Universität Augsburg
Institut für Geographie

Developing a framework to rate bikeability in cities using open source datasets.

Master thesis

Guidance: Prof. Dr. Jukka Krisp

Wendel, Tim

Matrikelnumber: 1462542

Master Geoinformatics

mit.ende@mailbox.org

Submission date: 21.10.2022

Danksagung

Vielen Dank Julia. Ohne dich wäre das alle nix geworden. #Team Milsch.

Abstract

Over the last 25 years, the number of cyclists in Germany has increased by over 40% (Hudde, 2022). Yet, the tools to assess the quality of cycling infrastructure on a broad scale are lacking. Currently, no unified evaluation technique exists that can be used to compare complete cities regarding their cycling infrastructure. The following thesis develops a framework to address this issue. The proposed framework uses *open source* and *open data* to conform with the ideas of *open science*. Furthermore, it aims to be easy to configure and simple to use, while still allowing for the technical complexity to run fast and efficient. To achieve this, the model uses microservice software architecture style and containerization. To be able to assess the bikeability citywide as well as on an individual street level, the framework employs a semi-random algorithm to model the relevancy of individual roads. The generated origin-destination pairs are connected by routes produced with the cycling-specific routing engine, Brouter. Lastly, the framework comes preconfigured with a proof-of-concept model, based on existing bikeability research. By comparing the citywide results with other indices, as well as comparing the score of individual roads with local experts, the underlying idea of the framework could be validated.

Table of contents

Table of figures

Table of maps

Table of tables

Table of equations

1 Introduction

Even though the phenomenon of climate change has been known since the 1960s, people tend to ignore the severity of the problem. Especially in Germany, where the car industry is prevalent, the importance of alternate modes of transport is neglected. Of course, the electric car is pushed and often seen as a saviour for the climate. Yet, it cannot be ignored that the technology is still expensive and still too far away from being really climate-friendly. In addition to that, cars take up space in crowded cities, and traffic jams harm not only the environment but also people's nerves. Some companies try to overcome such problems with even more sophisticated technology and strongly invest in the future technologies such as air mobility. Instead of the time and effort being put into such inventions, we could just reinvent a known means of transport: the bicycle. Compared to the Germans' favourite method of transport, the car, bicycles offer numerous advantages that not only benefit the environment but also peoples' health. Bicycles are affordable, practical, and rather safe. Second-hand models can be bought for less than 100 euros, and a tiny space to park the bike can be found everywhere. The safety aspect is something that is often argued, but it must be said that what makes cycling unsafe is cars and the current car-centric infrastructure we have in Germany. However, this is a problem that can be solved quickly and without great investment when the right technology and frameworks are used.

The bicycle has been around since the early 19th century. The original invention by Karl von Drais has undergone an incredible number of changes and improvements. We should finally start to cherish this reliable invention and make it even safer and even more practicable than it already is. Open source and open data are ways to contribute to the goal of improving German roads for cyclists, which, in the long run, will benefit society and an increasing number of individuals.

1.1 Benefits of cycling

Including cycling into one's daily life comes with a multitude of advantages. These include health benefits for the individual, the environment, and the economy.

To measure the relationship between health and cycling Oja et al. (2011) performed a systematic review study. They found 16 cycling-specific studies in which both cross-sectional and longitudinal studies showed a clear positive relationship between cycling and cardiorespiratory fitness. Furthermore, observational studies demonstrated a strong flipped relationship between commuter cycling and all-cause mortality, but especially cancer mortality and morbidity in elderly study participants. Studies among adults showed consistent improvements in fitness and improvements in strength of the hearth muscle due to commuting cycling. They conclude that the existing evidence reinforces the current efforts to promote cycling as an important contributor for better health among the population. Still, cycling is associated with risks, which are smaller when driving by car such as risk of crashes and a higher exposure to air pollution. A study by Johan de Hartog et al. (2010) compared the positive and negative side of cycling related to life expectancy. They found that daily commuting 7.5 km reduces your life expectancy by 0.8-40 days due to air pollution and 5-9 days because of traffic accidents. Yet, life expectancy gains attributed to cycling are between 3 to 14 months. Combined, this results in a net gain in life expectancy. The win loss ratio is evaluated at 9 times positive. This net positive effect is even higher, the older the person and the longer the daily commute.

This number can increase even more, if more people cycle. Y. Chen et al. (2022) found a positive relation between an increase in cyclists and an decrease in carbon dioxide and other air pollutants like nitrogen oxide. He links this to the reduction of cars. Cycling also offers economic benefits for the city and local shops. As well as an increase in job opportunities (Blondiau et al., 2016), the monetary advantage of cycling for cities comes from the fact that bicycles have less impact on road quality and do not require as much space. Additionally, the chance for walk-in customers is significantly higher when cycling or walking as when taking the car, benefiting local shops.

1.2 Open source and open data

The term *open* summarises two subsegments which will be introduced thereafter. In general, *open* implies that everyone can access, use, modify, and share free of cost or requirements, unaffected by the implied purpose (Open Knowledge Foundation, 2022).

Open source refers to the practise of sharing a project’s source code. Within the software industry, there are two typical approaches to distributing a software project. The first one shares the project with its source code protected. This is generally done by compiling the code to an executable which is then shared. As it is nearly impossible to decompile an executable back to the original source code, it is protected. The reasons for this approach could be to protect intellectual property, limit misuse, and increase security.

When distributing open source software, the source code is always sent to the recipient along with the compiled code, or sometimes even only the source code is distributed. Customers must then compile the project for themselves. Additionally, a certain licence is attached to the software that allows the user, with varying degrees of freedom, to manipulate, share, or even sell the software. This behaviour is always prohibited with closed source software (Fitzgerald, 2006). In addition to the access to source code, the Open Source Initiative (2007) defines nine other criteria that an open source project needs to fulfil. They deal with the licence that should be attached to each open source software project. The licence must allow any party to sell or give away the software as part of a larger project without requiring a royalty or a fee. It must allow others to share modifications and derived works under the same licence. However, it can restrict modifications, but only when *patch files* are allowed that modify the software at built time. The Open Source Initiative also requires the licence to be non-discriminatory against individuals or groups and open to all fields of applications. This also includes projects that might be regarded as immoral or unethical, e.g., in the context of war or genetic engineering. Additionally, the licence needs to be applicable when the software is distributed without the need to perform tasks or licence agreements on the receiver side. Furthermore, if parts of a project are extracted and shared on their own, the same licence agreements should apply. Similarly, if two projects with different licences are merged, both licences should still be applicable for their respective parts. Lastly, the licence should be technology neutral. As those limitations can still be interpreted differently, a wide variety of different licences have been developed and used. The most common ones are the *Apache Licence 2.0, the BSD-3 and BSD-2 Licence, the GNU General Public Licence,* and the *MIT Licence*. The main difference between them is how derived software needs to be licenced. Yet, these licences are the foundation of the advantages that open source offers. Firstly, the general assumption is that free software will result in an increased market share (Fitzgerald, 2006) and, related to that, a higher longevity of the tool, as more people use it and can participate in its development (Nyman et al., 2011). This also stems from the fact that many open source projects start out of necessity or as a creative outlet for a single author. These schemes are then presented to the public, which, due to the open licences, can freely participate in the development and improve the software (Heron et al., 2013). This idea that everybody who has the technical capabilities can participate in the development cycle of open source software has led to the success of this approach. Today, many of the biggest software projects, for example, Postgres, Kubernetes, or TensorFlow, are open source. These projects are developed in cooperation between employees working full time on those projects and volunteers who contribute in their free time. The reasons for the contributors spending their free time on work that others get paid for are manifold: perfecting their expertise, enhancing their reputation, altruism, filling a personal need, or simply for fun (Heron et al., 2013).

Regarding academia, *open source* aligns with the principal of *open research*. As algorithms and software diagrams published in scientific papers may show the general shape of a solution, implementation subtleties can make it difficult for others to recreate the software so that empirical results can be replicated and compared. These and related issues argue strongly in favour of the release of *open source* research code along with publications. Along with allowing access to the source code when publishing in a journal, fellow researchers and readers should be allowed to access the data that was used. This again aligns with the principles of *open data*.

Open data refers to data that everyone can access and use. Similar to *open source*, *open data* comes attached with a licence that should state the same attributes as open source software. Additionally, the data should be machine readable (E-Government-Gesetz - EGovG, 2017). The benefits of open data apply to both companies as well as society (Zuiderwijk et al., 2015).

Within the software industry the concepts and benefits are mostly accepted, and most of the large software projects are based on open source. However, within the science community, these concepts of open data and open source are only slowly becoming accepted and implemented. The concepts of open science are fundamental when conducting comparative measurements that compares multiple study sites.

1.3 Ideas behind the framework

As mentioned, cycling offers numerous advantages for health and the environment. However, many people in Germany still rely on cars, even for distances under 5km (Gebhardt et al., 2022). One of the reasons for the reluctance to use a bicycle is the infrastructure. It often feels uncomfortable or even dangerous to go by bike. This circumstance requires policymakers to encourage and, above all, support citizens to choose the bicycle as a means of transport. Providing good infrastructure that complies with modern standards on safety and design is crucial here. This has also been recognized within the scientific community which developed the concept of bikeability to measure cycling quality. Yet, as researchers cannot agree on a unified definition, they also lack an amalgamated approach to measure bikeability. Furthermore, relating studies differ in scope, the size of the study area, and the parameters used. Therefore, the results for a bikeability study in Munich (Schmid-Querg et al., 2021) cannot be compared with a study in Basel (Grigore et al., 2019). While these explementary cities have their differences in population size or topology, for example, the underlying city architecture is similar (old town, a river, European style buildings). Therefore, a comparison between cities should be possible, so that city planners can learn from positive as well as negative decisions.

For the previously stated reasons the goal of this thesis is to develop a framework with the following goals:

- The framework should enable researchers to generate a single citywide bikeability score.
- The index should be comparable between cities.
- The framework should generate a per road bikeability score.
- The road bikeability score should reflect the importance of this segment.
- Aligning with the ideas of open science, it should be reproducible.
- The underlying model should be highly configurable.
- The score should be calculated without major intervention.
- The framework offers a good overall performance.

Retrieved from these eight goals, five requirements can be deduced that the framework needs to be fulfil. These are:

- To be reproducible, only open source software can be used.
- To be reproducible, only open data can be used.
- To be reproducible, a platform independent underlying technology must be used.
- Data must be available with a spatial resolution in the meter range.
- Performance cannot be neglected.

Furthermore, the framework should be easily adaptable to other modes of traffic, such as walkability, therefore reflecting how inviting the environment is to walk, or focus only on certain groups within cycling, such as the cargo-, or e-bikes. While this had no direct influence during the conception of the model, it should be realisable due to the easy modification system.

To explain the framework, the thesis will be divided into four more chapters. The next chapter deals with the existing literature on bikeability, to offer a background on which features are important and should be integrated into the framework. Additionally, a literature review on distributed system architecture is attached. Whilst the framework as such does not need to be distributed, it uses the ideas and technologies for performance and scalability.

Having explained the scientific background, the next chapter deep dives into the design and functions of the framework. The framework as such is built as a foundation for future bikeability schemas. To be adapted to a researchers' needs, the herein supplied version comes with an integrated model that can be seen as a proof-of-concept and follows the design principles of sensible defaults. The rating of these parameters is adapted from the papers presented in the literature review but does not aim to be scientifically defendable. Still, the results for the study site of Augsburg, Germany are described in detail, followed by a bikeability score for all cities in Germany between 200,000 and 600,000 inhabitants. After the presentation of the results, the framework is discussed in detail, debating design decisions and trade-offs that have been taken to align with the above stated goals. The discussion will also detail parameters that could have been implemented but are not. To conclude the discussion chapter, the proof-of-concept model is compared with other indexes that rated either German or international cities, to get a sense for the general validity of the framework.

The thesis is concluded with a look into the future of the framework, reflecting on possible improvements and adaptations.

2 Literature review

The following section is split into two chapters. The first reviews existing literature on bikeability. This will be used to determine parameters that the framework should be able to address, and is used as the foundation for the proof-of-concept model. The second section reviews literature on designing a distributed software architecture. The best practices reported there will be the foundation of the framework's architecture.

2.1 Introduction to bikeability

2.1.1 From walkability to bikeability

The term "walkability", coined by Saelens et al. (2003), describes the relationship between walking and the urban environment. However, it lacks a unified definition, as many authors use it to describe the different aspects of walking through a city, for example the interaction with walking infrastructure. Furthermore, some authors use the term to refer to pedestrians and cyclists alike, simply as opponents to motorised traffic. Yet, several publications, e.g. (2016), and Krenn et al. (2015), call for a clear distinction between walkability and bikeability, which will be discussed later on.

Recent publications describe *walkability* as a multi-dimensional concept that can broadly be defined as the degree to which the built environment is pedestrian-friendly and enables walking (Habibian & Hosseinzadeh, 2018; Taleai & Yameqani, 2018). The discrepancies in the definition, stem from the changing number of variables in the urban environment that are used to evaluate walkability. The urban environment includes all physical aspects of infrastructure, services, as well as activities that can be found within a city (Fonseca et al., 2022). As the walkability of a city has a multitude of beneficial attributes, it has become an interdisciplinary field (Dovey & Pafka, 2020). The benefits range from a positive influence on the health of the population (Stevenson et al., 2016) to increased productivity in the context of an innovative information economy of idea formation, creativity, and knowledge spillover (Storper & Venables, 2004). The walkable city has also been recognised as mitigating differences and segregation of social class, ethnicity, gender, age, and ability (Massey, 2005). Additionally, walkability is the basis of a climate-change adapted, low-carbon city centred around transit-oriented neighbourhoods (Curtis et al., 2009)

The measurement of walkability is often based on seven broad categories: (1) land use density; (2) land use diversity; (3) accessibility; (4) street network connectivity; (5) pedestrian facility and comfort; (6) safety and security; and (7) streetscape design (Fonseca et al., 2022).

