



Enhancing Navigation in Large Language Models Through Qualitative Geographic Context

BACHELOR'S THESIS
in partial fulfilment of the requirements for the degree of
BACHELOR OF SCIENCE

University of Münster
Faculty of Geosciences
Institute for Geoinformatics

First assessor:

Prof. Dr. Reinhard Moratz

Second assessor:

Prof. Dr. Benjamin Risse

Submitted by:

Niklas Daute

Münster, December 2025

Enhancing Navigation in Large Language Models Through Qualitative Geographic Context

Verbesserung der Navigation in Large Language Models durch qualitativen geographischen Kontext

Abstract

Large Language Models (LLMs) are increasingly used for various daily tasks. While navigation is a common task, little research has been done on LLM navigation performance in real-world scenarios such as urban navigation. Additionally, there is no research on the effects of qualitative geographic context on LLM performance in this domain. In this thesis, we aim to provide insights into these areas by evaluating the navigation performance of three LLMs in an A/B testing setup across two geographic contexts. Our experiments include a retrieval-augmented (RAG-like) approach, which enables us to provide LLMs with qualitative geographic context in the form of symbolic dipole relations. Our results indicate that LLM performance in urban navigation tasks is severely limited. However, using our method we were able to produce a substantial performance increase of 62.5%, suggesting that improving LLM navigation performance via qualitative geographic context is possible. In addition to these results, we discuss the potential of navigation-specific LLM benchmarking as well as implications for practical applications and future academic research.

Acknowledgement

I want to express my sincere gratitude to Dr. Reinhard Moratz for supervising this thesis and providing invaluable feedback throughout the entire writing process, as well as sharing his deep expertise in qualitative spatial reasoning.

My gratitude extends to Prof. Dr. Benjamin Risse for agreeing to be the second reviewer of this thesis.

In addition, I want to thank James Odienki for his substantial contribution to the python implementation used in this project.

Lastly I want to thank my friends and family who have supported me during this journey.

Contents

1. Introduction	1
1.1. The Current State of Navigation using LLMs	1
1.2. Research Overview	4
1.3. Research Hypothesis	5
2. Background	7
2.1. Large Language Models	7
2.2. Emerging Abilities of LLMs	8
2.3. Context Enrichment for LLMs	9
2.4. Qualitative Geographic Data	10
2.5. Data sources and GIS Tools	12
2.6. LLM Benchmarks and Opportunities	14
2.7. Summary	16
3. Methods	17
3.1. Data Acquisition and Preparation	17
3.2. LLM Configuration	18
3.3. Experimental Design and Procedure	20
3.4. Evaluation Metrics	20
4. Results	23
4.1. Dataset Overview	23
4.2. Experiment Results	23
4.3. Summary of Findings	28
5. Discussion	31
5.1. Interpretation of Results	31
5.2. Limitations	35
6. Conclusion and Outlook	37

CONTENTS

A. Appendix	39
A.1. Hamburg Street Descriptions	39
A.2. Prompts	45
A.2.1. Control Group Prompt	45
A.2.2. Test Group Prompt	45
A.2.3. User Prompt	45
A.3. Example Navigation Tasks	45
List of Figures	47
List of Tables	49

1 | Introduction

In this chapter, we introduce the research topic of navigation using large language models. After briefly discussing the current behavior of LLMs in the evaluated setting, we discuss the existing body of research in this area. We then identify the research gap we aim to address in this thesis and formulate a corresponding research hypothesis.

1.1. The Current State of Navigation using LLMs

Today, users rely on large language models every day, with leading systems like ChatGPT processing over 2 billion user queries per day as of July 2025 (OpenAI, 2025). A recent journal article has highlighted the many unique use cases for LLMs, ranging from information retrieval to various other tasks such as drafting text or generating computer code (Chatterji et al., 2025). Given their widespread use, LLMs may already be used for navigational queries in practice. This motivates evaluating whether LLMs are currently capable of delivering accurate and reliable responses to user navigation queries. If performance can be shown to be insufficient, this motivates research exploring techniques to improve LLM navigation performance.

In the following example depicted in Figure 1.1, we prompted GPT-4o with a simple navigation task to observe whether the model could successfully provide a useful answer to a simple path finding problem. The prompt used to generate the response shown in Figure 1.1 was as follows:

I just arrived at Münster central station and need to get to the Hafenweg. Please provide me with step-by-step walking directions to get there.

Even though the model response shown in Figure 1.1 appears structurally sound and plausible at first glance, upon closer inspection there are several mistakes that would likely prevent the user from following the proposed route. To understand the mistakes, we analyze the response step by step, starting with the first instruction. To ensure followability, we provide Figure 1.2 showing one possible correct route as context.

1. *Exit Münster Hauptbahnhof through the main entrance onto Willy-Brandt-Allee.*

1. INTRODUCTION

Distance: ~2.5–3 km

Estimated walking time: 30–35 minutes

Directions:

1. Exit Münster Hauptbahnhof through the main entrance onto Willy-Brandt-Allee.
2. Turn right and walk southeast on Willy-Brandt-Allee toward the city center.
3. After about 500 meters, you'll reach Albersloher Weg. Turn left onto Albersloher Weg.
4. Continue along Albersloher Weg for about 1 km, passing the Aasee park area on your right.
5. When you reach the intersection with Hammer Straße, turn right.
6. Walk along Hammer Straße for around 700 meters. You'll pass a few residential and commercial buildings.
7. Look for Hafenweg on your left. Turn left onto Hafenweg.

You have arrived at **Hafenweg, Münster**.

Figure 1.1.: GPT-4o response to a simple route planning task.

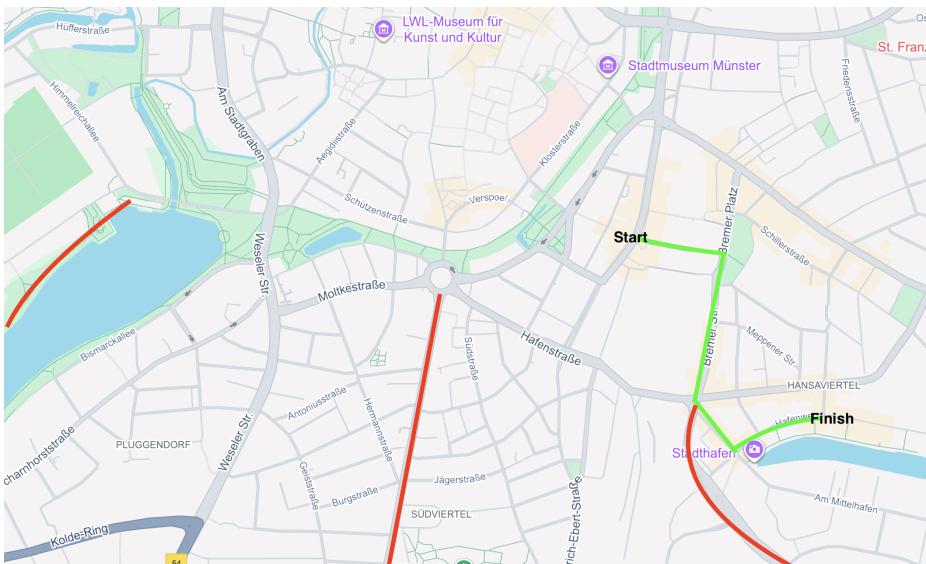


Figure 1.2.: One possible correct route (green) for the example navigation task compared to various segments found in the GPT-4o response (red).

Although exiting Münster Hauptbahnhof to reach the destination is certainly a sensible step, a more direct choice would be to use the secondary exit away from the city center and towards the actual destination. Also, the response introduces a street called “Willy-Brandt-Allee” which does not align with the map context shown in Figure 1.2. The second instruction is as follows:

2. Turn right and walk southeast on Willy-Brandt-Allee toward the city center.

In the example, the destination “Hafenweg” lies generally southeast of the starting point. The direction “toward the city center” is also inconsistent with the intended destination. The next instruction reads:

3. After about 500 meters, you'll reach Albersloher Weg. Turn left onto Albersloher Weg.

Next, the LLM response introduces the street “Albersloher Weg”. However, the map shown in Figure 1.2 suggests no visible connection between “Willy-Brandt-Allee” and “Albersloher Weg”. Continuing with the next instruction:

4. Continue along Albersloher Weg for about 1 km, passing the Aasee park area on your right.

This instruction raises many questions. For once, the destination does not lie one kilometer away from the Central Station, therefore the stated distance appears inconsistent with the approximate start-destination distance in this example. Further, the response introduces the “Aasee park area”, which is inconsistent with the route context shown in Figure 1.2. Additionally, the map shown in Figure 1.2 shows no visible connection between the two segments. With the next instruction, the mistakes made by GPT-4o continue:

5. When you reach the intersection with Hammer Straße, turn right.

The response then refers to “Hammer Straße”. Again, this instruction does not appear viable in the example situation, as Figure 1.2 does not show a visible connection between the two required streets. Thus, it doesn’t appear that this instruction could be followed in practice. Finally, the last two instructions read:

6. Walk along Hammer Straße for around 700 meters. You'll pass a few residential and commercial buildings.

7. Look for Hafenweg on your left. Turn left onto Hafenweg.

Again, these instructions do not appear viable, as the map context shown in Figure 1.2 does not show a visible connection between the two streets. Whether there are “residential and commercial buildings” along the Hammer Straße is irrelevant, although it may as well be true. In total, GPT-4o suggested several steps that may be difficult to follow in practice when compared against the map context shown in Figure 1.2.

Preliminary, informal tests discussed during thesis supervision suggested similar shortcomings in contemporary LLM navigation in urban environments. In line with these preliminary observations, the routing task shown above illustrates that such issues may occur in recent LLMs such as GPT-4o. In the remainder of this thesis, we will evaluate this question systematically across multiple models and geographic areas.

These observations motivate a systematic evaluation of LLMs and their abilities in spatial reasoning, and whether their performance can be improved using qualitative geographic context. We thus

conduct a series of experiments to evaluate the navigation capabilities of select LLMs across multiple geographic areas, both with and without the inclusion of qualitative geographic context. In order to gain a clear picture of the current state of research in this area, we must first review the relevant literature.

1.2. Research Overview

Within the scope of our literature review for this thesis, we did not identify prior work explicitly evaluating the impact of qualitative geographic context on LLM navigation performance. Without identified prior research explicitly addressing this issue, the viability of LLMs as natural language interfaces for navigation tasks remains insufficiently characterized in the existing body of research. While previous research has addressed the general capabilities of LLMs in reasoning tasks, it cannot automatically be assumed that these capabilities translate to strong navigation performance in real world scenarios. Many potential uses for LLMs in GIScience applications have been theorized, among those the possibility to act as natural language interfaces for spatial queries such as navigation. Several benchmarks indicate that LLMs struggle with planning- and navigation-related tasks.

