Institute of Software Engineering

Bachelorarbeit

AI-Assisted Authoring in the Software Engineering Domain: An Empirical Study on using ChatGPT

Markus Walter

Course of Study:	Softwaretechnik
Examiner:	Prof. Dr. Stefan Wagner
Supervisor:	Markus Haug, (Marvin Muñoz Barón, Kasra Habib, Tobias Eisenreich, Jonas Fritzsch)
Commenced:	2023-11-15
Completed:	2024-05-15

Abstract

ChatGPT is a popular AI tool with potentially many uses. This thesis looks at how ChatGPT 3.5 can be utilized to improve the academic writing process in the Software Engineering domain by interviewing ChatGPT users from this domain and performing a literature review. The result is a guideline that offers a general workflow and prompting templates for utilizing ChatGPT as an editing tool. An evaluation experiment was performed, suggesting, but not proving, major time savings and quality improvements for individuals with linguistic barriers. In addition, a final interview with the participants resulted in an improved version of the guideline.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Context and Motivation

With the release of the artificial intelligence (short AI) chatbot ChatGPT in November 2022, OpenAI was suddenly catapulted into the spotlight, with news coverage ramping up in early 2023 [JD]. While it was generally praised for its ability to speak like a human, it was regularly reported on as a gloom and doom technology, that is out to take your job and disrupt society [Loc; Roo]. Even though, the chatbot has been available for some time now, mass layoffs have not (yet) happened [AD]. Instead, ChatGPT has become another potential tool for your work, alongside many others. Looking into how ChatGPT can be used by Software Engineering researchers to help them write their scientific papers will be the main focus of this thesis.

What is ChatGPT, and how does it work?

ChatGPT stands for Chat Generative Pre-Trained Transformer. The 'Chat' describes the user interface, while the rest describes the technology behind it. The basic interface of ChatGPT is a chat window similar to any messaging app - you enter text, hit send and shortly after, you get a reply. It is quite similar to texting a friend or colleague, which allows people without any knowledge of artificial intelligence to use it, as demonstrated in fig 1.1.
ChatGPT is based on the transformer technology published by Google researchers in 2017, who developed a new way of analyzing text data with neural networks, a group of artificial intelligence algorithms. Contrary to earlier similar technologies, these new transformers did not need supervised training data, meaning data that has been labeled, e.g a poem receiving a 'poem' label. Instead, they were able to use unlabeled data, making training easier. The transformers take the text input by users and split it into tokens, who are analyzed for context and weighted for importance to create the output [VSP+17]. These weights and importance calculations are made with the help of the training data that has been analyzed before the model was published, hence the 'Pre-Trained' in GPT. Simplified, when you use ChatGPT, it looks up what words (tokens) are most likely to follow each other for your query (text input), based on the texts it analyzed in the training phase. Since ChatGPT can also create text, which it has not seen in its training set or input, it is a generative model.
However, this approach comes with its downsides: ChatGPT has no knowledge of facts or events that are dated after the model was trained, and false or made up statements that are simply a more likely output will be given to the user like they are facts [MČK+23].

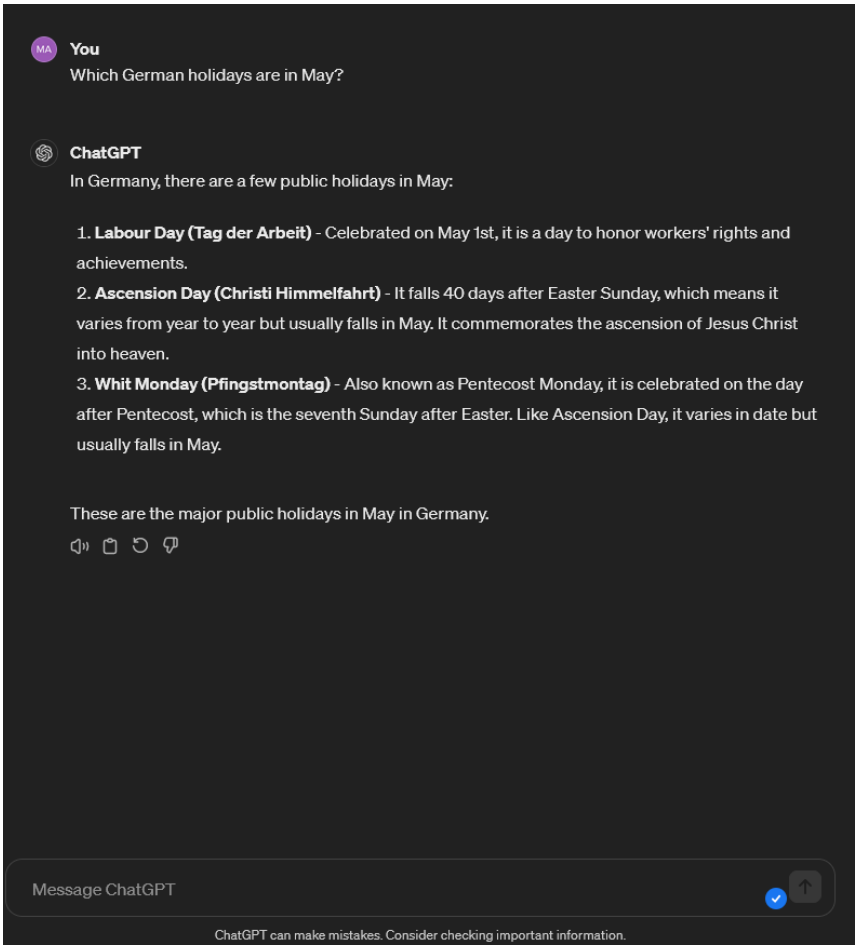


Figure 1.1: Simple conversation with ChatGPT

Writing scientific papers

Writing about your research is an important part of the research itself. It not only allows you to share your results, but is also needed to create accountability for and verifiability of your research, both being cornerstones of modern research[KG18]. In order to do so, standards on how to write academic papers have developed, each containing the same basic elements:

1. Providing context: What is your research topic and how does it fit into the field you work in.
2. Showing methodology: How have you conducted your research.
3. Evaluate findings: What is your result and how does it need to be interpreted.
4. Conclusion: What is the impact of your research and how can outside factors impact it.

1.2 Research Objectives

Given the requirements for academic papers shown in section 1.1 and the capabilities of ChatGPT for generating text, the idea to use the chatbot to help researchers write their papers arises. For analyzing how beneficial Large Language Models (LLMs) like ChatGPT can be, I looked at three Research Questions:

1. Which research tasks do researchers believe to be most suited to be supported by an LLM-based research assistant?
2. What prompting strategies and best practices can be used to support researchers in completing their research tasks?
3. Does an LLM-based research assistant help researchers in completing their research tasks more quickly and satisfactorily?

1.3 Structure of this Thesis

This thesis contains four major parts, the literature review, the first round of interview, the guideline itself and lastly the evaluation of the guideline. The literature review focuses on answering the research questions 1 and 2 via the works of other researchers. The first interview round instead focuses on answering the two research questions with user feedback. The guideline synthesis, and the guideline itself, describes how I used the knowledge gained from the literature review and first interview round and how it can be used. Lastly, the Evaluation interview, including the experiment, are meant to answer the research question 3 - if ChatGPT is a help. For each individual part, the workflow to conduct it is highlighted in the Study design chapter, while the result of the part is located in the Result chapter. In addition to these major parts, a modified guideline, based on the feedback from the second interview round was also added, but not evaluated.

2 Related Works

Hadi et al. surveyed several large language models, including ChatGPT and several plugins for it to develop general guidelines and identify tasks they can do [HQ+23]. Even though it is neither for the Software Engineering domain, nor for specifically for academic writing, the findings for ChatGPT are of value.
Arun HS Kumar analyzed the use of ChatGPT to generate papers in the Biomedical domain and found the texts to be of low academic quality. ChatGPT failed to follow instructions, especially for word counts, where it regularly fell short by 70%, had faulty citation, missed in-text references and lacked practical examples or personal experience [Kum23].
Salvagno, Taccone and Gerli summarized their finding about ChatGPT as 'extremely effective for the editing process; formatting and language editing, rewriting a particularly complex sentence in a clearer way, and even summarizing the entire text in order to compose a suitable abstract, are feasible using this approach, although the results are not always satisfactory, but they certainly save time.' [STG23].
Tafferner et al. looked at the use of ChatGPT for programming, it's knowable of hardware documentation and for literature review as part of a case study. For them, the documentation and programming by ChatGPT showed minor flaws, while the literature review contained many faulty or partially faulty paper suggestions [TBKG23].
Meyer at al. concluded that ChatGPT was beneficial to non-native speakers, for writing and editing tasks, however was concerned with the inaccuracies ChatGPT contained for generated text, which may not be present in 'new examples of LLMs developed specifically for providing accurate scientific information, such as perplexity.ai"[MUM+23].
Walters and Wilder analyzed the extent of citation problems by ChatGPT 3.5 and 4. They found that ChatGPT 4 faired significantly better, though still had major problems. Fabricated citations reduced from 55% to 18% and real citations with errors reduced from 43% to 24% [WW23].
Alshami et al. analysed ChatGPT as a tool for systematic reviews. Even though ChatGPT was making errors, they still found that it can save time [AEA+23].

3 Study Design

3.1 Literature Review

To gain an understanding of the current research landscape, I started by gathering potential related works by querying Google Scholar and OpenAlex. This was done with Publish or Perish [Har] on December 13th. For each of the two data sources, three requests were made, each using the top 50 results:

- Chat GPT writing
- Chat GPT scientific writing
- Chat GPT scientific writing Software Engineering

This resulted in a list of 300 papers, including duplicates, or 195 unique papers. For each paper, I then looked at the abstract to determine if they would help answer the research questions, resulting in a list of 74 papers with promising content, be it for the research question themselves or in providing background information that could be used for the interviews.
Reasons for papers being excluded:

- Focus on unrelated issues
  - Preventing/Detecting ChatGPT in publishing
  - ChatGPT for Education/Teaching
  - Ethics of ChatGPT in publishing
  - ChatGPT in Medicine
- Limited scientific value
  - 'I asked ChatGPT x' without evaluation
  - Opinion pieces with no references
  - Announcements

It is important to note that while some papers about the ethics of ChatGPT in publishing were excluded, not all of them were. Papers detailing how to prevent ethics violations (plagiarizing, faulty quoting, etc) were not excluded, while papers researching the effect on the scientific community or detectability of such violations were excluded. While there have been plenty of papers about the use of ChatGPT for writing, especially in the medical domain, where there appears to be a large research field for using AI to create reports, doctors notes, diagnoses etc. (hence the exclusion), none found were about the usage of ChatGPT as a writing assistant specifically in the Software Engineering domain.

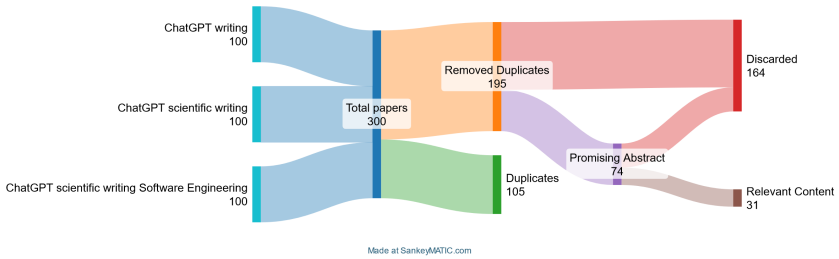


Figure 3.1: Literature exclusion overview

In the next phase, I looked at the entire papers to determine the usefulness. This resulted in the exclusion of further papers due to them not focusing on the topics highlighted in the abstract. 31 Papers remained, 8 of which provided background information.

3.2 Interviews

To gain a better understanding how ChatGPT can be used to write papers or aid research, interviews with researchers in the Software Engineering domain were conducted. This interview round focused on answering RQ 1, where can ChatGPT help, and RQ 2, how can ChatGPT help. To find suitable candidates the web presences of Institutes for Computer Science of the Technische Universität Berlin (TU Berlin), Rheinisch-Westfälische Technische Hochschule Aachen (RWTH Aachen), Karlsruher Institut für Technologie (KIT) and Universität Stuttgart were manually scraped for mail addresses of researches. Here, as far as possible, only researchers doing their doctorate or post-doctorate were included, however the inclusion criteria were highly dependent on the institute's site and what kind of information was available about the researchers. If only the currently held degree was listed, the minimum requirement was a Master's degree, if only the position was listed, all researchers were included, with other staff like tutors or administration being excluded. In case when no information was stated about their role in the institute, all staff members were included. In addition, it is important to note that only those researchers could be contacted who opted to have their mail address publicly listed on the web presence.
Due to this process, requests for study participation were sent to the following number of researchers:

- TU Berlin: 202
- RWTH Aachen: 293
- KIT: 496
- Universität Stuttgart: 167

For each University, except for Universität Stuttgart, one person replied. From the Universität Stuttgart none replied, however two candidates could be found through the intervention of my supervisor. With this pool of five candidates, I conducted the first interview round.

The interviews were conducted fully remote via WebEx on the same day in late February 2024, with each interview lasting between 15 minutes and 25 minutes. The interviews were either recorded or notes were made during the interview.

3.2.1 Participant demographics

The demographics of the interviewees were collected as part of the interview and, in case they responded to the participation request email, their reply email. The data between their field of study and experience has not been linked to the University to minimize the re-identification risk, since all three factors combined could result in such a small pool of potential interviewees, that an identification could be possible. In the table 3.1 the demographics and experience with ChatGPT are listed. The pseudo-names 'A', 'B', 'C', 'D' and 'E' will refer to the same person when used from now onward.

Participant	A	B	C	D	E
Field of Research	Automotive Software Testing	Empirical Software Engineering and Automotive Software with focus on safety	Domain specific languages, model driven development, web-applications, graphical user interface	Computer Security	Machine Learning, Active Learning
Years of experience	5	>1	almost 5	8	3.5
Use of ChatGPT in gerneral	several times per week	almost daily	multiple times a day	weekly	several times a day with month long breaks of no use
ChatGPT use during research	often	most of the time	very often	sometimes	sometimes
ChatGPT use for writing	most of the time	most common case	barely	mostly	most use
Use of other LLM	No/GPT 4	No	No but tried some	No	Yes, e.g DeepL Write, Grammarly

Table 3.1: Participant demographics and experience

3.2.2 Questionnaire

The questionnaire was composed of four question groups.

1. Differences between Software Engineering and other Domains
2. Concerns and Strategies
3. ChatGPT for Research
4. ChatGPT for Writing

The first question group was to make the answers of the participants comparable with the literature. In the literature, the use of ChatGPT was mostly without the focus of a specific domain, except for the medical domain. Concerns and Strategies covered the general experiences and workflow of using ChatGPT. The two other question groups are for directly answering RQ 1 and RQ 2.
In the question catalog 9 all potential questions are listed. Depending on previous answers, some were skipped or rephrased. Enumerated question were questions directly asked, with the bullet points being follow-up questions in case the interviewee was vague.

1. Is Research in the Software Engineering Domain different from other Domains (e.g. Medical, Engineering, Social Science. . .)?
    - Are certain aspects of research different weighted?
    - Are certain parts not needed?
    - Does research in the SE Domain need additional research aspects?
2. Does writing scientific papers for the SE Domain differ from other domains?
    - Are certain aspects of research different weighted?
    - Are certain parts not needed?
    - Does research in the SE Domain need additional research aspects?
3. Do you have concerns about using ChatGPT?
    - Which ones?
    - How do you attempt to mitigate them?
4. What prompting strategies have you used?
    - Can you recommend specific prompts /prompting patterns?
    - What strategies did not work?
    - Are prompting strategies dependent on the task at hand? E.g., does a strategy work for writing but not for a different task.
5. What task during your research would most likely benefit from assistance by ChatGPT?
    - Have you used ChatGPT for a literature review?
        - To find new sources?
        - To analyze known material (e.g. summarize paper)
    - Have you used ChatGPT to design or evaluate a study?
    - Have you used ChatGPT for programming /Algorithm design?
    - Have you used ChatGPT for Data Analysis tasks?
6. What was your experience using ChatGPT for research tasks?
    - Did it give desirable results?
    - Did you need to overcome certain hurdles to get good results? Which ones? How?