Land use diversity can include a multitude of measurements such as residential density (residences per area) (Adams et al., 2015), population density (persons per area) (Koohsari et al., 2016), amenity density (amenities per area) (Adams et al., 2015), building density (buildings per area) (Robinson et al., 2018) and job density (jobs per area) (Vargo et al., 2012). Somewhat similar, land use diversity measures the concentration of land uses within an area (Habibian & Hosseinzadeh, 2018).

The accessibility reflects the distance between the key amenities as well as public transport. Key amenities include doctors and supermarkets, but also parks, city centres, or other attractions. In general, the distance has a large influence on the decision to walk or to drive (Berke et al., 2007).

The connectivity of the street network is related to the directness of routes compared to the straight line, as well as to the possibility to choose alternative routes. Higher connectivity positively influences walkability (Ellis et al., 2016). To measure the connectivity, several options are used. Adams et al. (2015) use the intersection density, while others calculate the number of cul-de-sacs (Habibian and Hosseinzadeh (2018), or the number of street segments per area (Williams et al., 2018).

The category of pedestrian facility and comfort includes the sidewalk characteristics, slope, as well as environmental conditions. Characteristics of a sidewalk like width, support furniture, or even the bare existence of said sidewalk can be easily observed. The same can be said for the slope of a segment, which influences the energy needed to walk there (Vargo et al., 2012). Air pollution, canopy cover, and greenness level, on the other hand, change seasonally and are not as easily measurable. However, they also need to be noted since they also influence walkability (Lee et al., 2020).

Safety describes how the pedestrian is protected from motorised traffic, while security is related to crime (Williams et al., 2018). Traffic safety has been measured by the risk of having accidents, traffic exposure, and the adoption of traffic slowing measures. Moran et al. (2018) indicate that high traffic volumes were found to be a barrier to walking as well as the risk of accidents being associated with less physical activity. Measuring crime could be done by interviewing for local perceptions as well as using more visual features like the number of street lights, the number of graffiti, or thrown-in windows (Lee et al., 2020). Lastly, streetscape design elements that influence walking include the most significant aesthetics, the possibility to look a long way, complexity, and scale (Yin, 2017).

As noted above, cycling has sometimes been included in the conceptualization of the term walkability. However, this approach is outdated and based on the car-centric ideology that has dominated the perception of policy makers in North America and Europe since the Second World War (Fishman et al., 2013). Their approach was to differentiate between motorised travel, e.g., cars, and non-motorized travel, e.g., walking and cycling. This started to change as a result of the 1992 Rio de Janeiro Earth Summit and related events when a more modern approach to transit-oriented development (Padeiro et al., 2019) emerged. This approach focuses on trains, light-rail, or metros, which rely on passengers walking or cycling to the station. This paradigm shift led to the development of the concepts walkability and, later on, bikeability.

On the one hand, the similarities between walking and cycling are clear. As they are not motorised, the source of energy needs to come from the person itself. This limits travel distance to the amount of energy each person can spare. In addition, the walker/cyclist is fully exposed to the environment, meaning that weather, air pollution, and other road users have a high influence on the walk/cycle tour. Connected to that, the chance of crashes is higher, with a higher risk of possibly lethal injuries. Lastly, the percentage of trips made solely for recreational purposes is higher (Cao et al., 2009; Muhs & Clifton, 2016).

On the other hand, the differences are more significant. The most important ones are the travel speed and travel distance. The average person walks at a speed of 4.5 to 5.3 km/h (Bohannon, 1997), while cycling at around 16km/h (mean 18.4 km/h) (Boufous et al., 2018). With the advancement of electrical bikes, this difference will increase as e-bikes ride around 2 km/h faster than traditional bikes (Schleinitz et al., 2017). The average trip length, which is much shorter for pedestrians than for cyclists, can be deduced from that. Furthermore, walking is often only part of a trip and is used in conjunction with trains, buses, or even cars. Changing the travel mode is less likely when cycling (Krizek et al., 2009).

Another difference is the general obstacle that needs to be overcome in order to be able to cycle. A bike needs to be bought, it needs to be parked, it needs to be repaired and served, and additional equipment, like helmets need to be bought. Besides, before you can ride, you need to learn it first.

Next, the infrastructure needs are different. Pedestrians mostly walk on separated sidewalks and multi-use paths with minimal conflicts with other vehicles. Interactions with other road users for them happens mostly at intersections. Bikes, on the other hand, can be ridden within a multitude of infrastructure setups, with more points of conflict. Added to the previously mentioned multi-use paths, bikes can be ridden among motor vehicles on a road with no bike-specific infrastructure, on a road with painted arrows to indicate the travel lane is to be shared, in a striped bike lane adjacent to vehicles with no separation or painted separation, or in a facility separated from vehicle traffic by grade, a vehicle parking lane, bollards, planters, or other physical objects (Muhs & Clifton, 2016).

Additionally, it is possible to switch from biking to walking at any time. Once dismounted,

a bike can be pushed along the road and the cyclist becomes a pedestrian. Technically, the former cyclist can then switch to buses, trains, and subways as a new mode of transport, taking his bike with him. Yet, in practice it is rarely that simple.
There is also a difference between genders when comparing walking and cycling. While both genders walk equally often, even in countries with high gender equality like New Zealand, men cycle more than women (Shaw et al., 2020). Additionally, Zhao et al. (2019) argue that the effects of the current weather are greater for cyclists than for pedestrians. Lastly, the laws and policies associated with and the requirements for building new infrastructure are different (StVO, 2013/2013).

These differences make it obvious that a distinctive, independent term to assess the relationship between bikes, or rather cyclists, and the urban environment is needed. And as such *bikeability* will be discussed in more detail in the following chapters.

2.1.2 Different approaches towards bikeability

As with walkability a unified single definition for bikeability is nowhere to be found. As different authors include different components within it, only an ambiguous understanding of the term has emerged. Within dictionaries, the term *bikeable* is simply defined as "(an environment) suitable or safe for cyclists." [1] or as "suitable or fit for biking" [2]

A different definition comes from Lowry et al. (2012). They define bikeability as "an assessment of an entire bikeway network for perceived comfort and convenience and access to important destination". However, they also introduce the terms "bicycle suitability" and "bicycle friendliness". The *suitability* assesses the perceived comfort and safety of any section of path where cycling is allowed, including shared-paths and roads. *Friendliness* evaluates a group of various aspects of a bike ride. This includes bikeability, laws and policies related to cycling, the acceptance of cycling within a community, and the efforts undertaken to get people to cycle. The authors argue that these distinctions are important and need to be evaluated separately. A road that is suited to being ridden by bike but leads to no useful destination has a high suitability. However, the bikeability is low as the network is not complete. Another case would be a high bikeability and suitability but no laws and policies to protect cyclists and no encouragement to cycle. As a result, the low bicycle friendliness would also hinder cycling. Resulting from that definition is the understanding that bikeability is a subordinate concept which needs to be viewed within a spatial context and with additional content included.

[1] Oxforddictionaries.com (2022): BIKEABLE. Meaning & Definition for UK English. Available at https://www.lexico.com/definition/bikeable; 15.08.2022
[2] Merriam-Webster (2022): bikeable. Definition of bikeable. Availabe at https://www.merriam-webster.com/dictionary/bikeability; 15.08.2022

Nielsen and Skov-Petersen (2018) define bikeability as “the ability of a person to bike or the ability of the urban landscape to be biked”. Within this very broad definition, they argue for a spatial disaggregation. Whereas in the United Kingdom the term is related to the physical teaching of people and kids to ride a bike (Bikeability, 2022), in other areas it is applied in a more scientific research background.
Another definition from Grigore et al. (2019) describes it as “a measure of the ability and convenience in reaching important destinations by bike, based on the travel distance weighted by the perceived safety, -comfort, and -attractiveness of the streets and intersections along the routes”. They embed their definition in the context of cyclists trying to minimize the distance to be travelled but maximize the safety, comfort, and attractiveness of their chosen route.

The definition of Grigore et al. (2019) is the one that will be used as a basis for this work. In accordance with a wide variety of other definitions, this is a multitude of frameworks developed to measure the bikeability of an area. The following sections delve into the ideas behind those frameworks.

2.1.3 Analysing route choices

An important part when assessing the bikeability of an area is to understand where people ride their bikes and why they choose certain routes over others. With this information, general information about the weight of factors can be quantified. Studies related to this topic will be presented next.
In research, there are two major approaches to determining route choice and its determining factors (Sener et al., 2009). Aggregate-level studies cover a generalised relationship between the number of cyclists and the existing cycle network. For example, they observe the influence of new cycling infrastructure on the number of people who bike. Disaggregate-level studies try to understand each individual route decision performed by the individual bicycle rider.

2.1.3.1 Stated preference

The advantage of the latter approach is that the essential relationship between route choice and its determining factors can be analysed and interpreted. Additionally, data collection used to be easier or less expensive. One group was asked to recall actual routes whereas another group was equipped with GPS devices (at a time when smartphones didn't exist yet). Another advantage is that it is unnecessary to process detailed travel network data. There is no need to generate alternative routes on a real network. Models can also be simpler because the input data is very structured and there is only a limited number of alternatives or answers. Finally, employing this concept of stated preference, researchers can test for situations that are not yet in existence or occur rarely, which is not the case for revealed preference data (Broach et al., 2012).

A common way to generate disaggregated-level data is through surveys. Surveys can be designed simply so that respondents only need to rank bike infrastructure options. In a more complex setup, respondents need to actively choose between separate routes with certain trade-offs. In Stinson and Bhat´s (2003) setup, for example, the interviewees were asked to imagine a situation where they moved to a new city and needed to commute to a new job via bike. They have two options for the route: They could either choose the route that follows a minor arterial road with a wide right-hand cycling lane. The cycle lane is continuous, but they need to traverse one or two traffic lights. Alternatively, they could choose a wide residential road with no red lights. However, the cycling lane would be discontinuous. They defined a road as being continuous if the bike facility was available for the entire route without any interruptions. Consequently, they defined a bicycle facility as discontinuous if there was no bicycle facility for at least 25% of the route, meaning that the cyclist needed to share the lane with motorised vehicles.

Yet, there are also disadvantages to stated preference data. It is difficult to know how well a participant can map these textual or visual representations to his or her preferences for real facilities. When being confronted with a short route description or an image on a computer screen, many salient features of a route area will be missed. Also, although the choice set is, in a sense, controlled, it is likely that respondents have their own usual routes as points of comparison in mind (Sener et al., 2009).

2.1.3.2 Revealed preference

The availability of geographic information systems (GIS) enabled a new type of survey, the revealed preference. Whilst the data acquisition was still survey-based, participants did not choose between fictional situation but had to recall the rides they had previously undertaken. Those provided paths were then analysed via a GIS. Yet, Broach et al. (2012) argue that even though these findings prove valid and helpful, the first attempts to analyse bike route choices based on revealed preference data never resulted in the estimation of a full route choice model. In particular, the models lack a comprehensive choice set of paths since the recalled route is most of the time compared to the shortest path (Harvey et al., 2008). In addition, the various models focus on predicting certain aspects of route choice, for example, distance deviation from the shortest path or the presence of bike facilities. Therefore, they cannot be applied to predict path probabilities for a large set of routes. So, they are most useful when the behaviour of cyclists is analysed but not when the trip distribution in a network is in focus (Zimmermann et al., 2017).

2.1.3.3 Path-based choice sets

Menghini et al. (2010) proposed a method to disable the limiting factors of the revealed preference approach. The authors' main innovation came down to exploiting automatically processed GPS-based observations. This was previously done for car route choice models, because for cars, the data acquisition phase started earlier (Prato, 2009). The dataset the authors used was provided by a private company that generated the data in order to study the effectiveness of advertisement boards. Within the dataset, 2,435 residents of Zurich (the study area) were included and tracked for an average of 6.99 days in the year 2004. The dataset was filtered to contain only bike trips using mode detection and map matching to a street network built from different data sources.

The special feature of the choice model is that, for the available start-destination pairs, alternative routes are created, the choice set. These routes were created with the approach presented in the paper by Rieser-Schüssler et al. (2013). They use a breadth-first search link elimination approach, which searches for the shortest path between start and destination and then removes the segments from the link in turns. Based on these shortest paths, new starting points for the next round of elimination are generated. The resulting networks are unique and connected. This is repeated until the number of choices is as desired, or the network is exhausted.

The alternative routes were generated with the length of each segment as its cost. The additional factors were average slope, the maximal slope, the existence of bike paths, the number of traffic lights, and a path size measure. The researcher could show that the chosen tracks are, in comparison, shorter, less steep, involve fewer traffic lights, and more marked bike paths. In 35.9 % of the cases, the chosen route was also the shortest route. Based on these findings, they developed their model. Hood et al. (2011) based their research on this approach and transferred it to San Francisco; Broach et al. (2012) added a larger set of attributes.

A known problem with these path-based choice sets is the number of conceivable paths that are endless (if loops are allowed). Additionally, the actual number of selected and generated paths is unknown to the researcher. This can be overcome when using similar approaches like Menghini et al. (2010) that eliminate segments or by labelling route segments following the approach of Ben-Akiva and Bierlaire (1999). As a result, two different assumptions can be made about the choice set. First, the choice set contains all paths considered as alternatives. However, this has a large influence on parameter estimates as they can vary with the selected choice set. Alternatively, the choice set can be generated by sampling. Therefore, all possible routes between origin and destination are considered within a universal choice set, so that the parameter estimations can be corrected for bias that is inherent to sampling (Frejinger et al., 2009).