This research alone is however not sufficient to fully dismiss the idea of using LLMs for navigation. If research could identify techniques that substantially improve LLM navigation performance, the idea of their usage in this area could become practical in the future. Even though one study alone will not be sufficient to fully explore this area and answer all open questions, it could provide first evidence on the viability of such techniques and serve as a benchmark for future research.

Context enrichment is widely used to provide task-relevant external information to LLMs. Although this thesis does not test this explanation, one possible reason for poor navigation performance in LLMs could be such a blind spot in the training data, in this case regarding topological data suitable for navigation.

Any attempts to address the issue of poor LLM navigation performance using context enrichment must however answer the initial question of how to best provide geographic context in a way that is suitable for LLMs. Since LLMs are primarily trained on text corpora, we hypothesize that the geographic context should be present in the form of text as well. Many sources provide openly available spatial data online. Much of this data is however present in machine-readable formats, rather than in natural language.

A solution to address this related issue could be to utilize a framework for qualitatively describing topological data such as the dipole calculus. The dipole calculus is a framework that has been discussed in applications such as robotics. In the framework, purely qualitative relations are defined

to describe topological relations in an elegant manner. These relations could be translated into natural language statements, suitable for LLM context enrichment.

In this study we therefore aim to show first empirical evidence on whether this technique of improving LLM navigation using qualitative geographic context could prove viable. As mentioned, this research will address a gap in the existing literature, but will not be sufficient to fully explore the area. The testing framework we establish can also be utilized by future researchers studying additional ways of improving LLM navigation performance. In the end, this study will serve as a first step on the long-term path towards LLM usage in navigation tasks.

1.3. Research Hypothesis

Drawing from the current state of research, we propose that context enrichment using qualitative geographic relations may have a positive effect on LLM navigation performance. Context enrichment has been shown to make LLM responses less prone to hallucinations in other domains, while dipole relations provide a proven framework for qualitatively describing topological data. Combined, these techniques may aid the LLM in generating more accurate navigation responses. We thus formulate the following research hypothesis (\mathcal{H}_1):

The inclusion of qualitative geographic context in the form of natural language dipole relations substantially improves the navigation performance of large language models.

The alternative hypothesis can thus be formulated as follows (\mathcal{H}_0):

The inclusion of qualitative geographic context in the form of natural language dipole relations does not substantially improve the navigation performance of large language models.

A substantial improvement in this context is defined as a practically meaningful increase in success rate (in percentage points) relative to the control condition. In the following chapters, we will describe the methods used to test our hypothesis and present the results of our experiments. The thesis will then conclude with a discussion of the findings and their meaning for future research in this area.

2 | Background

This chapter provides the theoretical and technical background for LLM-based navigation in urban contexts. We outline core LLM concepts and summarize prior work on reasoning and planning-related tasks. In addition, we review qualitative representations of geographic data and context enrichment techniques for LLMs. Finally, we discuss whether these techniques have been applied in prior work on LLM navigation tasks.

2.1. Large Language Models

Large language models (LLMs) are the model class evaluated in this thesis. The history of LLMs is extensive, with early work on language modeling dating back to the 1950s (Minaee et al., 2025), and therefore out of scope for this thesis. Accordingly, this section provides a high-level overview of LLMs and summarizes their relevant characteristics to this study.

Large language models are a recent class of language models in general (Minaee et al., 2025). In practice, a common training objective for LLMs is next token prediction (Minaee et al., 2025). For example, given the input tokens (tokens are typically words or subword units) “Roses are”, a large language model may generate a plausible next token such as “red”. Adding the predicted token to the input sequence, the model can then repeat this process iteratively. This process is known as autoregressive generation and is also used in other domains such as computer vision (Xiong et al., 2025). In the running example, the next token after “Roses are red” could be “violets”. After further iterations this could eventually result in a full sequence such as “Roses are red, violets are blue”.

Training models to perform high-quality next-token prediction is non-trivial. In fact, a key limitation of pre-transformer architectures such as recurrent neural networks (RNNs) is that they can struggle to model long-range dependencies in long sequences. This difficulty has been cited as motivation to improve sequence modeling on long time horizons (Johnston et al., 2025).

Modern LLMs mitigate these limitations in part through the transformer architecture. The transformer architecture uses a mechanism called self-attention to model relationships between input

tokens (Vaswani et al., 2017). The models evaluated in this thesis are based on the transformer architecture.

Another key factor enabling large language models is the availability of large-scale training corpora. High-performing LLMs are typically trained on large amounts of data. Training corpora often include sources such as books and web pages. Although exact figures on training dataset sizes are difficult to obtain for current models, earlier models were trained on datasets containing billions of tokens (Minaee et al., 2025).

In the pre-training process, the model parameters are optimized to minimize a training objective. This means that the model, often consisting of billions of parameters, is adjusted iteratively. In each iteration, the training objective is evaluated and the model parameters are updated, until predefined stopping criteria are eventually met. In addition to pre-training, the model can also be fine-tuned to match a specific task or setting (Minaee et al., 2025).

Many additional details regarding LLM architectures and training processes are out of the scope of this thesis. The preceding overview provides the necessary concepts for the remainder of this thesis. Building on this overview, this thesis evaluates whether LLMs can be used to generate usable navigational instructions.

GPT-3 released by OpenAI in 2020 is often cited as an early example of a large language model (Minaee et al., 2025).

2.2. Emerging Abilities of LLMs

Beyond language understanding and generation, LLMs may exhibit additional abilities. The term “emerging abilities” is used to refer to LLM abilities that may arise with increased model scale (Wei et al., 2022). This means that these abilities are not just scaled up versions of existing capabilities of smaller models. Examples include in-context learning and multi-step problem solving (often referred to as “reasoning”) (Wei et al., 2022). With in-context learning, LLMs are able to learn new tasks from examples provided in the prompt, without any updates to the model parameters. The term “reasoning” is commonly used to describe multi-step problem solving (Wei et al., 2022).

These emerging abilities motivate evaluating LLMs on navigation tasks. Solving navigational tasks may require text generation of route descriptions as well as maintaining spatial context. If emerging abilities transfer to navigation tasks, LLMs may be able to solve them without task-specific training.

2.3. Context Enrichment for LLMs

LLMs can produce incorrect or misleading outputs, a phenomenon often referred to as “hallucination” (Huang et al., 2025). Hallucinated outputs may appear plausible at first, while being factually incorrect upon closer inspection (Huang et al., 2025). The navigation example given in the introductory chapter illustrates this phenomenon: a seemingly plausible set of route instructions may contain errors when validated against a map. This plausibility can distinguish hallucinations from nonsensical outputs. To reduce hallucinations, techniques have been developed to provide LLMs with additional information at inference time (Minaee et al., 2025).

In practice, this is often done by enriching the LLMs’ context. LLMs have a finite context window. Typically, this context window consists of the prompt and the current chat history. With context enrichment techniques, however, additional information can be included in the context window (Minaee et al., 2025).

Retrieval-augmented generation (RAG) is one such technique (Lewis et al., 2021). With RAG, an additional external knowledge base is connected to the LLM system. This external knowledge base contains a collection of information, for example company documents or other domain-specific knowledge. Given a user prompt, the system retrieves relevant information from this knowledge base using a retrieval query. The retrieval query is derived from the user prompt. Relevant snippets of the external knowledge base are then added to the LLMs’ context window. The LLM then generates an output conditioned on the prompt and the retrieved knowledge as an additional source (Lewis et al., 2021).

The effects of RAG on LLM hallucinations have been studied in previous research. In knowledge-intensive tasks such as question answering, RAG has been shown to reduce hallucinations (Gao et al., 2024). These findings motivate research exploring RAG techniques for navigation tasks. If navigation tasks require access to knowledge beyond the LLMs training data, RAG techniques could potentially improve their performance.

RAG has been implemented in several ways. One specific RAG implementation relevant to this thesis is graph-based RAG (Edge et al., 2025). This approach uses a graph structure rather than a document collection as the external knowledge base. Such graphs typically represent data as nodes and edges. Similar to traditional RAG, the user prompt is used to query the graph structure for relevant nodes and edges. These relevant nodes and edges are then converted to text snippets and appended to the LLM’s context window. In some cases, the graph structure may first have to be constructed from text, while in other cases a graph may be present in the first place (Edge et al., 2025). Recent work has applied graph-based RAG to LLM-based navigation (Moratz, Daute, et al., 2025).

Examples of graph structures include knowledge graphs, social networks or road networks. This suggests that a graph-based RAG system could be used for navigation tasks. However, this thesis does not implement a full graph RAG system. Instead, the technique is emulated by providing additional geographic context within the prompt.

2.4. Qualitative Geographic Data

LLMs can generate fluent text (Minaee et al., 2025) and emerging abilities have been proposed for more complex tasks (Wei et al., 2022). Additionally, context enrichment techniques such as RAG, have been reported to reduce hallucinated responses (Gao et al., 2024).

This thesis evaluates these ideas in the context of navigation tasks by providing qualitative geographic data as additional context. This motivates the question of how geographic data can be represented adequately for LLM consumption.

Geographic data can be represented in many ways. A key distinction can be made between qualitative and quantitative geographic data. While quantitative geographic data contains precise measurements such as distances or coordinates, qualitative geographic data describes properties and relationships of geographic entities (Lwin et al., 2012).

Aspect	Qualitative representation	Quantitative representation
Description	“Examplestreet” starts at the intersection with “Teststreet”. “Examplestreet” ends at the intersection with “Endstreet”.	Name: Examplestreet Length: 300 m Width: 6 m Starting coordinates: (1,1) End coordinates: (6,7)

Table 2.1.: Example of qualitative and quantitative representations of a street.

In the example given in Table 2.1, we illustrate the difference between qualitative and quantitative geographic data in the context of our research topic. While the quantitative representation focuses on precise numbers such as “Length: 300 m”, the qualitative representation describes the same road in terms of qualitative relations. This example does not imply that the two representations are interchangeable, contain the same information or are equally useful. Rather, they reflect different paradigms for representing geographic data. This motivates evaluating whether qualitative geographic data is suited for LLM context enrichment.

So far, only a small example of qualitative geographic data has been introduced. However, this thesis aims to represent entire street networks qualitatively. This requires a suitable framework

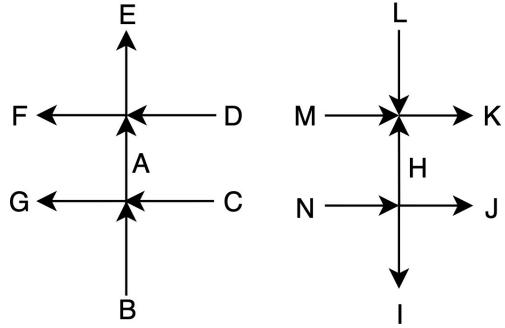


Figure 2.1.: All qualitatively different dipole relations that are necessary to describe a street network.

for qualitative representation of street network topology. The next section introduces one such framework, the dipole calculus (Moratz, Lücke, et al., 2011).