7. Which writing steps can be enhanced or sped up by ChatGPT?
    - Research of a topic
        - Find new sources
        - Analyze known material
        - Organization of the findings
    - Writing of a draft
    - Editing of a draft
    - Formatting
8. What sections of a paper are most likely to benefit from the use of ChatGPT?
    - Abstract
    - Introduction
    - Methodology
    - Main research
    - Conclusion
9. What was your experience using ChatGPT for writing tasks?
    - Did it give desirable results?
    - Did you need to overcome certain hurdles to get good results? Which ones? How?

3.3 Guide Synthesis

When analyzing the literature, I sorted each paper into either answering a research question or providing background information. For those who answering a research question, I created two lists, one for task benefiting from ChatGPT and one for techniques or prompt for using ChatGPT. Due to the semi-structured interview, the responses of the interviewees were already coded, with minimal need to assign an answer to a specific research question. For each question block, the answers from the interviewees were compared with the findings of the literature to find overlap. This overlap was then turned into the guideline, except for the prompt examples. Since the literature contained general advice or strategies with giving specific examples, these were taken from the interview only.

3.4 Evaluation Interviews

To evaluate the guideline, the participants of the first interview round were asked to partake in the second round. For this, all interviewees except for interviewee D could make it. The evaluation interview were between the 11th and 17th of Mai.

3.4.1 Evaluation design

I began with asking the interviewees about their initial thoughts on the guide. This was done before they used it in the experiment to record any first reader impression. Once they shared their initial thoughts, the evaluation experiment was conducted. The experiment was concluded with a set of questions, see 6, to assess user feedback.
In the evaluation experiment, the participants were asked to edit two texts, both seen in the appendix in chapter A. The first text was edited manually without any AI tool but with the help of any autocorrect feature built in an editing program of their choice, like Microsoft Word. The second text was edited with ChatGPT 3.5 and the guide. For both texts, the participants were told to assume that all statements in the text were correct, and all citations were accurate. The texts contained grammatical errors, 'false friends' expressions, run-on sentences and suboptimal phrasing. These defects were used to assess the quality of the edited texts. For each editing task, a time limit of 15 minutes was set. In the cases when the participants finished early, the time was tracked. When the participants did not finish, the progress was tracked. To eliminate any bias originating from using two texts, two participants were asked to edit text 1 manually and text 2 with ChatGPT, while the other two had to do it the other way around.

1. What are your initial thoughts on the guide?
    - Do you think any aspects are of limited use?
    - Do you think certain aspects need to be fleshed out more?
2. Do you think the guide helped you in using ChatGPT to edit the introduction.
    - Why/Why not
3. Do you think the guide helps user who have no or limited experience with ChatGPT to better utilize the tool?
    - Why / Why not?
4. Do you think the guide can help users write papers quicker?
    - Why / Why not?
5. Do you think the guide helps users improve the quality of their writing?
    - Why / Why not?
6. - Do you think the guide properly addresses the limitations of ChatGPT?
    - E.g. Warning of Hallucinations etc.
    - Why / Why not?

4 Result

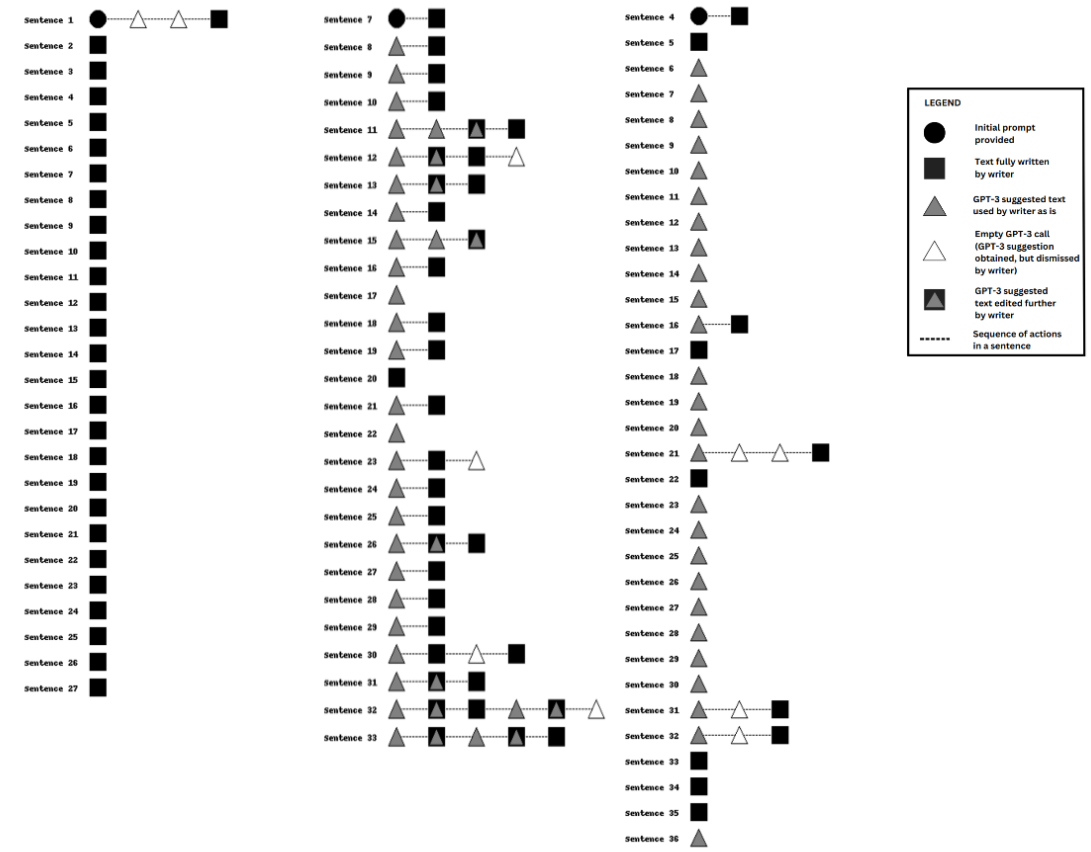
4.1 Literature Review

In the final list of papers, many included very similar information, though some had unique focuses or presented the information especially well. Research of ChatGPT looking at the Software Engineering Domain largely focuses on varying programming tasks, like code generation or debugging, but not on how ChatGPT can be used as a writing assistant in Software Engineering research. While programming is a skill needed for research in the Software Engineering Domain; therefore ChatGPT could help speed up research, researching this aspect is not part of this thesis, hence has been excluded.
Generally, when papers suggest the use of ChatGPT for writing, this is done without an evaluation or even testing the claim. Most often, the usability is simply suggested or tested with a very limited number of queries, often just one per use case, by the authors themselves. From the 31 papers selected, only 11 show use of ChatGPT and only 7 do some form of evaluation.
Some papers looked at focused on other tasks during the research process, like experiment design [TBKG23], data analysis [HS23] or science communication [Sch23]. While many papers already focus on the threads and concerns of using ChatGPT for research, few highlights how to use it ethically. The ethical concerns of using ChatGPT in scientific research, which are named in most publications, can be grouped into several main categories:

- Bias
  - ChatGPT's dataset predates 2021 [Opeb]
  - ChatGPT can be politically left leaning [Bud]
- Plagiarism
  - ChatGPT has no original ideas, but repeats content from its dataset, often paraphrased or mixed from different sources without giving credit[GE23]
  - ChatGPT can create entire, valid looking but fraudulent articles [MČK+23]
  - ChatGPT can give credit to existing sources for things never said by them [MČK+23]
- Hallucinations
  - ChatGPT can make facts up [MČK+23]
  - ChatGPT can make sources up [MČK+23]

These concerns must be considered in this work too, to ensure that the result can be used by future authors.
In addition to the ethical concerns, Grimaldi and Ehrler highlight that the use of ChatGPT and similar tools can result in a publishing flood, which would lead to an increasing number of publications with limited scientific value. Even today, paper mills designed to publish in high frequency, are a problem in the scientific community, however with ChatGPT or similar tools they ”could become supercharged” [GE23].
To combat the ethical concerns and fraudulent use of AI tools like ChatGPT, multiple papers [GE23; Joo23] mention the need for consistent guidelines for AI assisted writing and publishing, like what can the tools be used for or how do they need to be accredited.
Giglio and Costa suggest that authors utilizing AI assistance in the writing process must disclose the sections and way they were used in an acknowledgement section of a paper. In addition, the authors state that they checked the output of the AI and take responsibility for the entire paper [GC23]. Publisher like Elsevier have already implemented similar requirements [Els], such as not accepting papers with ChatGPT as co-author.
To better determine what uses of ChatGPT would not be fraudulent, Ciaccio looked at how non-AI help can be utilized during the writing process and how this help can be translated into AI usage. He suggests that spellchecking or grammar tasks should not require any disclosure of AI usage, since non-AI tools doing the same do not need to be disclosed either. For editing purposes, Ciaccio differentiates between different levels of editing. Copy editing, the lowest form, is described as seeking help for ”grammar correction, ensuring consistency in style and formatting, and checking for factual errors” [Cia23]. The need to disclose copy editing by a human is dependent on the publisher. Stronger forms of editing, such as substantive editing or ”content editing involves reorganizing entire paragraphs or sections, rewriting passages, and adding or deleting content” and developmental editing ”involving further rewriting, restructuring, or reordering of the material to improve clarity, coherence, and effectiveness” always need full disclosure. [Cia23]
To make the use of AI during the writing process more visible, Shibani et al. developed the CoAuthorViz framework that visualizes how sentences in the final result came to be. In figure 4.1 they show three example workflows with varying degrees of AI usage. The first is not using AI in the end result at all, the second uses AI but (heavily) modifies the result, and the last mostly uses the AI output [SRM+23]. Given that the last and potentially the middle case would be viewed as ghostwriting when done by a human, the legality of which is highly dependent on the kind of scientific work [Sut], the use of AI ghostwriting should be considered similarly.
In their evaluation of ChatGPT, Katar et al. broke down the writing process into the following steps [KÖ-+23].

- Choose a topic
- Research the topic
- Organize the findings
- Write the draft
- Edit the draft
- Format and add citations



(a) Case-1: Fully autonomous writer

(b) Case-2: Autonomous writer with GPT-3 assistance

(c) Case-3: GPT-3 dependent writer

Figure 4.1: CoAuthorViz Examples of three different AI co-authoring workflows by Shibani et al.

These steps are not necessarily separate from one another, for example adding citations during the draft process can help in not forgetting to accredit a source. Additionally, some steps, like the choice of the topic or the format of the work can be external inputs, e.g. for Bachelor thesis.
When it comes to researching the topic, the direct use of ChatGPT is problematic due to the aforementioned biases in the knowledge base and the tendency to make up sources. However, several authors point out that it can be used to summarize the findings of traditional literature reviews when the papers are entered as a prompt [AWL+23; GE23; HQ+23; NRS23]. While this would not eliminate the need to read papers at all, it can help in reducing the number of read papers that over-promise in the title or abstract.
When it comes to organizing the findings, [AWL+23; GE23; HQ+23; HS23] mention the potential of AI systems to easier organize data and similar 'tedious task', most simply suggest the usability for such tasks.
When it comes to writing the paper itself, either investigate the usefulness of using ChatGPT to create the entire article, which includes the aforementioned flaws of made up facts and sources [MČK+23] or suggest editorial uses such as grammar correction or phrasing improvements [STG23].

Information on how to achieve these tasks with ChatGPT is often not included, though Hadi et al.[HQ+23] offer general strategies about improving the utilization of ChatGPT, that can be integrated alongside the general information of other authors:

- Identify the task: What do you want to achieve, and how should the result look like? [HQ+23]
- Define a role: Give ChatGPT the proper context of how it should behave, e.g. provide a job title like 'programming tutor'[HQ+23]
- Ask ChatGPT on how to prompt to achieve the task[HQ+23]
- Provide explicit instruction, e.g. when asking for data formatting provide example data[HQ+23]
- Negative Prompting: Tell ChatGPT what not to do or what to exclude from the answer[HQ+23]
- Interact with the model: Answers can improve when asked to modify certain sections[HQ+23]
- Keep it short: ChatGPT has limits on content generation and evaluation size. Split section up if needed. [Kum]
- Start simple: Begin with basic instructions and then add to it [WTC22]

4.2 Interview Evaluation

Each interviewee answered very concisely, with helpful information, causing minimal need for follow-up questions. This cut the interviews 5–10 minutes shorter than the planned 30 minutes. While most of the use cases were quite similar, it was interesting to see that the Interviewees used different strategies and had a somewhat different focuses on concerns caused by utilizing ChatGPT. A full breakdown of the interviewees replies can be seen in table 4.1.

4.2.1 Differences between Software Engineering and other Domains

Some Interviewees noted that the basic process in research are similar, though each one highlighted that there are differences to other domains. The most prominent explanation was that Software Engineering deals with problems that require a direct solution. Interestingly enough, this was called both more theoretical and less theoretical than other fields by different interviewees, despite expressing the same thing. Such a difference can be from different points of reference, for example if you compare algorithm design to social sciences, where Software Engineering is more theoretical, however if you compare the implementation of your solution to mathematics, it seems less theoretical. Since Software Engineering generally aims at offering solutions to problems, the evaluation process often has a reduced focus according to multiple interviewees, with the example given that statistical significance is less frequently shown.

Another aspect named by two interviewees was the reduced need for safety, for example due to a lack of clinical trials. This was mentioned in comparison with the medical domain, where experiments can directly impact the health of any participant, something not typically the case in the Software Engineering domain.

When asked about how writing in Software Engineering Domain differs from other domains, four participants said that the structure is the same with different focuses or styles. Examples given were similar to the differences in research, namely less focus on evaluation or Software Engineering being less observatory and more about solving specific problems. One participant noted that the methodology can vastly differ between different domains, and even within the same domain can differ greatly depending on the field of study. As an example, he gave requirements engineering, where the focus vastly changes depending on the subject being analyzed.

4.2.2 Concerns when using ChatGPT

All interviewees remarked that ChatGPTs output cannot be trusted. ChatGPTs tendency to make data up or to falsify real information was the most named reason, however biases in the training data were also mentioned. This causes the need to always have enough domain knowledge for the request at hand to be able to tell if the output information is plausible and trustworthy. Interviewee B remarked that due to this behavior, an over reliance on the results is a great concern for the validity of the research.

The second most named concerns were data privacy based. This includes both the personal data, for example by study participants, or any other privileged information, for example if a non-disclosure agreement is in place. While the former was named by three participants, the latter was only named by one. This might be due to the kind of research the Interviewees partake in. When Interviewee E was asked if breaching non-disclosure agreements was a concern, his initial response was no, due to his research being public anyway, however he could see how this could be an issue.

An additional concern was the loss of data stated by Interviewee E. He described that ChatGTP can lose information that was provided in the prompt, but is no longer present in the reply.

4.2.3 Strategies for using ChatGPT

The most common strategy used by the interviewees was providing ChatGPT additional information about the specific task. Here three of the five Interviewees talked about assigning ChatGPT a role, to improve the output, while a fourth one said that such a role assignment would only change the tone but not the quality of the output. However, since the interviewee in question said that he creates his request with very specific instructions, it is possible that the definition of a role could still provide beneficial for less specific requests.