2.1.3.4 Node-based choice sets

As a result of the mentioned shortcomings, Fosgerau et al. (2013) developed a link-based network route choice model which has a unrestricted choice set. Their idea of a recursive logit model describes the route choice as a series of individual link choices instead of a singular path choice. The decider selects the utility-maximising outgoing link where the link utility is specified as the instantaneous cost, the expected maximum utility to the destination, and the error term. The choice to select a segment is then reported as a multinomial logit model and expected downstream utilities are identified from Bellman equations. The final path is therefore generated dynamically and step by step. Loops within the network can result in an infinite number of potential paths with an illogical length. Fosgerau et al. (2013) found that the probability of choosing a given path has the form of a multinomial logit model, except that the number of alternatives is infinite. This model also incorporates a link-additive correlation attribute, named *link size*. This is needed because in observed actual networks, paths to a destination share links. Due to this overlap, it is assumed that paths share unobserved attributes, so that path utilities correlate. If this aspect were ignored, the final result would have erroneous path probabilities. The Link Size employs the expected link flow as a substitution for the overlap, which can then be corrected for.

A further refinement of the link-based approach was introduced by Zimmermann et al. (2017). The authors adapt the model so that non-link-additive attributes, such as slope, can be incorporated into the link utilities of the model. Additionally, they provide a case study to estimate a cyclist’s route choice based on GPS observations in the road network of Eugene, Oregon. The data they used was generated by a special app that was developed by Broach et al. (2012) for his path-based approach in San Francisco. The respondents were asked to provide general information about their demographics and cycling habits. The rides were then tracked via GPS and users were asked to provide the purpose of the trip. In total, 648 trips cycled by 103 users were recorded. The provided road network was extended to include paths, minor alleys, and bike boulevards and, as a result, consisted of 16,352 nodes and 42,384 links. Information about the included network was length, slope, car volume, one-way restrictions, speed limits, presence of bike facilities, traffic signals, and stop signs. The results were similar to those by Menghini et al. (2010): length and slope limit the detours, but bike lanes increase the willingness to detour.

2.1.3.5 Purpose of trip

Based on GPS data Nair et al. (2019) propose a framework to also determine the reason for the trip. While there is plenty of research looking at what mode of transport was used or whether a person was walking or running (for example, Gong et al., 2012), the authors were the first to develop a setup for bicycles. Yet, this idea was previously applied to cars. They argue that existing and established methods for cars cannot simply be transferred to bikes because bike trips are significantly more open in terms of route choice.

As car-based studies have shown time, place, and land use at the destination can be used to infer the purpose of the trip; they were therefore used to analyse the trip. The land use was deferred by the *points of interest* available in OpenStreetMap. Additionally, the duration and length of the trip were added. A second advanced approach also used personal data, such as age, income, ethnicity, cycling type, history, and frequency. This, so far, is analogical to car-based approaches. Cyclist-specific was the inclusion of the travelled street segments. The authors argue that the selection of segments is related to destination and the type of cyclist. A commuter is more willing to take a shorter route with no bike infrastructure than a person that cycles once per week to go to, for example, a café.

The route segments were enriched with eight different attributes.

(1) Segment slope values for minimum, maximum, and average

(2) A graph-based characterization, using values like the centrality, degree, betweenness, and page rank

(3) Presence of bike facilities (based on OSM and provided by cities)

(4) Presence of different types of POI

(5) Crime statistics

(6) Crash statistics

(7) Parking violations

(8) Complaint reports issued to the city

To summarise, the purpose was to be deferred by its spatio-temporal features, the sum of the segments and the personal information. With this input, the authors could construct a common classification problem. They applied logistic regression, random forest, and devised a special-purpose classification they call XGBoost to solve it. This approach was also based on the existing car-based literature.

This framework was then applied to a dataset based in the city of Philadelphia, which offered the data as part of their open data initiative. The dataset was then categorised into the groups: commuting, social, exercise, errand, work-related, shopping, school, or other. The results show that for a successful classification, all available data is needed, especially personal data. As a result, this approach is not applicable to being used for anonymized data. With an F1 score of 0.83, their own proposed XGBoost algorithm resulted in the best classification.

2.1.3.6 Google Maps approach

In the same paper, the authors also propose a new method to analyse route choice. Instead of performing a path-based or node-based approach, they simply compare the actual routes to the three suggestions made by Google Maps (Gong et al., 2012). They then go through two steps to determine important features for the route selection. First, they apply a multivariate, two-sided Kolmogorov-Smirnov test to determine a significant difference. Secondly, they employ multivariate, two-sided Mann-Whitney tests to factor out parameters that influence route choices. They find that green areas, cycling facilities, and centrality are the most important factors.

2.1.4 Bicycle Level of Service

The term Bicycle Level of Service (BLOS) refers to measurable criteria of the physical environment that influences the perceived quality of a bike ride. This bases BLOS on the perception of individuals which can therefore vary greatly. As with bikeability there is no clear definition of the term and various authors define it differently, respectively include different sets of parameters (Navin, 1994). Regardless of their definitions most of these studies produced an index which focus on certain aspects e.g., (Botma, 1995; Jensen, 2007). To summarize the various papers will be divided into 3 subchapters. These deal with Bicycle Flow, infrastructure and exogeneous variables.

2.1.4.1 Bicycle flow

The first subchapter describes the flow of cyclists through an environment and how it affects BLOS. The flow is particularly important in BLOS studies because it describes how cyclists move and how this movement affects the rider's wellbeing. Yet, the number of papers published on the topic is relatively sparse. As a result, many observations are made with tools and methods that were originally developed for cars. However there are fundamental differences between the two modes of transport that limit the usefulness of applying those tools (Hoogendoorn & Daamen, 2016). For example, the speed, acceleration, and braking power are individual for everyone, whereas motorized vehicles behave similar in their braking or acceleration profiles.

The relationship between vehicle speed and distance form the basis for traffic flow models, this connection is called the fundamental relationship. The fundamental relationship is a useful concept to relate key variables such as speed, density, and flow, which can then be used as tools for LOS estimations (Y. Yuan et al., 2018). However, the general usefulness of this relationship has been doubted, as this connection can change significantly based on the individual characteristics of the elements affecting it.

Navin (1994) published a paper arguing that some of the car tools can and should be applied as there are similarities in certain situations. The flow parameter can therefore influence the BLOS in the form of *comfort zones*. He proposes two oval zones with various degrees of freedom for the cyclists. The comfort zone measures 1.7m by 3.4m and exists if there are no other cyclists in this area. The circulation zone is an area in which a cyclists can move freely. He defines it slightly larger at 2.3 by 4 meters. If the cyclist is within both of these zones he can manoeuvre freely around obstacles like cars, potholes or buildings sites. The LOS is high as the cyclists can flow freely.

Another important factor in flow analysis it the hindrance phenomena. Hindrance is defined as a degree in which one is unable to manoeuvre. For cyclists this happens regularly and in high frequency while overtaking others. This is also heavily influenced by the type of road user that is overtaken (e.g., regular bike, e-bike, e-scooter…). Botma (1995) showed that the higher the number of overtaking processes the lower the LOS. Additionally, he argues that there is difference between passing and meeting. Meeting occurs when a rider and the hindrance move in opposing directions. As both parties participate in the process and acknowledge each other, the LOS stays higher contrary to passing, when a rider overtakes (pass) a road user from behind. The speed at which both parties passed proved to be less of an influence.
This model was further advanced by Hummer et al. (2006) who differs between active and passive passing, meeting and delayed passing. Active passing occurs when you overtake somebody else whereas passive passing is to be overtaken. He defines meeting similar to Botma. Delayed passing occurs when you need to postpone your overtaking process due to road limitations.

As previously touched the modal interaction has also an influence on riding flow. The higher the number of different transport modes on the same road the higher the number of passings respectively overtaking's. This is increasingly more important because the average e-bike is 3km/h faster than the average cyclists (Schleinitz et al., 2017). Within the pedestrian level of service research community this already been acknowledged in papers like Kang et al. (2013), which argue for a distinction between e-bikers and classic cyclists.

Allowing motorized vehicles on bike infrastructure also lowers the BLOS and introduces stress.

2.1.4.2 Infrastructure

Cycling infrastructure plays a critical role in increasing the ratio of cycling within all modes of transport (Hull & O'Holleran, 2014). The overall benefits of increased cycling are worth nearly five times the cost of investing in new cycling infrastructure (Cavill et al., 2008). However, the decision to cycle is mostly based on the infrastructure close to the origin or destination of a trip. Furthermore, intersections play a crucial role in the perception of the overall cycling infrastructure. Also, the adaptation of new modes of transports, like e-bikes, e-scooters or electric vehicles influences the perception. Due to the heterogeneity of both off-road and on-road facilities, the characteristics of an infrastructure must meet the various needs of its users. Important aspects of this are elaborated in the following section.

Sharing Policy. A sharing policy decides which parts of the available infrastructure are available to a cyclist. The policy can either be *on-street* or *off-street.* On-street facilities are roads where bicyclists share lanes with motorized vehicles, including shared lanes, paved shoulders, on-street bicycle lanes or buffered bicycle lanes. Buffered lanes represent the infrastructure where a painted line separates the bicycle and the motorized vehicle lanes. Off-street facilities are those, where the cyclist has predominantly right of way. This also includes pathways, which must be shared with pedestrians. As this is conflict prone, different countries have implemented different policies to address this issue.

Traffic Enforcement. Traffic enforcement relates to signages, traffic lights, and drawn lines to indicate right-of-way. Whilst these enforcements are necessary to keep the overall traffic movements, Lu et al. (2018) argue to if cyclists are too frequently stopped, either due to red lights, or due to cars having right-of-way, this can lower the bike adaption rate. Furthermore, as signs need to transmit a complex message, unwisely erected signs, can confuse cyclists and other road participants. This happens especially often in cases, with shared by cyclists and pedestrians but no drawn line in the middle of the road (Hunter et al., 2011).

Pavement condition. The road condition influences a cyclists wellbeing as a bad condition leads to bicycles vibrating, which strongly influences the users' perceptions of a given cycle track, general cycling comfort and also the choice of route. Z. Wu et al. (2015) conducted a study in China to evaluate the impacts of pavement damage on the operation of bikes. They concluded that more than half of the bikes changed their trajectories, and the bike speeds significantly changed near the damaged road segments if the potholes where equal to, or greater than, 15 mm in depth.

Road furniture. The influence of road furniture on cyclists is twofold. On the one hand, they increase the safety as they lower the speed of cars on the road, on the other hand they also lower the speed and comfort of a cyclists. Moreover, high speed bumps can also increase the risk to fall (Mertens et al., 2016).

Bicycle Parking. Trip-end facilities can encourage cyclists and improve their comfort levels. Especially, in large urban areas, bicycle parking can be a problem, both for the cyclist who wants to store the bike safely but also for other people, when parked bikes are using up a lot of space (C. Yuan et al., 2017). The main implications a bicycle parking facility should fulfil are shade, security and that bikes are parked orderly.

Attractiveness. Another factor to Bicycle Level of Services is the attractiveness of the route. This includes greenery as in parks, trees, and forests, but also natural features like ponds, rivers, or lakes (Hull & O'Holleran, 2014).

Safety. The perception of safety can influence a cyclist in a multitude of ways (J. Wu et al., 2018). This includes all interactions with other road participants, e.g., how large is the street, how wide is the cycling lane…. But also, safety regarding crime (Alveano-Aguerrebere et al., 2017). This results from cyclists being more vulnerable compared to when using a car or a bus. Taking a car, emulates a safe space which locks out the outside and taking the bus, one is always safe in the crowd. Therefore, factors like crime statistics, but also if roads are illuminated influence the BLOS (Hull & O'Holleran, 2014).

2.1.4.3 Exogenous factors

Weather. The importance of the weather has been linked to a decrease in cycling by many researchers. Bergström and Magnusson (2003) surveyed over 1000 Swedish employees about the travel behaviour in different seasons. The results showed a decrease in cycling trips by 47% compared to summer. Car trips on the other hand increased by 27%. Depending on the category of cyclists, the opinion on what factors influenced the choice of transport mode for the journey to work changed. Temperature, precipitation, and road condition were the most important factors to those who cycled to work in summer but not in winter. Exercise was the most important to those who cycled frequently in winter, and travel time the most important to those who never cycled to work. Similar results were reproduced in other regions of the world, see for example Nankervis (1999) in Australia. A study from Parkin et al. (2007) link the decrease during winter not only to the shift in temperature but also the decrease with sun hours. Analog to the seasonal changes, a shift can also be measured over the week. People that cycle during the week, mostly commuters, are less affected by shifts in the weather, than recreational cyclists on the weekend. The overall negative effects of weather are attributed to an unpleasant feeling (cold, wet…) or a higher feeling of risk (slippery road, lowered sight…). Thru this combination the weather has been argued as one of the most influential parameters of bicycle level of service Ayachi et al. (2015).

Topography. Topography conditions have an impact on the comfort of cyclists. Still, only a few studies include the slope into the measurements of cycling comfort. The slope increases the need for physical activity and is therefore affecting the cyclists (Parkin & Rotheram, 2010). Moudon et al. (2005) argues that the slopes have no direct effect of bike-sharing and therefore also no direct impact on the overall population that cycles. Furthermore, other studies link the slope with a surplus of bike riders. They argue that the existence of hills and therefore slopes is linked with a more interesting and exciting environment which leads to a better visual cycling quality (Fernández-Heredia et al., 2014).

Sociodemographic aspects. Within many societies bike usage is depending on age, gender and educational level. The number of cycling men is greater than the number of women who cycle (Savan et al., 2017). This is generally attested to the risk of cycling. In countries with poor infrastructure, networks, and policies the discrepancy is highest. As such, gender, age, and education have a high influence on bikeway network studies (Buehler & Dill, 2016).