The dipole calculus is a qualitative spatial reasoning framework. In the framework, elongated, oriented objects (such as road segments) are represented as directed line segments with a start and end point, called dipoles (Moratz, Lücke, et al., 2011). Pairing two of these dipoles together results in a relation that can be classified into one of several relational categories. The set of these relational categories is finite. Examples include relations where one dipole continues another, crosses it, or branches off from it. The calculus abstracts from metric information such as distances and instead focuses on qualitative relations. This yields a framework allowing for qualitative reasoning on line-based geographic structures such as road networks (Moratz, Lücke, et al., 2011). Dipole relations have recently been used as context for LLM-based navigation (Moratz, Daute, et al., 2025).

Although extended versions of the dipole calculus exist, Figure 2.1 shows the set of dipole relations used in this thesis. In Figure 2.1, each dipole is represented by a unidirectional arrow. Each dipole has a unique label (A-N) for identification. For example, dipole G branches off from dipole B. Similarly, dipole A continues from dipole B, and dipole E continues from dipole A. In practice, a street can be represented by one or multiple dipoles under this representation. A single street may correspond to multiple dipoles depending on the segmentation of its geometry.

2.5. Data sources and GIS Tools

This thesis requires a data source for street network data which can be transformed into dipole relations. One such data source is OpenStreetMap (OSM). OpenStreetMap was created in 2004 and is a community driven, collaborative project providing free geographic data (Haklay et al., 2008). OpenStreetMap relies on Volunteered Geographic Information (VGI) provided by volunteers. This form of data collection is sometimes discussed under the term “citizen sensor” (Antoniou et al., 2017). Although this approach has benefits like removing corporate interests from the data, it also has downsides.

Since the data is provided by volunteers, the data quality may vary depending on the region of interest. Some studies have investigated the data quality of OSM. A 2013 review synthesized prior studies on OSM data quality. To summarize, the study found that OSM data was rapidly growing. While the overall data quality was found to be heterogeneous, meaning that it varied from region to region, the data quality was assessed to have improved over time (Sehra et al., 2013). In an article published in 2014, researchers specifically compared the OSM road data available for Germany to proprietary datasets and found a 9% difference in total network length, suggesting relatively high completeness in the investigated region (Neis et al., 2014). The same article also notes that many studies mention an urban bias in OSM data, meaning that more densely populated areas tend to have better data coverage than rural areas (Neis et al., 2014). While this may be a limitation for other use cases, for our study this may not be a major issue, since we are investigating navigation tasks in urban scenarios, where coverage is often reported to be better.

Even though we cannot ignore the fact that OSM data quality may vary, prior work suggests that it can be used for routing applications: in a conference paper from 2011, researchers investigated the use of OSM data for real time routing. The researchers successfully demonstrated that OSM data was a suitable choice for both server side and handheld routing implementations (Luxen et al., 2011). This supports the choice of OSM as the underlying data source for this thesis.

Many studies analyzing or utilizing OSM data have suffered from poor reproducibility and small sample sizes. To address these issues OSMnx was introduced in 2017. OSMnx provides an open source Python library, which allows users to easily download and visualize street networks from OpenStreetMap (Boeing, 2017). In this study, we will make use of OSMnx to obtain the street networks we require for our experiments.

Further, manual verification of the generated routes requires a reliable mapping tool. Since the validation in this study will be done manually, we need a tool with a user-friendly interface and reliable data coverage. To reduce dependence on a single data source, verification is done using a

different map provider. Instead we will use a widely used commercial mapping platform called Google Maps:

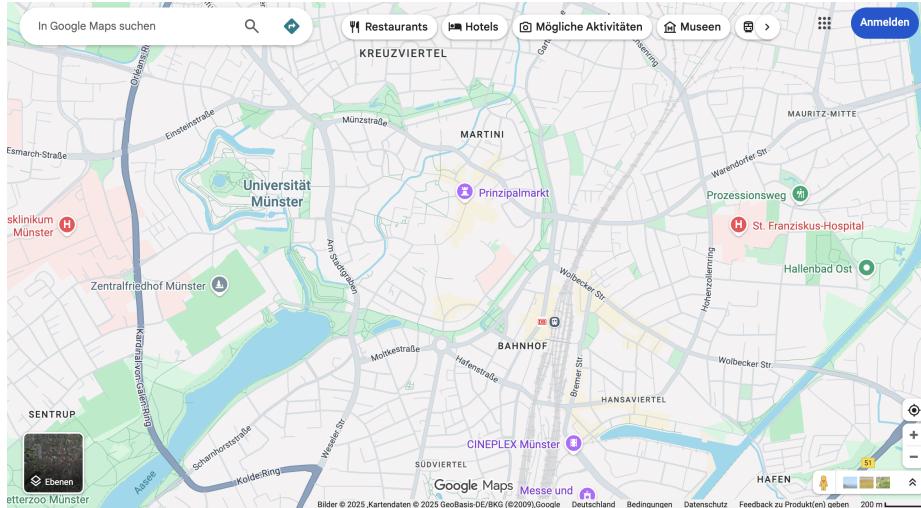


Figure 2.2.: A screenshot demonstrates the Google Maps web interface, map data © Google.

Google Maps is a web based mapping platform originating from the Australian company Where 2 Technologies, which was fully acquired by Google in 2004 (*Google Maps* 2025). In a recent report from 2020, Google, the company behind Google Maps, claims that the platform counts over 1 billion monthly active users (*A look back at 15 years of mapping the world* 2020). Additionally, a survey conducted on 511 smartphone users in 2018 found that 67% of respondents used Google Maps as their primary navigation app (*The Popularity of Google Maps: Trends in Navigation Apps in 2018* 2025).

In 2019 Google claimed that the data they use for Google Maps was sourced from over 1000 third party services globally (*Google Maps 101* 2019). Although many other competing commercial products such as Apple Maps, released in 2012 (*Apple Previews iOS 6 With All New Maps, Siri Features, Facebook Integration, Shared Photo Streams & New Passbook App* 2025), exist in the modern app landscape, Google Maps, backed by its long history and widespread popularity, will be the tool of choice for the validation process in this study.

By discussing LLMs, their emerging capabilities such as reasoning, context enrichment techniques, a framework for qualitative geographic representations and our choice of data sources, we have now covered most relevant concepts for this thesis. In the next section we will review existing research on LLM benchmarks and whether they cover navigation tasks.

2.6. LLM Benchmarks and Opportunities

LLM benchmarking spans a wide range of tasks and approaches. In this thesis, we focus on navigation-focused evaluation of LLMs. This motivates the question of whether prior LLM benchmarks have covered navigation specifically.

Some of the historical benchmarks for language models include GLUE (Wang, Singh, et al., 2019) and SuperGLUE (Wang, Pruksachatkun, et al., 2020). In GLUE, a model is evaluated on a set of nine different NLU (natural language understanding) tasks such as sentiment analysis or textual entailment. Sentiment analysis describes the task of classifying a given text by its sentiment. Textual entailment on the other hand describes whether one text (a hypothesis) is supported by another (a premise). For each of the nine tasks, a model produces predictions, and the individual task scores are aggregated into a final GLUE score. The GLUE score could then be used as a single metric to compare different models (Wang, Singh, et al., 2019).

A study conducted in 2019 showed that non-expert human annotators outperformed models on six of nine GLUE tasks, achieving an average score of 87.1 compared to then state-of-the-art (and fine-tuned) models scoring 83.9 (Nangia et al., 2019). As model performance improved, SuperGLUE was introduced in 2020 to provide more challenging tasks (Wang, Pruksachatkun, et al., 2020).

While GLUE and SuperGLUE provided standardized evaluation benchmarks, they do not cover many broader LLM abilities addressed in later benchmarks. To address this gap, new benchmark suites were proposed, including BIG-bench (Beyond the Imitation Game) introduced in 2022. The BIG-bench paper describes a benchmark of over 200 tasks, contributed by 450 researchers across 132 institutions. The included tasks span a diverse range of topics such as mathematics, physics and software development, shifting evaluation from NLU toward more general problems. A human baseline score was established, and in addition model scale was taken into account during performance evaluation (Srivastava et al., 2023).

A further shift occurred with the introduction of HELM (Holistic Evaluation of Language Models) in 2022. Language models were increasingly used as the basis for real-world applications, and this called for additional evaluation dimensions. Helm introduced a framework to evaluate model responses not only on their accuracy, but also on aspects such as fairness, robustness and efficiency. Fairness describes whether model responses differ systematically between certain groups. With robustness, a measure was introduced to evaluate whether a model's performance degrades when faced with adversarial inputs. The computational resources required to run a model were evaluated using the efficiency metric (Liang et al., 2023).

Beyond benchmark suites, evaluation practices have also shifted toward reusable infrastructure. An example of this is OpenAI's Eval framework. The framework is an open source registry of

existing evaluation suites, as well as tools to create new evaluations (*openai/evals* 2025).

In addition to benchmark suites and evaluation frameworks, crowd-sourced model evaluation methods have also emerged. The web-based platform LMArena (formerly Chatbot Arena) enables users to compare two models' responses side-by-side in anonymous pairwise "battles". These human preference evaluations are then aggregated into a leaderboard (Chiang et al., 2024).

Together, these developments show that the field of LLM evaluation has evolved to account for different factors: While benchmarks such as GLUE and SuperGLUE were used to benchmark models' NLU capabilities (Wang, Singh, et al., 2019; Wang, Pruksachatkun, et al., 2020), benchmark suites such as the BIG-bench aimed to evaluate the broader capabilities of language models (Srivastava et al., 2023). Since language models increasingly found their way into real-world applications, evaluation frameworks like HELM introduced additional metrics (Liang et al., 2023). Frameworks such as OpenAI's Evals allowed for reusable evaluation infrastructure (*openai/evals* 2025). Finally, to incorporate human preference signals into model evaluation, crowd-sourced platforms like LMArena have emerged (Chiang et al., 2024).

However, these benchmarks and evaluation frameworks do not specifically cover navigation tasks.

In a paper published in 2024, researchers introduced a benchmark called MANGO that evaluates mapping and navigation in maze environments. In their trials the LLMs were provided with textual walkthroughs covering the maze environments, and subsequently asked to answer navigation questions about these mazes. The authors report that models including GPT-4 struggled with the navigation tasks, while humans achieved high accuracy on the problems (Ding et al., 2024).

A paper from 2022 illustrated the shortcomings of LLMs in planning tasks. Several prominent LLMs were evaluated on a suite of action and change reasoning tasks. Across these planning tasks, the models achieved very low success rates. The models struggled even more to produce optimal results (Valmeekam et al., 2022). Since navigation can be framed as a planning problem, these results could indicate that LLMs could struggle with navigation as well, although the study does not specifically cover navigation.