Three of the interviewees also noted that using shorter texts, bullet points or specifying length constraints could also help with the output. Such constraints could help with ChatGPTs tendency 'to talk a lot' despite offering limited information.

Other commands used by the participants to improve the output included requests to exclude meta-texts like 'as a language model....' or the requests to not lose any information that was provided in the prompt.

When asked about using an iterative approach, like starting with a simple prompt and then requesting additions or changes to the output, one participant noted that such an approach can work, however in his experience, the performance degrades after multiple iterations. If such degradation occurs, using the chat history as background information for a new chat did help get back lost performance.

Another interviewee also mentioned that the risk of hallucinations can be reduces by not asking ChatGPT about sections of an earlier response that were vague, since such vagueness can indicate that ChatGPT does not have the knowledge to answer properly.

4.2.4 Research Tasks benefiting from ChatGPT

When asked what tasks during their research would benefit from the use of ChatGPT, all but one mentioned programming, but they noted that the solutions ChatGPT offered often needed further adjustments and debugging, with one participant even stating a 30-40% estimate on how much code needed modifying. Interviewee E also mentioned that ChatGPT can be used for debugging purposes, especially when public documentation for an interface exists. It is likely that the other interviewees who used ChatGPT for programming also used it for debugging, since these tasks are highly connected if not indistinguishable.

Outside of programming, the interviewees had limited experience. Interviewee B used ChatGPT to get ideas for a study design for a semi-structured interview once, while others mentioned minor tasks like helping with phrasing for mails or asking a simple knowledge question. The latter one can be a double-edged sword since any information obtained from ChatGPT must be manually verified, so either you need the domain knowledge yourself or it must be information that is easy to find through other means, like googling it, as the interviewees have pointed out. Interviewee C and E talked a bit more about their experiences with ChatGPT and how they use it for general brainstorming.

4.2.5 Writing Tasks benefiting from ChatGPT

All the interviewees used ChatGPT for editing or rephrasing purposes, though interviewee C does so rarely. For such purposes, the output has been described as 'good enough' even though some manual edits are still needed. However, even for editing purposes, the several interviewees noted that the output has to be checked for errors introduced by ChatGPT.

Two interviewees also mentioned successfully using ChatGPT for generating paper title ideas, where potential titles are generated with input of an abstract or bullet points.

None of the participants used ChatGPT to write parts of papers.

4.2.6 Paper parts benefiting from the use of ChatGPT

When asked what parts of a paper would benefit the most from ChatGPT, all interviewees named parts not directly involved with the main research itself. The title and abstract of a paper, both representing (very) short summarizations of the work, have been named by four of the five interviewees, with limitations like privacy concerns and the need to make manual adjustments.
The introduction of a paper was also named four times. Interviewee C gave additional details on why he considers that part of a paper being suitable, by stating that the introduction does not focus on the research itself, but rather related works.
Two participants have named the Conclusion as well. While it does contain strong summarization aspects, which, it is also requires you to put your work into the greater context, an analysis task. This

aspect might be the cause why fewer interviewees said that the conclusion benefits from ChatGPT. Regardless of what part was named, all interviewees once more noted that any output needed manual adjustments and verification. In addition, Interviewee E also stated that when we only look at the editing use case, all parts of a paper benefit from ChatGPT.

Table 4.1: Interviews answers round 1

Topic	A	B	C	D	E
Differences between research	Quite similar, process has different weights to aspects e.g. medical has higher safety	It is different, mostly dealing with software, few experiments, evaluation generally less important	It is different, but cannot say how, [lack of experience in other domains]	Different, e.g. no clinical trials, quality standard higher in other fields, e.g. statistical significance is shown less often	Close to mathematics, less so to social science, but basics are the same
Differences between writing	Methodology changes between different fields, even between different use cases, e.g. Requirements Engineering varies highly on the field for example non-functional requirements and User feedback	Writing feels similar in structure, but no experience with other domains	Believe so, evaluation is different, more practical compared to math	Structure is similar, but more create/flexible. E.g. no strict AIMRaD [Abstract, Introduction, Methods, Results, and Discussion]	Papers in social science are more observatory/descriptive, in SE its more about solving problems

Continued on next page

Table 4.1 – continued from previous page

Topic	A	B	C	D	E
General Concerns	Cannot rely on output directly, 'Misdirections' happen when pushed, Data privacy concerns	Hallucinations, only use when you have domain knowladge. Overreliance on results	Breaking non-disclosure agreements, unreliable output	Validity/Quality of responses, Data bias in training	Cannot rely on Domain Knowledge, loss of information from input, hallucinations, privacy concerns for personal data
Strategies	Directly asking shows poor output, Define roles e.g. 'You are a researcher', declare format of output, define limitations like language, length. Use commands to filter out phrases like 'as a language model I can...', Ask ChatGPT how to use ChatGPT	First command to define a role and give instructions, then input the actual request content in second command	Hallucinations can be avoided if areas where ChatGPT answers vaguely are not pushed, formulate request very specifically to what you want, Assigning a role does not really help, but changes the style	Ask for certain tone, give keywords instead of entire text	Use short paragraphs, to avoid loss of information, write 'Don't loose information'. Use simple Tasks. providing additional information (background knowledge, research area,...). Modified iterative process

Continued on next page

Table 4.1 – continued from previous page

Topic	A	B	C	D	E
Research Tasks benefiting from ChatGPT	Used for programming, 30-40% needed modification	Used to improve study design and semi-structured interview once, used for programming. Not perfect but gives some ideas to build upon	Often used for programming, Asking for literature does not help. Asking for Topics always yields similar results	Not enough experience	Programming, since it knows all the stack overflow answers and online API documentation
Writing Tasks benefiting from ChatGPT	Used for evaluating manually written text, with reasoning for judgment	Proof reading and help with phrasing	A lot of useless context. Software Engineers have certain writing style with 'precise, on point and simple to read' sentences. ChatCPT pads the text, but can help with editing (used rarely)	Give some keywords to generate title ideas, editing. For editing output is mostly satisfying	Mostly Editing or rewriting, but also asking for a source if a statement lacks a source, however any ChatGPT source must be manually verified, Title generation

Continued on next page

Table 4.1 – continued from previous page

Topic	A	B	C	D	E
Paper parts potentially benefiting from ChatGPT	Could help a lot with the abstract if no privacy concerns would exist	If other parts are already in place, abstract, introduction and conclusion could benefit from use of ChatGPT, however still needs manual refinement	Introduction, because less focus on the paper itself, but related work, since it does not require specific context. Other parts require you to formulate ’your specific idea’	Intro, Background and Conclusion could benefit, Title generation	Introduction and Abstract for example generating an Abstract from the introduction, or any other task shortening texts, title generation, but needs manual work
Other remarks			Used like google for simple problem like algorithms, creating simple web pages, finding inspiration, simple questions.		Brainstorming, like previously mentioned title generation

4.3 Guideline Synthesis

When comparing the concerns present in the literature with the ones mentioned by the interviewees, it is clear that they are very similar, though with a different focus. In the literature, a large focus on the ethics of using ChatGPT exists, something only directly mentioned by one interviewee. However, given that the interviewees use ChatGPT mostly for tasks that can be done by non-AI tools, and not text generation, it is likely that ethical concerns were a subconscious concern when choosing what tasks to do with ChatGPT. Similarly, the bias concerns that were often mentioned in the literature, were less present in the interviews, but so was the use case of using ChatGPT as a knowledge base, which could explain the reduced sensitivity. The problems of ChatGPT hallucinating or falsifying or forgetting data were equally present in the literature and the interviews.

When looking at the application potential during the research process, programming was the most named task, an area that is also covered by the literature but not further analyzed in this work. Except for using ChatGPT to draft mails, and the one time it was used during a study design, the interviewees did not utilize ChatGPT in the manners highlighted by the literature, like for translation, ghostwriting or data analysis tasks.

During the writing process, both the interviewees and the literature highlight the usefulness of ChatGPT for editing, spell checking and rephrasing tasks. Generating title ideas was also a use case that was either already utilized or a potential future application named by the interviewees, which is present in the literature. Other parts of the paper, such as the abstract, introduction, or the conclusion were named too, however only as potential future application rather than being generated right now. In the literature, generating the abstract has been analyzed as well, with generating other parts of the paper being suggested.

Comparing the strategies yielded little differences. Both literature and the interviewees provide additional information for the task at hand. This can include defining a role, input or output constraints such as text length, or giving other background information. One difference between the literature and the interviewees was when dealing with iterative approaches. In the literature, they are suggested without limitations, however one interviewee noted that when using many iterations, starting a second chat and providing the iterations from the previous one as background was needed to keep the quality of the ChatGPT responses high.

4.4 Guideline

ChatGPT can be a helpful tool during your research, but must be handled with caution to minimize the risk associated with generative AI. This guide aims at providing insight into how ChatGPT can be utilized and which limitations must be considered.

General Guidelines

ChatGPT is a tool that can misbehave and make errors. You must always be knowledgeable in the area that you ask ChatGPT about in order to spot mistakes.
When making your prompts, try to give instructions without spelling or grammar mistakes, since

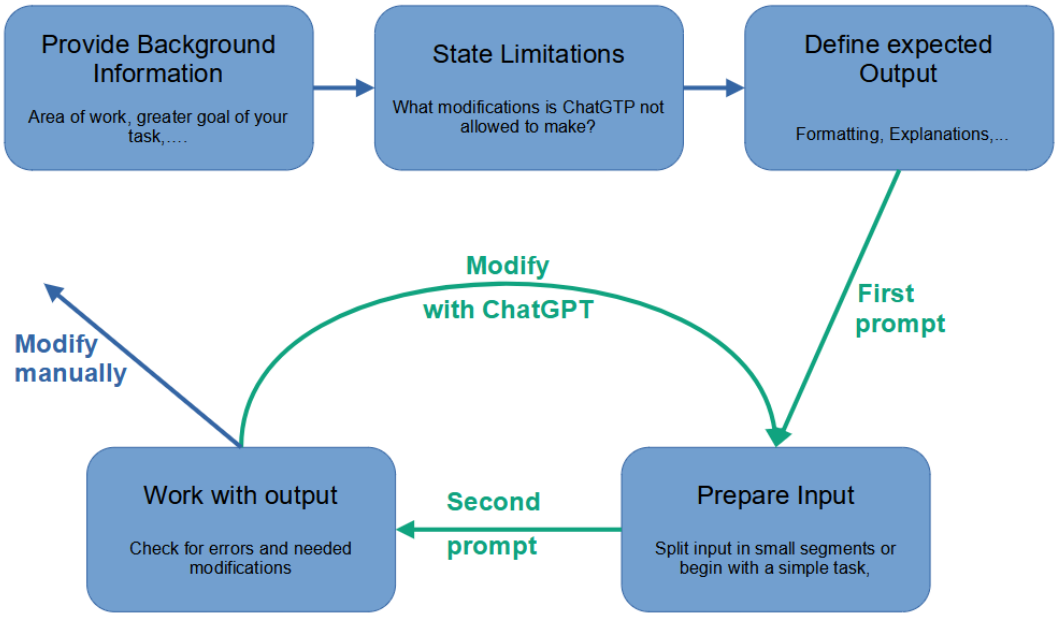


Figure 4.2: ChatGPT Prompting Workflow

either can lead to faulty interpretations of the task.
ChatGPT is good at providing '80%' solutions, but output is rarely perfect. Consider making the finishing touches manually.

Prompting Strategies

Provide Background Information

Before asking ChatGPT to do anything, you need to be aware of the task you want ChatGPT to perform. Here it is important to be as detailed as possible, with potential relevant information being the area of work you are doing, language selections (especially if code is involved) or any other outside input you already received. Telling ChatGPT to behave like a certain Role can help, but providing the same information in another form can yield similar results. Meta-information, like when and how you are providing the input, is also helpful. Potential prompts can include the following instructions:

- I need help editing a part of my paper about [Topic].
- I am writing a paper in [Field of Research] about [Topic] and need help with....
  - *Alternatively:* Pretend you are a researcher in [Field of Research] writing about [Topic]
- I need to make changes to my paper to improve [Property, like readability, understandability,...]
- I will provide one section of my text after another, starting with the next prompt.

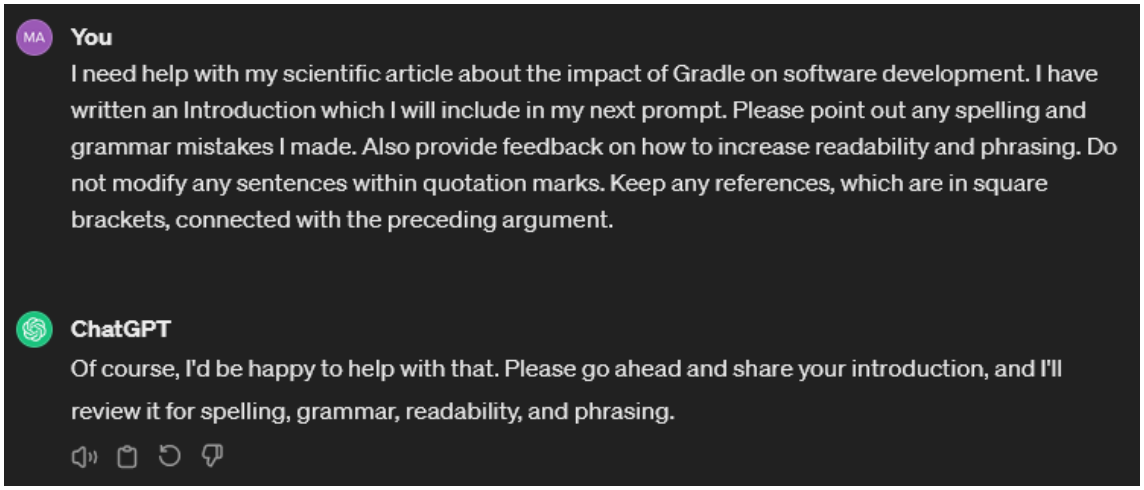


Figure 4.3: Initial prompt example containing background information, limitations and expected output

State limitations

Tell ChatGPT what it can and cannot do, to avoid undesired behavior. This is helpful to reduce the need for manual edits of the output, like removing modified quotes, however they are not a guarantee that ChatGPT will work as expected. Even with stated limitations, you must expect ChatGPT to break these constraints from time to time.

- Do not lose any information.
- Do not modify any statements surrounded by quotation marks.
- Do not add any new information.

Provide information about the expected output

This can include any desired output constraints, like formatting or explanations for the output. While you can also ask ChatGPT to highlight changes it made, the result is not to be trusted and will miss changed statements while also highlighting unchanged ones.

- Provide feedback why your changes improve the text.
- Keep the reply short.
  - *Alternatively:* Keep the reply within [Number] words or less.
- Provide multiple alternate solutions and explain the difference.

