



UHASSELT

KNOWLEDGE IN ACTION

Faculty of Business Economics

Master of Management

Master's thesis

Evaluation of modern tools for data visualization

Olsi Qatipi

Thesis presented in fulfillment of the requirements for the degree of Master of Management, specialization Data Science

SUPERVISOR :

Prof. dr. Koenraad VANHOOF

MENTOR :

De heer Maarten VANHOOF



UHASSELT

KNOWLEDGE IN ACTION

www.uhasselt.be
Universiteit Hasselt
Campus Hasselt:
Martelarenlaan 42 | 3500 Hasselt
Campus Diepenbeek:
Agoralaan Gebouw D | 3590 Diepenbeek

2023
2024



Faculty of Business Economics

Master of Management

Master's thesis

Evaluation of modern tools for data visualization

Olsi Qatipi

Thesis presented in fulfillment of the requirements for the degree of Master of Management, specialization Data Science

SUPERVISOR :

Prof. dr. Koenraad VANHOOF

MENTOR :

De heer Maarten VANHOOF

Table of Contents

1. ABSTRACT	1
2. LITERATURE REVIEW	2
2.1 Introduction	2
2.2 The evolution of NLP for visualization (NL2VIS)	2
2.2.1 Datatone (2015)	5
2.2.2 ADVisor (2021)	5
2.2.3 BERT vs GPT	5
2.2.4 Chat2Vis (2023)	6
2.2.5 LIDA	7
2.3 Prompt Engineering	7
2.3.1 Overview of Prompt Engineering	7
2.3.2 Prompt Engineering can also be done by users	8
2.3.3 Examples of Prompt Engineering in visualization tasks	8
2.4 Current market (usage) of visualization tools	9
2.5 Summary of Literature review	10
2.6 What comes next?	10
3. RESEARCH QUESTIONS	11
4. METHODOLOGY	12
5. RESULTS	15
5.1 Time recordings	15
5.2 Accuracy of the visualizations	18
5.3 User-Friendliness	26
6. DISCUSSION	28
7. REFERENCES	31
8. APPENDICES	35

1. Abstract

Since I want to start this thesis with something like, "In today's world every company seeks to make data driven decisions, which enhances even more the need for precise and modern tools for visualization," it is difficult for me to begin without running the risk of sounding repetitive.

Until now, I have encountered similar statements in the beginning of the articles I have been reading, but the idea for this work came from a specific one. Maddigan, P., & Susnjak, T. (2023), write about their tool Chat2Vis, which generates visuals from natural language. One of the possibilities for future work that they mention is the need for end-user evaluation of Chat2Vis.

So here is the main idea of the thesis. How can a user such as myself, a Master of Management student, with a background in finance, who is not a programmer, work with tools such as Chat2Vis (a Language to Visualization model), compared with the ones they normally use (i.e. Power Bi)? I will include in my paper also other tools, which stand as alternatives to Chat2Vis.

After diving into the documentation of the above mentioned tools, I will be presenting some theoretical background which will help me understand better and try to explain the way they work and the logic behind them.

Then, I'll use free databases to conduct practical tests, present my findings, and have a conversation about the things I discovered and others that still require research. The tests will be focused on comparing the performance of innovative NL2VIS tools (Chat2Vis) with commonly used tools (Power Bi), in terms of accuracy and user-friendliness. The final goal of the study would be to contribute to the study of NL2VIS tools, offering practical user experience and provide insights if they can become the new mainstream tools used from businesses for data visualization.

2. Literature review

2.1 Introduction

Developments in recent years have made it possible to create data visualizations directly from natural language. According to research, users find it simpler to have their visualizations derived from inputting a sentence than navigating on the menus, selecting the data and then the most appropriate chart (Wang Y. et al., 2022). This introductory part will provide a general overview of the technology behind natural language to visualizations process, the key concepts, as well as some of the main tools used. These tools will be also evaluated in more detail in the following part of the paper.

The general process of using natural language as a means of communication between human users and computers is called Natural Language Processing (NLP). A particular tool which is used in performing NLP tasks is the GPT (Generative Pretrained Transformer), known in the forms of GPT-3.5 or GPT-4, which stands at the core of ChatGPT. GPTs fall within a larger category, which is called the Large Language Models (LLMs). Language Models are able to produce new sentences based on the analysis of the data provided and the specific rules in their algorithm (Khurana et al., 2023, Liu et al., 2023).

To continue the logic of the above paragraph, it is clear that LLMs are able to produce text, but not visuals (Chen et al., 2023). However, this text output can serve as an input for other tools, which in turn can provide visualizations. For example they can produce code, which can be used by other software with the ability to produce visualizations (White et al., 2023). This last statement brings us to the concept of Prompt Engineering.

User input in the form of NL prompts can be too vague for the LLM to understand its intent and provide the most accurate output. Therefore, some techniques are needed to further refine the prompts and these techniques are included in the process of Prompt Engineering, to which some authors refer also as an art (Ekin, 2023). The combination of LLMs with Prompt Engineering, materialized in the form of specific tools which provide visualizations from NL will be the focus of this paper, first in the form of a literature review aiming to explain the main terms and developments in simpler words and further on as an evaluation of the LLM2VIS tools.

2.2 The evolution of NLP for visualization (NL2VIS)

The process of transforming the user requests expressed in Natural Language, to visualizations, is commonly referred as NL2VIS (Luo et al., 2021). Maddigan et al., 2023, in their work for Chat2Vis, which will be in the focus of this paper in the later stages, present the following classification of NL2VIS systems: Symbolic NLP, Deep-Learning and LLMs. This is also a timeline, with LLMs being the latest technology for NL2VIS tasks. In the following table, a summary of the methods will be presented.

Additionally, 3 specific examples (1 for each approach) will briefly be explained to illustrate their functioning. Additionally, an explanation of how the general rules of the category to which they belong apply to them is provided.

The first category, Symbolic NLP or Rule-Based NLP, relies on building a set of rules to treat the user input in natural language, in order to produce the output. In Table 1, the name "Symbolic NLP" is chosen for this category, because it includes also the Heuristic and Probabilistic sub-categories, besides the Rule-Based one. The Heuristic sub-category derives approximate solutions based on a set of rules. In the Rule-Based sub-category, the output is more accurate, but also the complexity of the model increases, as well as the need for computational resources. The third sub-category, Probabilistic NLP, relies on even-more complex rules based probabilities and grammar and although they offer a higher precision, this comes at the cost of more complex models requiring more computational resources (Li et al., 2024, Narechania et al. 2020, Maddigan et al., 2023).

The second category, Deep-Learning, moves away from the rules approach and uses learning via Neural Networks to understand user inputs. This makes tools based on this technology more flexible, compared to the ones using a static set of rules as was the case in the first approach. Deep-Learning based visualization tools (ADVisor, 2021), use also language transformers to produce visualizations, but not yet LLMs, which fall under the third category, explained in the next paragraph (Luo et al., 2020).

The third category, LLMs, use Large Language Models, which are bigger more complex than the Language transformer used in the previous category. Together with Prompt Engineering they explore LLMs ability to produce visualizations. Tools based on this technology (Chat2Vis), will be studied further in this paper.

Table 1: Evolution of NL2VIS

NAME	SUMMARY			DISADVANTAGES COMPARED TO LLMS	EXAMPLE
SYMBOLIC NLP	OTHER AUTHORS (LI ET AL., 2024, NARECHANIA ET AL. 2020) REFER TO THIS CATEGORY AS RULE-BASED, BUT THE SYMBOLIC APPROACH INCLUDES RULE-BASED, AS WELL AS HEURISTIC AND PROBABILISTIC APPROACHES. THEY ALL SHARE THE COMMON FEATURE OF USING PRE-DEFINED RULES IN TREATING THE USER INPUT (MADDIGAN ET AL. 2023).			BEING BASED ON A SET OF RULES, THEY LACK THE FLEXIBILITY TO HANDLE THE COMPLEX USER INPUTS IN NL. AS WE MOVE FROM THE APPROXIMATE HEURISTIC TO THE MORE PRECISE PROBABILISTIC APPROACH, THE PRECISION INCREASES, BUT ALSO THE NEED FOR COMPUTATIONAL RESOURCES, AS WELL AS THE COMPLEXITY TO BUILD THESE TOOLS.	<div> <div></div> <div></div> <div></div> <div></div> </div> DATATONE (2015)
	<u>HEURISTIC:</u> USE RULES TO FIND APPROXIMATE SOLUTIONS	<u>RULE-BASED:</u> MORE ACCURATE, EXPERT-DESIGNED PRE-DEFINED RULES..	<u>PROBABILISTIC:</u> USE GRAMMAR RULES AND PROBABILITIES.		
DEEP-LEARNING	THIS CATEGORY IS ALSO KNOWN AS THE NEURAL NETWORK APPROACH (LI ET AL., 2024), BEING NAMED AFTER THE TECHNOLOGY USED, OR DEEP-LEARNING, BEING NAMED AFTER THE PROCESS THEY USE TO PROCESS NL USER INPUTS (MADDIGAN ET AL. 2023, LUO ET AL., 2020).			UNLIKE THE PREVIOUS ONE, THIS APPROACH MAKES USE OF LEARNING TO BETTER UNDERSTAND USER INPUTS, RATHER THAN A PRE-DEFINED SET RULES. AS A RESULTS THE FLEXIBILITY AND ADAPTABILITY INCREASES. THESE TOOLS MAKE ALSO USE OF LANGUAGE TRANSFORMERS AS THE ONES IN THE THIRD CATEGORY, BUT LLMS ARE LARGER AND MORE SOPHISTICATED (MADDIGAN ET AL. 2023).	ADVISOR (2021)
<div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> LLMS	THESE TOOLS EXPLORE THE CAPACITY OF LLMS AND PROMPT ENGINEERING TO PRODUCE VISUALIZATIONS FROM NL (MADDIGAN ET AL. 2023).				CHAT2VIS (2023)

2.2.1 Datatone (2015)

This tool is based on the principle of the rule-based approaches of NLP. NL queries go to the Query Analyzer, which maps or translates the language queries into 'meaningful' context for data and visualizations, typically aiming to find column names, or relationship patterns. The 'final product' of this phase is a SQL query, generating a view from the database, which is then visualized (Gao et al, 2015).

Datatone was chosen as a representative of the first category in Table 1 above. The mapping of the queries is a feature these tools commonly share. However, the trend today is to move away from this technique, towards deep learning approaches, which can outperform rule-based tools in NL visualization tasks (Maddigan et al., 2023, Li et al., 2024).

2.2.2 ADVisor (2021)

ADVisor is one of the tools based on the Deep-Learning concept, which aims to resolve the problem of the user NL queries being restricted to a set of predefined templates that the rule-based approach has. Here, the user doesn't need to have knowledge about the data or the visualization, prior to entering the NL query, as it was previously needed. Instead of the pre-defined sets of rules, a pre-trained language model called BERT (will be detailed more in the following paragraph) is used. BERT makes sure that the user queries are converted into data attributes, which can be found in the tabular dataset (Liu et al., 2021).

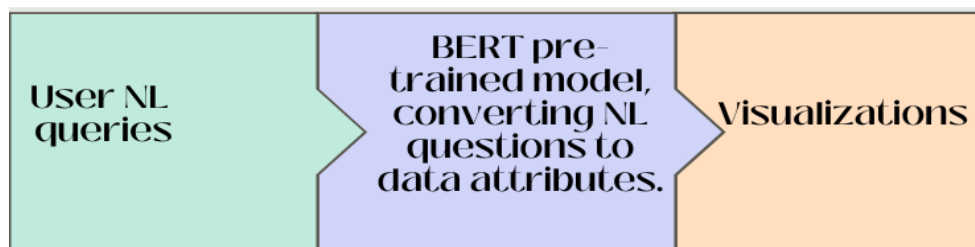


Figure 1: Simplified working model of ADVisor

2.2.3 BERT vs GPT

BERT stands for **B**idirectional **E**ncoder **R**epresentations from **T**ransformers. It was introduced by Google first in 2018. The "bidirectional" ability allows it to understand text from left to right, as well as from right to left, which in turn makes understanding words in context and tasks such as question answering, its main advantage (Devlin et al., 2018).

GPT stands for **G**enerative **P**re-trained **T**ransformer and it is developed by OpenAI. Unlike BERT, GPT analyzes text only in one direction (left to right). This feature makes GPT able to excel in tasks such as text generation in the form of sentences, paragraphs, stories and also code. Since the two models

have their own advantages and disadvantages the choice of one versus the other will depend on the specific NLP task (Rutherford, 2024).

For the context of this paper, it is needed to be understood why to choose one or another for data visualization tasks. Both models generate text and not visuals, but the above mentioned ability of generating code starting from NL prompts is further explored to build visualizations tools by combining GPT's code scripts with other layers (Maddigan et al., 2023).

2.2.4 Chat2Vis (2023)

Chat2Vis was introduced in 2023 by researchers Paula Maddigan and Teo Susnjak and it's a representative of the new tool exploring the capacity of LLMs to produce complex code scripts, mentioned above, in order to obtain data visualizations. After its first introduction, Chat2Vis was further fine-tuned with multilingual text and pre-trained models. Before delving into it a little deeper, let's represent a simple working model of it (Maddigan et. al., 2023).

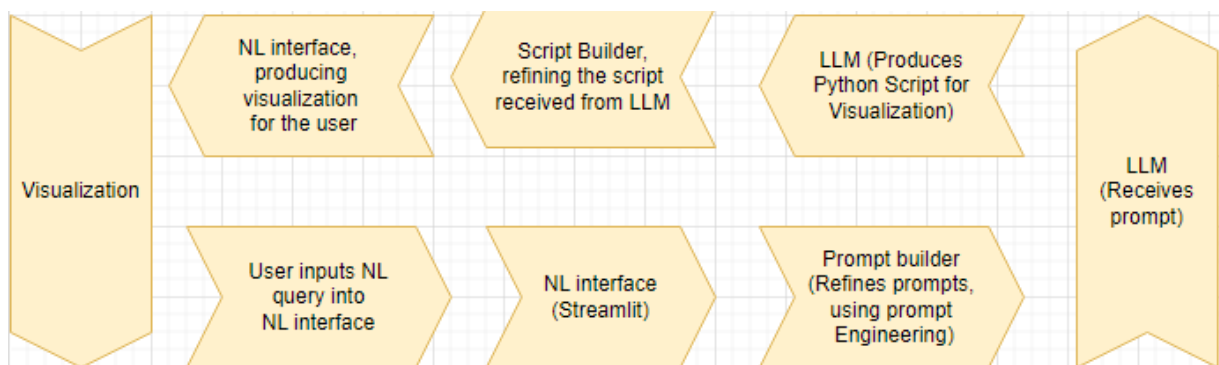


Figure 2: Simplified working model of Chat2Vis (Maddigan et. al., 2023)

The authors of Chat2Vis demonstrate in their study that this approach (using LLMs and prompt engineering), can produce accurate visualizations even when the query is underspecified, or misspecified. Chat2Vis is hosted on Streamlit cloud and in its user-interface allows the users to upload their own database to produce visuals (Maddigan et al., 2023, Katib et. al, 2023). Streamlit is a Python based framework, which makes it possible to convert Python language scripts into web apps (Dayanithi 2023). Through this technology, user NL queries, prompt engineering, LLMs and visualization are integrated all in a single system (Alsulami et. al., 2023). Chat2Vis has been used also in other studies, because of its ability to provide LLMs with both data and prompts at the same time (Sartori et. al., 2024).

Furthermore, Chat2Vis is able to produce 5 visualizations for each query, based on 5 different LLMs, allowing the user to compare their performance. These 5 LLMs are ChatGPT-4, ChatGPT-3.5, GPT-3, GPT-3.5 Instruct and Code Llama. Chat2Vis is able to understand complex queries. However the above mentioned models require some refinement, in order for their information to be more precise. Another

observation for Chat2Vis is that even though it is a chatbot-based tool, expected therefore to give users feedback for their query, justifying the choices made, it doesn't (Kavaz et. al., 2023).