In 2025, researchers investigating recent reasoning-variants of LLMs identified a sharp decline in performance as problem complexity increased. They demonstrated these effects in controllable puzzle environments, where both reasoning and non-reasoning models ultimately collapsed to near-zero accuracy. A counterintuitive discovery was made as well: As problem complexity increased, the models initially increased their thinking effort. However, after a certain point, reasoning traces became shorter again, even though token budget was still available (Shojaee et al., 2025). This suggests that performance on complex navigation tasks could also be limited, even though the study does not specifically cover navigation.

2.7. Summary

We began this chapter by introducing large language models and their emerging abilities. Next, we discussed context enrichment techniques such as retrieval augmented generation and one specific implementation called graph RAG. Afterwards, we presented how geographic data can be represented qualitatively using the dipole calculus. Thereafter, we discussed our choice of data sources and GIS tools.

Finally, we reviewed existing research on LLM benchmarks. Prior benchmarks on LLMs show that they struggle with tasks requiring planning (Valmeekam et al., 2022). Some research has also identified that even more recent reasoning variants struggle with complex problems (Shojaee et al., 2025). While navigation tasks have been studied in maze environments (Ding et al., 2024), real-world street-network navigation remains comparatively unexplored in existing LLM benchmarks. This motivates an opportunity to create a benchmark for LLM navigation tasks using real-world problems and to evaluate whether qualitative geographic context offers substantial performance improvements. Having this background knowledge firmly in place, we continue by presenting our experimental methods in the next chapter.

3 | Methods

In this chapter, we introduce the methods used in this thesis. Our research hypothesis states that the navigation capabilities of large language models may be substantially improved by the inclusion of qualitative geographic context. To test our hypothesis, we designed an experimental setup involving large language models, qualitative geographic context and navigation tasks. Our approach can be broken down into the data used, the configuration of the LLMs - including the context enrichment setup -, and ultimately the procedures used to conduct the experiments. To evaluate the results, we define a strict correctness criterion and determine success rates for the control and test groups.

3.1. Data Acquisition and Preparation

The first step in our experimental setup is the generation of dipole relations describing topological relationships in street networks which can then be used as qualitative geographic context for our LLMs. While high quality geographic datasets on street networks are available at no additional cost from various sources such as OpenStreetMap, these datasets typically do not include qualitative geographic descriptions. The most common way of encountering street network data is in the form of graphml files which describe the street network as a graph of nodes and edges:

As a basis for our experiments, we downloaded two of these graphs from the OpenStreetMap database using OSMnx, a powerful Python library capable of handling and processing OpenStreetMap data. In order to ensure a reasonable degree of geographic variation, we selected two different cities as sites for our datasets: Hamburg and Münster. This way, we end up with one dataset per city, each containing a varying number of nodes and edges. To extract qualitative descriptions from these datasets, an algorithm capable of generating dipole relations in natural language describing the topology of a given graph was used. In essence, the algorithm works by iterating over all nodes and creating dipole relations for each intersection. While the full algorithm is provided in the appendix, a conceptual description of the algorithm in pseudocode is given here:

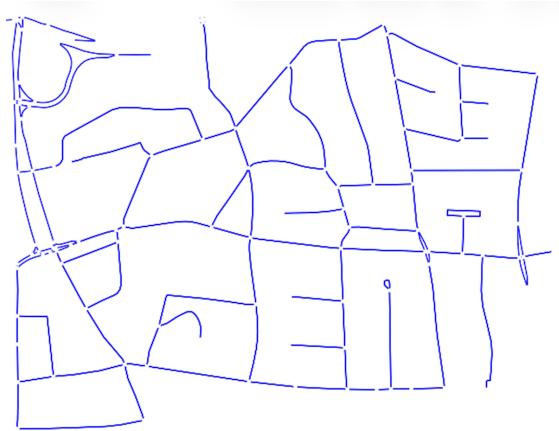


Figure 3.1.: Graph representation of area of interest in Hamburg, Germany as used in this study.

Following this method, we were able to generate a set of dipole relations for each of our initial street network datasets. These relations were stored in the form of simple .txt files, with one relation per line. An example of these relations is given in the appendix. To demonstrate the relations here, a small excerpt may be helpful:

Excerpt of qualitative relations (Hamburg)

```
Borchlingweg begins at the intersection with Ansorgestraße, Halbmondsweg.  
Stindeweg then branches off to the right.  
Langmaackweg then branches off to the right.
```

The python software used to generate the qualitative dipole relations was primarily implemented by James Odienki, and the code is available upon request. After discussing the data acquisition process, we now focus on a detailed description of the configuration of the LLMs used in this study.

3.2. LLM Configuration

While the most common way for users to interact with large language models is to use their proprietary web interfaces, we identified this approach to be impractical for our needs; namely to test multiple configurations of LLMs in a systematic manner. While most LLM providers also offer APIs to interact with their models, this approach would have introduced additional programming overhead for this project.

One additional solution to access a wide range of current or legacy LLMs through a unified interface is to use a platform like OpenRouter, which is the platform we have chosen to use for this study.

Algorithm 1 Generate Street Descriptions from OSM

Require: Bounding box
Ensure: Text descriptions for streets in bounding box

- 1: Download road network for bounding box
- 2: Remove motorways and isolated nodes
- 3: **for** each street name in network **do**
- 4: Collect all road segments with this name
- 5: Keep largest connected component
- 6: Find main end-to-end path
- 7: Orient path consistently (S→N or W→E)
- 8: **for** each junction along path **do**
- 9: Inspect connected streets
- 10: Use bearings to classify as left/right/crossing
- 11: Generate natural-language statement
- 12: **end for**
- 13: **end for**
- 14: Export all descriptions to file

Through its user-friendly web interface it is possible to query multiple LLMs concurrently without any additional programming effort. It is important to note however, that even though OpenRouter makes it possible to query multiple LLMs at the same time, this does not mean that the queried models share a common context window or share any other information between each other.

New models arrive on the market frequently, making it difficult to test “the latest” or “state of the art” models. After all, a model with significantly greater capabilities in various areas could be just around the corner at any given time. Nevertheless, we have selected three models which we consider to be widely used and capable of handling navigation tasks due to their advertised abilities of performing multi-step reasoning tasks. Our selection of models for this study therefore consists of OpenAI’s GPT-4o, Google’s Gemini 2.5 Pro as well as Anthropic’s Claude Sonnet 4.5.

The symbolic dipole relations generated by the process described in the previous section were stored in a simple .txt file. For the navigation tasks, this file was provided to the models using OpenRouter’s file upload (attachment) feature, allowing the models to reference the contained relations while generating a route. To ensure the model responses were free of any web-search result contamination, we switched off any web-search tools available and used a simple system prompt telling the models to use their own inherent knowledge to solve the given tasks only. The exact system prompts used in our experiments are provided in the appendix. In the trials conducted with additional qualitative geographic context, the system prompt was extended to instruct the models to use the provided dipole relations when figuring out a route. Other than that, no further changes were made to the model’s default configurations, making the setup easy to replicate and true to real-world usage scenarios.

3.3. Experimental Design and Procedure

Our experimental design follows a traditional A/B testing approach, where we compare the performance of a Control Group against the performance of a Test Group. In our case, the Control Group consists of LLMs performing navigation tasks on their own, without any additional context. Respectively, the Test Group consists of LLMs performing the same navigation tasks, but with access to our dipole relations. While the Test Group performs the exact same tasks as the Control Group, it was ensured that no memories of previous trials were available to the models in the Test Group sessions by starting each individual trial of both groups in a fresh session, disconnected from any previously conducted trials.

To reduce the risk of bias towards certain areas or street types, we designed navigation tasks which consist of two street names (start street and end street), both derived randomly from the available street names. The first step was to extract a list of unique street names from both datasets (Hamburg and Münster). These lists were then sampled uniformly at random 20 times with replacement each to generate start streets for the navigation tasks. To make sure that we receive 20 unique start streets per dataset, any duplicates were removed and replaced by new random samples. This process was repeated to generate the end streets for the navigation tasks. In case any start and end street pairs were identical, these pairs were also replaced by new random samples. In a final check, we ensured that all generated navigation tasks were unique within each dataset. In total, this results in 40 unique navigation tasks (20 per dataset). A subset of example tasks from the Hamburg dataset is provided in the appendix. In the following, we refer to a task as a single Start-to-End street navigation problem, while a trial refers to a single model response to a specific task. Each navigation task was evaluated once per model and context condition. This results in a total of 240 trials (3 models \times 2 context conditions \times 40 navigation tasks).

After trial execution, each model’s response was saved in a database. In the case of an exceedingly verbose model response, the response was truncated to contain only the relevant navigational instructions which we intend to investigate in this study.

3.4. Evaluation Metrics

All responses were then reviewed manually by comparing the generated instructions against Google Maps. When looking at any response, the objective was to determine whether the produced steps could be executed in sequence to solve the navigation task. If this was the case, the response was labelled as correct, otherwise it was labelled as incorrect. Since the responses were labelled manually by a single evaluator, they may contain some subjective judgement in a few borderline cases.

This means that just one incorrect instruction in any LLM response, control or test group, would consequently render the entire response incorrect. A trial was labelled as incorrect if the response contained at least one instruction that (i) referenced a non-existing street, (ii) required an impossible turn, or (iii) produced a disconnected route. Our validation does not take into account any further measures like distance measures or number of turns. While these additional measures could provide interesting insights, they are outside the scope of this study. The success rate is therefore defined as the proportion of navigation trials which were labelled as correct over the total number of navigation trials conducted.

$$\text{Success Rate} = \frac{N_{\text{correct}}}{N_{\text{total}}}$$

After labelling all model responses, we were thus able to calculate success rates for the control and test groups.

This concludes our description of the methods used to conduct this study. In the following chapter, we will present the results obtained from the experiments.

4 | Results

This chapter serves to present the results of our experiments. In total, 240 individual LLM responses were collected across various configurations to find out whether the inclusion of qualitative geographic context helps improve the navigation performance of LLMs. We are first going to provide an overview of the results obtained from our experiments, before breaking down the results by different factors such as test city or large language model tested. These results will then be summarized and serve as the basis for the discussion following in the Chapter 5.

4.1. Dataset Overview

City	# Individual Streets	Avg. Task Distance (m)
Hamburg	38	899
Münster	128	1272

Table 4.1.: Overview of test datasets.

Table 4.1 provides an overview of the two datasets used in our experiments. We were able to successfully derive symbolic dipole relations for both tested cities. For the Hamburg dataset, this resulted in a total of 38 individual streets and an average start-end distance of 899 meters. On the other hand, the Münster dataset contained a total of 128 individual streets, and also had a higher average start-end distance of 1272 meters (both measured using Google Maps).

4.2. Experiment Results

Group	# Trials	# Successful	# Failed	Success Rate (%)
Control Group	120	0	120	0%
Test Group	120	75	45	62.5%
Total	240	75	165	31.25%

Table 4.2.: Overview of trial outcomes in control and test groups with success and failure counts and corresponding success rates.