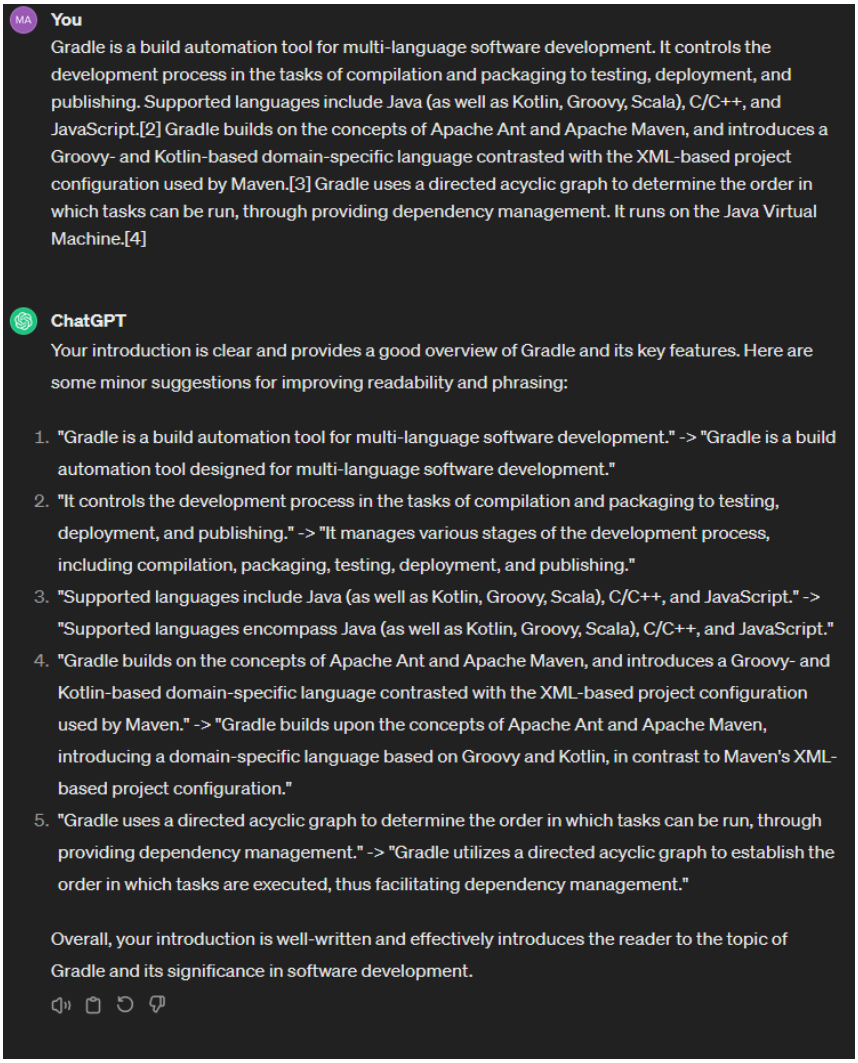


Figure 4.4: Editing suggestions, when only a paragraph is entered

Preparing the input

Due to ChatGPT being token-based with a maximum number of allowed input and output tokens, lengthy inputs will not be processed as expected. To mitigate this, splitting your input into multiple smaller segments can help. Depending on the task at hand, this can also influence the prompting strategy. For editing, spellchecking or even translation purposes, simply going one paragraph or even one sentence at a time is advised, and longer sections can lead to loss of information. For other tasks, like programming, starting with a simple request and then requesting additions or changes in additional prompts can help achieve the desired output. As can be seen in fig.4.4 when entering a single paragraph from the Gradle Wikipedia entry [Gradle24], ChatGPT can make suggestions for the entire paragraph. In contrast, entering the entire article, as seen in fig. 4.5, only suggested edits for the beginning few paragraphs, fig. 4.6, while ignoring the rest of the text.

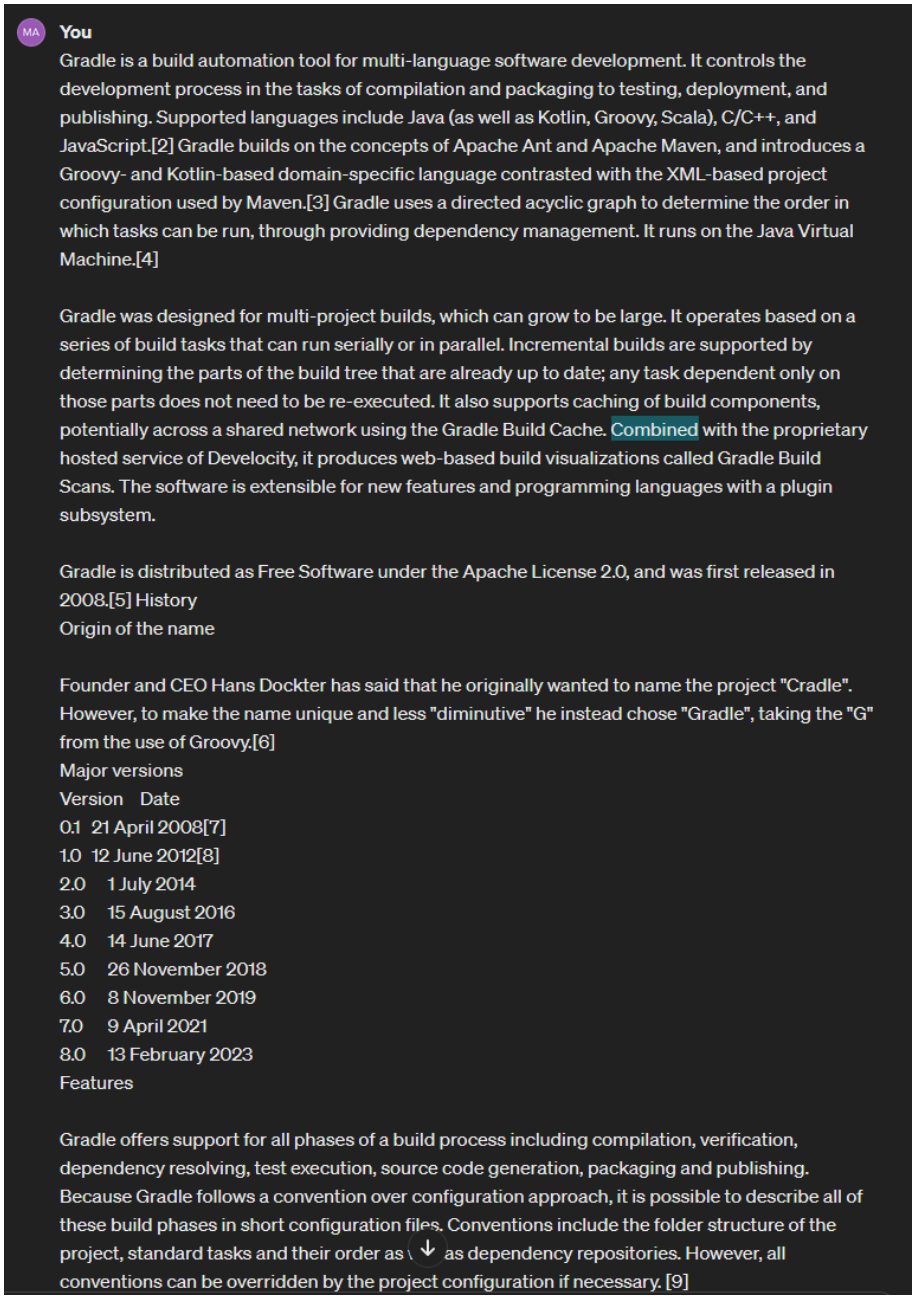


Figure 4.5: Input of entire texts at once, last edit suggestion highlighted

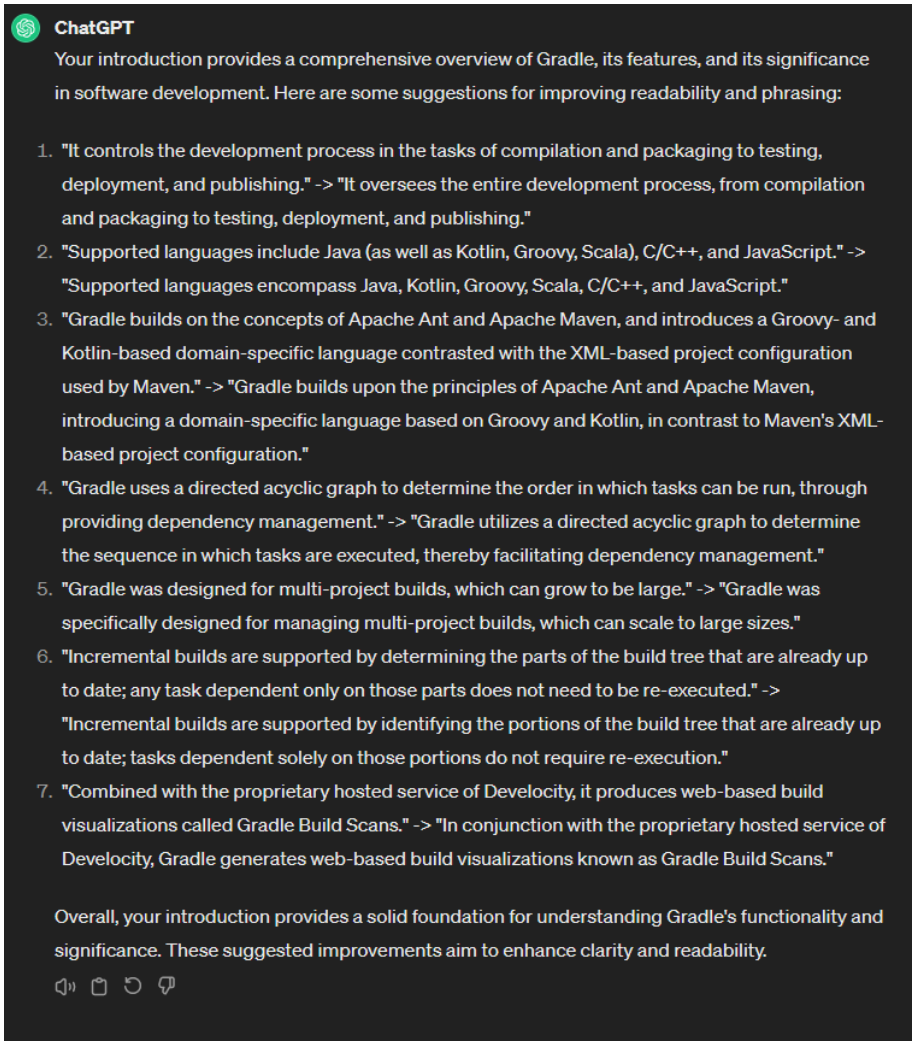


Figure 4.6: Editing suggestions for entire text, showing incomplete text analysis

Work with the output

After receiving the output from ChatGPT, make sure that all constraints are met and the result addresses your task. For editing tasks, manually fixing errors, like a modified quote, may be more efficient than prompting ChatGPT again. For more complex tasks, you can tell ChatGPT about the error and ask it to do the task again. While this can help, it can also lead to errors in different parts or no improvement at all. When working with text-based tasks where have multiple replies for the same section, combining suggestions by ChatGPT can also improve the end result.
If efforts to correct errors or improve the output are unsuccessful, don't hesitate to start over in a new chat. Here you can input the previous chat history as background information or begin from scratch. When modifying a response from ChatGPT, you can use these patterns:

- Please add this argument...
- Suggest alternatives for [Part you did not like]

- Modify the code to do [some constraint like return a status code instead of void]

4.5 Evaluation of the guideline

In order to evaluate the guideline, an experiment and an interview were conducted. The experiment was conducted as shown in the experiment design in 3.4.1.

4.5.1 Experiment Evaluation

The texts edited by the users were checked with the LanguageTool Pro version for spelling, grammatical and phrasing issues. The found issues were grouped into 3 categories:

1. Actual Errors: Anything not complying with spelling, grammar or inaccurate modifications; an objective measurement.
    - e.g. 'faster then' instead of 'faster than'
    - missing citations or modified quotes
2. Style Errors: Phrasings that made the text less understandable; a semi-objective measurement.
    - e.g. 'the git' instead of 'git'
    - run-on sentences
3. Style Choices: Phrasings that could be better according to LanguageTool but do not hurt as is, a subjective measurement.
    - e.g. 'in order to do sth.' instead of 'to do sth.'

All Participant except Participant C used editors with built-in spell checking, however Participant C was the only one who's manual edit did not contain any errors according to LanguageTool.

Comparison of Speed

In both editing tasks, only one out of four users finished the entire text. For the participants, who did not finish during either task, all but participant B managed to edit more text with ChatGPT. Those improvements were only minor, with one additional paragraph being edited. Participant B managed to finish when editing manually, but not with ChatGPT. When he was asked what slowed him down, he answered that when using ChatGPT he tried different phrasing, while his manual editing only aimed at making minor changes to improve the text. When looking at the text of the other participants, the same is true for the rest of them, where the ChatGPT edited texts contain more significant changes, except for participant E. Participant A, had the most significant speed up, from not finishing in 15 minutes to requiring only 6 minutes.

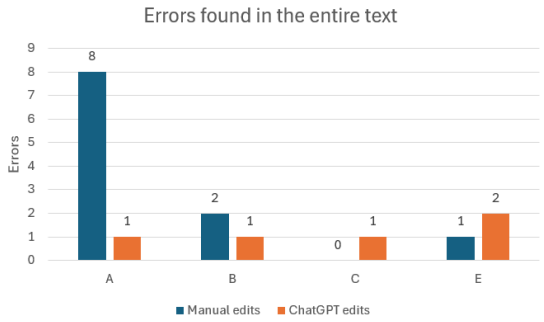


Figure 4.7: Number of errors by participant and task for entire text

Comparison in Quality

When comparing the number of errors in the edited text, on average, it decreased from 2,75 to 1,5 if the entire text is looked at, as can be seen in figure 4.7. When looking only at the parts that were edited, the number of errors decreases from 2,5 to 1 on average, figure 4.8. While this could show an improvement in quality, it is important to note that the improvements are from participant A alone, who made several new errors in his manual edit and commented that he noticed how reliant he is on AI assistants like Grammarly during editing. If we look only at the other three participants, the errors increase from 1 to 1,33 per text for the entire text, or increase from 0,67 to 1 when looking only at the edited text.
If we exclude style choices from the error pool, the result changes as follows. For the entire text and all participants, the errors decreased from 2,25 to 0,74 per text, see figure 4.9. For all participants and only the edited text, the errors decreased from 2 to 0,5 as seen in figure 4.10. Excluding participant A again saw in increase from 0,33 to 0,66 per text for the entire text or 0 to 0,33 per text for the edited parts.
Despite showing an increase, the numbers are too close to make any reasonable observation. A single error more during the manual editing or one less with ChatGPT could have changed the outcome for the analysis of participants B, C and E. In addition, if we look at the text edited by participant C, this error would not have happened without the time constraint put in place by the experiment. The participant lost a citation and was aware of the issue, but did not have the opportunity to put it back in the text during the allowed time.
In addition, it is worth pointing out that in while editing with ChatGPT all participants except participant E changed their editing from simply removing errors and oddities with minor adjustments to a larger scope of rewrites. Participant E chose larger rewrites for both manual and ChatGPT editing. While making little difference when checking with writing tools like Language Tools, such larger rewrites can increase the quality subjectively. After the experiment, the participants were asked additional questions. A detailed breakdown can be seen in table 4.3.