2.2.5 LIDA

LIDA is another visualization tool based on LLMs. Unlike Chat2Vis, LIDA is the product of Microsoft, testifying that also big companies are entering the field of data visualization with LLM-powered data visualization tools. It was introduced in 2023, during the 61st Annual Meeting of the Association for Computational Linguistics (Dibia, 2023). The approach divides visualization task into subtasks, therefore LIDA contains 4 parts:

- Summarizer: Converts the uploaded data into NL summary
- Goal explorer: Enumerates visualizations
- Viz Generator: Visualization code
- Infographer: Converts data into graphs

Similar to Chat2Vis, also LIDA offers a variety of LLMs to choose, but different from it, besides visualization, we can find also the Python code in LIDA. Anyways, as mentioned by the author, compared to other tools, offering a user interface, LIDA has code interface, which can bring difficulties to users with non-programming experience. LIDA has also a Demo UI interface in their Github page, but some installations are needed before using it.

2.3 Prompt Engineering

2.3.1 Overview of Prompt Engineering

LLMs have the fact of being large as a main advantage, implying training in many parameters. Therefore, the interaction with the human user can be improved by well-designed prompts. Improved prompts lead also to the improvement of the output the user receives, by enhancing its usefulness. This has also been defined as guidance, between the large data that LLMs contain and the specific output the human user desires. First, the user needs to define the specific response it wants to receive from the LLM. Then, the prompt is refined by Prompt Engineering and it may require an iterative (systematic) approach, in order to achieve the best results (Cain, 2024, Poddar, 2023).

Prompt Engineering today is also a programming challenge, because of the choice between simplicity and complexity and the specific advantages and disadvantages each approach has. For example, a complex approach of Prompt Engineering will bring more "nuanced" output from the LLM, but at the same time, it will become more difficult for the LLM to understand user intent. Conversely, with a more simple approach the NL inputs will be easier to understand, but the output may not be as specific as desired. Whichever the approach, Prompt Engineering remains critical in user interaction with LLMs, as it improves the quality and the relevance of the model's output (Cain, 2024).

2.3.2 Prompt Engineering can also be done by users

Besides programmers as mentioned above, Prompt Engineering can be achieved also by the user inputting the NL queries. OpenAI, suggests a number of strategies, which will be shown in a simplified and concise manner in Figure 3 below:

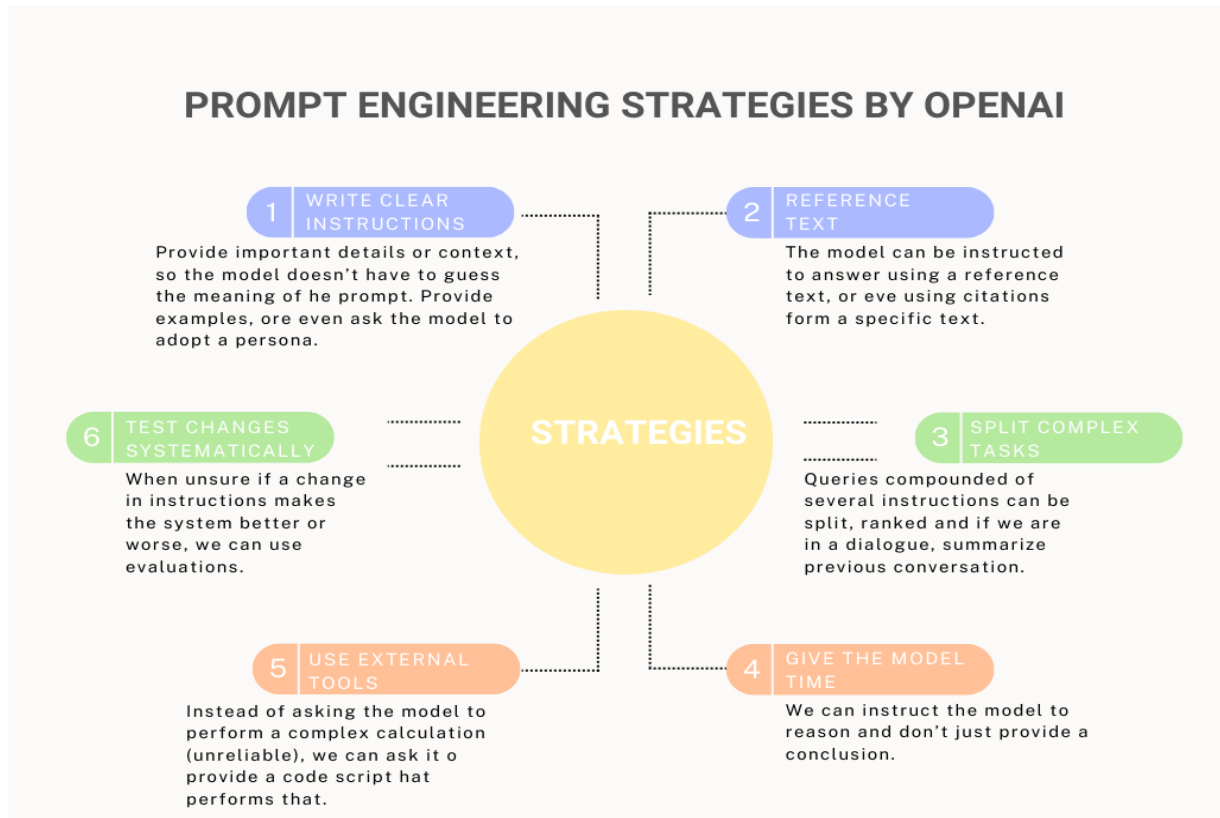


Figure 3: Overview of Prompt Engineering Strategies (OpenAI)

This concludes the general part of Prompt Engineering and the following paragraph will contain more relevant information on the focus of this for the applications of LLMs in data visualizations.

2.3.3 Examples of Prompt Engineering in visualization tasks

The above strategies can be applied to the context of data visualization tools, especially for obtaining the correct script of Python code, in order to produce the required graphs. The best technique will be the "show-and-tell" one, meaning to provide examples and instructions together with the prompt (Maddigan et al., 2023).

Similar to the software patterns, prompt patterns can be built, as reusable solutions to improve interaction with LLMs and customize output. An example of output customization is of course visualization. The correct approach in this case would be to tell the LLM to generate a text output to

be inputted to another tool, with the final goal to visualize it (White et al., 2023). This approach will be presented in a simplified way, in figure 4 below.

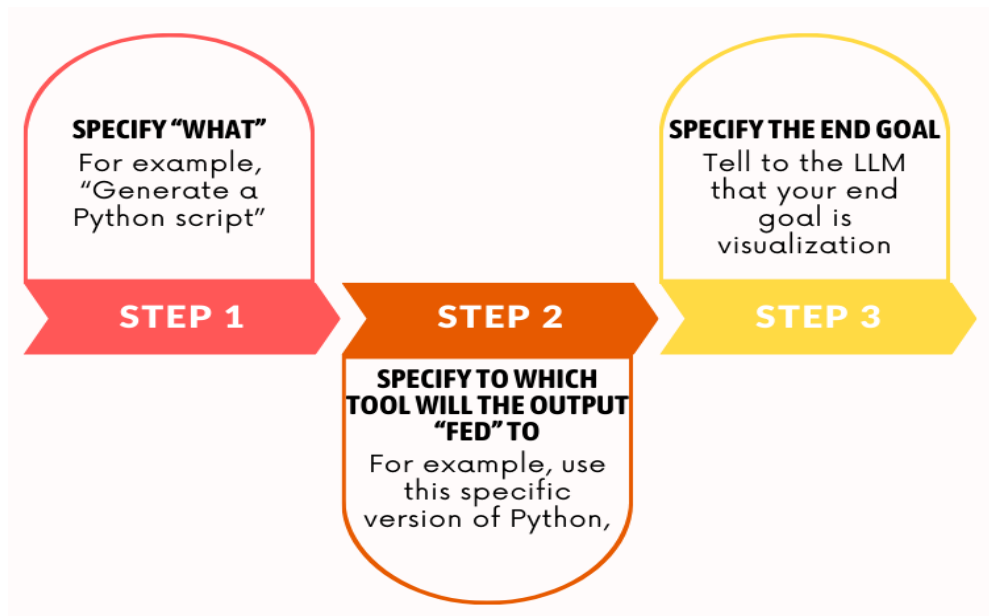


Figure 4: Representation of the Visualization generator pattern (White et al., 2023)

This is true also in the case of Chat2Vis. The authors (Maddigan et al., 2023) use a step-by-step process in tailoring the prompt they will input to the LLM. This process consists, of providing information about the data that will be used and after that they instruct the LLM to provide a Python script (Step 1 above) provide it with the needed version it (Step 2 above), together with their final intent to graph (Step 3 above) the prompt that will follow.

2.4. Current market (usage) of visualization tools

According to a study, approximately 70% of people prefer to have an AI chatbot (similar to Chat GPT) as an aid for visualization, instead of inserted software (Rajagopal, 2023). Furthermore, LLMs have been in the top 12 as a trend for the last 5 years, especially after the introduction of ChatGPT. There is also an increase in funds invested in the research about LLM, with more than 500 organizations currently engaged (TrendFeedr, 2024). This is also supported by the research from Fortune Business Insight. According to this research, there is a trend in current companies offering widely used visualization tools to invest in NLP, in order to enhance their product (Fortune Business Insight, 2024).

Even with the promising future, it would be useful to check the information about the most used data visualization tools from businesses currently, as well as in recent years. Forbes mentions Power Bi and Tableau among the best visualization tools, which are introducing components of AI, but are not LLM

based (Haan et al., 2024). This is supported also from the Markets and Markets research report, who also lists Salesforce (Tableau), Microsoft (Power Bi) as top players in the data Visualization market (Markets and Markets, 2021). These are also the two market leaders in the Gartner's magic quadrant of 2020 (Richardson et al., 2020). Based on the above data, it can be inferred that LLM based tools for data visualization, despite being in the focus of the companies, are not key players in the market yet.

As a closing paragraph of this short overview of the current visualization tools, let's bring some market data, from the mobile phone key players before and after the introduction of smartphones. This is to testify that the world has seen before a complete change in the key players of the market in the rise of a new technology. Before the introduction of the smart phones, Nokia was dominating the mobile phone market. After that, Samsung and Apple managed to surpass it, both with smartphone based technology (Cecere et al., 2015). Will LLMs also be the future of data visualization as smartphones were to the mobile phone technology? Will we soon need just a NL sentence to have our visualization, instead of navigating in the menus of our chosen data visualization tool? These questions will be answered in the following years hopefully, by future work, or personal experience of each one of us.

2.5 Summary of Literature review

In this section, part of the existing knowledge and work in the field of NLP, NL2VIS tasks, as well as LLMs usage for visualization was presented. The current research is focused in the LLMs, since these models hold some advantages, compared to previous approaches. Together with Prompt Engineering, they can achieve results in NL2VIS tasks. There are already some new tools exploring the field, (which will be in the focus of this paper later on), as well as investments from several companies in the field, but this type of tools are not key players in the market yet. Further exploration, will make sure to answer the question, if they will be the new mainstream tools for businesses in the future.

2.6 What comes next?

In the literature review part, we talked about LLM based-tools as the third approach to build data visualization tools. Based on this, the next section will focus on the presentation of the methods that will be used to explore a specific tool based on this technology, Chat2Vis. Our main aim is to compare Chat2Vis with a more traditional tool called Power Bi. We'll do this by doing some tasks (experiments) using both Chat2Vis and Power Bi. The outcomes of these experiments will be discussed in Results section, and the Discussion Section will be the one where we wrap up our final thoughts on this topic.

3. Research questions

While reading about visualization tools, my approach was to try and understand the current landscape of LLM-based visualization tools and the possibility to explore them. Initially I conducted a review of literature related to visualization tools, focusing particularly on those utilizing LLM (Large Language Models) technology.

While reading the various articles about LLM visualization tools, I was not able to find studies comparing them against more traditional visualization tools. This observation was the basis of the first research question: *"How do LLM-based visualization tools compare to the more commonly used ones?"*

After this, I shifted my focus to the user experience aspect. I noticed a pattern during the literature review. LLM visualization tools are designated for the common user of visualization tools, which may have limited programming skills, but they were usually tested by the programmers who built them. Based on this, I wanted to investigate the user-friendliness of these tools. This led to the second research question: *"How user-friendly are these tools for users with little or no programming experience?"*

In summary, through these two research questions I aim to contribute to LLM-based visualization tools by providing user insights on a specific tool, Chat2Vis.

4. Methodology

It is possible to evaluate LLMs and attempt to compare them with other tools, through standardized exams that concentrate on a particular application domain. For example, Chat GPT has been tested with law school exams (Choi et al., 2023) and on medical examinations (Nori et al., 2023). There is no standardized exam for evaluations of LLMs in data visualizations, but other tools can be used for this purpose (Chen et al., 2023).

In order to address the first research question, (*How do LLM-based visualization tools compare to the more commonly used ones?*) a **comparative evaluation** will be performed.

The chosen **tool** which will represent the “traditional” approach will be Power Bi. For the newer LLM-based technology, the chosen tool will be Chat2Vis. The comparative evaluation will be achieved via the completion of a series of visualization tasks on both tools and by recording results, in order to gain insights on the accuracy of Chat2Vis, compared to Power Bi. As regards to the **datasets**, publicly available ones, such as Iris, Titanic, etc. will be used (Sial et al., 2021, Kocon et al., 2023, Maddigan et al., 2023).

Next, we will focus on the **metrics** chosen to compare LLM-based tools with traditional ones. For this paper, assignments from the Data Science course at UHasselt will be used, and visualizations generated by the chosen LLM-based tool (Chat2Vis) will be compared with those from Power BI. Specifically, two key metrics will be used: accuracy and efficiency.

Accuracy will be assessed through the visual inspection and comparison of the correctness of visualizations produced by Chat2VIS and Power Bi. This measure ensures that the visualizations are not only technically correct but also effectively show the intended information. Specifically, the accuracy evaluation will be based on 5 criteria, as below:

- *Correctness of the graph – Does the graph show the correct information?*
- *Completeness of data – Does the graph include all the categories in the dataset?*
- *Scales – Check the scales of the axis.*
- *Clarity – Are the labels clear and readable?*
- *Relevance – Can the user identify the main groups or trends from the graph?*

Based on these criteria, we build a table for each of the graphs and show it in the Results section.

Efficiency will be measured by the time taken for each tool to complete the visualization tasks. This includes recording the start and end times of each task and calculating the total duration. The relevance of this measure lies in its ability to demonstrate the practicality and usability of Chat2Vis in performing practical visualization tasks. (Karat, J. 1997).

For the second research question (*How user-friendly are these tools for users with little or no programming experience?*) a **user experience study** will be conducted. The personal experience of

the author of this paper while performing the tasks will be provided (Karat, J. 1997). I will be performing the tasks of the UHasselt Data Science course with Chat2Vis and Power Bi and provide feedback for the ease of use (Wang Y. et al., 2022).

To make sure that the results are transparent and available to the other users, an online tool in the form of a website can be built (He et al. 2023), but this will require programming skills. Another approach would be to make all the materials used during the tests, publicly available on Github (Chen et al., 2023, Kocon et al., 2023, Wang et al., 2021). This will be the chosen tool for materials' publications in this paper. The screen recordings of the experiments may be accessed through this link: https://github.com/olsi2984/LLMs_evaluation_visualization_Master_thesis

In order to not overload the paper with information as well as to allow readers to focus and follow the main idea, making available the extended results in the appendixes will be used (Noever et al. 2023, Wang L. et al., 2021, Kocon et al., 2023). This technique will also be used in this paper and although some of the visualizations will be provided in the main text, readers can make use of more extended materials in the appendixes, if needed.

Before beginning with the visualization tasks, we set up Chat2Vis according to the protocol in Appendix 1. For Power Bi, the offline version: 2.127.1327.0 64-bit (March 2024), is used.

The visualization tasks are presented in Appendix 1 and consist of 3 main types of tasks:

Type 1: Creating visuals from data. The goal is to produce graphs from the datasets and compare them with each other for accuracy. Furthermore the time for performing these tasks is recorded. The specific task representing type 1 will be to *build a histogram of the Age variable, on the Titanic dataset*. This is followed by two other sub-tasks, *changing the name of the axis and the color of the graph*, so we can see the ability of Chat2Vis to perform this task as well as to record the time needed and compare it with Power Bi. Figure 33 shows a representation of this graph derived from the software used at UHasselt for this visualization task (KNIME).