4. RESULTS

All trials could be conducted successfully by the methods described in Chapter 3. This means that we got a valid LLM response for each navigation task under all test conditions. However, this does not imply that all responses were correct in terms of navigation success, but simply that we were able to collect all the necessary data. In total, 240 LLM navigation responses (trials) were collected and manually labeled as either correct or incorrect. Table 4.2 provides an overview of the trial outcomes.

There is a visible difference in navigation success rates between the control and test groups. In each group, the exact same 120 navigation trials were conducted. Out of these 120 trials, no LLM response in the control group could be labelled as successful. Consequently, all trials in this group were labelled as failures, leading to a success rate of 0%. However, in the test group, 75 out of 120 trials could be labelled as successful. With 45 remaining failures, this results in a success rate of 62.5%. Across both groups, this means that 75 out of 240 navigation trials were labelled as successful, while 165 trials were labelled as failed. This leads to an overall success rate of 31.25% across all experiments conducted in this thesis. To summarize, the inclusion of qualitative geographic context increased the success rate by 62.5 percentage points when compared to the control group.

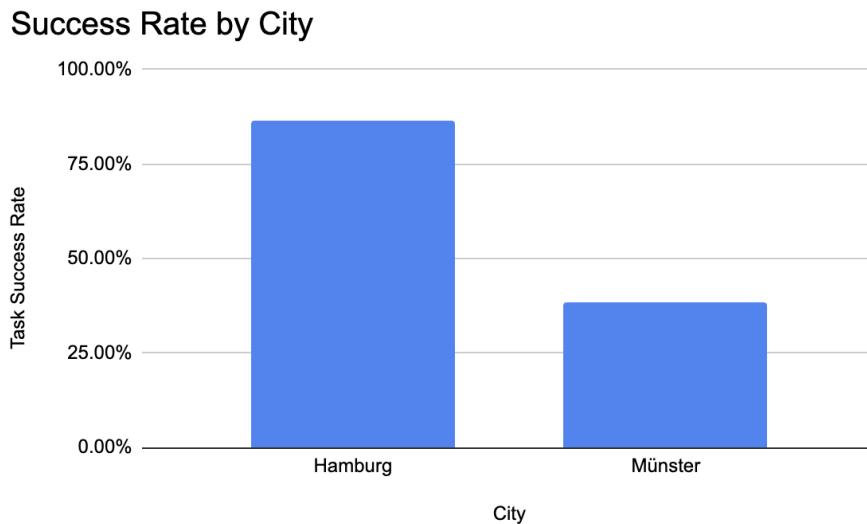


Figure 4.1.: Test Group success rate by City.

Figure 4.1 illustrates the success rates in the test group broken down by the two tested cities. When comparing test group performances under this condition, we can make several observations. First, the success rate increased in both cities compared to the control group (which had a success rate of 0% in both cities). In Hamburg, the success rate climbed to 86.6%, while in Münster it reached just 38.3%. This means that the success rate in Hamburg was more than double that of Münster. Out of the 75 correctly labelled navigation trials in the test group, 52 were conducted using the Hamburg

dataset, while 23 were conducted using the Münster dataset.

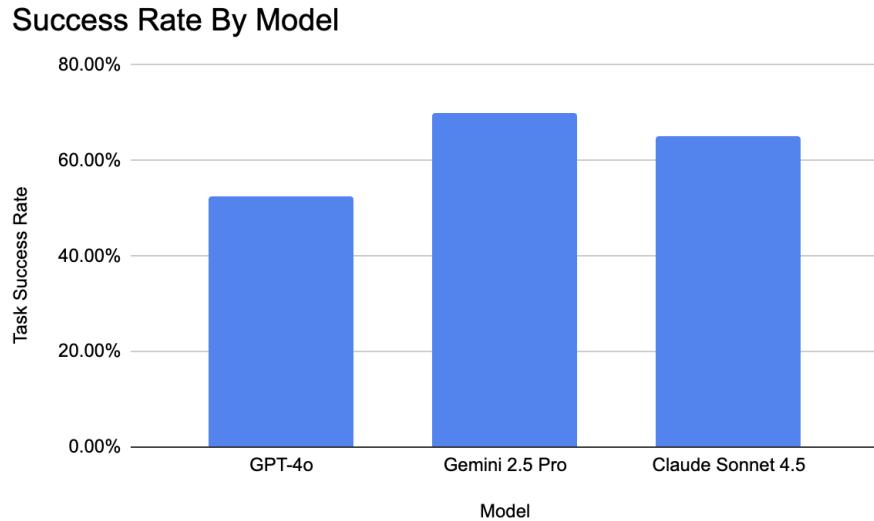


Figure 4.2.: Test group success rate by LLM.

Further, comparing the success rates across the three tested LLMs in the test group as depicted in Figure 4.2 yields additional insights. While all models performed better compared to their control group success rate of 0%, the best-performing model in the test group was Google’s Gemini 2.5 Pro. It achieved a success rate of 70%, with 28 out of 40 navigation trials labelled as correct. The next best-performing model was Claude Sonnet 4.5 with a slightly lower success rate of 65%. Claude answered 26 out of 40 navigation trials correctly. GPT-4o performed worst in the test group: with 21 out of 40 navigation trials labelled as correct, it achieved a success rate of 52.5%.

Breaking down the test group success rates by both test city and LLM as shown in Figure 4.3 leads to several additional observations. First, we can see that all models performed better with the Hamburg dataset compared to the Münster dataset. All models using the Hamburg dataset achieved a success rate of at least 75%, with Gemini 2.5 Pro ranking the highest at 95%. Claude Sonnet 4.5 followed with a success rate of 90%, five percentage points below Gemini 2.5 Pro. GPT-4o achieved a success rate of 75%, 20 percentage points lower than Gemini 2.5 Pro. With the Münster dataset, all models performed worse overall, although their relative rankings remained the same. Gemini 2.5 Pro again ranked highest among the three with a success rate of 45%, followed by Claude Sonnet 4.5 at 40% and GPT-4o at 30%. The lowest overall success rate in the test group was thus observed with GPT-4o using the Münster dataset at 30%, while the highest overall success rate was achieved by Gemini 2.5 Pro using the Hamburg dataset at 95%.

We may also examine the frequency distribution of correctly answered navigation tasks in the test group. Table 4.3 summarizes, for each navigation task, how many of the three tested models

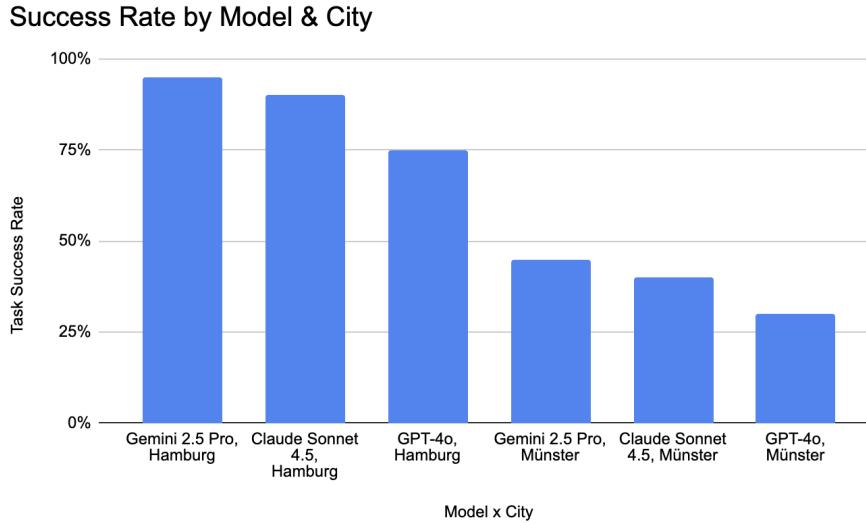


Figure 4.3.: Test group success rate by model and city.

Number of Correct LLM Responses	Frequency
0	6
1	9
2	9
3	16

Table 4.3.: Frequency Distribution of Correct LLM Responses

produced a response labelled as correct. Only six navigation tasks were answered incorrectly by all three models. Nine navigation tasks were answered correctly by just one model. Nine navigation tasks were answered correctly by two models. This means that the likelihood of all models failing given that at least one model failed was exactly 25% (the number of cases where all models failed (6) divided by the number of cases where at least one model failed (24)). The highest frequency could be observed for navigation tasks where all three models answered correctly, with a total of 16. Conversely, this means that the likelihood of all models answering correctly when at least one model answered correctly was approximately 47.06% (The number of cases where all models answered correctly (16) divided by the number of cases where at least one model answered correctly (34)).

Model	Frequency
GPT-4o	3
Gemini 2.5 Pro	4
Claude Sonnet 4.5	2

Table 4.4.: Distribution of Correct Models in Single-Success Tasks.

Additionally, we analyzed which model performed best in cases where exactly one model provided a response to a navigation task that was labelled as correct. The results of this analysis are summarized in Table 4.4. Although there were only nine such cases overall and the spread between the best- and worst-performing model was small, Gemini 2.5 Pro performed best in this category. It was able to provide a correct answer in four cases when GPT-4o and Claude Sonnet 4.5 both failed. GPT-4o then followed with three such cases. Claude Sonnet 4.5 performed worst, answering two navigation tasks correctly when both other models failed.

Model Failed	Frequency
GPT-4o	7
Gemini 2.5 Pro	1
Claude Sonnet 4.5	1

Table 4.5.: Distribution of Single Failed Model in Two-Success Trials

We also consider the complementary case: for tasks where exactly one model failed, we aim to identify which model was most likely to fail. The results of this analysis are summarized in Table 4.5. GPT-4o accounted for seven out of nine such cases, making it the model most likely to fail when both other models succeeded. Gemini 2.5 Pro and Claude Sonnet 4.5 accounted for exactly one failure of this kind each, so their performance in this category was identical.

Number of Failed Models	Hamburg	Münster
1	4	5
2	2	7
3	0	6
Total	6	18

Table 4.6.: Distribution of Failures by City and Number of Failed Models

Next, we analyzed the failures by the dataset they occurred on (Hamburg or Münster). Cases in which only one model answered incorrectly were distributed almost evenly across both city datasets: using the Hamburg dataset, there were 4 occurrences where one model failed to answer a navigation task. Using the Münster dataset, this number was just slightly higher at 5. For tasks in which exactly two models failed, we counted two such cases in Hamburg and seven cases in Münster.

The largest disparity between both cities could be observed in cases where all three models failed: No navigation task conducted on the Hamburg dataset resulted in all three models failing to provide a correct answer. Using the Münster dataset however, there were six such occurrences.

Aggregating across categories, there are 24 total cases where at least one model failed to answer a navigation task correctly. Out of these, 6 cases, or 25% occurred using the Hamburg dataset, while the remaining 18 cases, or 75% occurred using the Münster dataset.