Manual edits				
Contestant	A	B	C	E
Editor	Microsoft Word	Microsoft Word	Notepad	Google Docs
Text	Git-1	Git-2	Git-1	Git-2
Time	not finished	14 Minutes	not finished	not finished
Progress	last paragraph missing	entire text edited	last paragraph missing	first two paragraphs edited, some fixes in later sections
Errors	1 Actual Error missed + 2 new one, 1 Style Error missed + 4 new ones	2 Style choices	none	1 Style Error not found in unedited text
ChatGPT edits				
Contestant	A	B	C	E
Text	Git-2	Git-1	Git-2	Git-1
Time	6 Minutes	not finished	not finished	not finished
Progress	Entire text	Last paragraph not finished	Entire text, tried to fix error with ChatGPT	First three paragraphs done, started forth
Errors	1 new Actual Error, 1 missed Style choice	1 new style choice	1 new actual error	1 new Style choice, 1 missed style error in unedited tex

Table 4.2: Editing experiment results

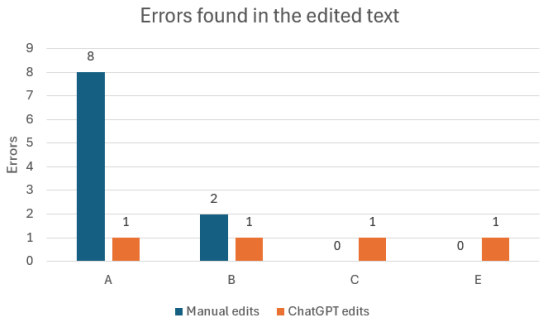


Figure 4.8: Number of errors by participant and task for edited text

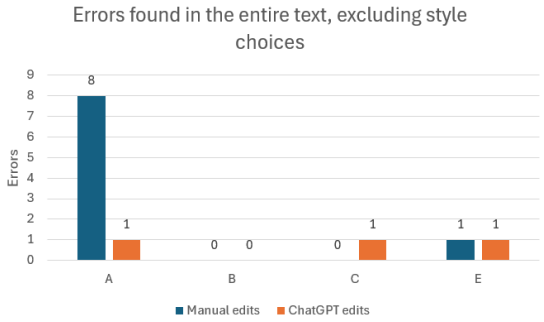


Figure 4.9: Number of errors, excluding style choices, by participant and task for entire text

4.5.2 First Impressions

This question was asked before the experiment. All participants noted that the guide seemed to be useful. Both participant A and C noted that they use less formal language during prompting, often times just bullet points. All participants also offered suggestions for improvements, some of which were also noted after the experiment.

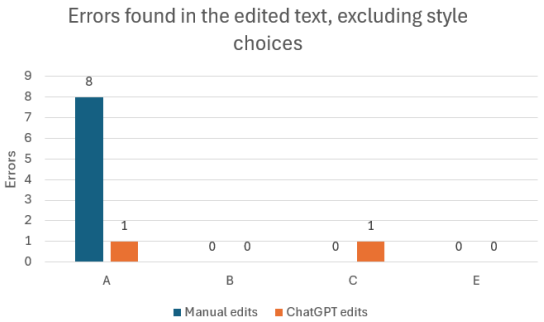


Figure 4.10: Number of errors, excluding style choices, by participant and task for edited text

4.5.3 Improvement Suggestions

All participants noted that additional information on how to deal with imperfect solutions should be provided. Some voiced this in the form of the decision-making process of when to switch to manual improvements instead of further ChatGPT prompting. Both participants A and C noted that the guide should provide a more profound understanding of the amount of text that can be input into ChatGPT at once. The need to have an explanation for the workflow diagram was raised by both participants C and E. The misbehavior of ChatGPT forgetting the task and requiring re-prompting was mentioned by both participant B and E.

4.5.4 Did the guide help the participants

All participants except for E agreed that the guide helps. Reasons stated were the inclusion of easy to use prompting patterns that could be copied directly from the guide. Having a 'Checklist' to work through was also mentioned. Participant E was unsure if the guide helped him.

4.5.5 Would the guide help new users

All participants agreed that the guide helps new users use ChatGPT more effectively. The inclusion of examples and a providing a structure were stated as reasons. One participant mentioned that this would reduce the need to develop your own strategies.

4.5.6 Speed Improvements

All participants said that ChatGPT can help increase the speed of writing papers, if editing is views as a part of writing a paper. Participant B was unsure how much the guide helped with writing, however in his opinion, it helps with using ChatGPT, which helps with writing. Interestingly, Participant B noted that the guide helps improve the speed of writing a paper despite being slower in the experiment with ChatGPT. When asked about it, he explained that he experimented more with different phrasings with ChatGPT, to see which one he liked best, while he only made minor changes when editing the text manually. Participant C noted that other AI editing tools like Grammarly were a greater help at speeding up the writing process by being able to integrate in the text editor.

4.5.7 Quality Improvements

All participants agreed that ChatGPT can help improve Quality, however participant E noted that the human author needed to be able to tell when a phrasing from ChatGPT is good or bad. Participant A noted that tools like Grammarly or DeepL Write are better suited, since they keep the written character of the original author, while ChatGPT always uses his character, thereby decreasing variation from different authors.

4.5.8 Warnings about ChatGPT

All participants agreed that the limitations need to be addressed more dominantly, with some suggesting further additions to the warnings, like what to do when ChatGPT hangs.

Table 4.3: Interviews Evaluation

Topic	A	B	C	E
Impressions on Guide	Workflow seems like the one he uses, he uses bullet points instead of full sentences for prompts	Workflow matches his usage, providing background information and output constraints most usefull, personally uses more positive constraints [E.g. 'Do X like this...' instead of 'Don't do it like this...']	Unclear what 80% solution means, Guide seems usefull, has examples, prefers light theme for ChatGPT screenshots, unclear when a strategy should be used, what kind of errors can happen. Uses keywords for prompting, unclear what guide is based on	Workflow diagram needs a short section to discribe content as a 'Quick Start Guide', When does ChatGPT forget the task, Needs hint that doing a task manually when ChatGPT refuses to do it properly is better.
Improvements for Guide	Limit of Characters for input and output not clear, sometimes Background information gets mixed up with content - how to deal with that	Working with the output can be more defined, e.g. what to do when ChatGPT is stubborn, Hint towards re-promiting, when GPT forgets the task	Typo in Workflow graphic, What does figure 5.1 refer to. What to do after modifiyng manually? Combine ChatGPT/Manual edit for new input? No need to write whole sentences for prompting. Add hint that pushing AI to much causes halucinations. Add more impromation on input length	When does ChatGPT forget the task, Needs hint that doing a task manually when ChatGPT refuses to do it properly is better. How do you handle the 80% solutions? How to decide when you should do the rest manually.

Continued on next page

Table 4.3 – continued from previous page

Topic	A	B	C	E
Guide for you	It helped, used snippets direclty from guide, like 'Do not loose Information', was more carefull when providing background information. In addition, used 'use objective language' to remove 'spicy words' [meant are fancy wordings, which ChatGPT regularly uses]	Helps, even though it is his normal workflow by being visable worklfow and providing 'a checklist in a way', Used some phrasing from Guide	Helped, used several sections directly from guide, like assigning a role, which made the output 'really good', but lost a reference	Guide might help, but hard to tell after a single use.
Guide for new users	For new users, guide is really really helpfull by providing structure and snippets	Good starting step, helps woth onboarding for ChatGPT. [No need to develop own expirience via 'hundreds of prompts']	It helps new users, but unsure how much impact the different sections have, thoug assigning a role definetly helps.	For new users it is helpfull, especially the warning in the begining, but should be more dominent

Continued on next page

Table 4.3 – continued from previous page

Topic	A	B	C	E
Improve Speed	Can improve speed, but not during writing itself, but for formating e.g. when another author of your research team used a different citation format.	ChatGPT helps speed up writing, not sure if guide istelf does, though it helps with using ChatGPT	It helps edit papers quicker, but other AI tools like Grammerly help better due to integration in the text editor. Result is comparable	helps for finalising a paper, hence increases speed
Improve Quality	Yes, but not as good as Grammerly. Problem: ChatGPT always has the same 'character' in texts, natural writers have variation. Grammerly is more natural	Yes, it does help remove odd phrasings and errors	Helps improve grammer and phrasing	Can help increase quality, if user has sence of quality [refering to fancy wording of ChatGPT, and being able to tell when to use a certain word].
Adressing of Limitations	Guide is partly informing but not warning. Limitations need more highlighting	It's mentioned but easy to forget, maybe add an example of halucination, make it more appearant. Might not be as important in rephrasing tasks.	What to do when ChatGPT hangs?	Warning about errors needs to be more pronounced

Continued on next page

Table 4.3 – continued from previous page

Topic	A	B	C	E
Other Remarks		Would be interesting to see how other AI Tools do (GPT 4.0, oterhs,...) with this guide. ChatGPT seems to get worse, forgetting the task quicker then when used a while back.	Interesting to know if editing prompt and regenerating output is worth it. Can ChatGPT check references?	Grammerly or DeepL Write are better at editing then ChatGPT. ChatGPT needs a lot of manual work [prompt, copying], while others work in editor.

4.6 Guideline improvements

Based on the suggestions by the evaluation interviews, several modifications were made to improve the guide, as seen in 4.7. These modifications include:

- Warnings how ChatGPT can misbehave
- Disclaimer how the guide was developed
- Description for the workflow diagram
- Updated workflow diagram
- Information on Token limitations
- More details on
- Citations
- Updated ChatGPT conversation in light mode

In addition, after confirming with my supervisor, the improved guideline was stylistically improved by a private friend, who used the content of my guide to create a standalone version. The changes to the guide made to form the standalone version, as well as the stylized version itself, are located in the appendix in chapter B.

4.7 Guideline version 2

ChatGPT can be a helpful tool during your research, but must be handled with caution to minimize the risk associated with generative AI. Such risks include ChatGPT falsifying data input by the user, loosing data or making up data. In addition, privacy concerns for all entered data must be considered [HQ+23]. This guide, based on user reports and a literature review, aims at providing insight into how ChatGPT can be utilized and which limitations must be considered.

General Guidelines

ChatGPT is a tool that can misbehave and make errors. You must always be knowledgeable in the area that you ask ChatGPT about in order to spot mistakes.
When making your prompts, try to give instructions without spelling or grammar mistakes, since either can lead to faulty interpretations of the task [Cia23].
Before entering any data you want ChatGPT to work on, start by generally providing background information, like your line of work or area of research [HQ+23]. This can be followed by telling ChatGPT how you want it to do, or not do your task, such as limitations of what it can change or how it talks. Lastly, define any output constraints, such as formatting restrictions[HQ+23]. Once you have done that, start by inputting the subject of your task. For text-based tasks like editing, use one paragraph at a time. Tasks, like programming, can be done by starting with a simplified version, that you modify in later steps [WTC22]. After receiving your first response by ChatGPT, check it for errors and needed modifications. You can ask ChatGPT to make modifications, do some manually, or combine both. After multiple worked on text passages, it may be necessary

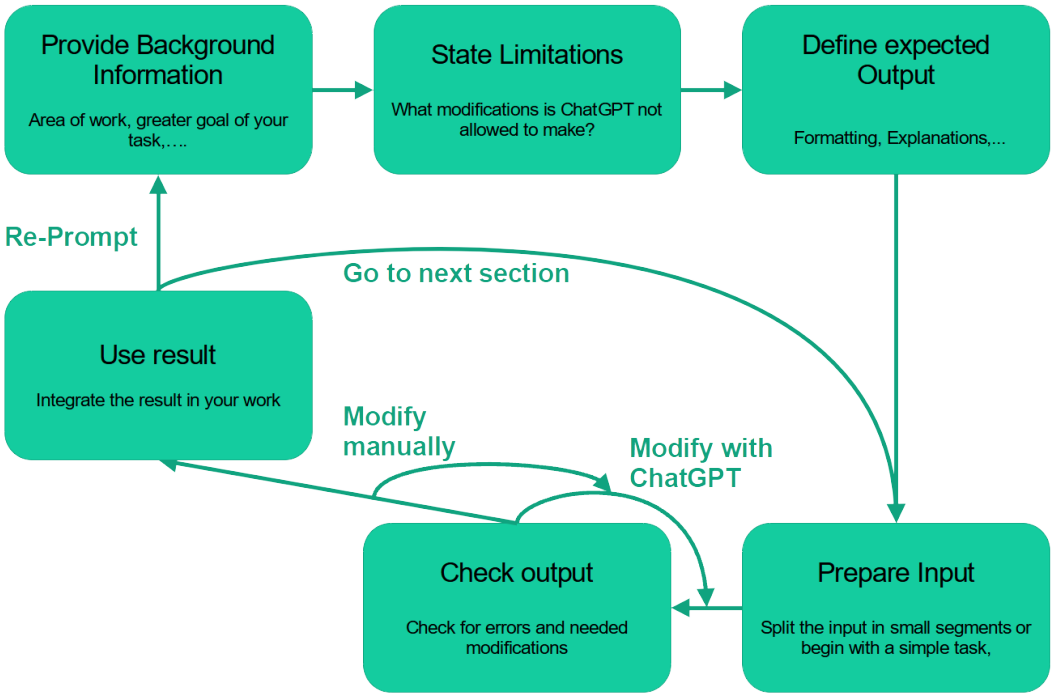


Figure 4.11: ChatGPT Prompting Workflow

to re-prompt ChatGTP with the background information, since it can forget them. Keep in mind, ChatGPT is good at providing '80%' solutions, which go a long way, but still need some work to be fully usable. Consider making the finishing touches manually. The workflow can be seen in figure 4.11.

Prompting Strategies

Provide Background Information

Before asking ChatGPT to do anything, you need to be aware of the task you want ChatGPT to perform. Here it is important to be as detailed as possible, with potential relevant information being the area of work you are doing, language selections (especially if code is involved) or any other outside input you already received. Telling ChatGPT to behave like a certain Role can help, but providing the same information in another form can yield similar results. Meta-information, like when and how you are providing the input, is also helpful. Potential prompts can include the following instructions:

- I need help editing a part of my paper about [Topic].
- I am writing a paper in [Field of Research] about [Topic] and need help with....
  - *Alternatively:* Pretend you are a researcher in [Field of Research] writing about [Topic]

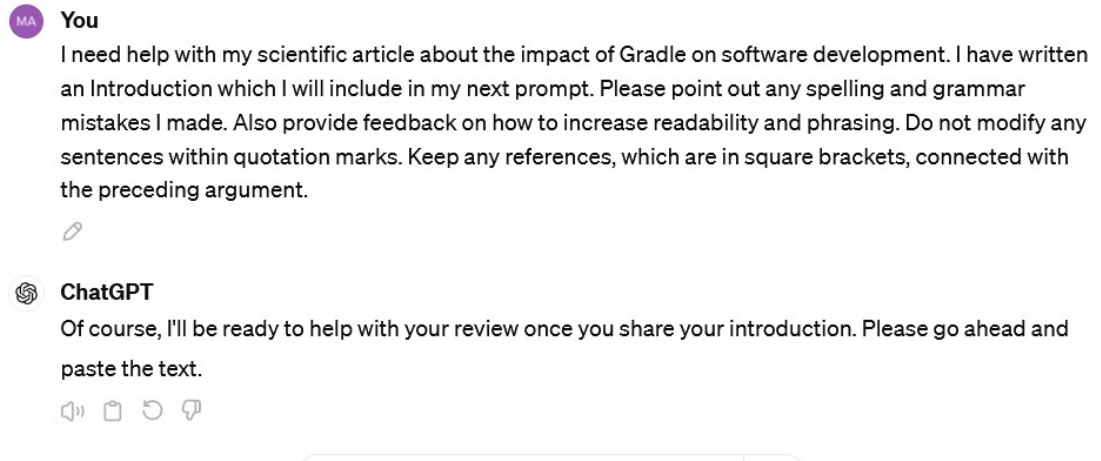


Figure 4.12: Initial prompt example containing background information, limitations and expected output

- I need to make changes to my paper to improve [Property, like readability, understandability,...]
- I will provide one section of my text after another, starting with the next prompt.

State limitations

Tell ChatGPT what it can and cannot do, to avoid undesired behavior. This is helpful to reduce the need for manual edits of the output, like removing modified quotes, however they are not a guarantee that ChatGPT will work as expected. Even with stated limitations, you must expect ChatGPT to break these constraints from time to time.

- Do not lose any information.
- Do not modify any statements surrounded by quotation marks.
- Do not add any new information.
- Use objective language.

4.7.1 Provide information about the expected output

This can include any desired output constraints, like formatting or explanations for the output. While you can also ask ChatGPT to highlight changes it made, the result is not to be trusted and will miss changed statements while also highlighting unchanged ones.

- Provide feedback why your changes improve the text.
- Keep the reply short.
  - *Alternatively:* Keep the reply within [Number] words or less.
- Provide multiple alternate solutions and explain the difference.