2.1.5 Existing frameworks to assess bikeability

Even though there is a multitude of assessment methods available, broad groups can be defined. A possible border can be drawn on which data is used (Moudon & Lee, 2003). The available options are to use only hard factors for bikeability or use both soft and hard factors. Hard factors include everything that can be measured and quantified, like the number of bicycle facilities and infrastructure. Soft factors include the policies and political agendas within an area. This separation is related to the distinction between hard factors as part of bikeability and soft factors as part of bicycle friendliness that is proposed by Lowry et al. (2012).

The second distinction is related to the process of data acquisition. It can either be through the technical analysis of infrastructure, using Geographic Information Systems (GIS), or by employing surveys to gather data. Using a GIS results in accurate data, whereas surveys convey a perception of the situation.

Grouping could also be done by the area that is interrogated. Related to that are the spatial resolution of the final rating as well as whether the result is a raster or a vector. Those decisions allow for a broad spectrum of results. For example, studies can either be city-wide but with a 100x100 m grid-based resolution (Krenn et al., 2015) or only a city quarter where each road segment is categorised (Schmid-Querg et al., 2021).

2.1.5.1 Munich Bikeability Index

In the recently published paper by Schmid-Querg et al. (2021), a bikeability index for the city centre of Munich is created and evaluated. The study area, which was chosen because of local knowledge and available data, has a densely structured road network frequently travelled by many cyclists as it is located between three universities. However, the number of available bike paths is said to be lower compared to most other parts of Munich. Additionally, the areas support multi-modal travel.

The authors surveyed cyclists about four predetermined factors, which they deemed sufficient to assess the bikeability. Those factors were the existence and type of bike path, the speed limit, the parking facilities for bicycles, and the quality of infrastructure at intersections for bicycles. Respondents were asked to rank different types within these categories between zero and ten. The types included ranking a broad bike path on both sides of the road versus a cycling street versus a regular street or ranking a traffic light with a designated cycling traffic light versus an intersection with no traffic light. Each segment of road was then assigned a value, for road type and speed limit. Parking facilities influenced connecting road segments and intersections influenced half the connection road segments. The resulting four layers were then overlayed and the final value for each segment was generated. The generated index showed a lack of suitable bicycle infrastructure, which, according to the authors, should be addressed (Schmid-Querg et al., 2021).

2.1.5.2 Bikeability in Basel

The area assessed, in the paper by Grigore et al. (2019) is located within the city centre of Basel, Switzerland. The east side of the area is bordered by the Rhine River. The area is densely developed and, as such, offers little possibility for new cycling infrastructure.

Using a formula, a weight for each road segment was determined. A road segment was

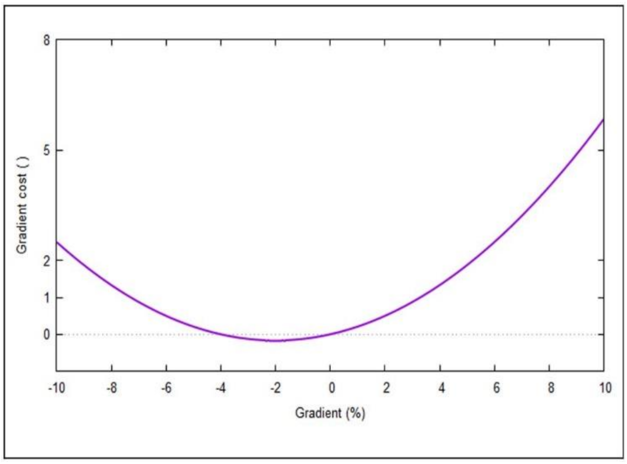


Figure 1 Gradient cost function *(Grigore et al., 2019)*

defined as a continuous stretch of road uninterrupted by intersections or road type changes. To calculate this cost multiplier per street segment, the gradient cost, the cost related to the type and dimensions of the cycling infrastructure, and the cost of additional hazards were added; the benefit of the riding environment was subtracted. The factors were based on existing research (Krenn et al., 2015; Winters et al., 2013). The resulting values ranged between zero and two, whereas lower values are better, and one is a neutral situation.

The cost of the gradient was calculated using a cost function visualised in Figure 1. The figure is derived from the Swiss norm regarding bicycle traffic (Swiss Association of Road and Traffic Professionals, 2017) which compares cycling speed and slope. Whilst the risk of injury is higher when going downhill, it is not associated with changing the route-finding behaviour but lowering the force needed to cycle and therefore includes a small benefit.

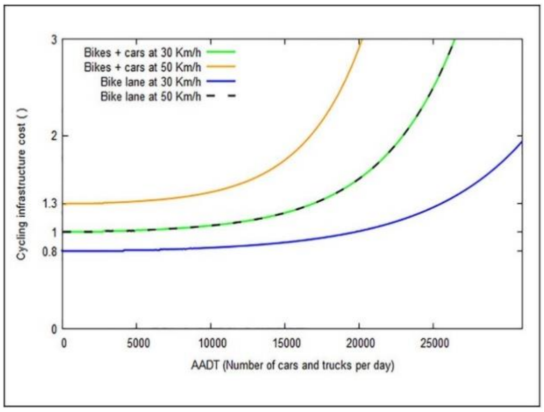


Figure 2 Cycling infrastructure cost functions "Bikes + Cars" and bike lanes at speeds 30 km/h and 50 km/h *(Grigore et al., 2019)*

The second chosen attribute was the cycling infrastructure, in conjunction with the annual average daily traffic (AADT). This attribute is more complex as there are various setups of shared or not shared infrastructure. The possible combinations are: bikes and pedestrians; bikes, cars and pedestrians; bikes and cars; bike lane; bus lane allowed for bikes; cycle track; and bike boulevard. The cost graph in Figure 2 was applied for the setup of bikes and cars. For bike boulevards, cycle tracks and shared pedestrian cyclist segments, the authors assumed constant values of 0.9, 0.8, and 1, respectively. Yet, distinctions were made influenced by the width of the cycle track. Additional hazards included parking (0.2), curb height (0.2), heavy traffic (0.2), or tram rails (0.5) in the same sections.

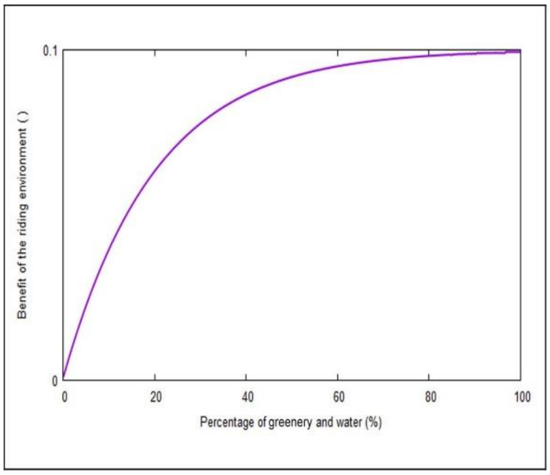


Figure 3 Benefit of the riding environment *(Grigore et al., 2019)*

Finally, the fourth attribute used to assess the cycling quality of segments was the riding environment. Greenery leads to a reduction of 20 % in the perceived travel time for pedestrians, in comparison to the actual travel time. Because directness is more important for commuting cyclists than for pedestrians, the authorsassume a maximum decrease in cost of 10 % (0.1). The benefit function was based on the coverage of greenery and water within a buffer of 20 m around the centre of the roadway and is shown in Figure 3.

The next step performed was the assessment of intersection quality. This was done using two metrics: turn cost components and a layout multiplier. For the first, the authors employed the following factors: basic turn, traffic light, AADT, and the existence of stop/yield signs. These were chosen based on the study performed by Broach et al. (2012). The basic turn cost was selected as cyclists tend to avoid turns. Depending on the presence of traffic lights, either a cost associated with that (except right turns) or an AADT cost was added. As a last step, a cost associated with stop or yield signs was added. The second input depends on the existence of bike lanes, bike boxes, spaces for indirect left turns, the number of car lanes (for direct left turns), and, lastly, the presence of a car lane turning right if there are no traffic lights. Because the authors did not find any quantification of these attributes, they quantified these attributes in comparison with each other. The result was a layout multiplier to quantify the influence of these factors.

With those two values, a perceived distance was calculated for each segment dependent on the route. This was done by adding the sum of all segments (calculated individually by multiplying the actual length with the cost multiplier) and the sum of all intersections (calculated individually from the turn cost multiplied by the layout value).

The bikeability was then calculated for one hectare squares. This was done because the authors had access to destination data, in this case the location of workplaces. It was assumed that cyclists choose the route with the shortest perceived distance. The authors computed bikeability as the average of the perceived distances along the paths with the least perceived distance. This was done for all destinations of interest, weighted by the number of workplaces. Normalisation based on the number of workplaces is necessary to ensure bikeability depends only on perceived distances and not on the absolute number of workplaces.

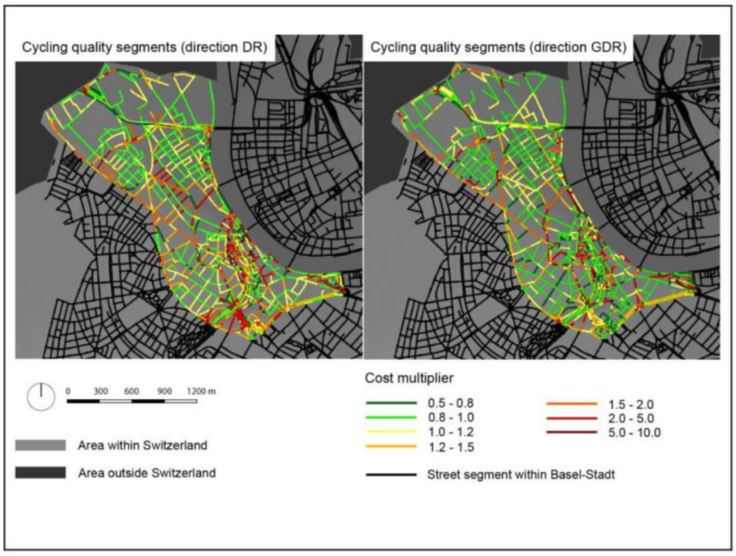


Figure 4 Cycling quality for segments in the case study area for both directions of travel *(Grigore et al., 2019)*

The result for the cycling quality per segment can be seen in Figure 4 in both directions of travel. The left side represents the direction of travel and the right side represents the opposite. The final bikeability raster showed a higher bikeability in the south of the study area. The authors argue that this is because there are more destinations and, therefore, a shorter distance to the destination. As a result, the authors employed another step to offer a better insight into the state of local biking quality.

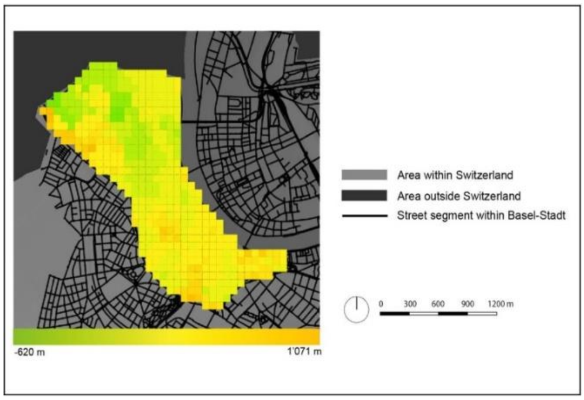


Figure 5 Subtraction of scaled average distance to workplaces from bikeability to workplace (unit: meter) *(Grigore et al., 2019)*

The subsequent calculations included, as discussed before, the computation of the bikeability within each grid cell. Next, the average distance to all destinations along the shortest perceived route is calculated. Afterwards, the average distance is scaled and finally subtracted from the bikeability within each individual cell. The general concept is that the actual distance is a linear value. The perceived distance, however, can increase at smaller rates (good infrastructure) or higher rates (bad infrastructure). This, when subtracted from the bikeability of a grid cell, results in values higher or lower than zero. Lower values (green) are good to cycle, while higher values (orange) are not well connected and therefore bad to cycle (Figure 5). As a last step, a sensitivity analysis was performed to measure the impact of improvements within the study area. The influence on the final score was minimal as the length and the slope were the most impactful variables.

However, the authors conclude that their approach is sufficient to determine areas that need to be improved, but it does not offer insight into what specific segment needs to be changed. Their framework can provide planners with insight about the situation; it uses data that is easy to access and can be applied to other areas. Nevertheless, refinement, like including actual GPS tracks and a better understanding of the influence of each quality attribute, needs to be performed (Grigore et al., 2019).

2.2 Introduction to distributed software architecture

A fundamental basis for the success of large, complex systems is a sophisticated software architecture (Clements et al., 2003). It defines the basic components and their interaction within the system. A system can be both distributed and undistributed. In the latter case, it is an amalgamation of several processes that together fulfil the expected function of the system (Clements et al., 2003).

In the context of distributed systems, two different architectural styles have become established. These rely on the concept of services for distribution: Service Oriented Architecture (SOA) and Microservices (Fowler, 2014; Sprott & Wilkes, 2006). In both these approaches, the goal is to solve the respective use case adequately and quickly by having the services work together and thus provide higher level functions for the overall system. Yet, theses styles differ in their implementation.

SOA prefers system decomposition into small services with a focus on an intelligent routing mechanism that includes the entirety of the enterprise IT. The advantage of this routing mechanism is centralised management.

Microservices, on the other hand, are based on smart services and simple routing mechanisms that require only a small number of resources. In addition, these services are executed within their own processes, which makes them more autonomous than the services in the SOA. However, this also results in a greater complexity for these services since they must take care of their communication with each other independently.

Microservices also allow the use of a so-called bounded context, a data model that adapts to the needs of the services. In this bounded context, however, microservices and SCS differ significantly. In microservices, the individual services are part of the overall system; communication between services of different contexts is possible.

In the following, the basic concepts and patterns when designing software architecture in connection with distributed systems are explained in more detail, and the most important properties of distributed systems are presented. Following that, the two most common styles are presented.