Type 2: Reproducing a visual. In this task, the figures of two graphs are presented and the goal is to reproduce them, using Chat2Vis and Power Bi. After this, compare the graphs for accuracy, as well as the time it takes to perform them with a tool and the other. For this type, there are two specific tasks. The first is to *reproduce a scatter plot from the Iris dataset*. The required graph is presented in Figure 5, Appendix 3. The second tasks, is to *reproduce a pie chart form the Wines Dataset*. The required charts are shown in figure 34.

Figure 33, Histogram of Age on the Titanic dataset

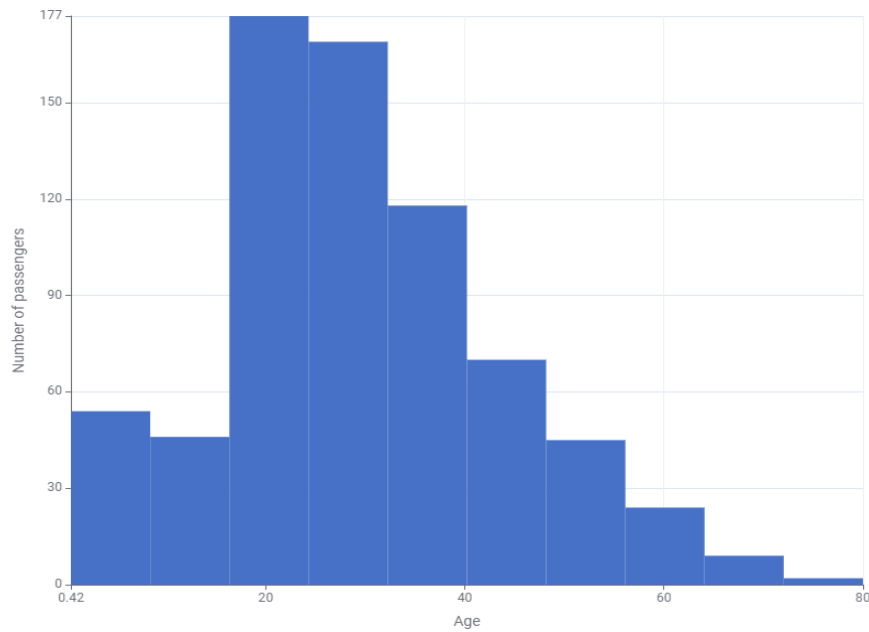
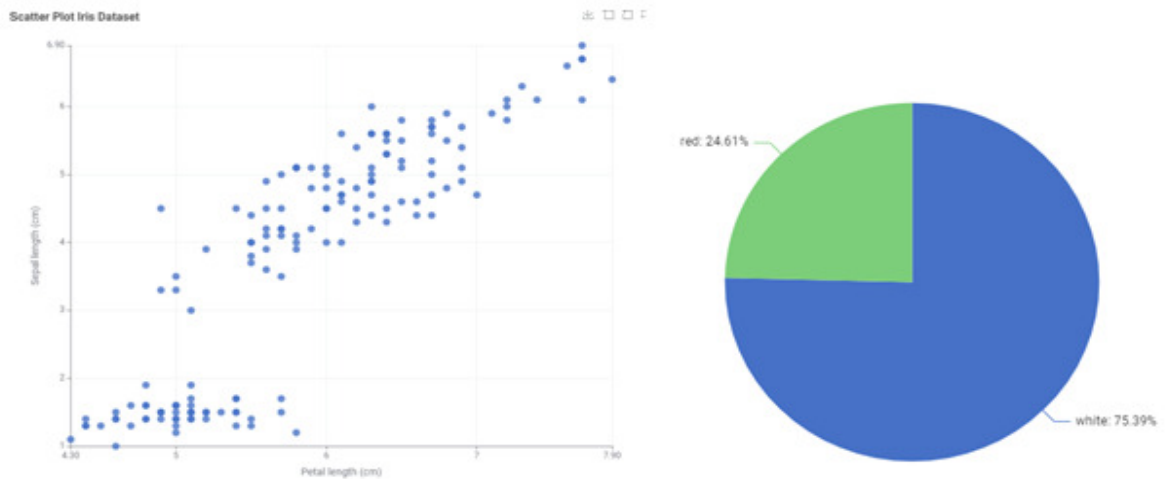


Figure 34: Scatter plot from the Iris dataset and pie chart from the Wines dataset



Type 3: Taking the visual from a tool to another. Part of the job of producing visuals is also taking them from one tool to another, for example take the charts from Chat2Vis or Power Bi and use them in a word document as part of a thesis for example. The goal here is to get insight on the user-friendliness of the selected tools (Chat2Vis and Power Bi). The specific task will be to *take all the created visuals from Chat2Vis and insert in the word document of this thesis.*

5. Results

This section will be divided in three parts. The first one will show time of the visualization tasks. The second one, will be dedicated to a presentation of the different visuals produced and their accuracy, aiming to answer the first research question (*How do LLM-based visualization tools compare to the more commonly used ones?*). The third part will be dedicated to the user experience in using Chat2Vis, in order to answer the second research question (*How user-friendly are these tools for users with little or no programming experience?*).

5.1. Time recordings

The times measured do not consider the time spent to find the appropriate menu on Power Bi, or the time to think about the NL query on Chat2Vis. The assumption is that the user knows before the menu where to click on Power Bi and the query that will be used in Chat2Vis. The type 3 task is performed repeatedly after each visualization build in type 1 and 2 and therefore the execution time is not measured.

Set-up time is considered the time from the beginning of the interaction with the tool, up to and including the data uploading. Therefore, set-up time for Chat2Vis is inputting the keys and uploading the data, while set-up time for Power Bi is uploading the data. Build time is the time needed to produce the visual. This means the time for writing the query in Chat2Vis and the time of the different LLM models to produce the visuals. For Power Bi, this time is consists of the time navigating the various menus plus the time for the tool to produce the graph.

For the first task, build a histogram of the Age variable, on the Titanic dataset, it can be seen that it takes more time on Power Bi than Chat2Vis (84 sec vs. 43 sec), with both set-up and build times being higher. However, this difference decreases in the two follow-up tasks which are performed with the data being already uploaded, with Power Bi doing better in changing the color of the graph. Anyways, for type 1 tasks the total time spend receiving the visuals from Chat2Vis is lower than the time spent in Power Bi ($84+39+15=138$ sec Power Bi VS $43+36+18=97$ sec Chat2Vis).

In type 2 tasks, the same trend is repeated, with Power Bi having higher set-up and build times than Chat2Vis, with the differences between the two tools being higher in the set-up than in the build time. Power Bi has a lower build time in the task reproducing the scatter plot from the Iris dataset due to the fact the NL query typed in Chat2Vis is longer.

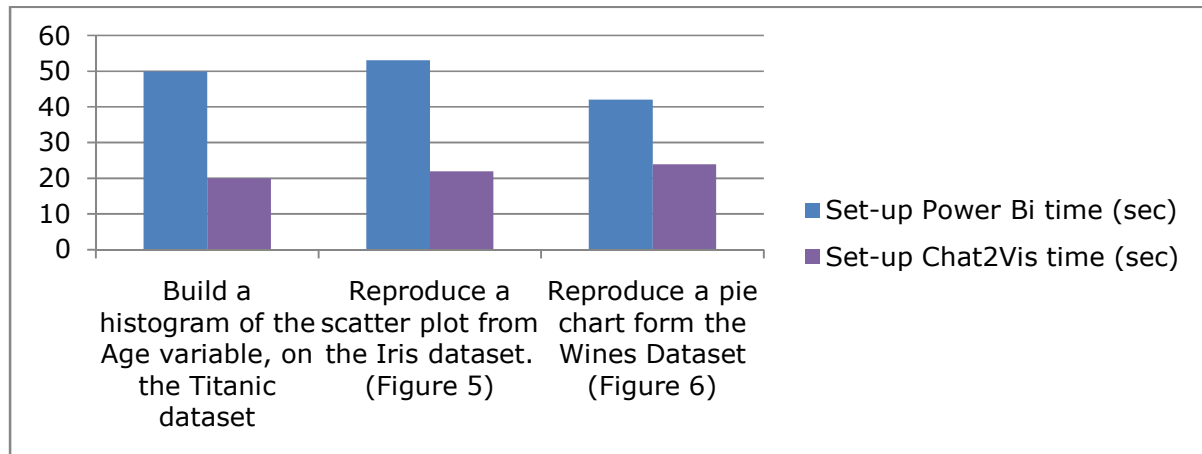
The results of the visualization tasks as regards to **time** are presented in Table 2 below.

Type	Task	NL query used in Chat2Vis	Power Bi time (sec)			Chat2Vis time (sec)		
			Set-up	Build	Total	Set-up	Build	Total
Type 1: Creating visuals from data	Build a histogram of the Age variable, on the Titanic dataset	Build a histogram of the Age variable, on the Titanic dataset.	50	34	84	20	23	43
	Changing the name of the axis	Name the horizontal axis Age of passenger and the vertical axis number of passengers.	0	36	39	0	36	36
	Changing the color of the graph	Use the color red for the graph.	0	11	15	0	18	18
Type 2: Reproducing a visual	Reproduce a scatter plot from the Iris dataset. (Figure 5)	Build a scatter plot on the iris dataset, showing sepal length in the horizontal axis and petal length in the vertical axis.	53	28	81	22	45	67
	Reproduce a pie chart from the Wines Dataset (Figure 6)	Build a pie chart by color on the wines dataset.	42	22	64	24	20	44
Type 3: Taking the visual from a tool to another	Take all the created visuals from Chat2Vis and insert in the word document of this thesis	-	-	-	-	-	-	-

Table 2: Visualization tasks times

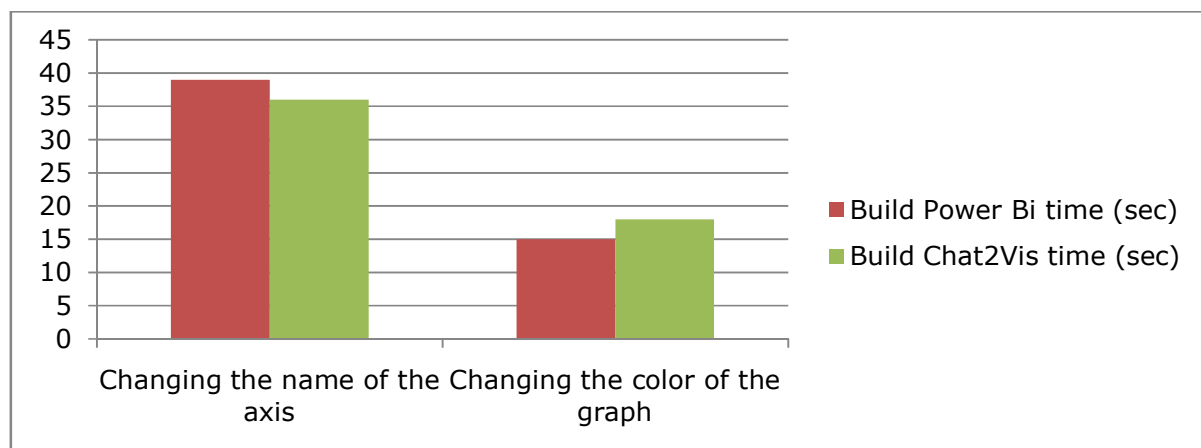
The most important trends noticed in the times are three. **Firstly**, it takes more time to set-up Power Bi than Chat2Vis, with set-up up times been higher in both tasks. The set-up time mainly consists on the data uploading. This can also be seen in the graph below.

Figure 35: Set-up and build times



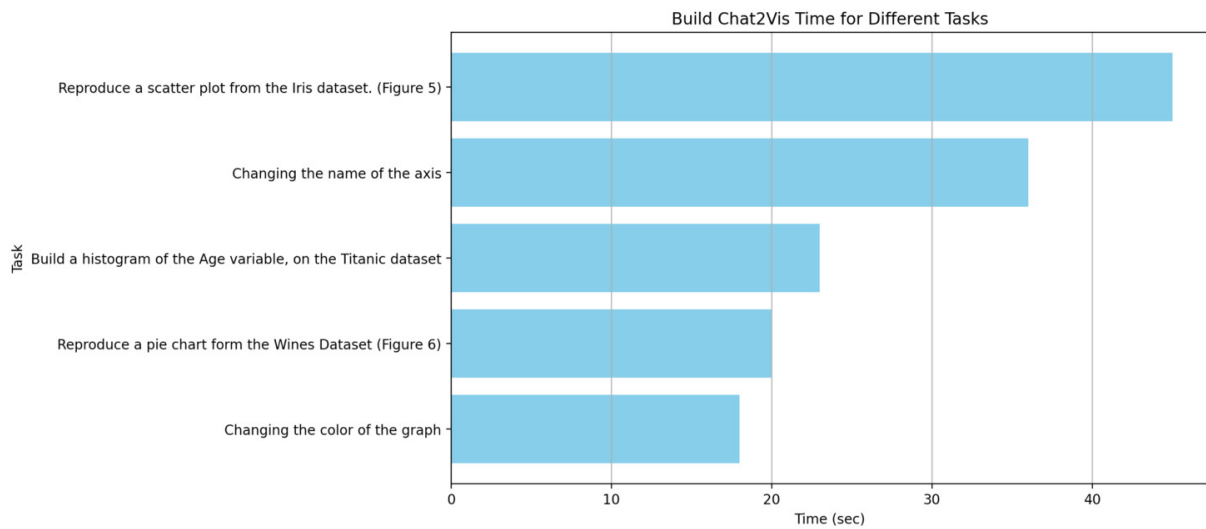
Secondly, if we perform other visualization tasks, with the data being already uploaded, such as changing the color, or changing the names of the axes, this difference lower. Power Bi does actually better in changing the color of the graph. This show that once the data is uploaded and you are familiar with the tool, you can achieve customizing tasks of a visual via a NL query (Chat2Vis) or menu clicks (Power Bi) in a similar time. This is shown also in the graph below:

Figure 36: Build times



Thirdly, if the length of the NL queries increases, the difference in build time decreases. This is because the time for writing the query, adds to the build time of Chat2Vis. If we sort the NL queries from the shorter one, to the longer, we get the graph below, showing an increase in build time for Chat2Vis, as the length of the NL queries increases.

Figure 37: Build times and query length, Chat2Vis



As a summary of Table 2, it can be said that Chat2Vis has lower set-up times than Power Bi and assuming that time is the only important factor this would make Chat2Vis more preferable than Power Bi, if the user needs to perform one single visualization task, from a specific database. If we need to perform more than 1 task in the same database, for example customizing the visual, than the differences in time between the two tools are lower. Another factor to consider is the length of the NL query needed. If the query is longer, it takes more time to build the visual in Chat2Vis, without considering the difference in set-up time.

5.2. Accuracy of the visualizations

The next part of this section will consider the results of the experiments as regards to the **accuracy** of the visuals produced by Chat2Vis, by putting them side by side with the visuals produced by Power Bi. For the first chart type, the histogram, we build table 3, with the relevant criteria present in the Methodology section, each customized for this specific graph, as well as the comments.

Regarding the correctness of the histogram graphs (see Figure 28), that proved difficult to assess, without specifying the number of age bins. For this reason, we re-perform this task, both in Power Bi and Chat2Vis and present the results in figure 38. For this task, we use the NL query "Build a histogram of the Age variable, on the Titanic dataset. Group Age in bins of 10 years each."

In Table 3, we can see that the graph rendered from the model GPT 3.5-instruct in Chat2 Vis, does not show the correct scale in the y-axis (The red cell in the table), but after re-performing the tasks, with age bins specifications, the rendered graph (Figure 38), shows the same scale and the same result for each bin, as Power Bi. The model Code Llama did not generate a graph in the second time we performed this task.