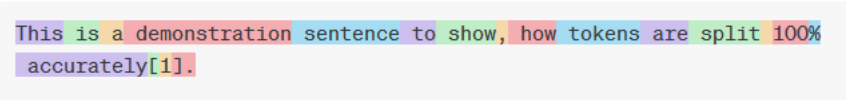


Figure 4.13: Example how words and numbers are split into tokens

Preparing the input

Due to ChatGPT being token-based, lengthy inputs will not be processed as expected. To mitigate this, splitting your input into multiple smaller segments, can help. Depending on the task at hand, this can also influence the prompting strategy. Even though the maximum number of 4096 allowed input and output tokens [Kum] is not small, a token being a (part of a) word, number or sign (see figure 4.13, [Opea]), missing content can be observed well before reaching the limit. For editing, spellchecking or even translation purposes, simply going one paragraph at a time is therefore advised. Any longer sections can lead to loss of information, as can be seen in fig.4.14 when entering a single paragraph from the Gradle Wikipedia entry [Gradle24], ChatGPT can make suggestions for the entire paragraph. In contrast, entering the entire article, as seen in fig. 4.15, only suggested edits for the beginning few paragraphs, fig. 4.16, while ignoring the rest of the text. For other tasks, like programming, starting with a simple request and then requesting additions or changes in additional prompts can help achieve the desired output [WTC22].

Work with the output

After receiving the output from ChatGPT, make sure that all constraints are met and the result addresses your task. For editing tasks, manually fixing errors, like a modified quote, may be more efficient than prompting ChatGPT again. For more complex changes, you can tell ChatGPT about the error and ask it to do the task again. While this can help, it can also lead to new errors in different parts or no improvement at all. Should this happen, making the final adjustments manually is advised. You can also combine manual and ChatGPT edits of the initial output, by editing it manually and the ask ChatGPT to do further adjustment. When working with text-based tasks, where you can have multiple replies for the same section, combining suggestions by ChatGPT can also improve the end result.
If efforts to correct errors or improve the output are unsuccessful, don't hesitate to start over in a new chat. Here you can input the previous chat history as background information or begin from scratch. When modifying a response from ChatGPT, you can use these patterns:

- Please add this argument...
- Suggest alternatives for [Part you did not like]
- Modify the code to do [some constraint like return a status code instead of void]

You

Gradle is a build automation tool for multi-language software development. It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing. Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript.[2] Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven.[3] Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management. It runs on the Java Virtual Machine.[4]

ChatGPT

Your introduction is clear and provides a good overview of Gradle and its key features. Here are some minor suggestions for improving readability and phrasing:

1. "Gradle is a build automation tool for multi-language software development." -> "Gradle is a build automation tool designed for multi-language software development."
2. "It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing." -> "It manages various stages of the development process, including compilation, packaging, testing, deployment, and publishing."
3. "Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript." -> "Supported languages encompass Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript."
4. "Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven." -> "Gradle builds upon the concepts of Apache Ant and Apache Maven, introducing a domain-specific language based on Groovy and Kotlin, in contrast to Maven's XML-based project configuration."
5. "Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management." -> "Gradle utilizes a directed acyclic graph to establish the order in which tasks are executed, thus facilitating dependency management."

Overall, your introduction is well-written and effectively introduces the reader to the topic of Gradle and its significance in software development.

Figure 4.14: Editing suggestions, when only a paragraph is entered

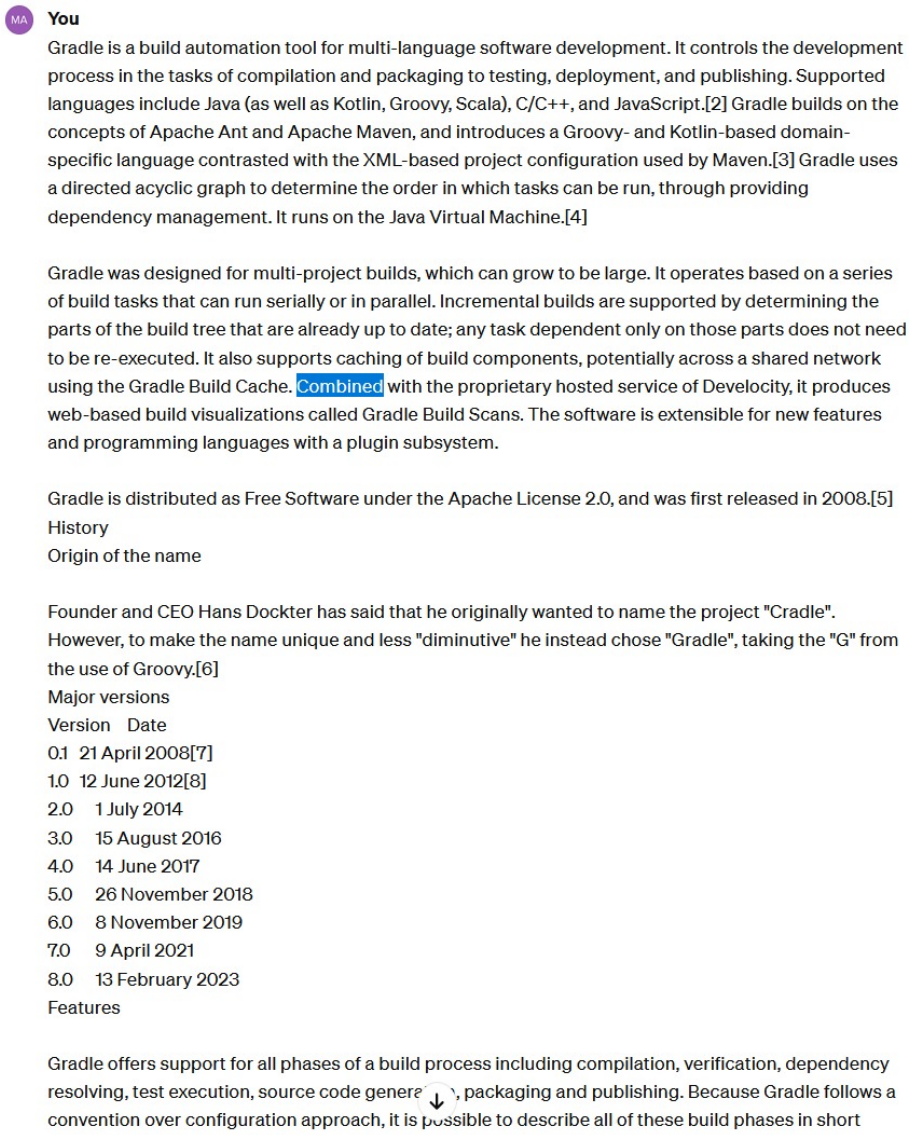
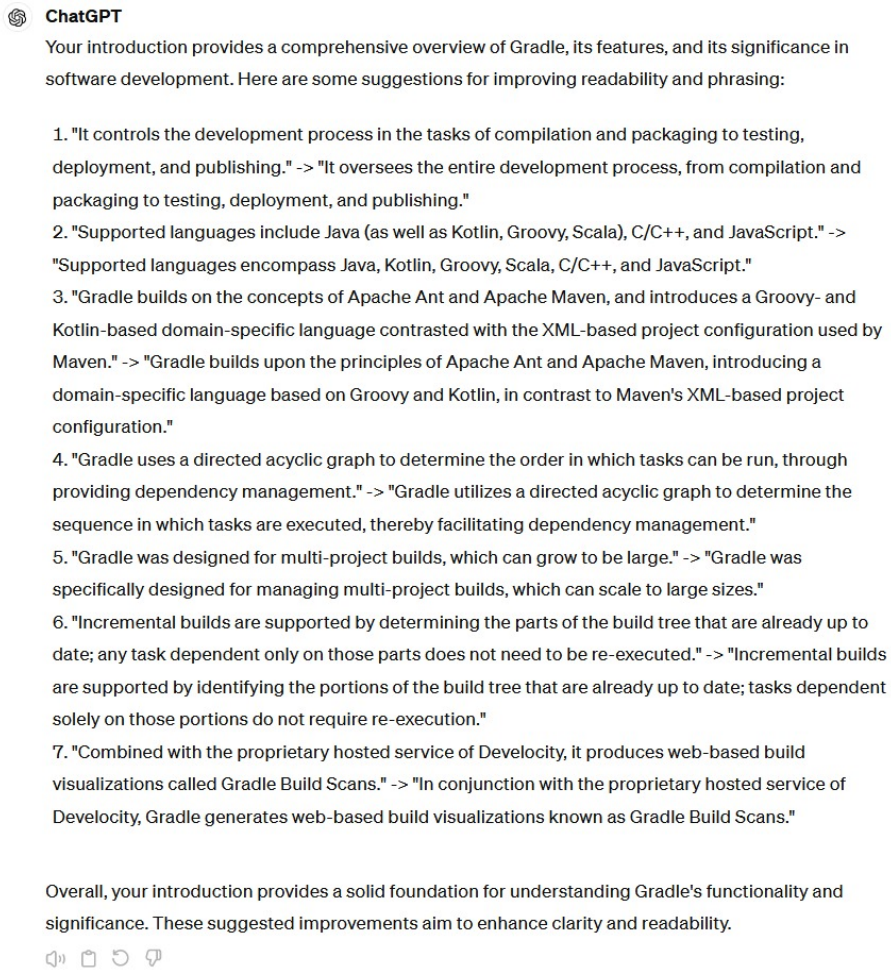


Figure 4.15: Input of entire texts at once, last edit suggestion highlighted



ChatGPT

Your introduction provides a comprehensive overview of Gradle, its features, and its significance in software development. Here are some suggestions for improving readability and phrasing:

1. "It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing." -> "It oversees the entire development process, from compilation and packaging to testing, deployment, and publishing."
2. "Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript." -> "Supported languages encompass Java, Kotlin, Groovy, Scala, C/C++, and JavaScript."
3. "Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven." -> "Gradle builds upon the principles of Apache Ant and Apache Maven, introducing a domain-specific language based on Groovy and Kotlin, in contrast to Maven's XML-based project configuration."
4. "Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management." -> "Gradle utilizes a directed acyclic graph to determine the sequence in which tasks are executed, thereby facilitating dependency management."
5. "Gradle was designed for multi-project builds, which can grow to be large." -> "Gradle was specifically designed for managing multi-project builds, which can scale to large sizes."
6. "Incremental builds are supported by determining the parts of the build tree that are already up to date; any task dependent only on those parts does not need to be re-executed." -> "Incremental builds are supported by identifying the portions of the build tree that are already up to date; tasks dependent solely on those portions do not require re-execution."
7. "Combined with the proprietary hosted service of Develocity, it produces web-based build visualizations called Gradle Build Scans." -> "In conjunction with the proprietary hosted service of Develocity, Gradle generates web-based build visualizations known as Gradle Build Scans."

Overall, your introduction provides a solid foundation for understanding Gradle's functionality and significance. These suggested improvements aim to enhance clarity and readability.

Figure 4.16: Editing suggestions for entire text, showing incomplete text analysis

5 Discussion

During the evaluation experiment, it was interesting to see that all participants, except participant E, chose to edit the text on a deeper level when using ChatGPT. Instead of focusing on finding and fixing flaws in the text, they attempted to create a better text in general. Participant E choose the deep editing style for both texts. The behavior of the other participants could have been caused by the experiment design, with the introduction of the 15-minute time cap and the participants knowing beforehand that there are flaws in the text. However, I think that the study design is still comparable to the real world, since publishing deadlines are a regular occurrence and many people make errors in their initial drafts. So even though the measured speedup was limited or even nonexistent for most participants, having the option to make larger rewrites, to improve the text within the same timeframe, can be beneficial to the final version of the paper.
The high levels of variations during the guides' evaluation experiment between participants A and the rest highlights a problem caused by the small sample size, namely is participant A an outlier or a member of a statistically significant group. Such groups could include non-native speakers or people with other linguistic barriers. Alternatively, it is possible that the perceived 'overreliance on Grammarly' which participant A mentioned during the experiment, could point towards another cause. However, the question remaining is the same: are people with similar overreliance a significant group, or is he an outlier. In my opinion, participant A should be included in the analysis as a token representative of non-native speakers with linguistic barriers, even though during the interviews he spoke without any perceivable flaws.
Furthermore, It is noteworthy that Grammarly or DeepL Write are AI tools like ChatGPT, though they use a different user interface. They are built for the very use case, I identified to be the most suited for ChatGPT. However, with their integration into editors, they are easier to use while, according to two participants, offering the same benefit. This calls into question the usefulness of ChatGPT as a writing assistant over other AI tools, but not AI tools in general.
When looking at the guidelines, it is also worth mentioning that multiple suggestions by the participants could not be verified by the literature. However, they are easily reproducible, such as ChatGPT using too many fancy words, or that it is forgetting the original task after several follow-up prompts. I myself made such observations during my private use with ChatGPT, leading me to include them in the guide, despite no scientific paper analyzing these issues yet. This was done, since these issues are things to look out for when using ChatGPT and can impact the quality of the result. However, the participants also mentioned the unverifiable observation, which I did not include, that the quality of ChatGPT responses declined over several months. I chose to exclude this observation because it had no impact on how to use ChatGPT right now, or how to use the tool in general. After all, a general decline in quality cannot be combated with a prompting strategy.

6 Conclusion and Outlook

6.1 Conclusion

With this thesis, I analyzed the potential benefit ChatGPT can have on scientific writing in the software engineering domain. By interviewing researchers and analyzing literature, I identified editing written texts for phrasing, grammar, and spelling improvements to be the most beneficiary aspect for the academic writing process. With the knowledge gathered this way, I compiled a guideline to help users utilize ChatGPT for said task and evaluated it with an experiment and interviews. The evaluation showed that the use of ChatGPT could help reduce the number of errors made by users with limited manual editing proficiency, while also providing a significant speedup. For other users, no reduction of errors was found and a limited speedup, if any, was measured. However, those users were able to increase the depth of their editing from error fixing and minor rewrites to major rewrites. This can potentially lead to an improvement of text quality beyond measurable errors.

6.2 Threats to Validity

As discussed before, the limited number of participants makes it difficult to generalize any findings, since only a single participant displayed a significant improvement. Additionally, with only five participants in the first interview round, it is likely that additional participants could have added new insights on how to better utilize ChatGPT as a writing assistant. When looking at the literature analyzed, another large problem was the high number of papers that can be found, with many having limited quality. Since the high number forced me to look only at a subset of papers, it is likely that several high-quality papers with valuable information were missed. The large percentage of low-quality papers, without any evaluation of the research, can be attributed to a rush of publications when ChatGPT was published. It is likely that the average quality of the papers about using ChatGPT for writing will improve.
Last but not least, changes to ChatGPT itself can impact the result of this thesis. Such changes could potentially happen due to the language model changing with use, like some users suggest, when they say that ChatGPT gets worse over time. Additionally, OpenAI, the creators of ChatGPT can change the behavior of the model, for example to comply with future legal requirements.

6.3 Outlook

My thesis analyzed the most likely use case of using ChatGPT as a writing assistant, however other use cases, such a generating certain parts, like the conclusion or abstract of papers, could be beneficial too. Additionally, it would be interesting to see in a larger scale study if and how

ChatGPT can impact the quality of scientific publications. This would allow a better statistical analysis of how beneficial ChatGPT is, since you could tell how often participants like A appear. Lastly, analyzing how different AI tools, like Grammarly, DeepL Write or ChatGPT compare to one another both in speed and quality could also go a long way in answering if ChatGPT can be utilized as a writing assistant.

Bibliography

[AD] Alex Kantrowitz, Douglas Gorman. *ChatGPT isn't good enough to take jobs and is unlikely to cause mass layoffs: 'The hot takes have run into reality'*. accessed on 02.05.2024. URL: https://www.businessinsider.com/chatgpt-isnt-good-enough-to-take-jobs-unlikely-mass-layoffs-2023-8 (cit. on p. 11).