4. RESULTS

In total, there are 15 total cases where at least two models failed to answer a navigation task correctly. Out of these, just 2 cases, or 13.33% occurred using the Hamburg dataset, while the remaining 13 cases, or 86.67% occurred using the Münster dataset.

Lastly, all of the six instances where all three models failed to answer a navigation task correctly occurred using the Münster dataset.

Model Pair	Disagreement Frequency (Out of 18 Trials)	Agreement Frequency
GPT-4o vs. Gemini 2.5 Pro	15	3
GPT-4o vs. Claude Sonnet 4.5	13	5
Gemini 2.5 Pro vs. Claude Sonnet 4.5	8	10

Table 4.7.: Pairwise Model Disagreement and Agreement in Non-Unanimous Trials

Finally, we analyze pairwise disagreement among the three studied models. To do so, we exclude all tasks which were answered unanimously, meaning that either all models answered correctly or all models answered incorrectly. This leaves us with a total of 18 non-unanimous tasks.

Within this subset of 18 tasks, we measure how often each pair of models disagreed. Disagreement is defined here as one model answering correctly while the other model answered incorrectly (or vice versa). The results of this analysis are summarized in Table 4.8.

Out of the 18 non-unanimous tasks, GPT-4o and Gemini 2.5 Pro disagreed in 15 cases, while agreeing on 3 cases. The next highest disagreement count was observed in the pairing GPT-4o and Claude Sonnet 4.5, which disagreed in 13 cases and agreed in 5 cases. With just 8 disagreements and 10 agreements, the pairing of Gemini 2.5 Pro and Claude Sonnet 4.5 showed the lowest amount of disagreements out of all combinations.

Accordingly, we conclude that Gemini 2.5 Pro and Claude Sonnet 4.5 agreed more often than they disagreed. On the other hand, any combination involving GPT-4o resulted in more disagreements than agreements.

4.3. Summary of Findings

Overall, our experiments indicate that LLM navigation performance with additional symbolic geographic context improved compared to the control group without context in our test setup. While the control group achieved no successful trials, the control group achieved non-zero success rates across all tested LLMs and datasets. The magnitude of performance gains varied by the two city datasets: success rates in Hamburg were higher for all models compared to Münster. Among

4.3. Summary of Findings

the tested LLMs, Gemini 2.5 Pro achieved the highest success rate, followed by Claude Sonnet 4.5 and GPT-4o. These results will serve as the basis for the discussion following in Chapter 5.

5 | Discussion

Our study aimed to investigate the impact of qualitative geographic context on the navigation performance of LLMs. We hypothesized that our approach would improve the ability of LLMs to provide accurate navigation instructions. In order to test our hypothesis, we conducted a series of experiments by means of the methods described in Chapter 3. The results of these experiments presented in Chapter 4 serve as the basis for the following discussion.

5.1. Interpretation of Results

Our experiments suggest that the inclusion of qualitative geographic context had a positive effect on LLM navigation performance in the tested setup. The overall success rate increased from 0/120 (0%) in the control group to 75/120 (62.5%) in the test group (see Table 4.2). This is a substantial improvement in task success rate, supporting our initial research hypothesis within the scope of our experiments. However, since the evaluation in this thesis does not include formal statistical significance testing, these results should be interpreted as empirical evidence rather than a general claim about all possible cities, models, or navigation scenarios.

The success rate of 0% in the control group is striking, and raises multiple questions. First: was the experimental setup appropriate to test LLM navigation performance in real world scenarios? Since the navigation tasks were designed by combining two real world streets in each city with each other to form a routing task, the setup realistically reflects potential real world user scenarios. Additionally, the sole use of LLMs for navigation without any additional tools may be questioned. However, since a goal of this study was to evaluate LLM navigation performance in the first place, this setup was necessary to isolate the LLMs' capabilities from any external sources or information. Further, our correctness criteria could be considered too strict by some. This does not align with typical navigation use cases however, where a single wrong instruction can lead to failed navigation, regardless of the system used. Under this evaluation framework, the 0% success rate indicates that the tested LLMs were not reliable for navigation in the tested areas.

Further, as depicted in Table 4.1, the performance increase across the two tested cities was not equal. In Hamburg, the success rate in the test group was 86.6%, while in Münster it was only

38.3%. Several factors could explain this discrepancy. One possible explanation is the size difference between the two tested datasets. The Hamburg dataset contained just 38 individual streets, while the Münster dataset contained 128 individual streets. Perhaps the smaller dataset was easier for the LLM to process in its context window, leading to better performance. Additionally, the navigation tasks in Münster spanned a larger distance (1272m) compared to Hamburg (899m), as measured using Google Maps. Thus, another possible explanation is that the LLMs struggled to find effective solutions for these longer routes. Since the control group performance was equally poor in both cities, it seems unlikely that any intrinsic model bias towards one of the cities played a role in the observed results. However, such biases cannot be entirely ruled out without further experiments. In total, it may be possible that multiple factors contributed to the observed performance difference between the two cities.

We may also consider the performance differences between the three tested LLMs shown in Figure 4.2 and discuss whether it made sense to test multiple models, and whether the choice of models was appropriate. The results indicate that while all three models benefited from the provided geographic context, their performance varied in the tested setup. This suggests that navigation performance using our approach may depend on the specific LLM used, justifying our decision to test multiple models. While the exact reasons for the performance differences remain speculative, differences in model architecture, training process, or fine tuning could have played a role. The best performing model in our experiments was Gemini 2.5 Pro. This does not imply however, that it is the best model using our approach in general: other models not evaluated in this study may perform better or worse than our tested models. However, the substantial increase in success rates across all three models indicates that our approach has the potential to improve navigation performance in a variety of current LLMs. The development of new models is also rapid and ongoing: areas of poor performance today may be addressed by a new model tomorrow. This rapid model progress may reduce the marginal benefit of context enrichment as presented in this study over time, motivating repeated evaluation as new models are released.

In Chapter 2 we discussed several LLM related benchmarks. This raises the question whether our framework could be integrated or expand on existing benchmarks. As we have seen, all models in the control group had a success rate of 0%. Therefore, it seems unlikely that our framework would provide interesting insights when purely testing LLM performance without any additional techniques to boost navigation performance. For any techniques aiming to improve LLM navigation however, our framework could serve as a navigation-specific task suite to compare performance improvements.

We also discussed the suitability of OpenStreetMap data for our approach in Chapter 2. In this thesis, OSM data was used to extract street networks for Hamburg and Münster. Whether variance in OSM data quality could influence our results was not explicitly tested in this study and remains

speculative. Issues in OSM data quality are unlikely to be the main cause for all test group mistakes however, since the provided context was sufficient to improve results across all tested models and cities. Future research could explore different data sources in order to better understand the influence of data sources on LLM based navigation.

In addition, we may consider the suitability of the dipole relations presented in Chapter 2 for our approach. In this study, these relations were used to represent geographic context in a qualitative manner. The results indicate that this representation was effective: While the previously discussed variations in performance across cities and models cannot be ignored, the overall improvements show that dipole relations can be a useful tool for LLM navigation tasks. Whether alternative frameworks for spatial representation could yield comparable or even better results remains an open question for future research.

Our results indicate that poor LLM navigation performance is not necessarily inevitable in the tested setup. By providing qualitative geographic context, navigation performance improved substantially in our experiments. This indicates that LLM navigation performance can be improved with additional techniques. In practice, this does not imply that established navigation solutions are obsolete. Based on the results of this study, no claim of LLM based navigation entirely replacing established navigation solutions can be made. The many mistakes made by the LLMs in the control group as well as the mistakes still present in the test group highlight the current limitations of LLM-based navigation and justify alternative solutions.

Mistakes in navigation can lead to serious consequences in time, safety and cost in real world scenarios. Although certainly encouraging, a success rate of 62.5% is therefore not sufficient for many use cases, while established navigation systems typically achieve high reliability in everyday use. If the gap could be closed further in the future, LLMs could someday seriously be considered as alternatives to current navigation solutions.

So far, we have only considered topological information when providing geographic context to LLMs. In practice, there are countless other types of geographic information that could be supplied for various use cases as well. Future work could explore the inclusion of additional types of geographic context: accessibility information, data on points of interest or data on real-time dynamic conditions could potentially be valuable avenues for future implementations aiming to differentiate themselves from traditional navigation solutions and making use of the unique capabilities of LLMs. These additional data types were however not a part of this study, and their potential benefits remain unclear from the results presented here.

Our results can also be interpreted in another way: If it takes additional, non-trivial work to achieve better navigation performance with LLMs, maybe they are currently not the best tool for the job after all. With our context enrichment setup, we were able to achieve higher trial success rates in

our test group compared to the low success rate of the control group. At present, implementing a comparable pipeline in user navigation applications could prove to be non-trivial however. It is difficult to imagine users, especially those without a technical background, recreating our setup on mobile devices such as smartphones or smartwatches for example, which are so commonly used for navigation in this era of mobile computing. Following this interpretation, LLM systems may benefit from further calibration, and explicit caveats about their limitations when answering users' navigation queries could be an appropriate measure. This study does not concern itself with mechanisms like these however. Many mistakes of various nature were made by the tested LLMs, even though they are widely used state-of-the-art models today. And since these models are used so frequently today, it is not implausible that some users have already tried to navigate using them.

It would be interesting to know whether users would be aware of the potentially poor results. In the best case, users would quickly realize the mistakes before following them and move to other existing navigation solutions to get accurate navigation results. Even then however, this could lead to undermined trust in LLMs in general, which is certainly not to be desired by their providers. In adverse cases, users could face delays or safety-relevant situations if they were to follow incorrect navigation instructions provided by LLMs.

Research on LLM-based navigation could also benefit from improvements in validation techniques. Manual validation as done in this study does not scale well and is not suitable for larger datasets or the coverage of many models. Eventually, manual validation may become unrealistic as it becomes too time-consuming and labor-intensive. Automated validation methods could help address this issue, and also help the reproducibility of results.

In this study, we demonstrated one possible way to improve LLM navigation performance. Although the results were promising, this doesn't mean that other methods could not achieve similar or even better results than ours. We encourage researchers to explore the development of other alternative methods. The testing framework described in Chapter 3 was designed to be easily reproducible and could serve as a basis for future research.

Beyond navigation, our approach could be used in similar scenarios involving graph-like structures. One interesting avenue for additional research could thus be to study whether knowledge graphs could be represented using qualitative relations as well, and whether this could allow LLMs to reason about knowledge graphs in a similar way as we have shown for geographic networks.

This study should thus be interpreted as a proof of concept: LLM navigation performance can be improved by providing qualitative geographic context. This does not suggest however that LLMs are ready to replace traditional navigation solutions yet, but that does not mean that the idea of LLM-based navigation can be fully dismissed either.