2.2.1 Concepts and patterns

Within a system, a service is a reusable function that can be used by different clients for different purposes and is defined by controlling a set of rules. There is a defined interface, and the service can be used independently over the network. By using communication protocols (HTTP, SOAP etc.), the service is platform-independent (Cerny et al., 2017). This can become clearer by envisioning the system *house construction*. The system *house construction* consists, among other things, of the service *cover roof*. This service can be used by different clients in different ways. For example, the client can implement it as a flat roof or a pitched roof something in between. The process of building a roof follows defined procedures (control rules). To make use of the service, the customer calls the office of the roofing company using their defined interface. The roofing company can be commissioned by anyone (independently) in Augsburg as well as in Dachau (platform-independent).

Distributed systems can be deployed in different ways. In a monolithic deployment, the entirety of all services and systems is replaced by the updated version (Fowler, 2014). In this process, the old version is deleted from the server and uploaded again in a new version. This is because all services are closely connected and there are no independent, delimited processes. This is also reflected in the versioning. The previous version must always be retained, otherwise there is a risk that functionalities based on the old version will be destroyed. When distributed systems are deployed as individual services, the previous version can simply be replaced instead of co-existing, as the systems acts largely independent from each other. This type of deployment is predestined to be cloud-based due to its easy extensibility and ease to update.

Distributed systems also differ in their user interface. This can be integrated externally via a portal or exist as an additional internal service in the system, which acts as an interface for another or a group of other services.

The scope of an application is closely related to the deployment and the bounded context (Evans & Fowler, 2019). Here, too, two major distinctions can be made. The scope can either encompass all IT-relevant areas of a company, such as the billing system, web shop, process automation and system security, or there is a separate system per project. The advantage of a project-based approach is that changes can be made quickly and locally. In the enterprise-based approach, the change process is more complex, but existing systems can be easily reused and integrated with each other.

Integration mechanisms and technologies can be simple and primitive, or smart and complex (Schmidt et al., 2005). The intelligence refers to the communication level. In smart and complex systems, intelligence is transferred to the communication layer and is responsible for transferring data to the right place. This simplifies services, as they do not have to worry about the transfer. This approach is mostly based on enterprise service busses as the central system of the service architecture. The bus is responsible for transport, transformation, and monitoring. Contrary to this are simple and primitive communication mechanisms mainly based on HTTP, RPC, and message queues. With the latter approach, the intelligence must be in the services as they become more complex, because they are responsible to ensure that the communication works.

Data storage is either centralised or shared. In centralised storage, there is one database where everything is stored. With shared storage, there is one database per use case. However, a distinction must be made here as to whether a database is available for each service or for a group of services. The advantage of a central database is the lower initial and maintenance costs, since only one database must be administered. The disadvantage is the limited choice of technology. With shared data storage, more flexibility is possible. Consistency is often cited as a disadvantage of this form of storage. However, this can be minimized thanks to the concept of eventual consistency.

Figure 6 Orchestration versus choreography *(Cerny et al., 2017)*

The scalability of a service infrastructure is limited by their implementation. When using an ESB or a central database, only vertical scaling is possible. By employing a container-based architecture style near “infinite” scaling is possible, especially when deploying into the cloud.

Communication patterns between services can be divided into two groups: orchestration and choreography. These patterns refer to the way in which the sequence of activities takes place or how a business process is structured. Orchestration is used for centralised business processes. Activities are coordinated through different services and the results are then combined. When using the orchestration model, a canonical data model is often introduced, in which all services agree on a superordinate data model (Xiao et al., 2016). This makes subsequent changes to this data model difficult since all subordinate services must also be retrofitted. The choreographic approach, on the other hand, allows a simplified change to the data model or the individual service since there is no central element for the service composition. The approach originates from Domain Driven Design (Evans & Fowler, 2019) and describes exchange of messages, agreements on interfaces between interacting services, and rules for interaction. Services can thus be separated along their respective context, which makes a canonical data model obsolete. Structuring the communication like this also transfers to how teams interact. Depending on the pattern, these must be centralised (orchestration) or can work independently (choreography) of each other.

Most services are incorporated from other services. The service size describes the functionality that a service represents, i.e., how far or how "finely" a service is divided into further sub-services or how much logic is contained in a service. Services can cover only very small functions, such as 'show time', or larger functions, such as 'show weather'.

2.2.2 Service Oriented Architecture

SOA is a style of architecture that was developed primarily for building systems in larger companies. Services form the basis of the implementation (The Open Group, 2011). More precisely, SOA is about the independent construction of business-oriented services that can be combined into meaningful, higher-level business processes and solutions (Rosen et al., 2012).

SOA provides a schema for the organization and use of these services, which can be under the control of different ownership areas/domains (Oasis, 2005). SOA distinguishes between consumers and providers (Sprott & Wilkes, 2006). However, these different interests in a service are aligned in a business-centric SOA approach. A service can therefore be composed of a multitude of services. SOA offers a powerful framework within which the provision, combination, consumption, and reuse of services can be described and organised. This enables a simplified approach to solving these complex systems. Visibility, interaction, and impact are the key terms to describe SOAs (Oasis, 2005).
Visibility refers to the ability of those who have needs and those who have capabilities to find each other. This is usually done through descriptions of properties such as functions, technical requirements, associated constraints, policies, and the mechanisms used to request and/or respond. Through visibility, the need for a service can be compared to the capabilities that are available.
The use of a capability is called an interaction. It is usually initiated by the exchange of messages. In the process, the interaction can go through a variety of other actions. However, these interactions are all embedded in a certain context. This describes the series of elements that are passed through on the path between the providing service and the consuming service. The resulting path provides a clear decision point that justifies and represents policies and decisions.
The purpose of using an interaction is to achieve an outcome that satisfies the user's request. This outcome can be the return of information or the change of state within the system. A distinction can be made between public and private actions. Private actions cannot be recognised by other users and services. Public actions, on the other hand, lead to a change of state, which has an influence on other services.

Additionally, to a subdivision into SOA key terms, SOA can be divided into four elementary levels. Figure 7 shows the important elements that form the basis of the Service Oriented Architecture (Rosen et al., 2012). The diagram, which is divided into four layers, distinguishes between functional and informational concepts. Functional concepts form the basis to create systems and processes. Informational concepts describe the data used by the functional concepts on the different layers. These four layers will now be explained in ascending order.

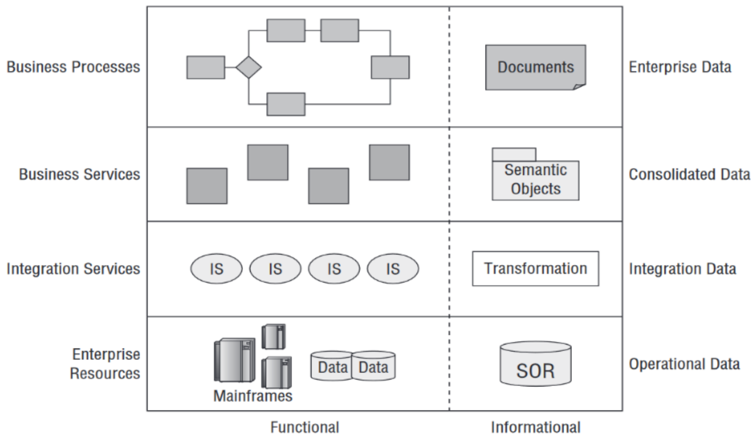


Figure 7 Architectural elements SOA *(Rosen et al., 2012)*

The Enterprise Resources and Systems layer consists of existing applications, legacy software and/or purchased commercial software, as well as older object-oriented implementations. These applications map business processes, i.e., transactions, which represent individual logical steps in the operational systems of the enterprise. The execution of a transaction usually results in one or more persistent records being read, written, or modified in a database (SOR).

Integration services provide integration and access to existing applications, and between applications. This often involves the transformation of data and functionality from what is desired at the business service level to what is possible in existing systems. This service is usually provided by an enterprise service bus.

Business services provide enterprise-wide business functions with a high level of abstraction. This layer provides an interface to the abstraction and integration of the layer below, removing the direct dependency between processes and existing systems. Business services provide business functionality through logical groupings of operations on underlying layers. Not all operations necessarily have to take place on the same operational system; operations can also be replicated across several similar systems. Business services work with semantic data objects, i.e., virtual data that describe the information that must be exchanged or passed on between the services.

A business process consists of a series of operations that are executed in an ordered sequence according to a set of business rules. Business processes represent long-lasting groups of actions or activities. They are composed of business services and usually include several service calls. Business processes work with business documents. The processes and documents are combined from the services and objects of the underlying layer according to a business process model and a common semantic data model. The scope of these processes is often the entire company.

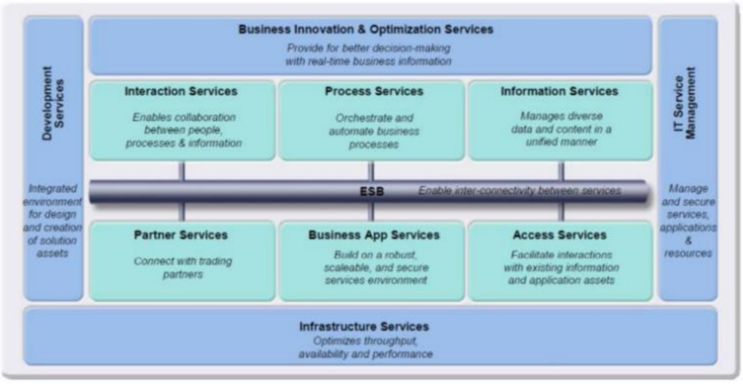


Figure 8 Cross-section of a SOA reference architecture ***(Korthaus et al.)***

For clarity, Figure 8 shows a cross-section of the SOA reference architecture looking at the integration layer and the business services and processes. In this example, business processes such as accessing assets outside the system or connecting to external partner services are shown in turquoise. The business processes in light blue are higher-level and represent, for example, processes for business-relevant decisions. The integration layer is represented by the Enterprise Service Bus. This connects the services and transforms the data to enable communication.

2.2.3 Microservices

Lewis and Fowler (2014) describe microservices as an architectural approach in which an application consists of many small services. These are characterised by the fact that they run in their own process and communicate with each other via simple protocols. This definition from 2014 has also been adopted in most scientific publications (Di Francesco et al., 2017).

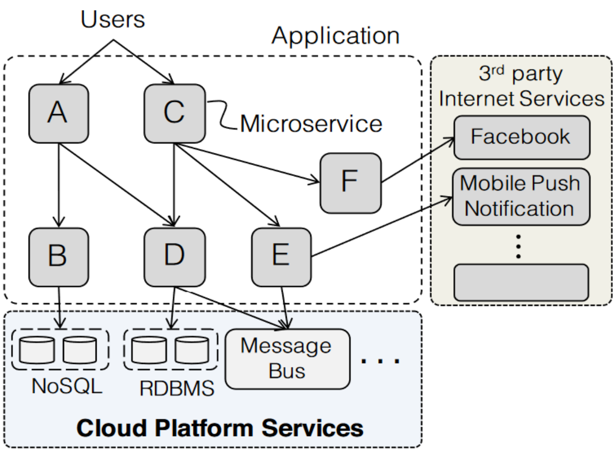


Figure 9 Typical microservice based architecture *(.NET foundation, 2022)*

Figure 9 shows a typical microservices-based web application. Boxes A to F represent the microservices themselves. They communicate with each other via simple HTTP calls, usually using a Rest API. Services A and C represent the input to the application and call further services on request of a user. These can then call either further services of their own, or external services, or access a database. In this diagram, the databases are the managed services of a cloud platform.

In detail, Lewis and Fowler describe nine characteristics that can be found in most systems based on the microservice architecture:

Component separation through services. Components are represented as a software unit that can be replaced and extended without affecting others. The components are not limited to a specific technology, but a single building block of a system. They can be changed, removed, or dismantled at any time by a user of the architecture (Johnson, 2012). In concrete terms, components as services means that components such as a Java library or a Ruby Gem are not executed within the main process of the application (see SOA), but that each library is separated into its own service with its own process. Communication then takes place via a Remote Process Call (PRC) or a REST interface.

Structuring according to business capabilities. In older software architectures, teams are strictly divided. Depending on the sub-area of the application and the specialty, a team consisted, for example, only of front-end or only of database specialists. This division made communication considerably more difficult since agreements had to be reached across team boundaries. Based on Conway's Law (Conway, 1968), this does not create the best possible software, but only reflects the structure of the company. The microservice architecture breaks with this separation. Teams are assembled along business competencies, creating a diverse mix of specialists from several disciplines, which promotes faster and more efficient communication and thus better systems.

Products, not projects. In this respect, microservices also differ from other software projects. The approach is particularly well known through Amazon Web Services (Orban, 2015). Instead of handing the system over to another team after development, the original development team maintains the system throughout its entire production and life cycle. The developers thus have a much deeper understanding of the scope and efficiency of the system and can continuously improve it.

Smart endpoints and dumb pipes. While in SOA the enterprise service bus (Schmidt et al., 2005) represents a very complex infrastructure that has to fulfil a multitude of functions, microservices rely on the opposite. Most of the protocols used are simple and standardised. For example, a service only receives an HTTP request, processes it accordingly and sends a response. Alternatively, message brokers are used.

Decentralised governance. In contrast to centralised governance, microservices do not require the use of a single technology (programming language, database, etc.), but developers can decide individually for each service. In this way, individual problems can be solved as quickly as possible in an equally individual way.

Decentralised data management. A separation of different data models is integrated in the architecture and the separation of different data storage options is allowed. This separation of data models is based on the concept of bounded context and domain-driven design (Rademacher et al., 2018). The bounded context defines the area of application of the specific business logic. Domain-driven design enables the decomposition of domains into contexts, each of which combines coherent domain concepts. These contexts then in turn correspond to functional microservices that map specific business capabilities. Based on this, the developer of a service is also given freedom in the choice of the database used. A service can use only one other instance of a database or also another technology, for example a graph database instead of the classic SQL database.

