural language interfaces for data visualization," in *Proceedings of the 28th annual acm symposium on user interface software & technology*, 2015, pp. 489–500.
[8] V. Setlur, S. E. Battersby, M. Tory, R. Gossweiler, and A. X. Chang, "Eviza: A natural language interface for visual analysis," in *Proceedings of the 29th annual symposium on user interface software and technology*, 2016, pp. 365–377.
[9] X. Qin, Y. Luo, N. Tang, and G. Li, "Deepeye: Visualizing your data by keyword search." in *EDBT*, 2018, pp. 441–444.
[10] A. Narechania, A. Srinivasan, and J. Stasko, "Nl4dv: A toolkit for generating analytic specifications for data visualization from natural language queries," *IEEE Transactions on Visualization and Computer Graphics*, vol. 27, no. 2, pp. 369–379, 2020.
[11] B. Yu and C. T. Silva, "Flowsense: A natural language interface for visual data exploration within a dataflow system," *IEEE trans. on visualization and computer graphics*, vol. 26, no. 1, pp. 1–11, 2019.
[12] E. Loper and S. Bird, "Nltk: The natural language toolkit," *arXiv preprint cs/0205028*, 2002.
[13] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The stanford corenlp natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55–60.
[14] H. Voigt, M. Meuschke, K. Lawonn, and S. Zarrieß, "Challenges in designing natural language interfaces for complex visual models," in *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, 2021, pp. 66–73.
[15] C. Liu, Y. Han, R. Jiang, and X. Yuan, "Advisor: Automatic visualization answer for natural-language question on tabular data," in *2021 IEEE 14th Pacific Visualization Symposium (PacificVis)*. IEEE, 2021, pp. 11–20.
[16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol. 1, 2018, pp. 4171–4186.
[17] Y. Luo, J. Tang, and G. Li, "nvbench: A large-scale synthesized dataset for cross-domain natural language to visualization task," *arXiv preprint arXiv:2112.12926*, 2021.
[18] J. Tang, Y. Luo, M. Ouzzani, G. Li, and H. Chen, "Sevi: Speech-to-visualization through neural machine translation," in *Proc. of the 2022 Int. Conf. on Management of Data*, 2022, pp. 2353–2356.
[19] Y. Song, X. Zhao, R. C.-W. Wong, and D. Jiang, "Rgvisnet: A hybrid retrieval-generation neural framework towards automatic data visualization generation," in *Proc. of the 28th ACM SIGKDD Conf. on Knowledge Discovery and Data Mining*, 2022, pp. 1646–1655.
[20] P. Maddigan and T. Susnjak, "Chat2vis: Generating data visualisations via natural language using chatgpt, codex and gpt-3 large language models," *IEEE Access*, pp. 1–1, 2023.
[21] A. Srinivasan, N. Nyapathy, B. Lee, S. M. Drucker, and J. Stasko, "Collecting and characterizing natural language utterances for specifying data visualizations," in *CHI '21: Proc. of the 2021 CHI Conf. on Human Factors in Computing Systems*, 2021, p. 1–10.
[22] Y. Luo, N. Tang, G. Li, C. Chai, W. Li, and X. Qin, "Synthesizing natural language to visualization (nl2vis) benchmarks from nl2sql benchmarks," in *Proceedings of the 2021 International Conference on Management of Data, SIGMOD Conference 2021, June 20–25, 2021, Virtual Event, China*. ACM, 2021.
[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Info. Processing Syst.*, pp. 6000–6010, 2017.

APPENDIX A
BENCHMARK EXAMPLES

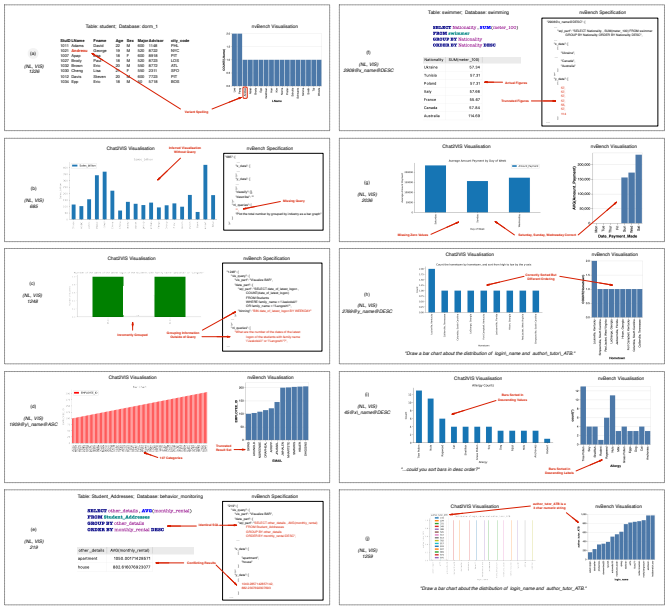

Fig. 11. Examples of Mismatched Visualisations Between Chat2VIS and nvBench

Fig. 12. nlvUtterance Movies Dataset Examples from Chat2VIS