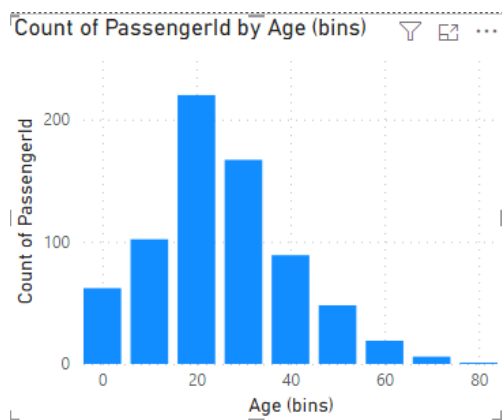
Table 3: Accuracy evaluation, histogram

Criteria	Power Bi	Chat2Vis (GPT 3.5 – Instruct)	Chat2Vis (Code Llama)
Correctness of the Graph			
Correct number of passengers for each group of age	Difficult to assess without specifying the number of bins	Difficult to assess without specifying the number of bins	Difficult to assess without specifying the number of bins
Completeness of data			
The graph includes all age categories	Ages from 0 to 80	Ages from 0 to 80	Ages from 0 to 80
Scales			
Size of the bins (x-axis)	bins of 10 years, starting from 0	bins of 10 years, starting from 0	bins of 10 years, starting from 0
Scale of y-axis	From 0 to 200	From 0 to 177	From 0 to 25
Clarity			
Reading the labels	Labels are clear and readable	Labels are clear and readable	Labels are clear and readable
Relevance			
Peaks and trends	The largest group of passengers is in the age group 20 - 30	The largest group of passengers is in the age group 30 - 40	The largest group of passengers is in the age group 30 - 40

As regards, to the completeness of the data, both Chat2Vis rendered graphs include all the categories of age, form 0 to 80 years. The labels are clear and readable. For the relevance, it can be said that at first, the model rendered from GPT 3.5 – Instruct, did not show the correct group, but when we retrieve the graph the second time, it correctly shows the largest group of passengers having the age 20 – 30 years old.

Figure 38: Charts from Power Bi and GPT 3.5 –Instruct, with Age variable bins of 10 years each.

Power Bi



Chat2Vis (GPT 3.5 – Instruct)

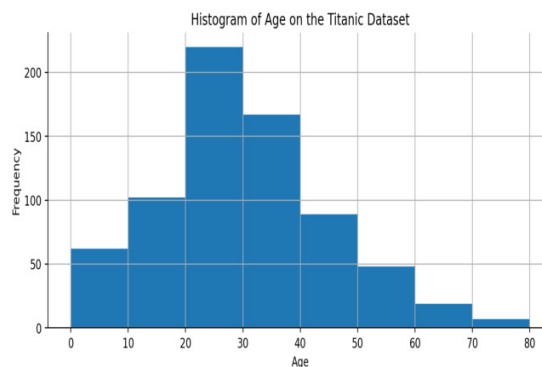
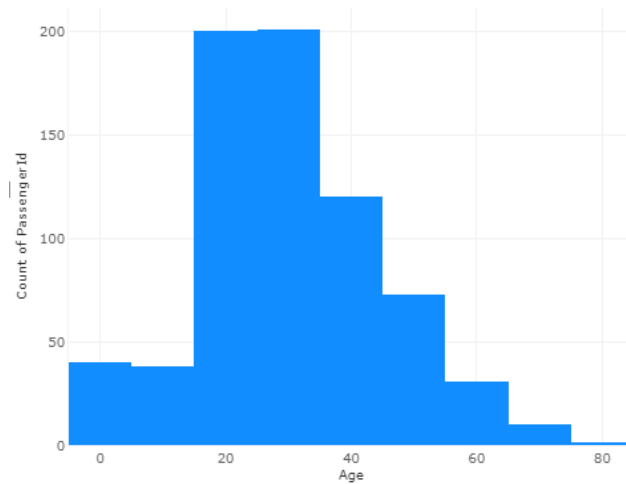
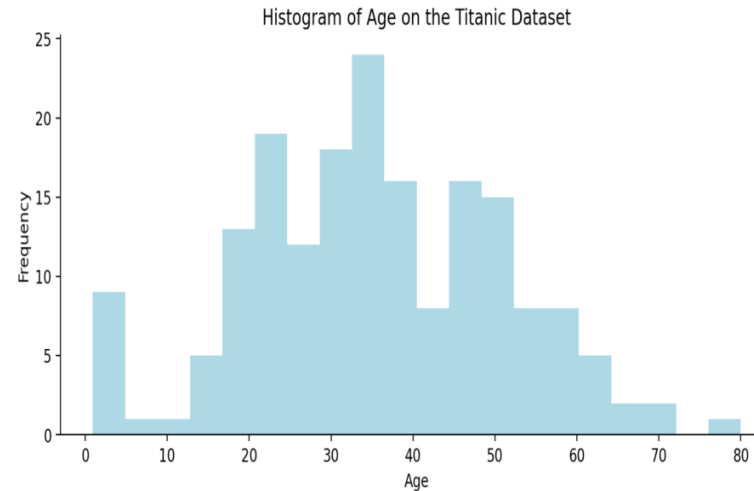


Figure 28: Build a histogram of the Age variable, on the Titanic dataset.

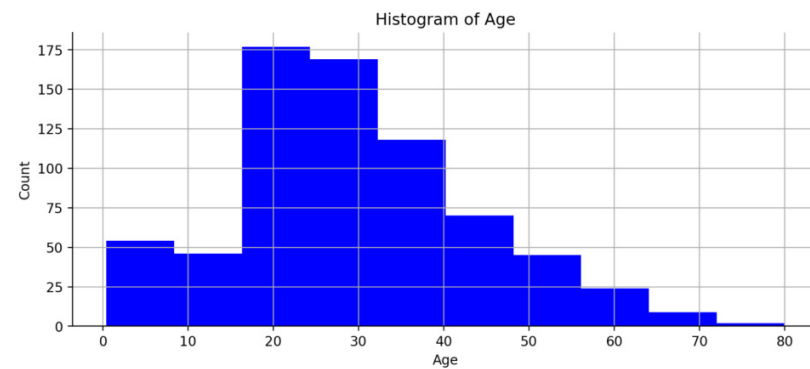
Power Bi:



Chat2Vis (GPT 3.5 – Instruct)



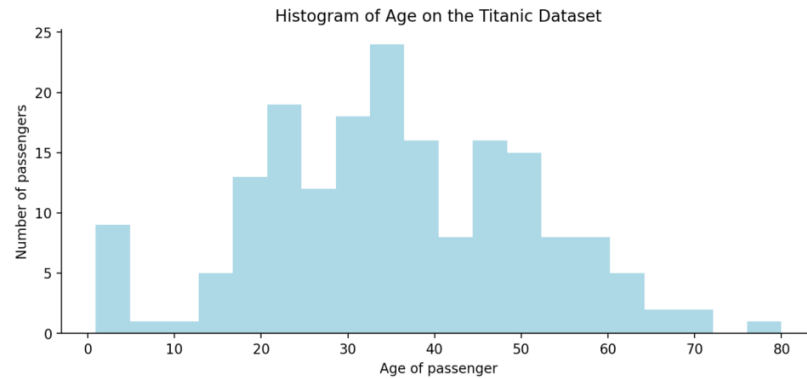
Chat2Vis (Code Llama):



The picture shows the results of the first task, with Chat2Vis producing graphs from 2 LLM models. The graph produced from Code Llama in Chat2Vis gives the idea of the major concentration of the between ages 20 – 30 years, while the graph from GPT 3.5- Instruct incorrectly shows the largest group in the age 30-40 years. However, this changes, when we perform the task for the second time and specify the number of bins.

Figure 29: Changing the name of the axis: Both LLMs were able to change the names on the axis, following the NL query.

Chat2Vis (GPT 3.5 – Instruct):



Chat2Vis (Code Llama):

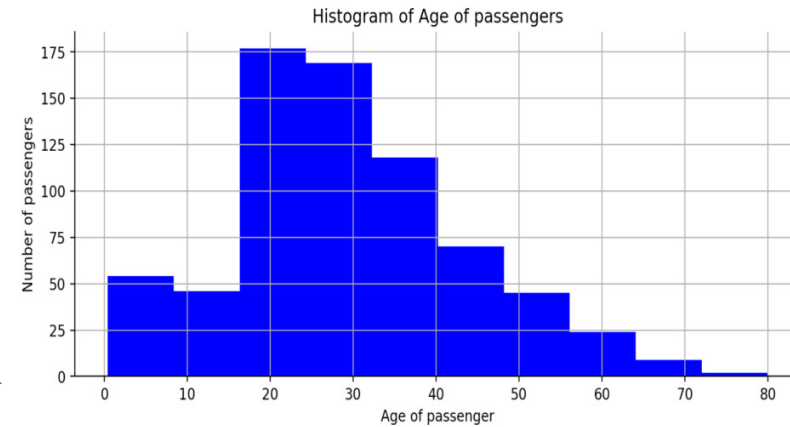
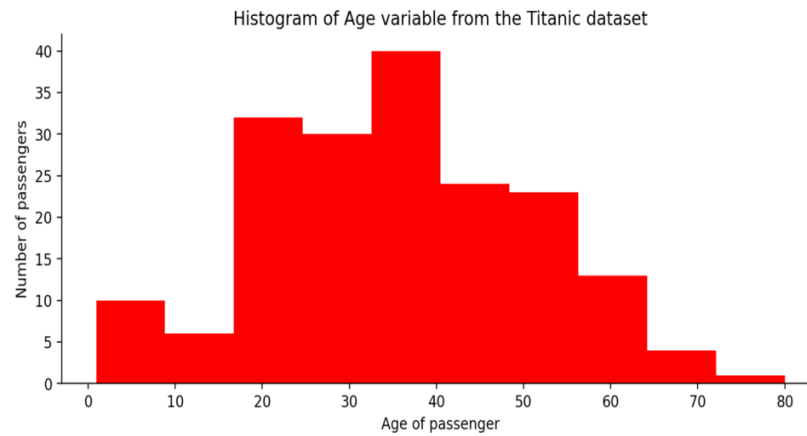


Figure 30: Changing the color of the graph: Both models were able to change the color of the graph from blue to red.

Chat2Vis (GPT 3.5 – Instruct):



Chat2Vis (Code Llama):

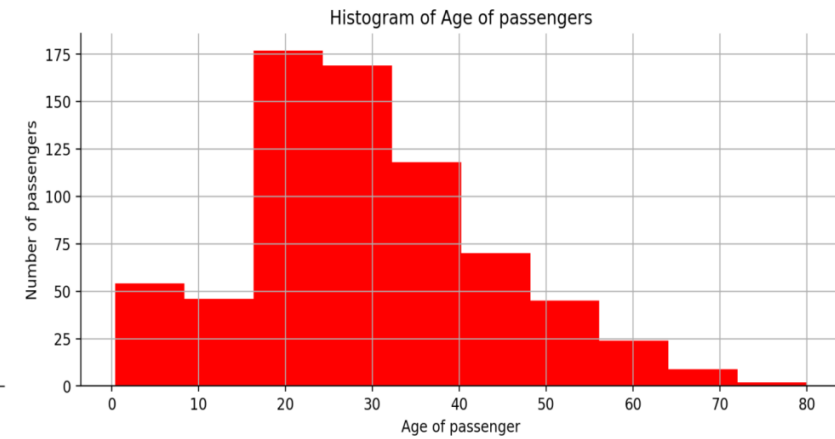
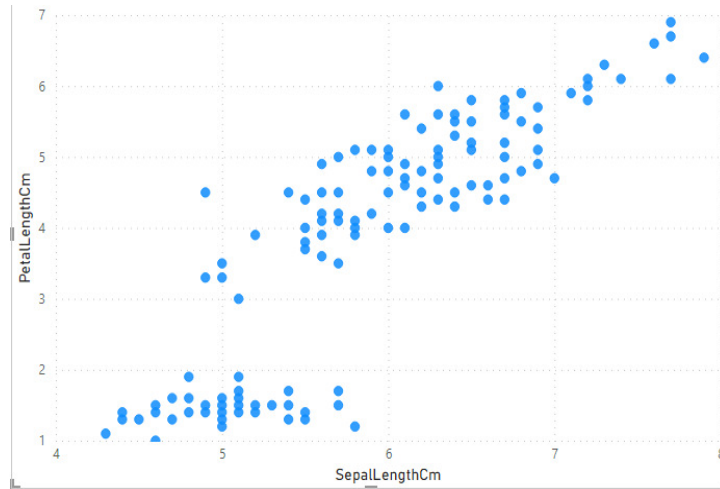
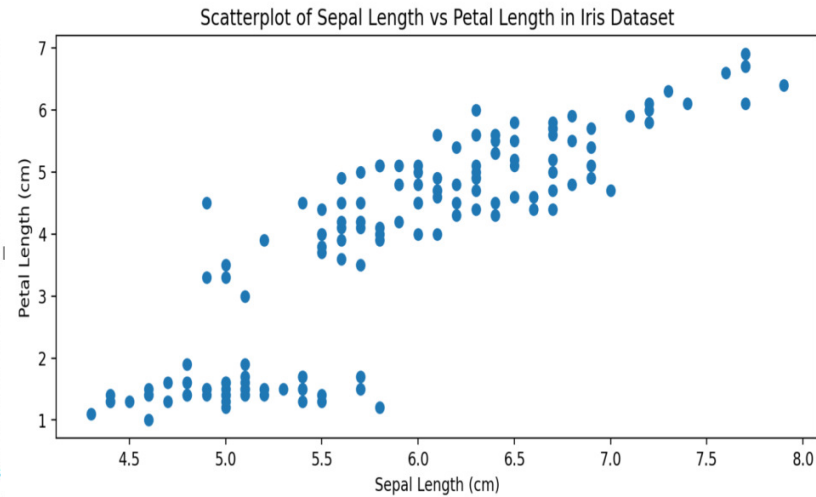


Figure 31: Reproduce a scatter plot from the Iris dataset according to Figure 5 in Appendix 3

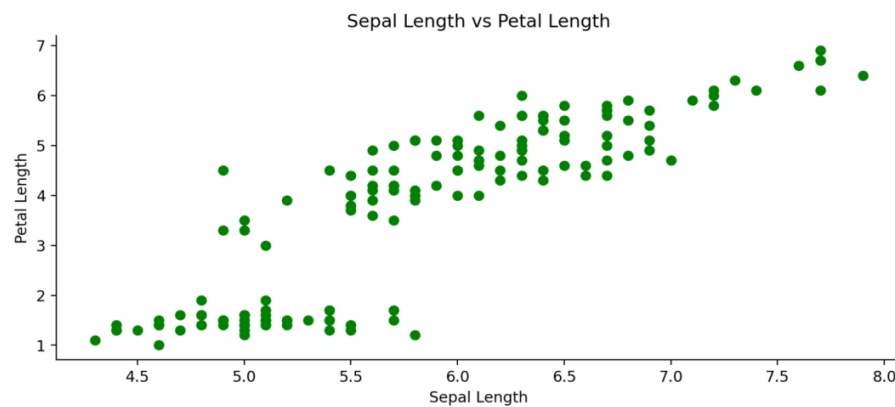
Power Bi:



Chat2Vis (GPT 3.5 – Instruct)



Code Llama:



All the three graphs give the idea of the two clusters forming. The chart from GPT 3.5 - instruct contains also the measure "cm" in both axes. For this graph, we build table 4, in the next page, with the relevant criteria and comments.

Table 4: Accuracy of the scatterplot.