5.2. Limitations

While our methods have shown promising results, there are limitations. First, our study considered only two distinct areas of interest within just two European cities. Although we observed improvements in both cities, it is possible that other cities may perform worse using our approach. Data quality can also differ when considering different regions. While OpenStreetMap coverage was sufficient for our selected cities, this may not be the case in other cities or regions. Thus, generalization to other cities or regions remains an open question.

Second, we only evaluated three LLMs in this study. Today, many different LLMs are available, and it is likely that many more will be released in the future. While we were able to observe improvements across all three tested models, this does not guarantee that performance improvements will occur with any LLM. Results may also vary with future architectures, augmentation techniques, or model versions.

6 | Conclusion and Outlook

In this study, we have explored the effects of qualitative geographic context on LLM navigation performance. Our research hypothesis stated that giving language models access to qualitative geographic context would improve their navigation capabilities. To test this hypothesis, we designed a series of experiments comparing LLM performance on navigation tasks in two independent conditions: one with qualitative geographic context and one without. We utilized randomly generated navigation tasks in our two chosen test areas (Münster and Hamburg) and let both conditions generate answers to each task. The answers were then evaluated manually based on a correctness criterion.

The results of our experiments show that LLMs perform poorly in navigation tasks without qualitative geographic context, answering 0 out of 120 tasks correctly, corresponding to a success rate of 0%. In contrast, with additional geographic context, the models achieved a substantially higher performance, answering 75 out of 120 tasks correctly, corresponding to a success rate of 62.5%. These findings support our initial research hypothesis: All three tested LLMs (GPT-4o, Gemini 2.5 Pro, Claude Sonnet 4.5) demonstrated a higher success rate on navigation tasks when provided with qualitative geographic context.

Our setup involved just two specific urban areas from the cities of Münster and Hamburg. In addition, out of many available LLMs, we only tested three models. In future research, these limitations could be addressed by including more variety both in the tested geographic areas and in the selected LLMs. Further, our evaluation was based on human labeling of a large number of generated answers. This time-consuming process could be improved in future studies by developing methods for automated evaluation of navigation task answers. Thus, while our approach showed promising first results, these limitations can hopefully be addressed in future research. Overall, this study establishes a foundation for context-enriched LLM navigation and motivates continued research in this domain.

A | Appendix

A.1. Hamburg Street Descriptions

Agathe-Lasch-Weg

Agathe-Lasch-Weg begins at the intersection with Holmbrook, Paul-Ehrlich-Straße.

Emkendorfstraße then branches off to the right.

Roosens Weg then branches off to the right.

Halbmondsweg then branches off to the left.

Klein Flottbeker Weg then branches off to the right.

Reventlowstraße then branches off to the right.

Ansorgestraße

Ansorgestraße begins at the intersection with Emkendorfstraße, Liebermannstraße.

Roosens Weg then branches off to the right.

Behringstraße

Behringstraße begins at the intersection with [‘Walderseestraße’, ‘Behringstraße’].

Bernadottestraße

Bernadottestraße begins at the intersection with Droysenstraße.

Halbmondsweg then branches off to the right.

Halbmondsweg then branches off to the right.

Poppes Weg then branches off to the right.

Roosens Weg then branches off to the left.

Schlagbaumtwiete then branches off to the right.

Elbblöcken then branches off to the right.

Borchlingweg

Borchlingweg begins at the intersection with Ansorgestraße, Halbmondsweg.

Stindeweg then branches off to the right.

Langmaackweg then branches off to the right.

Corinthstraße

Corinthstraße begins at the intersection with Bernadottestraße.

Elbblöcken then branches off to the right.

Elbchaussee then branches off to the right.

Droysenstraße

Droysenstraße begins at the intersection with Jungmannstraße.

Preußerstraße then branches off to the left.

Elbblöcken

Elbblöcken begins at the intersection with Bernadottestraße.

Leipoldstieg then branches off to the right.

Elbchaussee

Elbchaussee begins at the intersection with Schlagbaumtwiete.

Corinthstraße then branches off to the left.

Emkendorfstraße

Emkendorfstraße begins at the intersection with Agathe-Lasch-Weg.

Röpers Weide then branches off to the left.

Lavaterweg then branches off to the left.

Reventlowstraße then branches off to the right.

Ernst-August-Straße

Ernst-August-Straße begins at the intersection with Liebermannstraße.

Roosens Weg then branches off to the left.

Halbmondsweg

Halbmondsweg begins at the intersection with Bernadottestraße.

Bernadottestraße then branches off to the left.

Ansorgestraße then branches off to the left.

Borchlingweg then branches off to the right.

Agathe-Lasch-Weg then branches off to the right.

Klein Flottbeker Weg then branches off to the left.

Reventlowstraße then branches off to the right.

Hammerichstraße

Hammerichstraße begins at the intersection with Jungmannstraße.

Walderseestraße then branches off to the left.

Holbrook

Holbrook begins at the intersection with Agathe-Lasch-Weg, Paul-Ehrlich-Straße.

Jungmannstraße

Jungmannstraße begins at the intersection with Reventlowstraße.

Preußerstraße then branches off to the left.

Droysenstraße then branches off to the left.

Klein Flottbeker Weg

Klein Flottbeker Weg begins at the intersection with Agathe-Lasch-Weg, Halbmondsweg, Reventlowstraße.

Zickzackweg then branches off to the left.

Droysenstraße then branches off to the left.

Langmaackweg

Langmaackweg begins at the intersection with Borchlingweg.

Lavaterweg

Lavaterweg begins at the intersection with Emkendorfstraße.

Leipoldstieg

Leipoldstieg begins at the intersection with Elbblöcken.

Liebermannstraße

Liebermannstraße begins at the intersection with Ansorgestraße, Emkendorfstraße.

Ernst-August-Straße then branches off to the right.

Bernadottestraße then branches off to the right.

Lobsienweg

Lobsienweg begins at the intersection with Droysenstraße.

Meistertwiete

Meistertwiete begins at the intersection with Bernadottestraße.

Olshausenstraße

Olshausenstraße begins at the intersection with Reventlowstraße.

Reventlowstraße then branches off to the right.

Walderseestraße then branches off to the left.

Paul-Ehrlich-Straße

Paul-Ehrlich-Straße begins at the intersection with Agathe-Lasch-Weg, Holmbrook.

Poppes Weg

Poppes Weg begins at the intersection with Bernadottestraße.

Ansorgestraße then branches off to the left.

Preußerstraße

Preußerstraße begins at the intersection with Jungmannstraße.

Droysenstraße then branches off to the right.

Reventlowstraße

Reventlowstraße begins at the intersection with Olshausenstraße.

Walderseestraße then branches off to the left.

Roosens Park

Roosens Park begins at the intersection with Ansorgestraße.

Roosens Weg

Roosens Weg begins at the intersection with Ansorgestraße.

Agathe-Lasch-Weg then branches off to the left.

Reventlowstraße then branches off to the right.

Röpers Weide

Röpers Weide begins at the intersection with Walderseestraße.

Emkendorfstraße then branches off to the right.

Schlagbaumtwiete

Schlagbaumtwiete begins at the intersection with Elbchaussee.

Bernadottestraße then branches off to the right.

Slevogtstieg

Slevogtstieg begins at the intersection with Elbblöcken.

Stindeweg

Stindeweg begins at the intersection with Borchlingweg.

Taxusweg

Taxusweg begins at the intersection with Halbmondsweg.

Walderseestraße

Walderseestraße begins at the intersection with Reventlowstraße.

Reventlowstraße then branches off to the right.

Zickzackweg

Zickzackweg begins at the intersection with Klein Flottbeker Weg.

Zypressenweg

Zypressenweg begins at the intersection with Halbmondsweg.

[’Walderseestraße’, ’Behringstraße’]

[’Walderseestraße’, ’Behringstraße’] begins at the intersection with Behringstraße.

Walderseestraße then branches off to the right.

Ovelgönner Hohlweg

Ovelgönner Hohlweg begins at the intersection with Elbchaussee, Halbmondsweg.

A.2. Prompts

A.2.1. Control Group Prompt

You are an expert navigator for [city]. Your sole task is to generate walking routes.
Use your internal knowledge to derive the route.

A.2.2. Test Group Prompt

You are an expert navigator for [city]. Your sole task is to generate walking routes.
Base your answer on the geographic relations provided.

A.2.3. User Prompt

Generate a walking route from the start point [start] to the end point [end].

A.3. Example Navigation Tasks

Start	End
Stindeweg	Slevogtstieg
Lobsienweg	Schlagbaumtwiete
Behringstraße	Röpers Weide
Övelgönner Hohlweg	Taxusweg
Corinthstraße	Olshausenstraße

Table A.1.: Example navigation tasks (start and end streets).

List of Figures

1.1.	GPT-4o response to a simple route planning task.	2
1.2.	One possible correct route (green) for the example navigation task compared to various segments found in the GPT-4o response (red).	2
2.1.	All qualitatively different dipole relations that are necessary to describe a street network.	11
2.2.	A screenshot demonstrates the Google Maps web interface, map data © Google. .	13
3.1.	Graph representation of area of interest in Hamburg, Germany as used in this study. .	18
4.1.	Test Group success rate by City.	24
4.2.	Test group success rate by LLM.	25
4.3.	Test group success rate by model and city.	26

List of Tables

2.1.	Example of qualitative and quantitative representations of a street.	10
4.1.	Overview of test datasets.	23
4.2.	Overview of trial outcomes in control and test groups with success and failure counts and corresponding success rates.	23
4.3.	Frequency Distribution of Correct LLM Responses	26
4.4.	Distribution of Correct Models in Single-Success Tasks.	26
4.5.	Distribution of Single Failed Model in Two-Success Trials	27
4.6.	Distribution of Failures by City and Number of Failed Models	27
4.7.	Pairwise Model Disagreement and Agreement in Non-Unanimous Trials	28
A.1.	Example navigation tasks (start and end streets).	45