[AEA+23] A. Alshami, M. Elsayed, E. Ali, A. E. E. Eltoukhy, T. Zayed. "Harnessing the Power of ChatGPT for Automating Systematic Review Process: Methodology, Case Study, Limitations, and Future Directions". In: *Systems* 11.7 (2023). ISSN: 2079-8954. DOI: 10.3390/systems11070351. URL: https://www.mdpi.com/2079-8954/11/7/351 (cit. on p. 15).

[AWL+23] A. Ahmad, M. Waseem, P. Liang, M. Fahmideh, M. S. Aktar, T. Mikkonen. "Towards Human-Bot Collaborative Software Architecting with ChatGPT". In: *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering*. EASE '23. Oulu, Finland: Association for Computing Machinery, 2023, pp. 279–285. DOI: 10.1145/3593434.3593468. URL: https://doi.org/10.1145/3593434.3593468 (cit. on p. 27).

[Bud] T. Budig. *Welche Partei würde ChatGTP wählen?* accessed on 09.01.2024. URL: https://www.linkedin.com/posts/tobias-budig_welche-partei-w%C3%BCrde-chatgtp-w%C3%A4hlen-um-das-activity-7016784489624342528-nxTs (cit. on p. 25).

[Cia23] E. J. Ciaccio. "Use of artificial intelligence in scientific paper writing". In: *Informatics in Medicine Unlocked* 41 (2023), p. 101253. ISSN: 2352-9148. DOI: https://doi.org/10.1016/j.imu.2023.101253. URL: https://www.sciencedirect.com/science/article/pii/S2352914823000953 (cit. on pp. 26, 53).

[Els] Elsevier. *The use of generative AI and AI-assisted technologies in writing for Elsevier*. accessed on 09.01.2024. URL: https://www.elsevier.com/about/policies-and-standards/the-use-of-generative-ai-and-ai-assisted-technologies-in-writing-for-elsevier (cit. on p. 26).

[GC23] A. D. Giglio, M. U. P. d. Costa. "The use of artificial intelligence to improve the scientific writing of non-native english speakers". In: *Revista da Associação Médica Brasileira* 69.9 (2023), e20230560. ISSN: 0104-4230. DOI: 10.1590/1806-9282.20230560. URL: https://doi.org/10.1590/1806-9282.20230560 (cit. on p. 26).

[GE23] G. Grimaldi, B. Ehrler. "AI et al.: Machines Are About to Change Scientific Publishing Forever". In: *ACS Energy Letters* 8.1 (2023), pp. 878–880. DOI: 10.1021/acsenergylett.2c02828. eprint: https://doi.org/10.1021/acsenergylett.2c02828. URL: https://doi.org/10.1021/acsenergylett.2c02828 (cit. on pp. 25–27).

[Gradle24] Wikipedia. *Gradle*. 11.04.2024. URL: https://en.wikipedia.org/wiki/Gradle (cit. on pp. 39, 56).

[Har] A.-W. Harzing. *Publish Or Perish*. https://harzing.com/resources/publish-or-perish (cit. on p. 17).

[HQ+23] M. U. Hadi, q. a. tashi q. a., R. Qureshi, A. Shah, a. muneer a., M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, S. Mirjalili. "A survey on large language models: applications, challenges, limitations, and practical usage". In: (2023). DOI: 10.36227/techrxiv.23589741.v1 (cit. on pp. 15, 27, 28, 53).

[HS23] H. Hassani, E. S. Silva. "The Role of ChatGPT in Data Science: How AI-Assisted Conversational Interfaces Are Revolutionizing the Field". In: *Big Data and Cognitive Computing* 7.2 (2023). ISSN: 2504-2289. DOI: 10.3390/bdcc7020062. URL: https://www.mdpi.com/2504-2289/7/2/62 (cit. on pp. 25, 27).

[JD] Jem Bartholomnew, Dhrumil Mehta. *How the media is covering ChatGPT*. accessed on 02.05.2024. URL: https://www.cjr.org/tow_center/media-coverage-chatgpt.php (cit. on p. 11).

[Joo23] P. Joo-Young. "Could ChatGPT help you to write your next scientific paper?: concerns on research ethics related to usage of artificial intelligence tools". In: *jkaoms* 49.3 (2023), pp. 105–106. DOI: 10.5125/jkaoms.2023.49.3.105. eprint: http://www.e-sciencecentral.org/articles/?scid=1516083059. URL: http://www.e-sciencecentral.org/articles/?scid=1516083059 (cit. on p. 26).

[KG18] Karl Popper, Guiseppe Franco. *Handbuch Karl Popper*. Springer VS Wiesbaden, 2018. ISBN: 978-3-658-16242-9. DOI: https://doi.org/10.1007/978-3-658-16242-9 (cit. on p. 12).

[KÖ-+23] O. KATAR, D. ÖZKAN, G. -3, Ö. YILDIRIM, U. R. ACHARYA. "Evaluation of GPT-3 AI Language Model in Research Paper Writing". In: *Turkish Journal of Science and Technology* 18.2 (2023), pp. 311–318. DOI: 10.55525/tjst.1272369 (cit. on p. 26).

[Kum] M. Kumar. *Understanding Tokens in ChatGPT*. accesses on 29.04.2024. URL: https://medium.com/@manav.kumar87/understanding-tokens-in-chatgpt-32845987858d (cit. on pp. 28, 56).

[Kum23] A. H. Kumar. "Analysis of ChatGPT Tool to Assess the Potential of its Utility for Academic Writing in Biomedical Domain". In: *Biology, Engineering, Medicine and Science Reports* 9.1 (Feb. 2023), pp. 24–30. DOI: 10.5530/bems.9.1.5. URL: https://www.bemsreports.org/index.php/bems/article/view/132 (cit. on p. 15).

[Loc] S. Lock. *What is AI chatbot phenomenon ChatGPT and could it replace humans?* accessed on 02.05.2024. URL: https://www.theguardian.com/technology/2022/dec/05/what-is-ai-chatbot-phenomenon-chatgpt-and-could-it-replace-humans (cit. on p. 11).

[MČK+23] M. Májovský, M. Černý, M. Kasal, M. Komarc, D. Netuka. "Artificial Intelligence Can Generate Fraudulent but Authentic-Looking Scientific Medical Articles: Pandora's Box Has Been Opened". In: *J Med Internet Res* 25 (2023), e46924. ISSN: 1438-8871. DOI: 10.2196/46924. URL: http://www.ncbi.nlm.nih.gov/pubmed/3725668 (cit. on pp. 11, 25, 27).

[MUM+23] J. G. Meyer, R. J. Urbanowicz, P. C. N. Martin, K. O'Connor, R. Li, P.-C. Peng, T. J. Bright, N. Tatonetti, K. J. Won, G. Gonzalez-Hernandez, J. H. Moore. "ChatGPT and large language models in academia: opportunities and challenges". In: *BioData Mining* (2023). DOI: 10.1186/s13040-023-00339-9 (cit. on p. 15).

[NRS23] M. Neumann, M. Rauschenberger, E.-M. Schön. ""We Need To Talk About ChatGPT": The Future of AI and Higher Education". en. 2023 (cit. on p. 27).

[Opea] OpenAI. *Tokenizer - Learn about language model tokenization*. Manual input of shown text, accesses on 29.04.2024. URL: https://platform.openai.com/tokenizer (cit. on p. 56).

[Opeb] OpenAI. *What is ChatGPT?* accessed on 09.01.2024. URL: https://help.openai.com/en/articles/6783457-what-is-chatgpt (cit. on p. 25).

[Roo] K. Roose. *The Brilliance nad Weirdness of ChatGPT*. accessed via WayBackMachine on 02.05.2024. URL: https://web.archive.org/web/20230118134332/https://www.nytimes.com/2022/12/05/technology/chatgpt-ai-twitter.html (cit. on p. 11).

[Sch23] M. S. Schäfer. "The Notorious GPT: science communication in the age of artificial intelligence". In: *JCOM 22(02), Y02.* (2023). DOI: 10.22323/2.22020402 (cit. on p. 25).

[SRM+23] A. Shibani, R. Rajalakshmi, F. Mattins, S. Selvaraj, S. Knight. "Visual Representation of Co-Authorship with GPT-3: Studying Human-Machine Interaction for Effective Writing". In: *International Educational Data Mining Society, Paper presented at the International Conference on Educational Data Mining (EDM) (16th, Bengaluru, India, Jul 11-14, 2023)* (2023). URL: https://eric.ed.gov/?id=ED630873 (cit. on p. 26).

[STG23] M. Salvagno, F. S. Taccone, A. G. Gerli. "Can artificial intelligence help for scientific writing?" In: *Critical Care* (2023). DOI: 10.1186/s13054-023-04380-2 (cit. on pp. 15, 27).

[Sut] SutdiBucht. *Sind Ghostwriter legal?* accesses on 10.01.2024. URL: https://studibucht.de/blog/ist-ghostwriting-legal/#Die_rechtliche_Basis_fuer_das_Ghostwriting (cit. on p. 26).

[TBKG23] Z. Tafferner, I. Balázs, O. Krammer, A. Géczy. "Can ChatGPT Help in Electronics Research and Development? A Case Study with Applied Sensors". In: *Sensors* 23.10 (2023). ISSN: 1424-8220. DOI: 10.3390/s23104879. URL: https://www.mdpi.com/1424-8220/23/10/4879 (cit. on pp. 15, 25).

[VSP+17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin. "Attention is All you Need". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf (cit. on p. 11).

[WTC22] T. Wu, M. Terry, C. J. Cai. “AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts”. In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. CHI ’22. , New Orleans, LA, USA, Association for Computing Machinery, 2022. ISBN: 9781450391573. DOI: 10.1145/3491102.3517582. URL: https://doi.org/10.1145/3491102.3517582 (cit. on pp. 28, 53, 56).

[WW23] W. H. Walters, E. I. Wilder. “Fabrication and errors in the bibliographic citations generated by ChatGPT”. In: *Scientific Reports* (2023). DOI: 10.1038/s41598-023-41032-5 (cit. on p. 15).

A Texts for experiment

A.1 Git-1

Managing software during development, be it for research, public or commercial use, can be challenging on its own. Trying out different avenues to solve a problem or changes in requirements can force you to revert written codes and creates the challenge of keeping track of the changes you made. In addition, the code for a project is rarely written by a single person, but rather a team of different developers who need to share their individual progress among each other.
To solve these Challenges, version control software has been developed.
Currently, the most used version control software is git [1], a versioning tool first developed by Linus Torvalds in 2005 [2] with a strong focus on speed, support of non-linear development, fully distributed workflow and the ability "to handle large projects like the Linux kernel efficiently (speed and data size)" [3].
In order to achieve these requirements, git works with local repositories in addition to the remote repository, where each local repository contains the full development history. Changes from other repositories can be imported as development branches, branches being an alternate version of the project, and merged just like local branches [4]. Due to the local repository, tasks like comparing changes or switching branches are also much faster then centralized versioning software like SVN [5].
In addition, git is temper proof, allowing quick identification of any changes made after a commit occurred, including who made the change. This is made with the utilization of hash trees. Here, each commit the hashes of files are used to calculate the hash of the entire commit, thereby allowing quick detection of modifications of even a single file [6].
When using git for the first time, developers need to 'pull' a repository from the remote location, thereby downloading it to the local machine. From here, the developer can use the 'checkout' command to move to a different development branch or a previous commit. Changes from one branch can be added to another with 'merge' and changes within the working files can be added to your local repository with a 'commit' after which they can be uploaded to the remote repository with a 'push'.

[1] https://stackoverflow.blog/2023/01/09/beyond-git-the-other-version-control-systems-develo pers-use/
[2] https://github.com/git/git/commit/e83c5163316f89bfbde7d9ab23ca2e25604af290
[3] https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git
[4] https://git-scm.com/book/en/v2/Distributed-Git-Distributed-Workflows
[5] https://digitalvampire.org/blog/index.php/2006/11/16/oh-what-a-relief-it-is/
[6] https://mirrors.edge.kernel.org/pub/software/scm/git/docs/user-manual.html#trust

A.2 Git-2

Software development is often made in larger teams, making collaboration a key requirement to work on the same source codes. This, paired with the reality that written code has to be changed frequently, be it due to requirement changes or to fix problems, creates a need to keep proper track of any changes you make.
To solve these challenges, version control software has been developed.
Git, a versioning tool first developed by Linus Torvalds in 2005 [1], is currently the most used version control software [2]. It focuses on speed, non-linear development, a fully distributed workflow and the ability "to handle large projects like the Linux kernel efficiently (speed and data size)" [3].
It also features security concepts to ensure that once a change is committed, it cannot be manipulated. This is done by calculating a hash for each file, who are then used to calculate a hash of the entire commit as a fingerprint. This tree structure, called a hash tree, allows easy detection even if a single file has been modified [4].
The other requirements are achieved by using local repositories in addition to a remote repository, where each local repository contains the full development history. By using local repositories, changes between branches are easier calculated and switching is also much faster then centralized version software like SVN [5]. Transfers between different repositories are treated like merges of local branches, once they have been pulled from the remote repository [6].
The general use cases for git are the following: to download a project to your local machine, you do a 'checkout' from the remote repository. To change branches or move to a different commit, a 'checkout' can be performed. Two branches can be combines with a 'merge' and changes made to the working files can be saved to the local repository with a 'commit'. Committed changed must be 'pushed' to a remote repository in order to be accessible for other developers.

[1] https://github.com/git/git/commit/e83c5163316f89bfbde7d9ab23ca2e25604af290
[2]https://stackoverflow.blog/2023/01/09/beyond-git-the-other-version-control-systems-develo pers-use/
[3]https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git
[4]https://mirrors.edge.kernel.org/pub/software/scm/git/docs/user-manual.html#trust
[5]https://digitalvampire.org/blog/index.php/2006/11/16/oh-what-a-relief-it-is/
[6]https://git-scm.com/book/en/v2/Distributed-Git-Distributed-Workflows

B Stylized Guide

The stylized guide was designed by Jasmin Seitz. The content is from my thesis, with minor changes to allow the use of the document without the rest of this thesis.
These changes were made to the content:

- Added Title page
- The sentence 'ChatGPT is a tool that can misbehave and make errors' was placed more dominantly
- Highlighting of potential errors to look out for
- Updated in-document references
- Added sources referred to by guideline

„ … “

Using ChatGPT for scientific authoring

in the Software Engineering Domain

GUIDE BY
Markus Walter

DESIGNED BY
Jasmin Seitz

ChatGPT can be a helpful tool during your research, but must be handled with caution to minimize the risk associated with generative AI.

ChatGPT is a tool that can misbehave and make ERRORS.

Such risks include ChatGPT falsifying data input by the user, loosing data or making up data. In addition, privacy concerns for all entered data must be considered [HQ+23]. This guide, based on user reports and a literature review, aims at providing insight into how ChatGPT can be utilized and which limitations must be considered.

GENERAL GUIDELINES

You must always be knowledgeable in the area that you ask ChatGPT about in order to spot mistakes. When making your prompts, try to give instructions without spelling or grammar mistakes, since either can lead to faulty interpretations of the task [Cia23].

Before entering any data you want ChatGPT to work on, start by generally providing background information, like your line of work or area of research [HQ+23]. This can be followed by telling ChatGPT how you want it to do, or not do your task, such as limitations of what it can change or how it talks.

Lastly, define any output constraints, such as formatting restrictions [HQ+23].
Once you have done that, start by inputting the subject of your task.
For text-based tasks like editing, use one paragraph at a time. Tasks, like programming, can be done by starting with a simplified version, that you modify in later steps [WTC22]. After receiving your first response by ChatGPT, check it for errors and needed modifications.