Infrastructure automation. The concepts of continuous integration and continuous delivery (CI/CD) can be implemented more easily with microservices than with monolithic software infrastructures. Continuous integration (CI) is the process of software development in which an individual's changes to the current software are regularly committed to a project version control system (Duvall et al., 2007). In the process, all changes are tested automatically. Based on this, continuous deployment is a software engineering approach in which teams produce software in short cycles and can thus update the production system in an automated way (L. Chen, 2015).

Design for failure. The possibility of a service failing is always considered during implementation. This makes sense because the division into many individual services increases the error frequency of the entire system. Monitoring and recovery of a service should always be integrated into the creation of a service (Heorhiadi et al., 2016). To test a system, special tools have been developed that spoof many possible points of failure.

Evolutionary design. Microservices are in a constant state of flux, as services are continuously improved, split up or removed completely. This is especially possible because microservices are easily interchangeable and thus offer great potential for improvement.

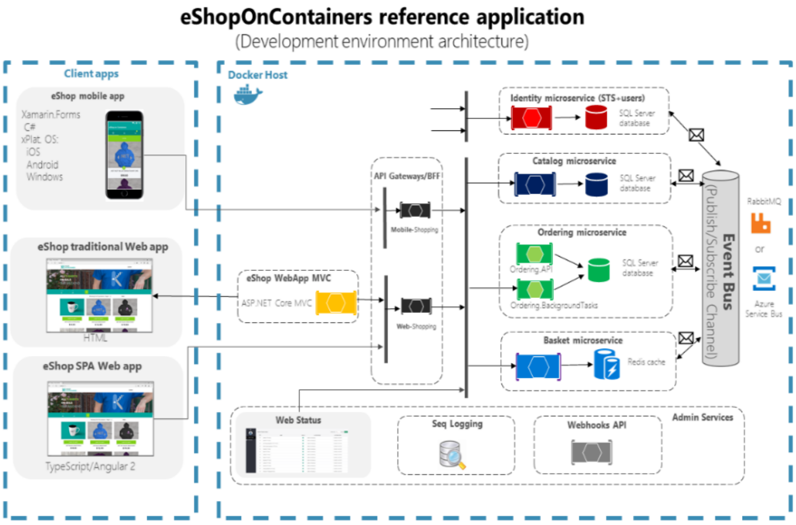


Figure 10 Reference architecture for an E-Commerce shop *(.NET foundation, 2022)*

To exemplify the previously discussed points a simple e-commerce shop application is explained. The implementation in Figure 10 consists of a client and a server. The client has three options for accessing the application: via a mobile client, via a classic website or via a single-page website. The request is then forwarded to the corresponding services by means of an API gateway. An API gateway is a service that represents a single entry point into the system and then forwards the requests within the system (Montesi & Weber, 2016). The redistribution is done by means of HTTP requests. The network is constantly monitored to be able to permanently control the load and functionality of the system (see Admin Services). The implemented services then cover the functions of the shopping basket (blue), the order (green), the catalogue (blue) and the authentication (red). The services are assigned to different contexts according to their colour. Each service in the system has its own database to hold the data that is used. An event bus is connected to the services. This uses channels in which the services can publish data or be notified of data changes. The services themselves run within a Docker container. Through the implementation of Docker containers, the use of API gateways and the constant monitoring of the system, the number of services can be flexibly controlled to increase reliability. Likewise, the load on certain services can be easily regulated by dynamically making resources available in the event of load peaks.

3 Case study

In the following section, the particulars of the proposed framework will be explained. With the framework, two main results are generated.

First is a map which shows, on an individual street level, how cyclable this street is. This is based on the source dataset. After that, this score is weighted by the number of routes that traverse it. Second is a single value, which describes the bikeability of the chosen study area. As all pseudo-random routes are generated, they are assigned a score based on the road segments. The final score is defined as the average of all route score results in the city. The final score is not simply the average of all road segments, because the number of variables affecting a route is higher, than the number of factors influencing the individual segment. The resulting values for both approaches range between -1 and 1, whereas -1 is a negative influence, 0 is neutral, and 1 is positive.

Even though the framework is developed to be applied to all major cities in Germany, this section will use Augsburg to visualise and explain various decisions affecting the model. Augsburg is a Bavarian city located at the confluence of the two rivers Lech and Wertach. It is the regional seat of the administrative region of Swabia and the third largest city in Bavaria with nearly 300,000 inhabitants (Augsburg Stadt, 2022). It is home to the University of Augsburg and the University of Applied Sciences Augsburg. The city consists of 42 different districts. The total area for the county of Augsburg is 146.85 km². The main reasons for choosing Augsburg are that the author has a vast knowledge of cycling in Augsburg due to a previous job as a bike courier, and that he can better relate between abstract dataset and reality as he can easily visit the sites in reality and therefore draw conclusions about the accuracy and functionality about the framework.

The general procedure within the framework is the same for all cities. It is schematically described in Figure 11. All the visualized steps will be described in the following chapters.

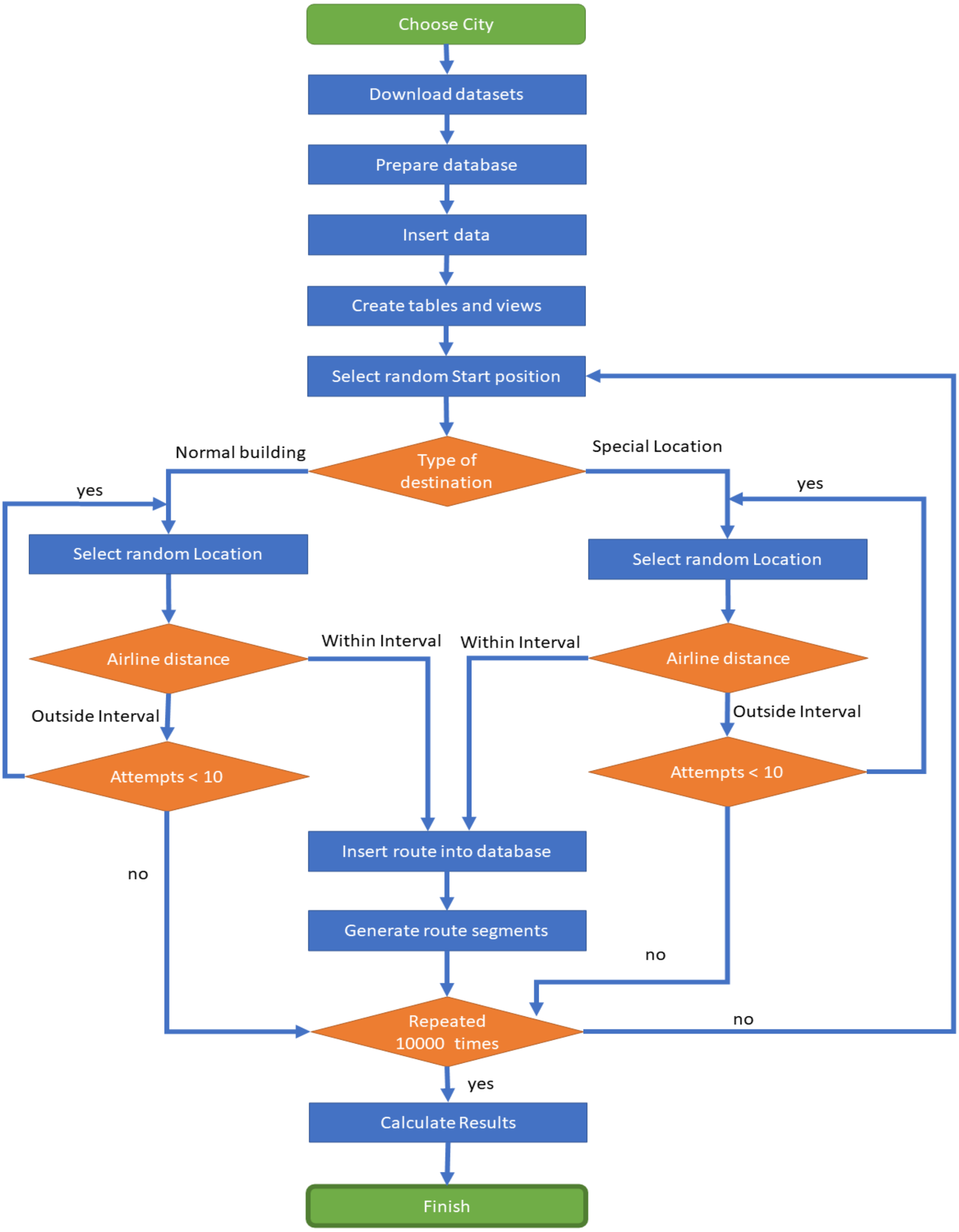


Figure 11 Flow diagram representing the data flow of the proposed framework

3.1 OpenStreetMap – the foundation of the framework

Based on the design goal that the framework can be applied all over Germany, the requirement for all datasets to be available nationwide arises. Additionally, they need to be spatially precise on a street level and they need to be *open*. To search for possible datasets, an extensive search through the web has been performed.

The city of Cologne, for example, offers an API where everybody can query the traffic load averaged over the last five to ten minutes. However, as this dataset would likely add to a quality assessment of cycling infrastructure in Cologne, it cannot be used anywhere else and therefore must be discarded (Amt für Verkehrsmanagement Stadt Köln, 2022).
Similar to this are all datasets offered on the new Mobilithek (2022) website hosted by the *Bundesministerium für Digitales und Verkehr* [3]. Whilst the website proclaims that data is the basis of a new modern mobility and more important than ever, the number of available datasets is limited. Additionally, there is a large discrepancy between the origins of the datasets. Germany-wide datasets do not exit, and most datasets come from North Rhine-Westphalia, especially Cologne (93) and Neuss (21). Additionally, even large cities like Munich have only seven related datasets and Augsburg has zero.

Another, probably useful, data source would be real time traffic providers like Google Maps or TomTom[4]. They provide real-time traffic data even on an individual street level. However, these datasets are protected, and access needs to be acquired.

As those datasets are the only ones that provide data at street level, aggregate datasets were considered. These included crime datasets for perceived safety, population density, or employment and education statistics. However, to be of any use, these datasets should not be at a city-wide level but separated into minimum city district. This limitation ruled out all freely available German-wide datasets, from sites such as *Deutschlandatlas (2022)* but also from paid services like *Living atlas* provided by Esri [5]. Still, the data at district level exists and is provided by cities like Augsburg with their *Kurzmitteilungen aus Statistik und Stadtforschung (2021)*. However, using it within an automated pipeline is not possible due to two reasons. First, each city shares this data under a different name, at a different place, updated at a different time, so that each city would need to get their own data acquisition pipeline. And, secondly, the data is not in a machine-readable format, but in pdf. This would also warrant individual data processing per city.

[3] Mobilithek (2022). Available at https://mobilithek.info/imprint (22.09.2022)
[4] TomTom (2022) LIVE Services. Available at https://www.tomtom.com/de_de/drive/maps-services/live-services/ (220.9.2022)
[5] Esri (2022) Living Atlas of the World. Available at https://livingatlas.arcgis.com/en/home/ (22.09.2022)

As detailed above, finding suitable datasets is not easy. The three prerequisites limit the available datasets to exactly one. Only the OpenStreetMap (OSM) [6] dataset complies with the requirements.

OpenStreetMap is an *open* collaborative project that aims to store a free and editable geographic database of the world (OpenStreetMap Authors, 2022). It is a project built by communities around the world that edit, create, and use the data. Since its inauguration in 2004, millions of users have contributed to the data pool. Today, it is one of the best data sources for geographic data. Whilst OpenStreetMap in essence is only a large database, thanks to its Open Database License, everybody can access all data and use it for a multitude of applications. The most important one is the generation of maps. These maps are the basis of many modern applications. Yet, these maps are not limited to street networks, as the name would suggest, but can display many different things. Features like buildings, rivers, and land use but also routes for ferries and busses, spots for drinking water, or park benches are included in OpenStreetMap. These features can also be tagged with a multitude of different key-value pairs that give a deeper insight into the actual location. For example, a road can be tagged in 28 different ways, with over 50 optional recommended tags[7]. Consider a residential road; it provides access to housing and is frequently lined with it, but it serves no purpose in connecting settlements. This already gives a very good impression of the actual road, but the impression can be further improved by tags like incline, cycleway, surface, or width. These tags can also be used as a basis for another application of the OSM dataset: routing.

The actual spatial position is displayed in an OSM-specific data format. The data is represented within connections between nodes, ways, and relations. A node is a point with a latitude and longitude value as well as an OSM specific ID. A simple point like this can then be annotated with tags to be displayed if it for example represents a park bench. Ways are built on top of nodes, so that they have the option of being tagged or are simply an OSM id themselves with a list of OSM IDs that represent the ID of nodes. It is then the task of the user to build up a geometry from all the nodes which represents the way. Ways can either be lines or polygons. This is decided by the first and last attached ID of a node. If it is the same then it is a polygon because the geometry represents a closed loop. Ways can therefore be lines like roads, or polygons like buildings. Relations are built on top of both nodes and ways to represent larger objects like a bus route. As the bus typically follows the road, it is more efficient to store the bus route as a list of ways than as a list of nodes, or

[6] OpenStreetMap Contributors (2022). Available at https://www.openstreetmap.org (21.09.2022)
[7] OpenStreetMap Contributors (2022): Key:highway. Available at https://wiki.openstreetmap.org/wiki/Key:highway (21.09.2022)

even duplicate geometries. The same principal of using nodes applies for ways. A node has a minimum size of 24 bytes: the ID represented as an int64 (8 bytes) and latitude and longitude as a float64 (8 bytes). The size of a way with five base points can then be as low as 36 bytes (1 * 8 bytes for id, 5 * 8 bytes id of the nodes). Storing a way as a combination of all node points would result in a disk requirement of 88 bytes (5 * 2 * 8 for the nodes + 8 for the id). This method is, however, only effective if the node is used by multiple ways or relations. It is safe to assume that this encoding method reduces the file size by more than 40%.