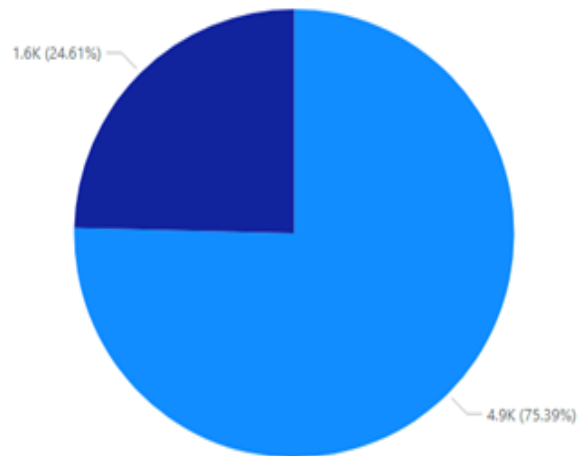
Criteria	Power Bi	Chat2Vis (GPT 3.5 – Instruct)	Chat2Vis (Code Llama)
Correctness of the Graph			
Each point has the right coordinate (approximate, by splitting the graphs area into smaller squares and observing the number and positioning of the dots)	Serves as benchmark for comparison	Correct number of points and positioning compared to Power BI	Correct number of points and positioning compared to Power BI
Completeness of data			
The graph includes all categories	Includes the range, up to petal length of 7 cm and sepal length of 8 cm	Includes the range, up to petal length of 7 cm and sepal length of 8 cm	Includes the range, up to petal length of 7 cm and sepal length of 8 cm
Scales			
Scale of x-axis (Petal length)	Starts from 1, up to 7, with steps from 1 cm	Starts from 1, up to 7, with steps from 1 cm	Starts from 1, up to 7, with steps from 1 cm
Scale of y-axis	Starts from 4, up to 8, with steps from 1 cm	Starts from 4.5, with steps from 0.5 cm	Starts from 4.5, with steps from 0.5 cm
Clarity			
Reading the labels	The labels are clear and readable, show also the unit (cm)	The labels are clear and readable, show also the unit (cm)	The labels are clear and readable However, it does not show the unit (cm)
Relevance			
Identifying groups (clusters)	There are 2 clusters visible. One above petal length 2 cm, and the other below	There are 2 clusters visible. One above petal length 2 cm, and the other below	There are 2 clusters visible. One above petal length 2 cm, and the other below

For the scatterplot, first we observe the various points in the graphs produced by Chat2Vis by considering smaller sections one by one. Both models (GPT 3.5 – Instruct and Code Llama) in this case, give the correct number of points. Next we check if the graphs include all the categories by looking at the ranges of the axis. Both graphs show all the range. Further down the table, we check also the scales used in the axis and we notice that 0.5 cm

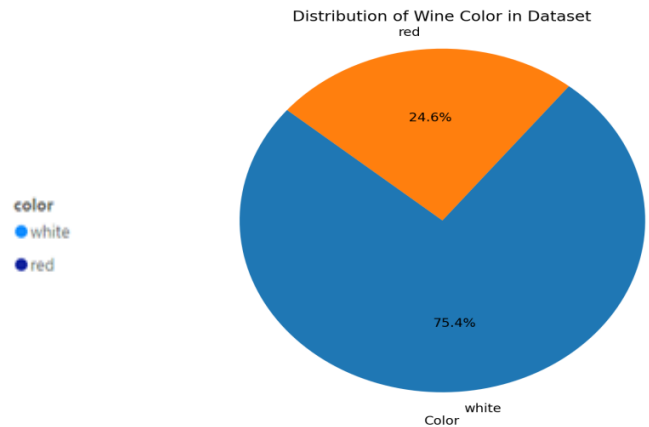
steps, instead of 1 cm, but we consider it acceptable as it does not affect the general view of the graph. The labels also here are clear and readable, but we notice that Code Llama does not show the units of the axis (cm), making it a little more difficult for the user. Lastly, we consider the relevance of the graphs and as we can see, both graphs show the two clusters forming above and below the petal length 2 cm.

Figure 32: Reproduce a pie chart form the Wines Dataset according to Figure 6 in Appendix 3

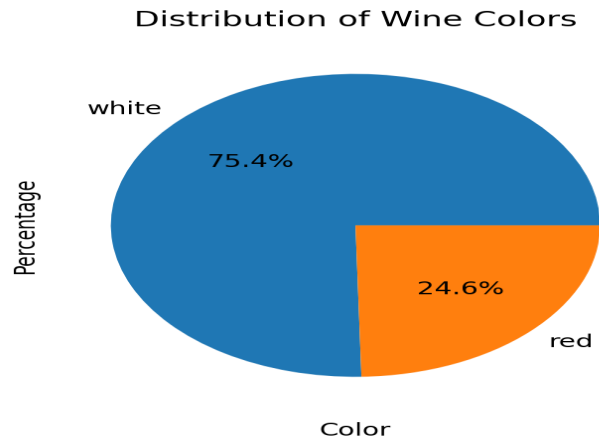
Power Bi:



Chat2Vis (GPT 3.5)



Chat2Vis (GPT 3.5 – Instruct)



For this task, the pie chart was produced from GPT 3.5 and GPT 3.5 Instruct. Both models give the correct percentages, but they position the red color wines in different parts of the circle. For this graph, we build table 5, in the next page with the relevant criteria and comments.

Table 5: accuracy of the pie chart

Criteria	Power Bi	Chat2Vis (GPT 3.5)	Chat2Vis (GPT 3.5 – Instruct)
Correctness of the Graph			
Correct percentages for each group	75.39% white, 24.61% red	75.4% white, 24.6% red	75.4% white, 24.6% red
Completeness of data			
The graph includes all categories (colors) of wines	The graph shows 2 groups	The graph shows 2 groups	The graph shows 2 groups
Scales			
Decimal points of the percentages	2 decimal points	1 decimal point	1 decimal point
Clarity			
Reading the labels	The labels are clear and readable, includes also the actual number of wines, besides percentages	The labels are clear and readable	The labels are clear and readable
Relevance			
Identifying groups of wine	2 groups of wines, as regards to color, white and red	2 groups of wines, as regards to color, white and red	2 groups of wines, as regards to color, white and red

We evaluate the correctness of the pie charts by looking at the percentages of each group. Power Bi shows 75.39% white and 24.61% red wines, while both models on Chat2Vis show 75.4% white and 24.6% red. We consider this small change a rounding and thus, acceptable. For the completeness, we can see that all graphs show two groups of wines, so we can say that they show the complete range of data. For the scales, we notice that Power Bi offers more details by showing two decimal points instead of one, and furthermore, it shows also the number of wines contained in each group. However, the labels of both graphs rendered by Chat2Vis are clear and readable. Lastly for the relevance, both graphs correctly show two groups of wines.

After presenting side-by-side all the graphs, it can be said that in all the visualization tasks, the LLMs in Chat2Vis were able to produce accurate graphs, with some minor changes from the ones in Power Bi. The most precise measure of accuracy was in the last task, where LLMs showed the correct percentages in the pie chart. However, for the histogram chart, we needed to re-perform the task in order to specify the number of bins, in order to evaluate accuracy better.

A final statement about the first research question, after taking in consideration time and accuracy of the graphs is what comes next. So, our question was how do LLM based tools compare to more traditional one, in terms of time and accuracy. In our case, the selected tools were Chat2Vis and Power Bi. Therefore, the answer of the first research question is that yes, from our visualization tasks

it resulted that Chat2Vis as a LLM-based tool can produce accurate visuals faster than Power Bi. Of course, this study does not intend to provide an exhausting number of visualization tasks. The tasks we performed were of a simple difficulty, typically used in the beginning of a Data Science task, aiming to get a feel of the data, by checking for outliers, balance of the dataset, etc.

5.3. User-friendliness

The following and last section of this part, will discuss the user-friendliness of Chat2Vis, compared to Power Bi. The use of NL queries, as compared to menu navigation, makes it more comfortable to produce graphs, even when you compare it to a familiar tool for the user, of which you know the menus. We show the various criteria used, in table 6 as well as the comments for each.

Table 6: User-friendliness criteria

Criteria	Chat2Vis comment	Power Bi comment
Data uploading time	Faster than Power Bi	Slower than Chat2Vis
Graph customization	Possible	Also possible, via the various menus
Transform the data after uploading	Not possible, you need to perform the changes in the CSV file and then upload again.	Possible
Transferring the visuals to another document	Possible. The procedure is straightforward, copy-paste.	Possible, but it is easier with Chat2Vis.
Uploading more than one file of data	Possible.	Also possible.
Building dashboard-like reports with multiple graphs.	Not possible. You can build only one visual at a time.	Possible. This is a key feature of Power Bi.

Data uploading in Chat2Vis is faster than Power Bi, making it a preferred tool when the user needs to build simple visuals that help to get a feel of the data. Chat2Vis supports also customizations, such as axes renaming or color, by adding sentences to the NL query. However, in tools such as Power Bi the user can also edit and transform the data once uploaded. This is not possible in Chat2Vis. If the data needs cleaning, this should be done previously in the CSV file.

Another preferred feature of Chat2Vis is the easiness of copying the visuals from the tool to the word document. Chat2Vis allows right-clicking the graphs and copying them to another tool or report. It should be mentioned that one of the tasks the users can achieve with Power Bi is building reports by putting different graphs, from different databases, in a single page and exporting it to PDF. This is not possible for the moment with Chat2Vis, because although the user can upload multiple datasets, the

data from which you intend to produce visuals should be selected before. It allows only one selection, meaning that it is possible to receive only one visual at a time.

After the above analysis, it is opportune to give an answer to the second research question, how user-friendly is Chat2Vis, for users with little or no programming experience. The answer is that yes, according to my experience in performing the visualization tasks, Chat2Vis has a friendly user-interface. You can easily find where to upload the data and where to input the NL query and the answer from the system for a NL query, it is also fast, being that in the form of a visual, or of an error message.

User-friendliness of LLM-based visualization tools (Chat2Vis) was the last part of the Results section. The next session will be about a more general discussion about the contributions and limitations of this study, as well as a final and broader discussion about LLMs and their visual-building potential in the future. Discussion will be also the final section of this paper. Interested reader can find some more detailed information in the appendices, after the References part.

6. Discussion

After performing the visualization tasks, we come to this part of the final discussion. In my research about other similar works on Chat2Vis, I was able to find several mentions of it as an LLM-based tool for visualization, but only one (Kavaz et al., 2023) actually used it to build visuals. This is where this work aims to give its contribution, by adding insights and practical experience from an independent user.

In reflecting critically on my research findings, it's important to consider how they align with the existing literature. While the literature emphasizes the potential of LLM-based tools like Chat2Vis for user-friendly visualization creation, my study provides practical evidence supporting these claims. However, it also highlights the limitations that come with relying solely on such tools for comprehensive data analysis and reporting.

Methodological Reflections

The methodological choices in this study were largely effective in achieving the research goals. Using simple visualization tasks from the Data Science course of UHasselt allowed for a straightforward assessment of Chat2Vis's capabilities in generating visuals from NL queries. However, this choice also introduced limitations. The simplicity of the tasks did not allow for an evaluation of Chat2Vis's performance with more complex, real-world data visualization needs. Future studies should consider incorporating more complex and varied tasks to better assess the tool's capabilities and limitations.

Another methodological aspect to reflect on is the exclusive use of English NL queries. This choice was made to standardize the evaluation and because English is widely used in data science. However, Chat2Vis supports multiple languages (Madigga et al., 2023), and testing with other languages could provide a more comprehensive evaluation of its usability and accessibility. Future research could include a broader range of languages to understand better the tool's applicability in a global context.

Lastly, future work should consider the use of experts and software for a more detailed evaluation of the accuracy of the graphs rendered by the LLM tool. We use only 5 specific criteria and visual comparison in this paper, but experts can add even more criteria, as well as more detailed evaluation of the accuracy of the graphs.

Limitations and Future Research

One of the primary limitations of this study is the assumption that users are familiar with Power BI, which was used as a benchmark for comparison. This assumption may not hold true for all potential users of Chat2Vis. A more comprehensive study could include comparisons with other visualization tools that users might not be as familiar with, to more accurately assess the user-friendliness of LLM-based tools.

Another limitation is the exclusion of lengthy NL queries and complex prompt engineering. While the simplicity of the tasks used provided a clear demonstration of Chat2Vis's basic functionality, it did not challenge the tool's ability to handle more complex queries that may be necessary for advanced data analysis. Future research could explore the limits of Chat2Vis's NL processing capabilities and its effectiveness in handling more intricate and detailed prompts.

Furthermore, future researchers on LLM-based visualization tools should consider recruiting different users from different professions, or even a specific group, who normally use visualization tools in their jobs and record their findings, in order to add more value to the user experience study, which is done by a single user in this paper.

Expectations and Findings

When starting this research, I expected to find that Chat2Vis would significantly simplify the process of creating data visualizations. This expectation was met to an extent, as the tool indeed facilitates the creation of visuals through NL queries. However, the study also revealed that while Chat2Vis is useful for generating basic visuals, it lacks the advanced data manipulation and integration capabilities required for more comprehensive data analysis and reporting.

Scientific Value and Suggestions for Future Research

This study contributes to existing work by providing practical insights into the use of LLM-based tools for data visualization, specifically Chat2Vis. It underscores the potential of these tools to enhance user experience by simplifying the visualization process. However, it also highlights the need for further development to incorporate advanced features such as data cleaning, transformation, and integration of multiple datasets.

Future research should focus on several areas to build on these findings:

- **Expansion of the range of tasks and queries** by incorporating more complex and varied visualization tasks to better understand the tool's capabilities.
- **Multi-language evaluation**: Assessing Chat2Vis's performance with queries in different languages to evaluate its capabilities in a more global environment.
- **Comparison with a broader set of tools**: Including a variety of visualization tools in the comparison to provide a more comprehensive evaluation of user-friendliness.
- **Include a more comprehensive user experience study**, by recruiting more users to assess more correctly the user-friendliness of the chosen tool.
- **Include experts and software** for the accuracy evaluation of the graphs.
- **Exploring advanced features**: Investigating the integration of data manipulation, cleaning, and reporting functionalities to enhance the tool's utility for comprehensive data analysis.

Personal Reflections

I would like to close this paper with my personal thought after conducting this research, as a user of data visualization tools in my daily work as a finance specialist. The possibility to build accurate visuals by using NL is indeed very helpful, but the needs of a normal user go beyond producing a visual. They include also the possibility to clean and transform the data, as well as to build complex reports from multiple datasets, which Chat2Vis does not offer for the moment. So my final thought is that rather than replacing traditional visualization tools, LLM-based tools can offer a powerful addition to them, allowing user to have in the same place the easiness of building graphs from NL as well as the possibility to edit, transform, data and build complex reports.

7. References

- Alex Rutherford. (2024). BERT vs. GPT. Powerbrainai. Retrieved March 8, 2024, from <https://powerbrainai.com/bert-vs-gpt/>.
- Alsulami, S., Alsobhi, A., Alabdli, A., Alafif, T., Jassas, M., & Albishre, K. AutoGrader: Automatic Test Grader Based on ChatGPT and Term Frequency-inverse Document Frequency.
- C. Liu, Y. Han, R. Jiang, and X. Yuan, "ADVISor: Automatic visualization answer for natural-language question on tabular data," in Proc. IEEE 14th Pacific Vis. Symp. (PacificVis), Apr. 2021, pp. 11–20.
- Cain, W. (2024). Prompting Change: Exploring Prompt Engineering in Large Language Model AI and Its Potential to Transform Education. TechTrends, 68(1), 47-57.
- Chat2Vis. (2024). Creating Visualisations using Natural Language with ChatGPT and Code Llama. Retrieved April 6, 2024, from <https://chat2vis.streamlit.app/>
- Cecere, G., Corrocher, N., & Battaglia, R. D. (2015). Innovation and competition in the smartphone industry: Is there a dominant design?. Telecommunications Policy, 39(3-4), 162-175.
- Chen, Z., Zhang, C., Wang, Q., Troidl, J., Warchol, S., Beyer, J., ... & Pfister, H. (2023). Beyond Generating Code: Evaluating GPT on a Data Visualization Course. arXiv preprint arXiv:2306.02914.
- Choi, J. H., Hickman, K. E., Monahan, A. B., & Schwarcz, D. (2021). ChatGPT goes to law school. J. Legal Educ., 71, 387.
- Dayanithi. (2023). Streamlit in 3 Minutes. Medium. Retrieved March 11, 2024, from <https://medium.com/data-and-beyond/streamlit-d357935b9c>
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Dibia, V. (2023). Lida: A tool for automatic generation of grammar-agnostic visualizations and infographics using large language models. arXiv preprint arXiv:2303.02927.
- Ekin, S. (2023). Prompt engineering for ChatGPT: a quick guide to techniques, tips, and best practices. Authorea Preprints.
- Fortune Business Insight. (2024). *Data Visualization market*. Fortune Business Insight. Retrieved March 18, 2024, from <https://www.fortunebusinessinsights.com/data-visualization-market-103259>