You can ask ChatGPT to make modifications, do some manually, or combine both. After multiple worked on text passages, it may be necessary to reprompt ChatGTP with the background information, since it can forget them.

Keep in mind, ChatGPT is good at providing ’80%’ solutions, which go a long way, but still need some work to be fully usable. Consider making the finishing touches manually.

The workflow can be seen in figure 1.

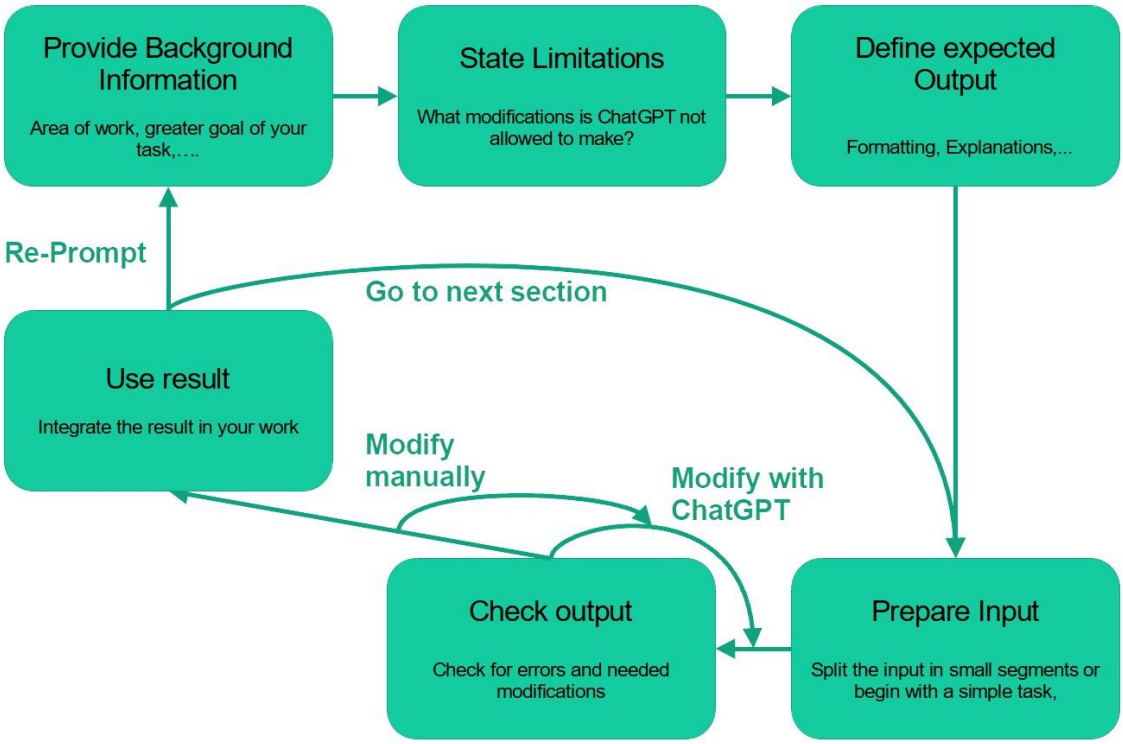


Figure 1: ChatGPT Prompting Workflow

PROMPTING STRATEGIES

Provide Background Information

Before asking ChatGPT to do anything, you need to be aware of the task you want ChatGPT to perform. Here it is important to be as detailed as possible, with potential relevant information being the area of work you are doing, language selections (especially if code is involved) or any other outside input you already received.
Telling ChatGPT to behave like a certain Role can help, but providing the same information in another form can yield similar results. Meta-information, like when and how you are providing the input, is also helpful.

Potential prompts can include the following instructions:

- I need help editing a part of my paper about [Topic].
- I am writing a paper in [Field of Research] about [Topic] and need help with....
  - *Alternatively*: Pretend you are a researcher in [Field of Research] writing about [Topic]
- I need to make changes to my paper to improve [Property, like readability, understandability,...]
- I will provide one section of my text after another, starting with the next prompt.

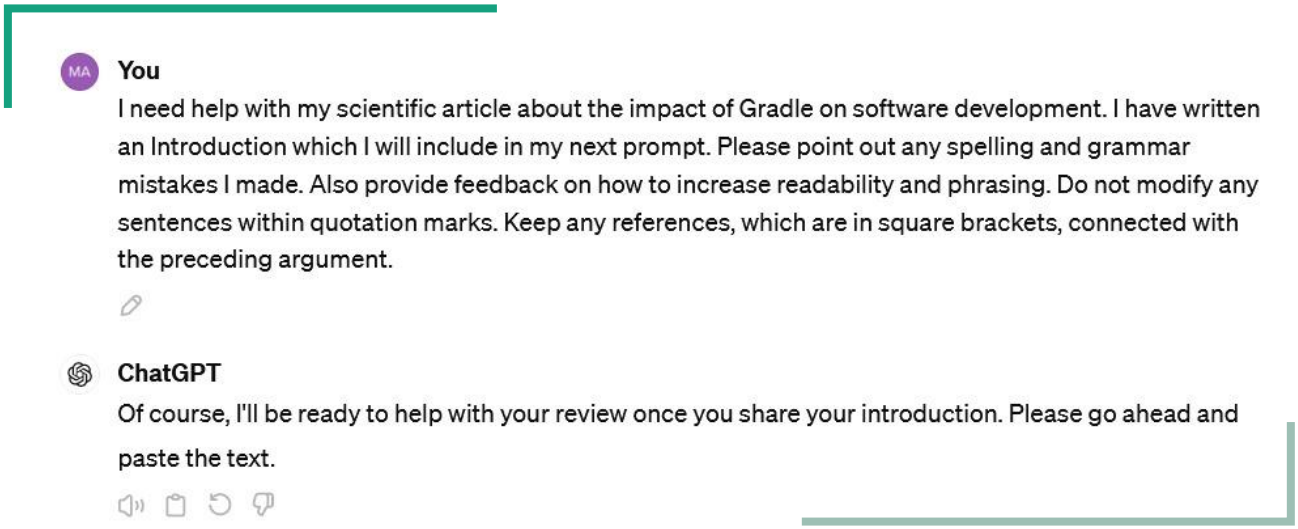


Figure 2: Initial prompt example containing background information, limitations and expected output

State limitations

Tell ChatGPT what it can and cannot do, to avoid undesired behavior. This is helpful to reduce the need for manual edits of the output, like removing modified quotes, however they are not a guarantee that ChatGPT will work as expected. Even with stated limitations, you must expect ChatGPT to break these constraints from time to time.

- Do not lose any information.
- Do not modify any statements surrounded by quotation marks.
- Do not add any new information.
- Use objective language.

Provide information about the expected output

This can include any desired output constraints, like formatting or explanations for the output. While you can also ask ChatGPT to highlight changes it made, the result is not to be trusted and will miss changed statements while also highlighting unchanged ones.

- Provide feedback why your changes improve the text.
- Keep the reply short.
  - *Alternatively*: Keep the reply within [Number] words or less.
- Provide multiple alternate solutions and explain the difference.

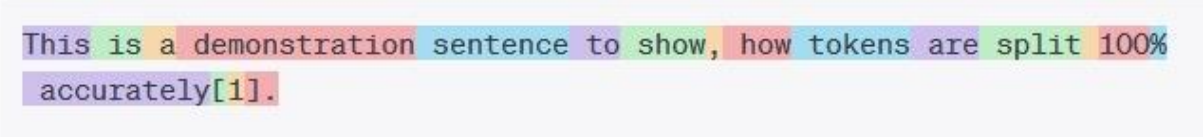


Figure 3: Example how words and numbers are split into tokens [Opea]

Preparing the input

Due to ChatGPT being token-based, lengthy inputs will not be processed as expected. To mitigate this, splitting your input into multiple smaller segments, can help.

Depending on the task at hand, this can also influence the prompting strategy. Even though the maximum number of 4096 allowed input and output tokens [Kum] is not small, a token being a (part of a) word, number or sign (see figure 3), missing content can be observed well before reaching the limit. For editing, spellchecking or even translation purposes, simply going one paragraph at a time is therefore advised.

Any longer sections can lead to loss of information, as can be seen in fig. 4 when entering a single paragraph from the Gradle Wikipedia entry [Gradle24], ChatGPT can make suggestions for the entire paragraph. In contrast, entering the entire article, as seen in fig. 5, only suggested edits for the beginning few paragraphs, fig. 6, while ignoring the rest of the text. For other tasks, like programming, starting with a simple request and then requesting additions or changes in additional prompts can help achieve the desired output [WTC22].

Work with the output

After receiving the output from ChatGPT, make sure that all constraints are met and the result addresses your task. For editing tasks, manually fixing errors, like a modified quote, may be more efficient than prompting ChatGPT again.

For more complex changes, you can tell ChatGPT about the error and ask it to do the task again. While this can help, it can also lead to new errors in different parts or no improvement at all. Should this happen, making the final adjustments manually is advised. You can also combine manual and ChatGPT edits of the initial output, by editing it manually and the ask ChatGPT to do further adjustment. When working with text-based tasks, where you can have multiple replies for the same section, combining suggestions by ChatGPT can also improve the end result.

If efforts to correct errors or improve the output are unsuccessful, don't hesitate to start over in a new chat. Here you can input the previous chat history as background information or begin from scratch.

When modifying a response from ChatGPT, you can use these patterns:

- Please add this argument...
- Suggest alternatives for [Part you did not like]
- Modify the code to do [some constraint like return a status code instead of void]

You

Gradle is a build automation tool for multi-language software development. It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing. Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript.[2] Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven.[3] Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management. It runs on the Java Virtual Machine.[4]

ChatGPT

Your introduction is clear and provides a good overview of Gradle and its key features. Here are some minor suggestions for improving readability and phrasing:

1. "Gradle is a build automation tool for multi-language software development." -> "Gradle is a build automation tool designed for multi-language software development."
2. "It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing." -> "It manages various stages of the development process, including compilation, packaging, testing, deployment, and publishing."
3. "Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript." -> "Supported languages encompass Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript."
4. "Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven." -> "Gradle builds upon the concepts of Apache Ant and Apache Maven, introducing a domain-specific language based on Groovy and Kotlin, in contrast to Maven's XML-based project configuration."
5. "Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management." -> "Gradle utilizes a directed acyclic graph to establish the order in which tasks are executed, thus facilitating dependency management."

Overall, your introduction is well-written and effectively introduces the reader to the topic of Gradle and its significance in software development.

Figure 4: Editing suggestions, when only a paragraph is entered

MA **You**

Gradle is a build automation tool for multi-language software development. It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing. Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript.[2] Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven.[3] Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management. It runs on the Java Virtual Machine.[4]

Gradle was designed for multi-project builds, which can grow to be large. It operates based on a series of build tasks that can run serially or in parallel. Incremental builds are supported by determining the parts of the build tree that are already up to date; any task dependent only on those parts does not need to be re-executed. It also supports caching of build components, potentially across a shared network using the Gradle Build Cache. Combined with the proprietary hosted service of Develocity, it produces web-based build visualizations called Gradle Build Scans. The software is extensible for new features and programming languages with a plugin subsystem.

Gradle is distributed as Free Software under the Apache License 2.0, and was first released in 2008.[5]
History
Origin of the name

Founder and CEO Hans Dockter has said that he originally wanted to name the project "Cradle". However, to make the name unique and less "diminutive" he instead chose "Gradle", taking the "G" from the use of Groovy.[6]
Major versions

Version	Date
0.1	21 April 2008[7]
1.0	12 June 2012[8]
2.0	1 July 2014
3.0	15 August 2016
4.0	14 June 2017
5.0	26 November 2018
6.0	8 November 2019
7.0	9 April 2021
8.0	13 February 2023

Features

Gradle offers support for all phases of a build process including compilation, verification, dependency resolving, test execution, source code genera[illegible], packaging and publishing. Because Gradle follows a convention over configuration approach, it is possible to describe all of these build phases in short

Figure 5: Input of entire texts at once, last edit suggestion highlighted

ChatGPT

Your introduction provides a comprehensive overview of Gradle, its features, and its significance in software development. Here are some suggestions for improving readability and phrasing:

1. "It controls the development process in the tasks of compilation and packaging to testing, deployment, and publishing." -> "It oversees the entire development process, from compilation and packaging to testing, deployment, and publishing."
2. "Supported languages include Java (as well as Kotlin, Groovy, Scala), C/C++, and JavaScript." -> "Supported languages encompass Java, Kotlin, Groovy, Scala, C/C++, and JavaScript."
3. "Gradle builds on the concepts of Apache Ant and Apache Maven, and introduces a Groovy- and Kotlin-based domain-specific language contrasted with the XML-based project configuration used by Maven." -> "Gradle builds upon the principles of Apache Ant and Apache Maven, introducing a domain-specific language based on Groovy and Kotlin, in contrast to Maven's XML-based project configuration."
4. "Gradle uses a directed acyclic graph to determine the order in which tasks can be run, through providing dependency management." -> "Gradle utilizes a directed acyclic graph to determine the sequence in which tasks are executed, thereby facilitating dependency management."
5. "Gradle was designed for multi-project builds, which can grow to be large." -> "Gradle was specifically designed for managing multi-project builds, which can scale to large sizes."
6. "Incremental builds are supported by determining the parts of the build tree that are already up to date; any task dependent only on those parts does not need to be re-executed." -> "Incremental builds are supported by identifying the portions of the build tree that are already up to date; tasks dependent solely on those portions do not require re-execution."
7. "Combined with the proprietary hosted service of Develocity, it produces web-based build visualizations called Gradle Build Scans." -> "In conjunction with the proprietary hosted service of Develocity, Gradle generates web-based build visualizations known as Gradle Build Scans."

Overall, your introduction provides a solid foundation for understanding Gradle's functionality and significance. These suggested improvements aim to enhance clarity and readability.

Figure 6: Editing suggestions for entire text, showing incomplete text analysis.

SOURCES:

[HQ+23] M. U. Hadi, q. a. tashi q. a., R. Qureshi, A. Shah, a. muneer a., M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, S. Mirjalili. "A survey on large language models: applications, challenges, limitations, and practical usage".
DOI: 10.36227/techrxiv.23589741.v1

[Cia23] E. J. Ciaccio. "Use of artificial intelligence in scientific paper writing".
In: Informaticsin Medicine Unlocked 41 (2023), p. 101253.
ISSN: 2352-9148.
DOI: https://doi.org/10.1016/j.imu.2023.101253.
URL:https://www.sciencedirect.com/science/article/pii/S2352914823000953

[WTC22] T. Wu, M. Terry, C. J. Cai. "AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts". In: Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems. CHI '22., New Orleans, LA, USA, Association for Computing Machinery, 2022.
ISBN: 9781450391573.
DOI:10.1145/3491102.3517582.
URL: https://doi.org/10.1145/3491102.3517582

[Kum] M. Kumar. Understanding Tokens in ChatGPT. accesses on 29.04.2024.
URL: https://medium.com/@manav.kumar87/understanding-tokens-in-chatgpt-32845987858d

[Opea] OpenAI. Tokenizer - Learn about language model tokenization. Manual input of shown text, accesses on 29.04.2024.
URL: https://platform.openai.com/tokenizer

[Gradle24] Wikipedia. Gradle. 11.04.2024.
URL: https://en.wikipedia.org/wiki/Gradle

Declaration

I hereby declare that the work presented in this thesis is entirely my own and that I did not use any other sources and references than the listed ones. I have marked all direct or indirect statements from other sources contained therein as quotations. Neither this work nor significant parts of it were part of another examination procedure. I have not published this work in whole or in part before. The electronic copy is consistent with all submitted copies.

---

place, date, signature