3.2 Brouter – routing engine

Generating the best route from A to B is a difficult task, because the multitude of factors influence routing decisions. Additionally, the definition of the *best* route also varies by mode of transport, time, place, and per individual. As a result, a multitude of routing engines has been created. These vary in the parameters, mode of transport, or even in the language they are implemented. The most famous one is *graphhopper*[8] which was one of the first and is now also commercially available. However, as with most other routing engines, its focus is on car routing.

Implemented with a special focus on bikes are for example *OsmAnd*[9], *Valhalla*[10] and *Brouter*[11]. They all included paths and trails into their network and incorporate parameters like road surface quality. Yet only *Brouter* fully integrates elevation into the routing process. Additionally, it allows the specification of user-specific routing profiles and comes preconfigured with 14 of these profiles. It also runs offline on Android, offers alternative route calculations, supports interim targets and no-go zones. It can consider long-distance cycle routes, is based on open-source MIT license, and, finally, thanks to being based on sophisticated routing-algorithms, implemented in Java it is performant. For these reasons, the Brouter was used to generate routes. The standalone codebase of *Brouter* has no graphical interface and can therefore only be used via the command line. Even though it offers the detailed advantages, other projects have been initiated to create various front

[8] GraphHopper GmbH (2022): GraphHopper Directions API with Route Optimization. Available at https://www.graphhopper.com/ (21.09.2022)
[9] Osmandapp (2022): OsmAnd. Available at https://github.com/osmandapp/OsmAnd, (21.09.2022)
[10] Valhalla (2022): Valhalla. Open Source Routing Engine for OpenStreetMap. Available at https://github.com/valhalla/valhalla (21.09.2022)
[11]abrensch (2022): Brouter. configurable OSM offline router with elevation awareness. Available at https://github.com/abrensch/brouter (21.09.2022)

ends that enable more people to use the routing engines. Those include Brouter Web[12]and bikerouter.de[13], which offer over 30 different profiles.

3.2.1 .rd5 files – OSM data extended with height information

As the OpenStreetMap dataset does not include elevation data, simple OSM extracts are not sufficient for the routing engine. Additionally, these extracts are large and include information that is unnecessary for routing. For those reason the Brouter authors supply *.rd5* files on the project's website[14]. These are generated worldwide for 5x5 degree squares and include only the routing-essential data from OSM and the elevation data based on the SRTM 90m Digital Elevation Database v4.1 as provided by NASA. For comparison, the complete area of Germany can be covered by four 5x5 squares with a combined file size of 566 MB, whereas the OSM extract for Germany, as can be found on geofabrik.com [15], is 3.7 GB. The .rd5 files include all tag information encoded in a binary bitstream which is each encoded with a lookup table that contains all the tags and values that are needed for encoding. In addition, there are two special cases "empty" for tag is not set and "unknown" if the tag is not in the lookup table. As the tags system of OSM has evolved over time, the file also uses "aliases" for tags that are different but mean the same. The elevation data is also conserved in the form of tags. However, an elevation buffer rule is applied. During the generation, each way (e.g., an edge in the network graph) gets assigned a minimal elevation value and a maximum elevation value. The average gradient is also assigned. As the elevation dataset has a spatial resolution of 90 meters, elevation changes within a single segment are ignored.

3.2.2 Profiles – configuring Brouter

The profiles can then draw on all the information within the .rd5 files to calculate the best route. The profile mapping file is built like a simple key-value pair where either a way with a certain segment can be allowed into the routing network or disallowed, or a tag is assigned a certain value for the cost calculation. For example, the profile *trekking.brf* sets the value "allow_ferries" to true, which results in ferries being acceptable in the route. Or it assigns a way tagged as motorway a cost of 10,000 (which is so high that these ways are never used) and a cycleway tagged segment 1. This high level of configuration results in the quantity of profiles available which find a route suitable for every use case, and are even adaptable to weather conditions.

---

[12] nrenner (2022): brouter-web. Web client for BRouter, a routing engine based on OpenStreetMap. Available at https://github.com/nrenner/brouter-web (29.09.2022)

[13] Jaschen Marcus (2022): bikerouter.de. der perfekte Weg um Bikerouten online zu planen und basiert auf BRouter-Web. Available at https://brouter.m11n.de (29.09.2022)

[14] Brouter (2022): Segments. Available at http://brouter.de/brouter/segments4/ (17.10.2022)

[15] Geofabrik (2022). Available at https://download.geofabrik.de/ (17.10.2022)

All routes generated in this thesis are based on the profile “trekking.brf” which is optimized for slow travel, with a tendency to avoid car traffic, but still focuses on approaching the destination efficiently. It can be found in the linked GitHub[16] repository. This is the default in most web based Brouter front ends as it best represents the average commuter.

3.3 Data preparation

The OSM data used in this analysis was downloaded from geofabrik.de[15]. The German company provides extracts of the full OSM dataset updated daily. They offer subsets of the datasets split at state or sometimes district levels, depending on the state size. The data is provided as *osm.pbf* files, an encoding method developed specifically for OpenStreetMap extracts. This offers two advantages. Firstly, the data is very compact, as the underlying technology, *protocol buffers*, is developed with a small file size as the main target. The second advantage is processing speed, because the files are organized so, that the content can be decoded in parallel, as that is the main time-consuming factor.

The datasets are then uploaded into a PostgreSQL database. PostgreSQL[17] is a powerful, open source object-relational database system that has been actively developed for over 30 years. It is widely used in the industry and has built a reputation for reliability, feature robustness, and performance. Additionally, it has been designed to be easily extensible both with new types and new functions. For this reason, a wide variety of extensions to extent PostgreSQL have been created. One of these extensions is PostGIS(2022)[18], which adds support for geographic objects and functions to be run through SQL. PostGIS is built atop of GEOS[19], a C/C++ library for computational geometry with a focus on algorithms used in GIS software. PostGIS also implements the *Open Geospatical Consortium Simple Features* geometry model. Within the following framework, PostgreSQL 14.5 and PostGIS 3.3 are used.

The data is decoded and sent to the database with the tool osm2pgsql[20], a tool specifically built for this. The 1.4 version is used with the chosen option to recreate the tables, including all tags as a *hstore* column (PostgreSQL extension for key-value data) and in memory only mode. This needs to be noted due to possible memory problems when inserting data. This problem should affect however only PC with an available memory limit of 8 Gb or lower. Another option used is to set the schema for a certain area. A schema, in the PostgreSQL specific context, is a collection of logical structures of data which use the same namespace.

[16] Wendel (2022): Bikeability. Available at https://github.com/Timahawk/Bikeability (17.10.2022)
[17] PostgreSQL (2022): PostgreSQL. Available at https://www.postgresql.org/ (01.10.2022)
[18] PostGIS (2022): PostGIS. Available at https://postgis.net/ (01.10.2022)
[19] GEOS (2022): Available at https://libgeos.org/ (01.10.2022)
[20] osm2pgsql (2022): osm2pgsql. Available at https://osm2pgsql.org/ (01.10.2022)

The database allows for multiple schemas within the same database. This is relevant because we want to compare different cities and need to use all data available. The city-specific data is then not referenced by table name e.g., table *augsburg_roads* and *ulm_roads*, but by schema name which is accessed via the structure *schema.table* nominator e.g., table *augsburg.roads* and *ulm.roads.* This greatly improves readability, structure, and automatability. For each analysed city, the smallest available *geofabrik* dataset was used and inserted in a schema named after this city

After this, four SQL scripts were executed that generate indexes, views, or new tables based on the three tables that are created by *osm2pgsql*. These tables are *world_line, world_point, world_polygon*. As previously discussed, OSM data is structured into nodes, ways, and relations. However, these are not OGC Simple Features and cannot be uploaded into the database. Osm2pgsql changes these in the following way: nodes are transferred into points; ways are transferred into lines and polygons respectively and relations are also transferred into lines and polygons. Additionally, as the data gets transferred from unstructured to structured data, osm2pgsql creates several columns with common tags that are then either filled with the respective key or a Null value.

The first scripts adds a new integer column to the three tables which is used as a unique key throughout all processing. This step can theoretically be omitted, as each geometry in the OSM dataset has a unique *osm_id*. However, most likely to some clipping process, there are sometimes duplicates ids. For example,

The second script creates a multitude of views and materialized views that are used for further processing. These represent the parameters that are presented in the next chapter. The decision if a view will be materialized was based on the performance implications for the system. Views that were used frequently, for example the intersections, are generated as materialized. Based on the smaller table size, the performance could be increased. The Views were created by searching for specific key-value combinations, for instance a polygon view displaying the study area is created by selection everything that has the key *boundary* (used to mark borders of areas) and a key-value pair *admin_level:6. Admin_level* represents country specific border types, in Germany[21].

The third script uses the views generated before to create new tables that don´t use the inserted geometry but generate derived ones. Most notably is the table *roadsegs* which splits the original lines which were identified as roads into smaller road segments. This is visualized in Figure 6; the original blue line is a continuous road with three intersections. This road is then split at each intersection. The new segments all have the same attributes

[21] OpenStreetMap Contributors (2022): Key:boundary. Available htttps://wiki.openstreetmap.org/wiki/Key:boundary (21.09.2022)

and can be related to the road via a unique ID. These segments build the basis of all following analysis. The last step is the generation of views that show possible weak links in the road network and present the final score.

3.4 Explaining route generation

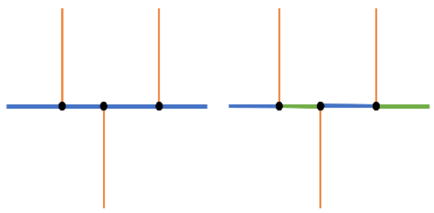


Figure 12 Schematic road network diagram

After the preparation step, the data is used to generate the pseudo-random routes that are the foundation of all scores and results that are produced. These routes are based on top of a table that was previously generated. The table contains a point location for all buildings and all shops, offices, and restaurants within the study area. Categorised as buildings are all polygons tagged as *building* with an area larger than 100 qm². The limit on size is needed to exclude garages and sheds. As this table determines the origin and destination pairs for routes, leaving them in leads to an unreasonable number of destinations in suburban areas. The shops, offices, and restaurants were selected by their associated tags. For each city 10,000 routes are generated by the following steps. First, the database is queried for the number of rows n within the origin/destination table. Then the script chooses a random number in the interval of zero and n, that represents the start point. Following that, the scripts generate a number m between 0 and 1. If m is smaller than 0.75, the destination can only be a point that has been tagged *shop*, *office*, or *restaurant*. If it is higher, the destination must be a "normal" building. This decision is based on existing research papers to cycling purposes (Goel et al., 2022). This result is wired to the database which selects a random row that fulfils the result of m. Additionally, the database calculates the distance between origin and destination, and returns the result. The result is then checked if the straight line is within a predefined interval. The interval limits the distance to a minimum of 500 m and a maximum of 10 km. This interval represents the average cyclists commute trip (Goel et al., 2022). If this limitation cannot be fulfilled, it is retried ten times with the same origin. If still no pair is generated, the origin is discarded.

After this step, the script has two longitude-latitude pairs which are then send to the Brouter routing engine. The engine returns either an annotated *geojson* response with the generated route or an error. These errors can either be that the origin or destination are too far away from a road, or that there is no routable connection. The successful response contains a 3D line and some properties about the route, for example length, ascent, or travel time. This result is then inserted into the database. As a last step, the route is matched to the previously calculated road segments. As both the segments and the route are created by the same dataset, the matching can be performed by creating a buffer of 1 meter around the route and selecting all segments within this polygon. This selection is then inserted into a table which contains an identifier for the road segment and an identifier for the route.

3.5 Parameters included in the framework

As has been described within the literature review, the factors influencing cycling in the urban space are manifold. In this framework, over 20 have been included. All of them are normalised between minus one and one. Minus one represents a negative influence on bikeability, zero symbolises no effect, and one characterises a positive effect. The general evaluation process follows typical switch-case or match-case approaches that can be found in most programming languages. The considered value is compared with each case and if it fulfils this requirement, this case is evaluated. The following cases are not evaluated. For clarification, read the example in the next subchapter.

3.5.1 **Cycling infrastructure**

Within the datasets, there are many different key-value pairs that affect the cycling infrastructure and thereafter its score. To be able to properly categorise them, the values are summarised into groups. The first group is designated cycling infrastructure. This category contains infrastructure that is clearly and legally designed for cyclists. It contains cycling streets, cycling lanes that are structurally separated from the road, and shared cycling-walking lanes, if they are proclaimed as that via a sign. The value assigned is a 1. This category does not contain marked bike lanes. These are assigned a value of 0.7. This distinction is made because research (e.g., Lowry et al. (2012)) has shown that cyclists rate designed infrastructure higher. The next group is roads that were labelled as motorway, or as federal, or national road (tagged primary, or secondary). As these are very unsafe for cyclists, they are awarded a -1. Roads tagged tertiary, which represent connecting roads, are awarded a 0.7 because a distinction can be made between how bad they are to ride compared to larger roads. All other road types (residential roads, tracks, …) are awarded a zero score. This includes sidewalks. The assumption is that they have no positive but also no negative influence on bikeability.

In the switch case approach for cycling infrastructure, two tertiary roads are evaluated. The first one has no additional associated tags. Therefore, the first match is the tag representing the road. A value of -0.7 is assigned. The second road has an associated tag which informs that the road has a marked cycle lane. As this case is tested before the one asking for the road type, it matches, and the road gets a 0.7 value. For this to work, the order of these cases must always be considered.