Bibliography

- A look back at 15 years of mapping the world* (Feb. 2020). en-us. URL: <https://blog.google/products/maps/look-back-15-years-mapping-world/> (visited on 12/11/2025) (cit. on p. 13).
- Antoniou, VYRON and Geographic Directorate, PAPAGOU Camp, GR (Sept. 2017). *Mapping and the Citizen Sensor*. en. Ed. by University of Nottingham, GB et al. Ubiquity Press. ISBN: 978-1-911529-16-3. DOI: 10.5334/bbf. URL: <https://www.ubiquitypress.com/site/books/10.5334/bbf/> (visited on 12/10/2025) (cit. on p. 12).
- Apple Previews iOS 6 With All New Maps, Siri Features, Facebook Integration, Shared Photo Streams & New Passbook App* (2025). en-US. URL: <https://www.apple.com/newsroom/2012/06/11Apple-Previews-iOS-6-With-All-New-Maps-Siri-Features-Facebook-Integration-Shared-Photo-Streams-New-Passbook-App/> (visited on 12/11/2025) (cit. on p. 13).
- Boeing, Geoff (Sept. 2017). “OSMnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks”. en. In: *Computers, Environment and Urban Systems* 65, pp. 126–139. ISSN: 01989715. DOI: 10.1016/j.comenvurbsys.2017.05.004. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0198971516303970> (visited on 12/10/2025) (cit. on p. 12).
- Chatterji, Aaron et al. (Sept. 2025). “How People Use ChatGPT”. en. In: (cit. on p. 1).
- Chiang, Wei-Lin et al. (Mar. 2024). *Chatbot Arena: An Open Platform for Evaluating LLMs by Human Preference*. arXiv:2403.04132 [cs]. DOI: 10.48550/arXiv.2403.04132. URL: <http://arxiv.org/abs/2403.04132> (visited on 12/12/2025) (cit. on p. 15).
- Ding, Peng et al. (Mar. 2024). *MANGO: A Benchmark for Evaluating Mapping and Navigation Abilities of Large Language Models*. arXiv:2403.19913 [cs] version: 1. DOI: 10.48550/arXiv.2403.19913. URL: <http://arxiv.org/abs/2403.19913> (visited on 12/12/2025) (cit. on pp. 15, 16).
- Edge, Darren et al. (Feb. 2025). *From Local to Global: A Graph RAG Approach to Query-Focused Summarization*. arXiv:2404.16130 [cs]. DOI: 10.48550/arXiv.2404.16130. URL: <http://arxiv.org/abs/2404.16130> (visited on 12/10/2025) (cit. on p. 9).
- Gao, Yunfan et al. (Mar. 2024). *Retrieval-Augmented Generation for Large Language Models: A Survey*. arXiv:2312.10997 [cs]. DOI: 10.48550/arXiv.2312.10997. URL: <http://arxiv.org/abs/2312.10997> (visited on 12/10/2025) (cit. on pp. 9, 10).
- Google Maps* (Dec. 2025). en. Page Version ID: 1325235575. URL: https://en.wikipedia.org/w/index.php?title=Google_Maps&oldid=1325235575 (visited on 12/11/2025) (cit. on p. 13).

- Google Maps 101 (July 2019). *Google Maps 101: how we map the world*. en-us. URL: <https://blog.google/products/maps/google-maps-101-how-we-map-world/> (visited on 12/11/2025) (cit. on p. 13).
- Haklay, M. and P. Weber (Oct. 2008). “OpenStreetMap: User-Generated Street Maps”. en. In: *IEEE Pervasive Computing* 7.4, pp. 12–18. ISSN: 1536-1268. DOI: 10.1109/MPRV.2008.80. URL: <http://ieeexplore.ieee.org/document/4653466/> (visited on 12/10/2025) (cit. on p. 12).
- Huang, Lei et al. (Mar. 2025). “A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions”. In: *ACM Transactions on Information Systems* 43.2. arXiv:2311.05232 [cs], pp. 1–55. ISSN: 1046-8188, 1558-2868. DOI: 10.1145/3703155. URL: <http://arxiv.org/abs/2311.05232> (visited on 12/10/2025) (cit. on p. 9).
- Johnston, Liam et al. (Mar. 2025). “Revisiting the problem of learning long-term dependencies in recurrent neural networks”. en. In: *Neural Networks* 183, p. 106887. ISSN: 08936080. DOI: 10.1016/j.neunet.2024.106887. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0893608024008165> (visited on 12/09/2025) (cit. on p. 7).
- Lewis, Patrick et al. (Apr. 2021). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. arXiv:2005.11401 [cs]. DOI: 10.48550/arXiv.2005.11401. URL: <http://arxiv.org/abs/2005.11401> (visited on 12/10/2025) (cit. on p. 9).
- Liang, Percy et al. (Oct. 2023). *Holistic Evaluation of Language Models*. arXiv:2211.09110 [cs]. DOI: 10.48550/arXiv.2211.09110. URL: <http://arxiv.org/abs/2211.09110> (visited on 12/12/2025) (cit. on pp. 14, 15).
- Luxen, Dennis and Christian Vetter (Nov. 2011). “Real-time routing with OpenStreetMap data”. en. In: *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*. Chicago Illinois: ACM, pp. 513–516. ISBN: 978-1-4503-1031-4. DOI: 10.1145/2093973.2094062. URL: <https://dl.acm.org/doi/10.1145/2093973.2094062> (visited on 12/10/2025) (cit. on p. 12).
- Lwin, Ko Ko, Yuji Murayama, and Chiaki Mizutani (2012). “Quantitative versus Qualitative Geospatial Data in Spatial Modelling and Decision Making”. en. In: *Journal of Geographic Information System* 04.03, pp. 237–241. ISSN: 2151-1950, 2151-1969. DOI: 10.4236/jgis.2012.43028. URL: <http://www.scirp.org/journal/doi.aspx?DOI=10.4236/jgis.2012.43028> (visited on 12/10/2025) (cit. on p. 10).
- Minaee, Shervin et al. (Mar. 2025). *Large Language Models: A Survey*. arXiv:2402.06196 [cs]. DOI: 10.48550/arXiv.2402.06196. URL: <http://arxiv.org/abs/2402.06196> (visited on 12/09/2025) (cit. on pp. 7–10).
- Moratz, Reinhard, Niklas Daute, et al. (Dec. 2025). *Bilateral Spatial Reasoning about Street Networks: Graph-based RAG with Qualitative Spatial Representations*. arXiv:2512.15388 [cs]. DOI: 10.48550/arXiv.2512.15388. URL: <http://arxiv.org/abs/2512.15388> (visited on 12/18/2025) (cit. on pp. 9, 11).

- Moratz, Reinhard, Dominik Lücke, and Till Mossakowski (Oct. 2011). “A condensed semantics for qualitative spatial reasoning about oriented straight line segments”. en. In: *Artificial Intelligence* 175.16-17, pp. 2099–2127. issn: 00043702. doi: 10.1016/j.artint.2011.07.004. url: <https://linkinghub.elsevier.com/retrieve/pii/S0004370211000890> (visited on 11/07/2025) (cit. on p. 11).
- Nangia, Nikita and Samuel R. Bowman (June 2019). *Human vs. Muppet: A Conservative Estimate of Human Performance on the GLUE Benchmark*. arXiv:1905.10425 [cs]. doi: 10.48550/arXiv.1905.10425. url: <http://arxiv.org/abs/1905.10425> (visited on 12/12/2025) (cit. on p. 14).
- Neis, Pascal and Dennis Zielstra (Jan. 2014). “Recent Developments and Future Trends in Volunteered Geographic Information Research: The Case of OpenStreetMap”. en. In: *Future Internet* 6.1, pp. 76–106. issn: 1999-5903. doi: 10.3390/fi6010076. url: <https://www.mdpi.com/1999-5903/6/1/76> (visited on 12/10/2025) (cit. on p. 12).
- OpenAI (July 2025). *Unlocking Economic Opportunity: A First Look at ChatGPT-Powered Productivity*. Tech. rep. OpenAI (cit. on p. 1).
- openai/evals* (Dec. 2025). original-date: 2023-01-23T20:51:04Z. url: <https://github.com/openai/evals> (visited on 12/12/2025) (cit. on p. 15).
- Sehra, Sukhjit Singh, Jaiteg Singh, and Hardeep Singh Rai (Aug. 2013). “Assessment of OpenStreetMap Data - A Review”. In: *International Journal of Computer Applications* 76.16. arXiv:1309.6608 [cs], pp. 17–20. issn: 09758887. doi: 10.5120/13331-0888. url: <http://arxiv.org/abs/1309.6608> (visited on 12/10/2025) (cit. on p. 12).
- Shojaee, Parshin et al. (Nov. 2025). *The Illusion of Thinking: Understanding the Strengths and Limitations of Reasoning Models via the Lens of Problem Complexity*. arXiv:2506.06941 [cs]. doi: 10.48550/arXiv.2506.06941. url: <http://arxiv.org/abs/2506.06941> (visited on 12/12/2025) (cit. on pp. 15, 16).
- Srivastava, Aarohi et al. (June 2023). *Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models*. arXiv:2206.04615 [cs]. doi: 10.48550/arXiv.2206.04615. url: <http://arxiv.org/abs/2206.04615> (visited on 12/12/2025) (cit. on pp. 14, 15).
- The Popularity of Google Maps: Trends in Navigation Apps in 2018* (2025). url: <https://themanifest.com/app-development/trends-navigation-apps> (visited on 12/11/2025) (cit. on p. 13).
- Valmeeekam, Karthik et al. (Nov. 2022). “Large Language Models Still Can’t Plan (A Benchmark for LLMs on Planning and Reasoning about Change)”. en. In: url: <https://openreview.net/forum?id=wUU-7XTL5X0> (visited on 12/12/2025) (cit. on pp. 15, 16).
- Vaswani, Ashish et al. (2017). “Attention is All you Need”. en. In: (cit. on p. 8).
- Wang, Alex, Yada Pruksachatkun, et al. (Feb. 2020). *SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems*. arXiv:1905.00537 [cs]. doi: 10.48550/arXiv.1905.00537. url: <http://arxiv.org/abs/1905.00537> (visited on 12/12/2025) (cit. on pp. 14, 15).
- Wang, Alex, Amanpreet Singh, et al. (Feb. 2019). *GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding*. arXiv:1804.07461 [cs]. doi: 10.48550/arXiv.

BIBLIOGRAPHY

- 1804.07461. URL: <http://arxiv.org/abs/1804.07461> (visited on 12/12/2025) (cit. on pp. 14, 15).
- Wei, Jason et al. (Oct. 2022). *Emergent Abilities of Large Language Models*. en. arXiv:2206.07682 [cs]. DOI: 10.48550/arXiv.2206.07682. URL: <http://arxiv.org/abs/2206.07682> (visited on 12/09/2025) (cit. on pp. 8, 10).
- Xiong, Jing et al. (May 2025). *Autoregressive Models in Vision: A Survey*. arXiv:2411.05902 [cs]. DOI: 10.48550/arXiv.2411.05902. URL: <http://arxiv.org/abs/2411.05902> (visited on 12/09/2025) (cit. on p. 7).

Declaration of Academic Integrity

I hereby confirm that this thesis, entitled

Enhancing Navigation in Large Language Models Through Qualitative Geographic Context

is solely my own work and that I have used no sources or aids other than the ones stated. All passages in my thesis for which other sources, including electronic media and AI Tools, have been used, be it direct quotes or content references, have been acknowledged as such and the sources cited. I am aware that plagiarism is considered an act of deception which can result in sanction in accordance with the examination regulations.

Niklas Daute, Münster, December 18, 2025

I consent to having my thesis cross-checked with other texts to identify possible similarities and to having it stored in a database for this purpose.

I confirm that I have not submitted this thesis in part or whole as an examination paper before.

Niklas Daute, Münster, December 18, 2025