- Gao, T., Dontcheva, M., Adar, E., Liu, Z., & Karahalios, K. G. (2015, November). Datatone: Managing ambiguity in natural language interfaces for data visualization. In *Proceedings of the 28th annual acm symposium on user interface software & technology* (pp. 489-500).
- He, Y., Cao, S., Shi, Y., Chen, Q., Xu, K., & Cao, N. (2024). Leveraging Large Models for Crafting Narrative Visualization: A Survey. *arXiv preprint arXiv:2401.14010*.
- HuggingFace. (2024). User access tokens. What are User Access Tokens? Retrieved April 6, 2024, from <https://huggingface.co/docs/hub/en/security-tokens>
- Karat, J. (1997). User-centered software evaluation methodologies. In *Handbook of human-computer interaction* (pp. 689-704). North-Holland.
- Katherine Haan. (2024). *The Best Data Visualization Tools Of 2024*. Forbes. Retrieved March 18, 2024, from <https://www.forbes.com/advisor/business/software/best-data-visualization-tools/>
- Katib, I., Assiri, F. Y., Abdushkour, H. A., Hamed, D., & Ragab, M. (2023). Differentiating Chat Generative Pretrained Transformer from Humans: Detecting ChatGPT-Generated Text and Human Text Using Machine Learning. *Mathematics*, 11(15), 3400.
- Kavaz, E., Puig, A., & Rodríguez, I. (2023). Chatbot-based natural language interfaces for data visualisation: A scoping review. *Applied Sciences*, 13(12), 7025.
- Khurana, D., Koli, A., Khatter, K., & Singh, S. (2023). Natural language processing: State of the art, current trends and challenges. *Multimedia tools and applications*, 82(3), 3713-3744.
- Kocoń, J., Cichecki, I., Kaszyca, O., Kochanek, M., Szydło, D., Baran, J., ... & Kazienko, P. (2023). ChatGPT: Jack of all trades, master of none. *Information Fusion*, 101861.
- Li, G., Wang, X., Aodeng, G., Zheng, S., Zhang, Y., Ou, C., ... & Liu, C. H. (2024). Visualization Generation with Large Language Models: An Evaluation. *arXiv preprint arXiv:2401.11255*.
- Liu, Y., Han, T., Ma, S., Zhang, J., Yang, Y., Tian, J., ... & Ge, B. (2023). Summary of chatgpt-related research and perspective towards the future of large language models. *Meta-Radiology*, 100017.
- Luo, Y., Tang, J., & Li, G. (2021). nvBench: A large-scale synthesized dataset for cross-domain natural language to visualization task. *arXiv preprint arXiv:2112.12926*.
- Maddigan, P., & Susnjak, T. (2023). Chat2vis: Fine-tuning data visualisations using multilingual natural language text and pre-trained large language models. *arXiv preprint arXiv:2303.14292*.
- Maddigan, P., & Susnjak, T. (2023). Chat2vis: Generating data visualisations via natural language using chatgpt, codex and gpt-3 large language models. *IEEE Access*.

- Manish Poddar. (2023). Prompt Engineering for Large Language Models. Medium. Retrieved March 11, 2024, from <https://manish-poddar.medium.com/prompt-engineering-for-large-language-models-9f62cf4a00d7>
- Markets and markets. (2021). *Data Visualization Tools market*. Forbes. Retrieved March 18, 2024, from <https://www.marketsandmarkets.com/Market-Reports/data-visualization-tools-market-94728248.html>
- Microsoft. (2023). *Microsoft Lidademo*. Github. Retrieved March 18, 2024, from <https://github.com/c17hawke/quickstart-microsoft-lida-demo>
- Narechania, A., Srinivasan, A., & Stasko, J. (2020). NL4DV: A toolkit for generating analytic specifications for data visualization from natural language queries. *IEEE Transactions on Visualization and Computer Graphics*, 27(2), 369-379.
- Noever, D., & McKee, F. (2023). Numeracy from Literacy: Data Science as an Emergent Skill from Large Language Models. *arXiv preprint arXiv:2301.13382*.
- Nori, H., King, N., McKinney, S. M., Carignan, D., & Horvitz, E. (2023). Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.
- OpenAI. (2023). Prompt Engineering. OpenAI. Retrieved March 11, 2024, from <https://platform.openai.com/docs/guides/prompt-engineering>
- OpenAI. (2024). API keys. OpenAI. Retrieved April 6, 2024, from <https://platform.openai.com/api-keys>
- Rajagopal, D. (2023). THE FUTURE OF DATA VISUALIZATION IN THE AGE OF ARTIFICIAL INTELLIGENCE (AI).
- Richardson, J., Sallam, R., Schlegel, K., Kronz, A., & Sun, J. (2020). Magic quadrant for analytics and business intelligence platforms. Gartner ID G00386610.
- Sartori, C. C., Blum, C., & Ochoa, G. (2024). Large Language Models for the Automated Analysis of Optimization Algorithms. *arXiv preprint arXiv:2402.08472*.
- Sial, A. H., Rashdi, S. Y. S., & Khan, A. H. (2021). Comparative analysis of data visualization libraries Matplotlib and Seaborn in Python. *International Journal*, 10(1).
- TrendFeedr. (2024). *Behind the AI Boom: Large Language Model (LLM) Trends*. TrendFeedr. Retrieved March 18, 2024, from <https://trendfeedr.com/blog/large-language-model-llm-trends/>
- Wang, L., Zhang, S., Wang, Y., Lim, E. P., & Wang, Y. (2023). LLM4Vis: Explainable visualization recommendation using ChatGPT. *arXiv preprint arXiv:2310.07652*.

Wang, Y., Hou, Z., Shen, L., Wu, T., Wang, J., Huang, H., ... & Zhang, D. (2022). Towards natural language-based visualization authoring. *IEEE Transactions on Visualization and Computer Graphics*, 29(1), 1222-1232.

White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., ... & Schmidt, D. C. (2023). A prompt pattern catalog to enhance prompt engineering with chatgpt. *arXiv preprint arXiv:2302.11382*.

8. Appendices

8.1 Appendix 1, Visualization tasks

The tasks will be divided in 3 types:

In type 1 tasks, we will be producing visuals from the datasets. In type 2 tasks we will be reproducing two charts and in type 3 tasks we will try to take the produced visuals from the respective tools (Power Bi and Chat2Vis) and bring them to the main word document of the thesis.

Main task type 1: Creating visuals from data

- Build a histogram of the Age variable, on the Titanic dataset.

Follow up on task 1, to customize

- Try to change names of the axis
- Try to change the colors of the graph

Main task type 2: Reproducing a visual

- Reproduce the scatter plot in Figure 5, using Iris dataset.
- Reproduce the visual in Figure 6, using the Wines Dataset.

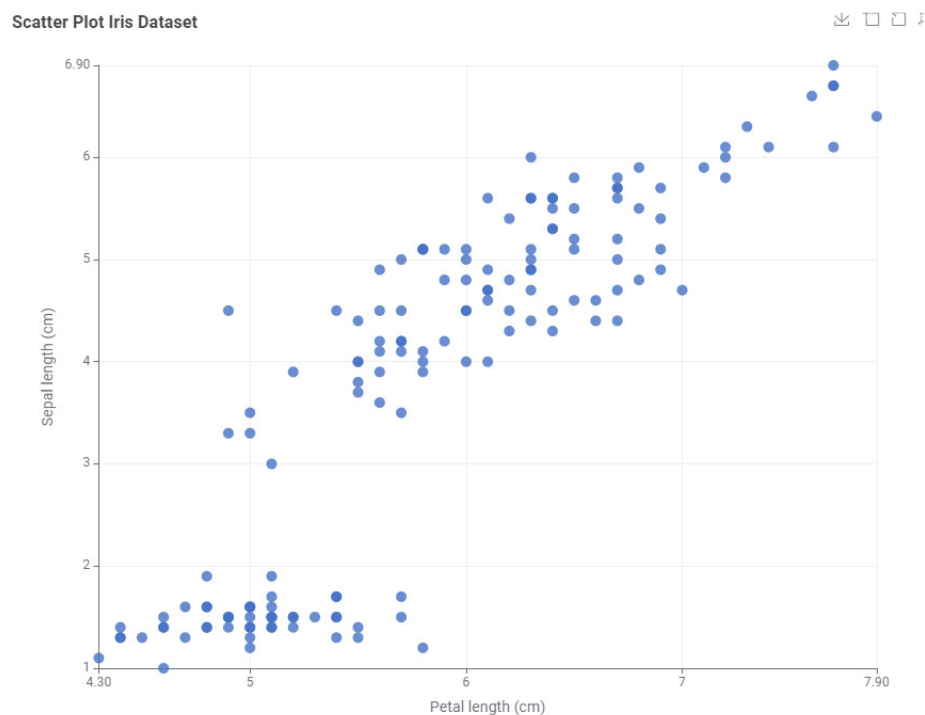


Figure 5: Scatter Plot from the iris dataset

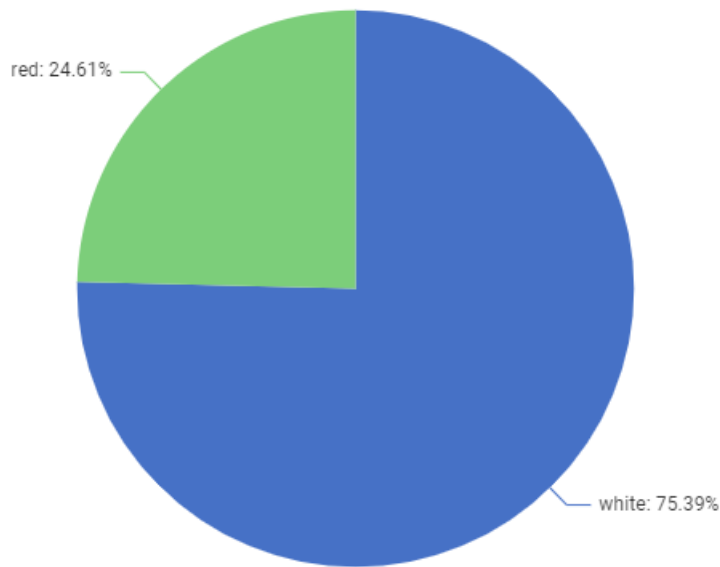


Figure 6: Pie Chart Wine dataset


Main task type 3: Taking the visual from a tool to another.

- Take all the created visuals from Chat2Vis and insert in the word document of this thesis. Do the same with Power Bi.

8.2 Appendix 2, Setting up Chat2Vis

Before proceeding with the set up of Chat2Vis, a short presentation of the interface of this tool will be shown.

Chat2Vis interface:

 Paula Maddigan and Teo Susnjak

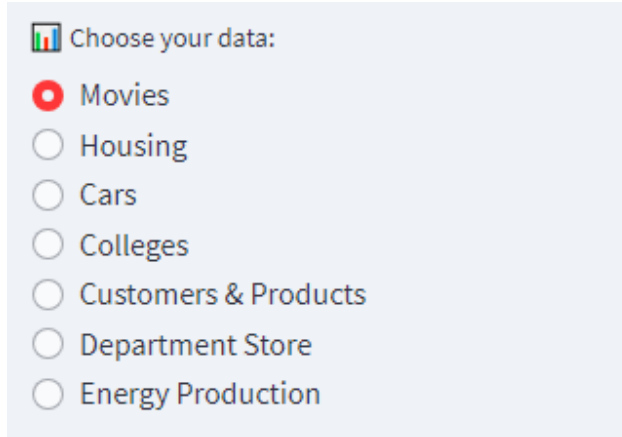
Chat2VIS: Generating Data Visualisations via Natural Language using ChatGPT, Codex and GPT-3 Large Language Models
(<https://doi.org/10.1109/ACCESS.2023.3274199>)

Chat2VIS: Fine-Tuning Data Visualisations using Multilingual Natural Language Text and Pre-Trained Large Language Models
(<https://doi.org/10.48550/arXiv.2303.14292>)

[Blog](#) by Paula Maddigan

The link where we can find Chat2Vis is: <https://chat2vis.streamlit.app/>. Chat2Vis interface consists of two parts; there is a grey column to the left of the screen, showing the names of the authors, the link to their 2 published papers, as well the link to the blog of one of the authors Paula Maddigan. This is shown also in Figure 7.

Figure 7: Chat2Vis interface (1 of 6)



By continuing below, it can be seen that Chat2Vis has already 7 datasets uploaded and allows the user to choose one of them and create visualization. This is shown in Figure 8.

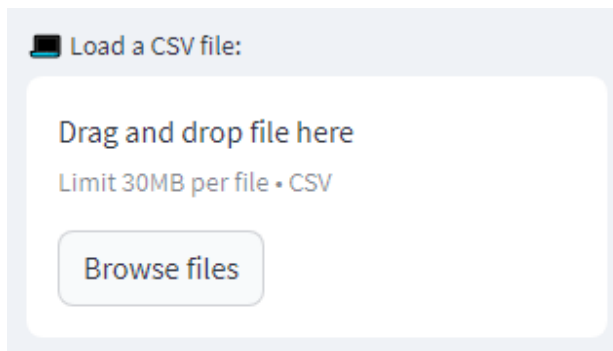
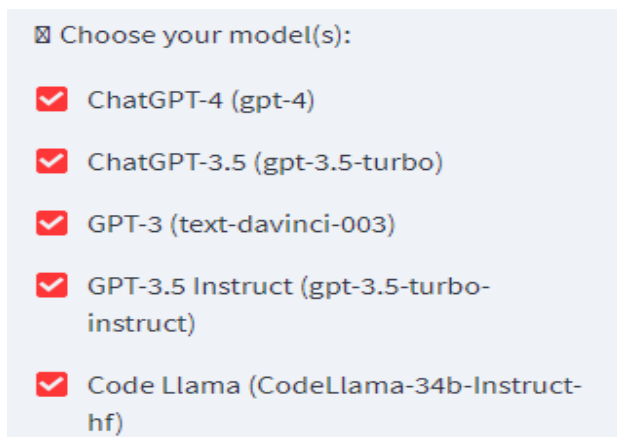


Figure 8: Chat2Vis interface (2 of 6)

By continuing below in the grey column, we come to the section where the users can upload their own datasets. As of April 6 2024, we can see that Chat2Vis accepts only CSV files. This is shown in figure 8.

Figure 8-bis: Chat2Vis interface (3 of 6)



The next session of the grey column presents the 5 LLMs from which Chat2Vis is able to produce visuals, as of April 6, 2024. This is shown in Figure 9.

Figure 9: Chat2Vis interface (4 of 6)

The main section of the Chat2Vis interface (on the right of the grey column), allows the users to input their NL query and click then click on “go” to see the visual. Above this sub-section there are two smaller sub-sections for the OpenAI Key and the HuggingFace Key. The little question mark on the top-right shows for which LLM these keys are needed, when you hover with your mouse above them. This section is shown in figure 10, below.

Chat2VIS

Creating Visualisations using Natural Language with ChatGPT and Code Llama

🔑 OpenAI Key: ?

🔑 HuggingFace Key: ?

🗣️ What would you like to visualise?

Figure 10: Chat2Vis interface (5 of 6)

The last section is located below of the “go” button and allows the user to have an overview of the uploaded datasets, each displayed in a separate window. The datasets uploaded by the user get displayed here as well, after uploading them. This is shown in Figure 11, below.

Movies
Housing
Cars
Colleges
Customers & Products
Department Store
Energy Production

Movies

Title	Worldwide Gross	Production Budget	Release Year	Content Rating	Running Time	Genre	Creative Type
From Dusk Till Dawn	25,728,961	20,000,000	1,996	R	107	Horror	Fantasy
Broken Arrow	148,345,997	65,000,000	1,996	R	108	Action	Contemporary F
City Hall	20,278,055	40,000,000	1,996	R	111	Drama	Contemporary F
Happy Gilmore	38,623,460	10,000,000	1,996	PG-13	92	Comedy	Contemporary F
Fargo	51,204,567	7,000,000	1,996	R	87	Thriller	Contemporary F
The Craft	55,669,466	15,000,000	1,996	R	100	Thriller	Fantasy
Twister	495,900,000	88,000,000	1,996	PG-13	117	Action	Contemporary F
Dragonheart	104,364,680	57,000,000	1,996	PG-13	108	Adventure	Fantasy
The Phantom	17,220,599	45,000,000	1,996	PG	100	Action	Super Hero
The Rock	336,069,511	75,000,000	1,996	R	136	Action	Contemporary F

Datasets courtesy of NL4DV, nvBench and ADVISor

Figure 11: Chat2Vis interface (6 of 6)

Chat2Vis set-up

In order to use Chat2Vis we need 2 kinds of keys, an OpenAI Key and a HuggingFace Key. The first one is needed for ChatGPT and the second for Code Llama.

OpenAI key:

In order to obtain the OpenAI key, I logged in to the link <https://platform.openai.com/api-keys> and logged in with my Google account. The next step is to verify the phone number. You receive a SMS with a code from OpenAi, which you need to put on the website. Next, you create the key, which you can copy and paste to the Chat2Vis website. The user may proceed directly to produce visuals only with OpenAI key, by "un-clicking" the Code Llama option from the "Choose your model(s):" section at the bottom of the grey column.

To check if Chat2Vis is working we use the following NL query to produce the visual, by using the datasets that are already uploaded: "Show a pie chart of the Movies by genre." Chat2Vis was able to produce the required visual from the GPT-3.5 instruct LLM version. The other OPENAI LLM versions produced error messages. This is shown in Figure 12, below:

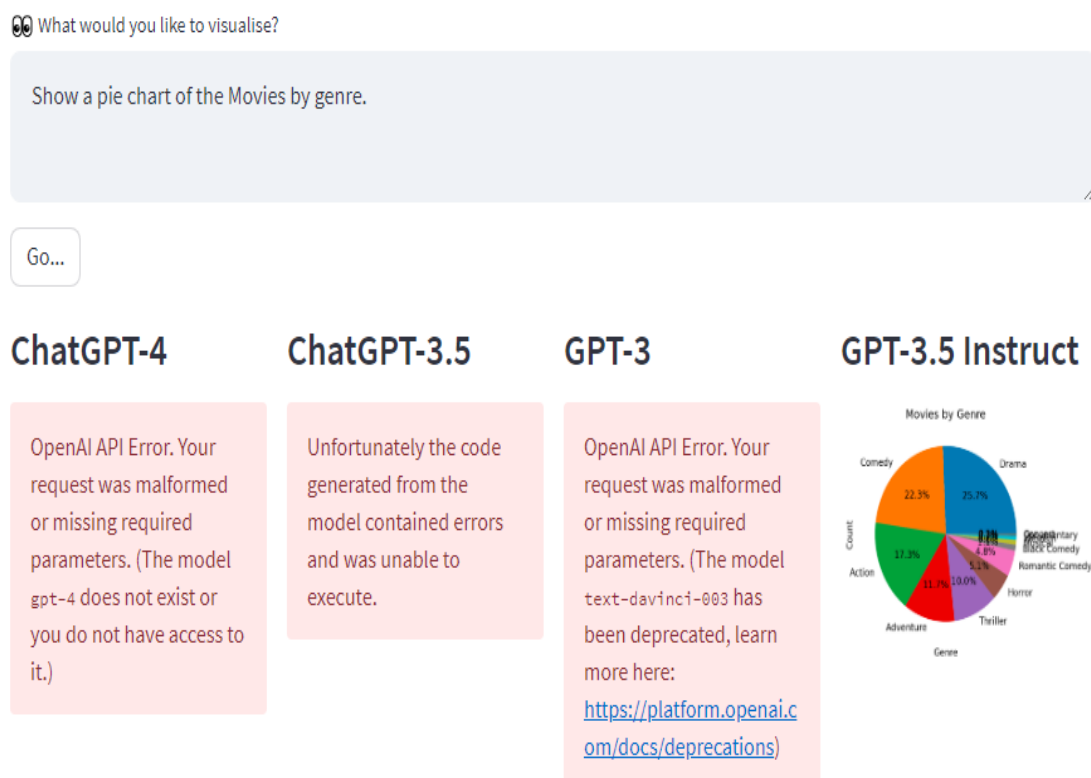


Figure 12: Pie chart of the Movies dataset, by genre.

HuggingFace Key:

In order to obtain a HuggingFace key I clicked on the link:

<https://huggingface.co/docs/hub/en/security-tokens>. Then I clicked on settings to generate the token to copy to the Chat2Vis website. We use the same NL query, "Show a pie chart of the Movies by genre." to check the results. This time ChatGPT 3.5 and ChatGPT 3.5 instruct produced the visual, but all the other models produced error messages. This is shown in Figure 13 below.

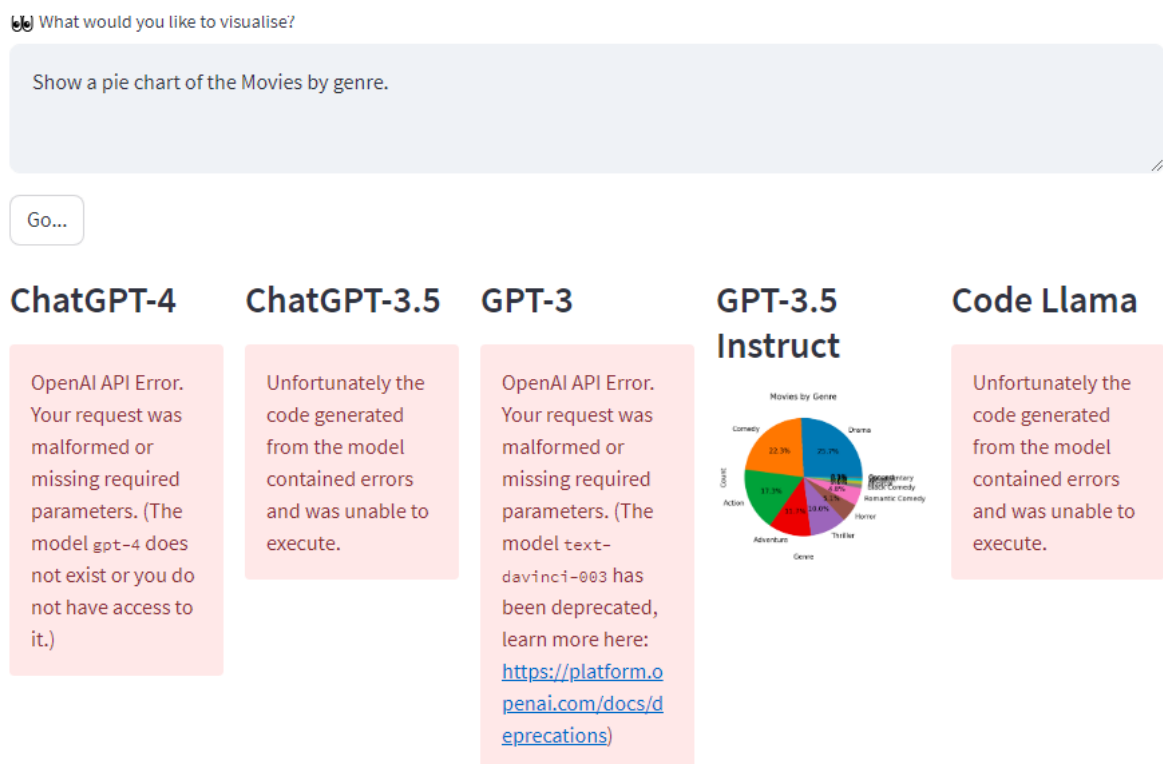


Figure 13: Pie chart of the Movies dataset, by genre.

8.3 Appendix 3, Experiments

Main task type 1: Creating visuals from data

- Build a histogram of the Age variable, on the Titanic dataset.

Figure 14: Histogram of Age, Power BI

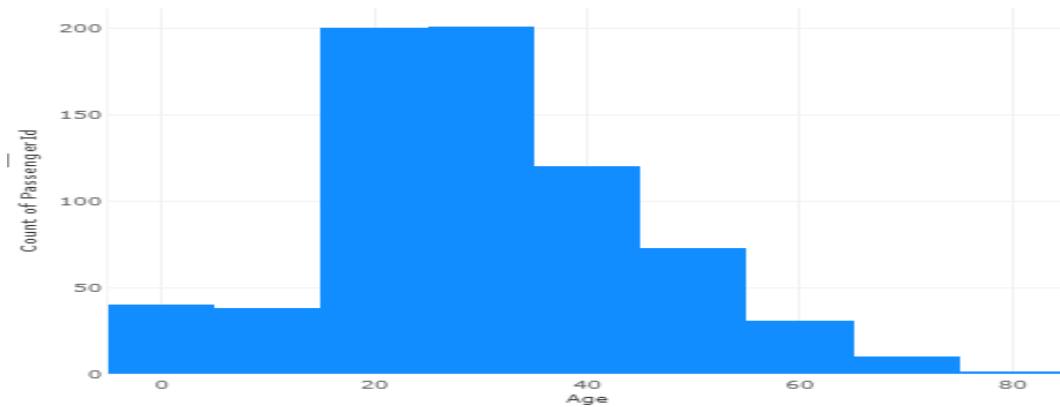


Figure 15: Histogram of Age, Chat2Vis, GPT-3.5 Instruct

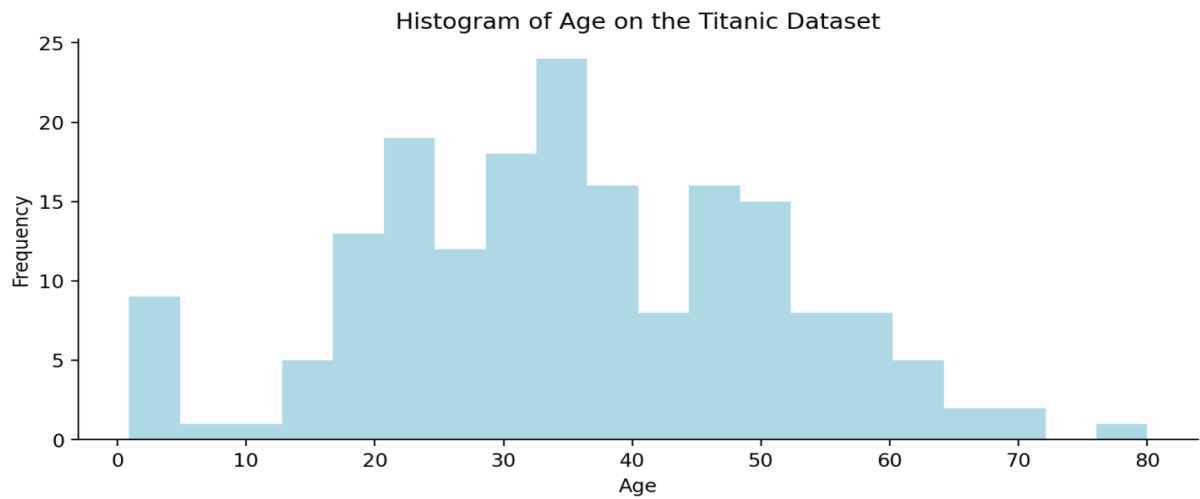
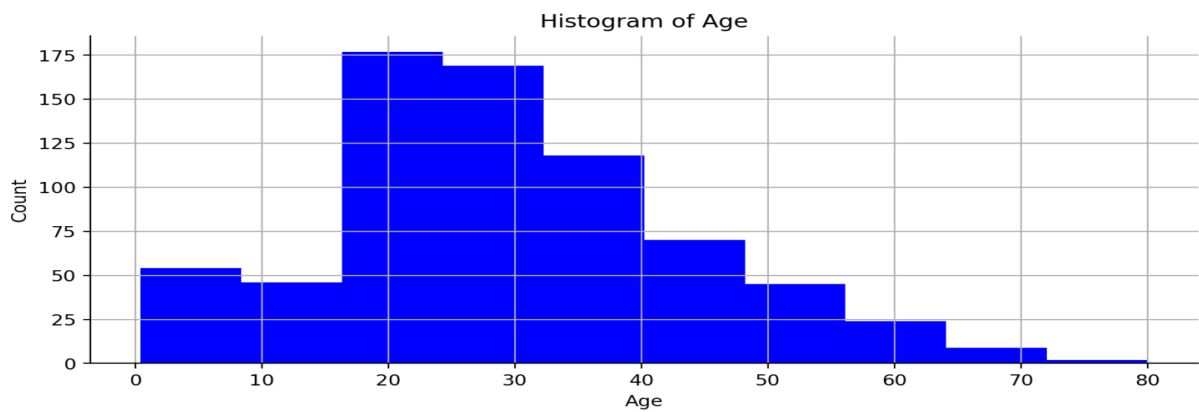


Figure 16: Histogram of Age, Chat2Vis, Code Llama:



- Try to change names of the axis for the histogram of Age

Figure 17: Changing the names of the axis, Power Bi

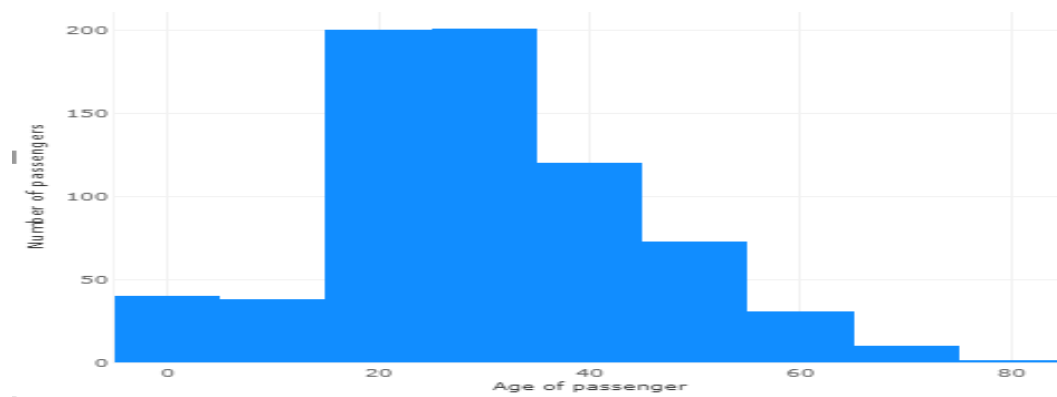


Figure 18: Changing the names of the axis, Chat2Vis (GPT-3.5 Instruct)

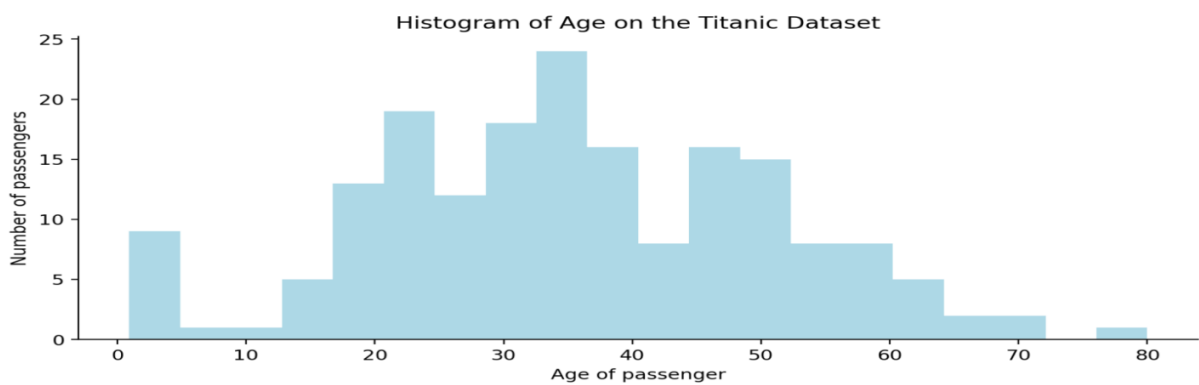
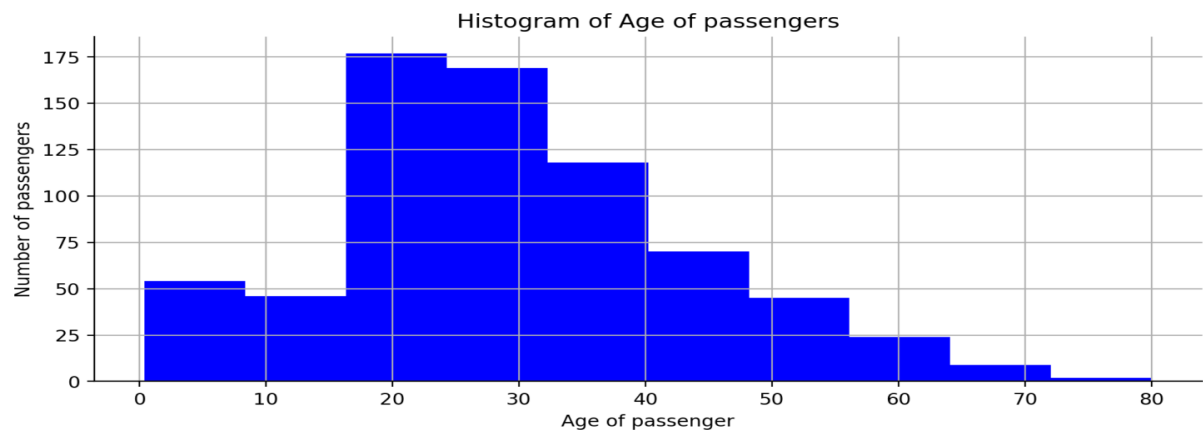


Figure 19: Changing the names of the axis, Chat2Vis, Code Llama:



- Try to change the colors of the graph for the histogram of Age

Figure 20: Changing the color of the graph, Power BI

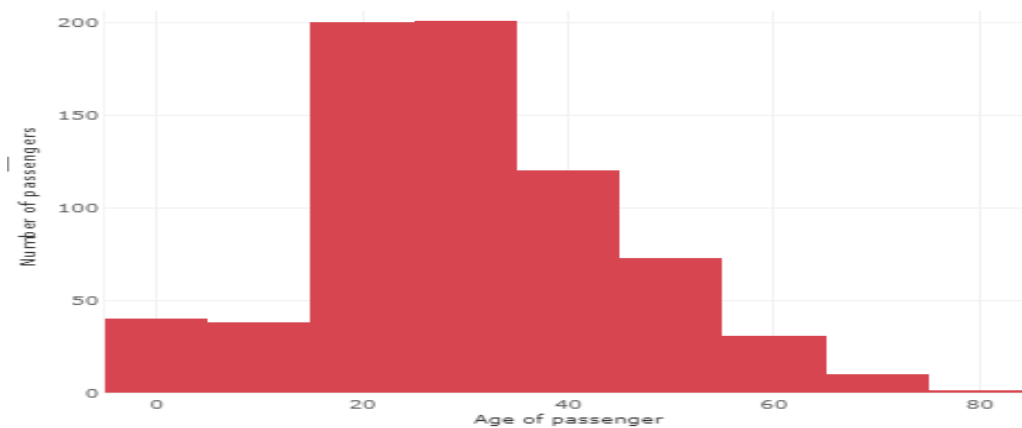


Figure 21: Changing the color of the graph, Chat2Vis (GPT-3.5 Instruct)

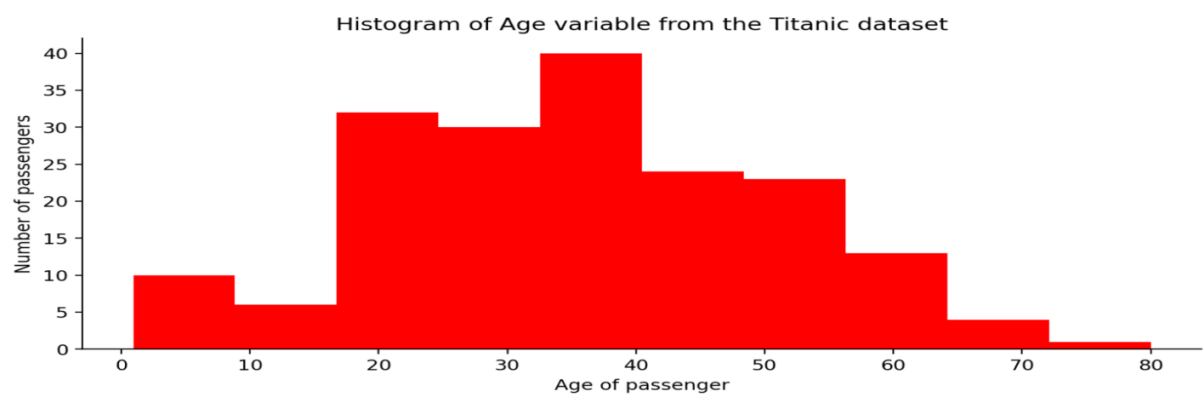
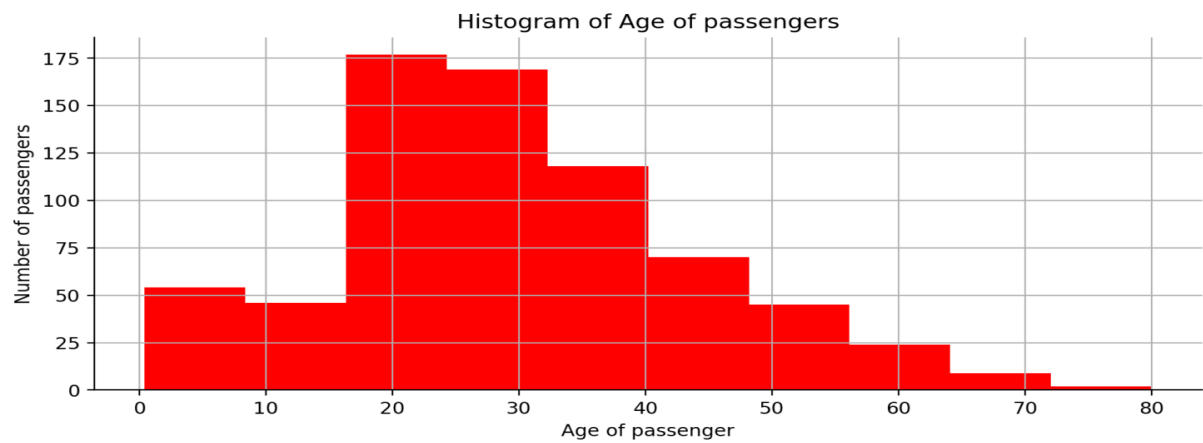


Figure 22: Changing the color of the graph, Chat2Vis, Code Llama



- Reproduce the scatter plot in Figure 5, using Iris dataset.

Figure 23: Scatter plot, Power Bi:

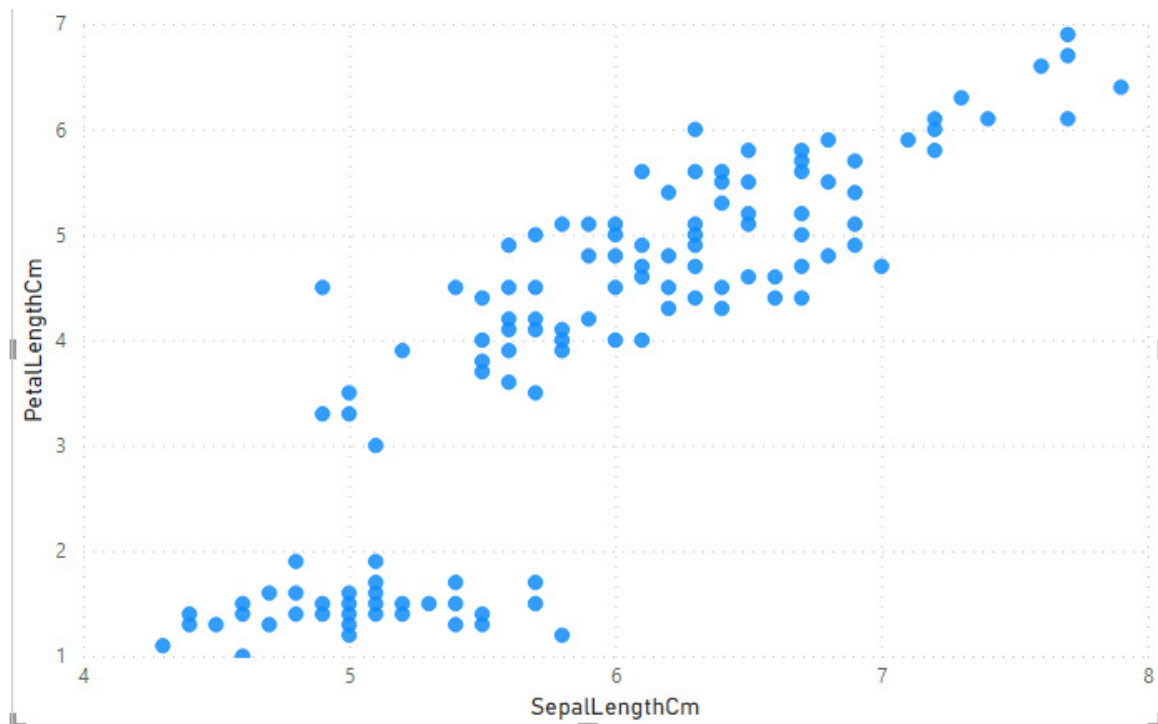


Figure 24: Scatter plot, Chat2Vis (ChatGPT-3.5)

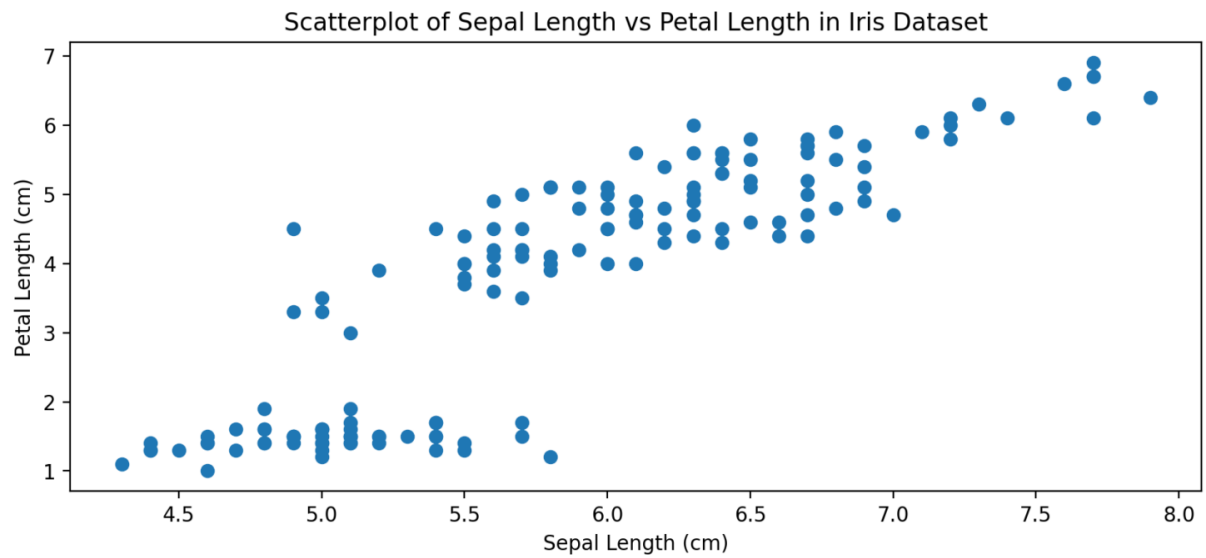
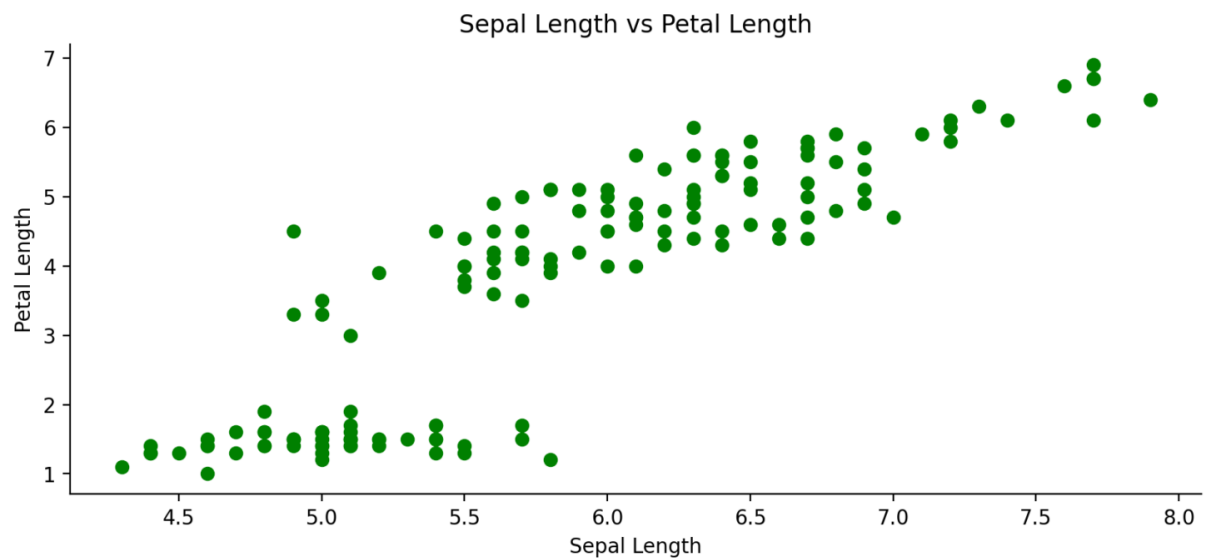


Figure 25: Scatter plot Chat2Vis, Code Llama



- Reproduce the visual in Figure 6, using the Wines Dataset.

Figure 26: Pie chart, Power BI:

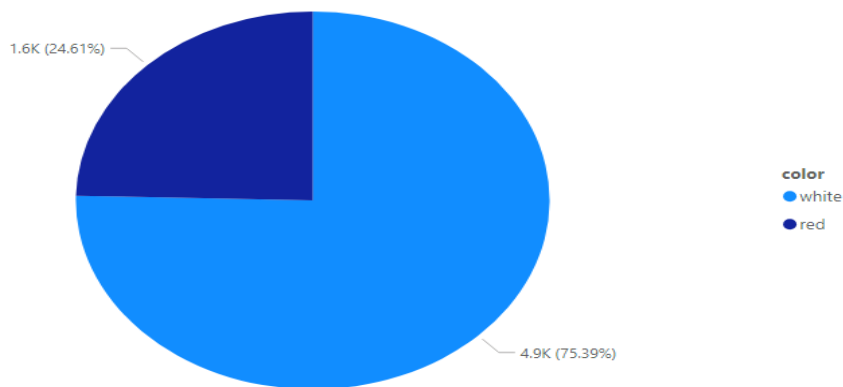
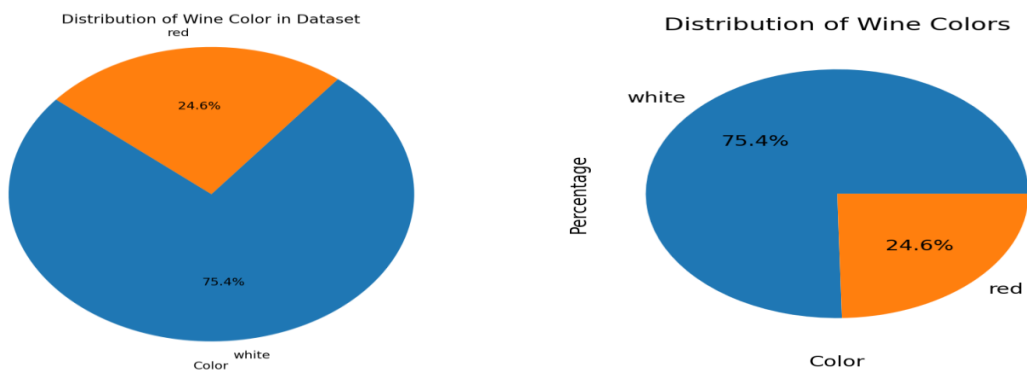


Figure 27: Chat2Vis (ChatGPT-3.5 and GPT-3.5 Instruct)



- Take all the created visuals from Chat2Vis and insert in the word document of this thesis. Do the same with Power Bi.

In order to copy a graph from Chat2Vis, is straightforward. You can right-click the graph, select "copy" and then paste it to the word document. I wasn't able to do the same with Power Bi. In order to get the produced visuals to the word document, a print-screen was needed.