3.5.2 Speed limit

The next influential factor is the speed limit. As there is no data as to how fast the cars drive, the maximum speed limit is used as a general assumption. Depending on the speed, eight different values are assigned. The first case is a speed limit smaller than 30 km/h or a designated cycling path. This pattern of evaluating a value, in this case speed limit, but still being influenced by the infrastructure will be prevailing through most included parameters. The assumption behind this approach is, that for cycling on a designated cycling path, the speed of the cars does not affect the cycler due to the physical separation. As this separation does not happen for bike lanes, they are not included. The speed limit of 30 km/h reflects a traffic calmed area in which most cyclist enjoy cycling. Therefore, a score of one is assigned. Speed limit of under 40 km/h results in a score of 0.5. Following the same reasoning as before, it has a positive influence but to a lesser extent. Starting from speed limits of 60 km/h, a negative value is assigned ranging from -0.2 at 60 km/h to -1 for everything faster than 100 km/h. Roads labelled with no speed limit, or a speed limit of 50 km/h are awarded a zero score. Most of these roads are residential roads or tracks, and the framework typically assumes no positive or negative influence for bikeability on these road types, hence a value of zero.

3.5.3 Surface type

The parameter surface type consists of three summarised groups. First, all roads either asphalt or tar are awarded zero. The next category cyclocross is evaluated a -0.2. This category includes cobbles and paved sections, good quality field paths, e.g., very condensed dirt with few potholes, but also roads that are tagged for bad quality. Additionally, it is the default value. Segments tagged for mountain biking, logging paths, or bad tracks are awarded a zero. As can be seen with this parameter, values do not necessarily have to spread from -1 to 1 but can also be purely negative or purely positive.

3.5.4 Trees, parks, and forests

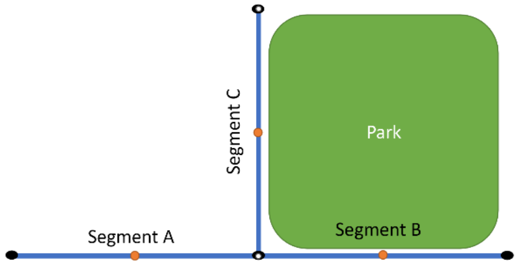


Figure 13 Schematic road network with park

The parameters trees, parks, and forests have a purely positive value. For trees, the value is influenced by the number of trees on a segment. The values range from a positive value of 0.25 for one tree up to the value of 1 for 10 trees. A tree is counted as being on a segment if it is within a 10 m buffer of the segment. It must be noted that there is a high spatial discrepancy within cities, if trees are tagged or included in the dataset.

Parks and forests are also searched within a 10 m buffer of the middle of the segment. Therefore, the score is always 1, independent of the length of the segment relative to the park size or amount of time spent in the forest. This decision was made due to the faster and easier implementation. Additionally, in most cases the value is 1 anyway. This is because road segments are short, and parks are typically framed by roads or paths which split the segments. The middle point is chosen to prevent situations where segment A would be related to the park because its buffer intersects the park polygon. Now, the 10 m buffer around the orange middle does not intersect. The limits of this approach are small side roads where the middle is close to the edge, or long segments with small parks.

Another important point is that these three parameters are not exclusive. Therefore, a road segment with a forest on one side, a park on the other and marked trees within the park would be rewarded threefold. As park include tags like playground and not all parks necessarily have trees, this is by choice. Trees within forests should not tagged. The general bonus of trees, parks, and forests is based on three effects found in literature. The better air quality and better microclimate near those locations and the general positive influence of green both visually and behaviourally.

3.5.5 Bus and tram routes

Bus and tram routes influence cycling quality negatively in multiple ways. First, busses as well as all larger vehicles lower the level of service because they have a higher risk of accidents. Due to their larger size, large blind spots can hide cyclists, which leads to crashes. This is especially the case when vehicles want to turn left on an intersection with a bike line on the left side. As a result, average cyclists try to avoid roads with lots of vehicles. For busses, there is the additional risk of pedestrians leaving or running for a bus with limited sense for cyclists. Furthermore, and this applies also for trams, people can wait for their choice of traffic on bike lanes, when the station is full, or they are simply unaware. For trams, the choice of turning is limited through tram rails. These, however, pose a significant risk of falling when crossing them or, even worse, when travelling parallel to them. The wheel can be caught within the rails guide rail and the intuitive sudden countermovement leads to a sudden stop which can throw the cyclists over the handlebar. For those reasons, the segments with parallel bus and tram routes get awarded minus one.

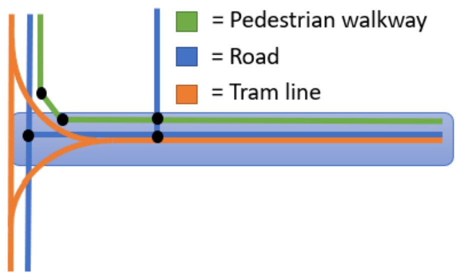


Figure 14 Simplified schematic road network of Augsburger Straße, Augsburg

Figure 15 Simplified schematic road network of intersection Stadtberger Straße–Bürgermeister-Bohl Straße

For bus routes, the process of finding these segments is simple. As they are based on the roads (see OSM encoding schema), the roads are buffered by 10 cm, and all roads that contain a bus route are awarded a minus one. For tram routes, a different approach is used which is necessary for situations like in Figure 14. First, the geometric representation of roads and tram lines is not on top of each other, therefore a native approach would just be to use a larger buffer. This is because tram lines are mostly (exceptions are, for example, the Bismarkstraße in Stadtbergen) depicted as one way roads as can be seen in Figure 15. Additionally, because tram lines have, due to their design, a rounder turning angle than cars, very big buffers need to be used to find road segments with parallel tram lines. This results in many small roads that get included in the result set. The chosen approach is therefore as follows: First, all roads get buffered 3 m. The 3 m limit is chosen because it is the smallest buffer that still matches tramlines in when the road is a single line in the middle. This would still match all segments shorter than 3 m, located in the buffer. Similar to the situation depicted in Figure 15. The road in question is the one between road and pedestrian walkway. To address this issue the angle between 0, 0 of the coordinate system and the road is calculated. The angle is based on the start and end point. Only if the difference between the angle of the tram line and the road is smaller than 45°, the road is kept in the result set. Lastly, the tram is buffered by 15 m and checked if it contains the road segment. Only then, the road is labelled as parallel to a tram. The last step is needed to limit edge cases. Otherwise, if the east-west road in Figure 15 was extended straight further east, the buffered road would touch the tram line and the angle would be the same. Finally, it needs to be noted that these complete calculations only get applied if the road has no dedicated, or painted cycle lane. It is assumed that nobody would build a dedicated or painted cycle lane on top of tram lines due to obvious safety reasons. Also, tram or bus stops have no direct influence on this, as there cannot be made a clear testimony on their influence based on the data.

3.5.6 Long-distance cycle routes

Figure 16 Cycling-specific sings

When travelling by bike or any other mode of transport, it is always a plus when there are a lot of signs that help to reach the destination. For cycling, the typical white signs with green font are used. These depict either a destination, like a village, city centre, or city district along with the length in kilometres, and a forest symbol if the route traverses field trails, or just simple arrows explaining where to go at an intersection. The OSM-tagging system provides key-value pairs for exactly these signs. However, the data source is very sparce. For Augsburg, only 37 of these signs exists. This is also the case in many other cites of similar size, what makes this a bad metric to evaluate.

There is, however, an alternative approximation method. Instead of using the individual signposts, the model uses long-distance cycle routes to evaluate the level of signs. Underlying to this is the general assumption that these routes are very well signed. This is typically the case, because the route signs are always attached to the bike way signs. Therefore, a road segment can get a one rating in signage if it is on top of a cycle route. This is calculated by buffering the route (10 cm) and annotating all roads within.

3.5.7 Permission

Another parameter that influences the result is *permission*. It describes whether the cyclist has the right to cycle where the routing algorithm sends him. This is evaluated by checking for certain tags on roads. The Brouter engine already includes those tags, so that, most of the time, this gets evaluated to zero. There are, however, special cases like pedestrian zones. For Augsburg, cyclists are only allowed to push their bikes through these areas between 08:00 am to 10:00 pm, but they can cycle through them outside of this time period. The time of travel is not incorporated into the model which makes the evaluation somewhat pointless. Still, road segments within the pedestrian precinct get awarded -0.5. Segments that are clearly labelled as no permission get -1.

3.5.8 On-street parking and road lighting

The road-associated parameters *parking* and *lit* are both evaluated by using the tags for each segment. As with routings signs, very few streetlights are tagged at all. The same applies for individual parking spots. Within the OSM dataset, there are large parking spaces, like car sharing parking lots, but no individual parking spots along a road. These are mostly included by tagging the road segment.

The calculation of the score is thus simple. Roads either get a score of 1, if they are tagged as illuminated, or 0, if they are not. As touched on in the chapter above, this is a highly temporal influence, both over the course of a day but also over the course of a year. On a side note, it must be mentioned that due to the current geopolitical events, illuminated streets are kept dark, which is very unlikely to ever be updated within the OSM dataset.

For parking, the approach is the same. Roads tagged with street side parking are awarded -1. The negative influence of street side parking is based on the many accidents that happen when car doors get opened into cycle lanes (can happen both with dedicated and painted on cycle lanes), or cars pull in or out of parking spots.

3.5.9 Straightness

The parameter *straightness* describes how straight a road segment is. The underlying research indicates that a higher average speed results in a higher level of service. One limiting factor on cycling speed is curves. Depending on the variables previous speed, curve angle, skill, and others, this limits the overall speed on a section. Consequently, this is highly influenced by the individual, and therefore hard to evaluate. Furthermore, the termination of a curve angle is computational intense. It is highly influenced by the number of curve points, which can highly vary between roads in OSM.

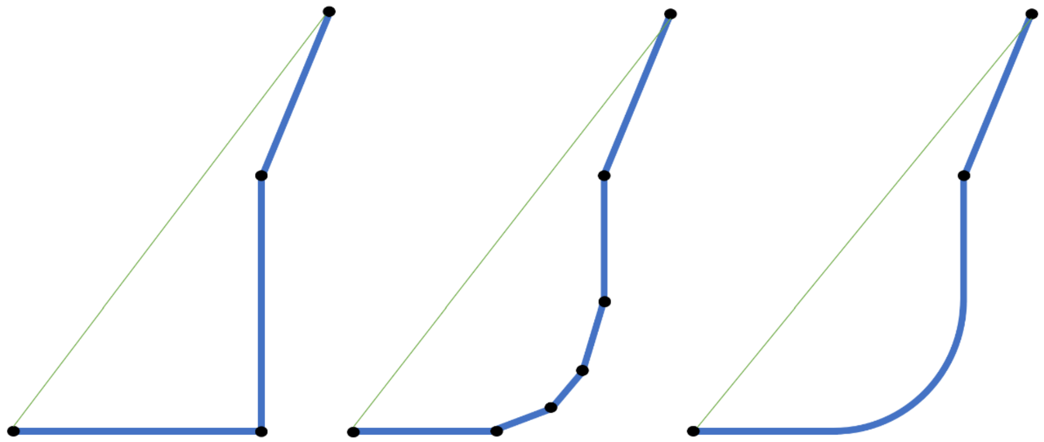


Figure 17 Simplified schematic representation of a single-.curved road segment

Figure 17, for example, depicts three times the same simplified road segment, which has two curves, with a different number of node points. The second curve is rather small and should have a low impact on the cycling speed. The first has an 90° angle, so slower speeds are necessary. The black points represent node points. There are no node points in last example, because this should depict a situation where thousands of node points are. This can happen when transforming angle geometries to simple OGC features.

As a viewer, it is obvious that this is in fact a 90° angle. However, the evaluation algorithm cannot see that immediately. In the first simplification, the algorithm could measure the angle between every node. The result would be that the segment has 2 curves, one with high angle and one with a low angle. This, however, fails in case 2 and 3. In case 2, the algorithm would calculate all five curves with a minimum impact. In case 3, thousands of curves would have no impact. The result would therefore greatly vary. A second approach would be to sum up all angles until there is a zero degree divergence between 3 points. Then, when the divergence is larger than zero, a new angle starts. This could work for 2 and 3 but would detect only a single curve in the first case. Also, especially for case 3, this would call for thousands of calculations. To sidestep problems associated with curves, a simplified algorithm is implemented. To decide the curviness of a road segment, the model simply compares the difference in length between the straight line from start to end point with the actual length. The higher the difference the curvier the road; hence, the lower the final score. Roads with differences smaller than 10 % of their length get a 1; 20 % get a 0.5; 60 % get a 0; 40 % get a -0.5; and lower than that get -1. This evaluation is attached once to the individual road segment score and once to the route score.

3.5.10 Bicycle parking and bicycle renting

All previously discussed parameters are applied on individual road segments. For parameters like *bicycle parking* and *bicycle renting*, this approach does not work. On the one hand, if you travel from Hochzoll to Pfersee in Augsburg, it makes sense to check each road on the trip if it is lit or has a bike lane and route accordingly. The routing decision is, however, not influenced by bicycle parking or renting in the middle of the route, at the Königsplatz for example. These parameters only influence the quality of the ride, if they are within a certain distance of the origin and destination. For bicycle parking, it already has a positive influence if it appears on either side. Bicycle renting facilities, however, need to be available at both origin and destination; at least when the bike sharing system forces the user to return a bike to a station. This applies for most cities nowadays and is included as such in the model.

For parking, the score is based on a 50 and 100 m buffer. If there is a parking spot within the buffer, a score of 1 (100 = 0.5) is awarded. The score is not influenced by the number of parking spots. The 50 and 100 m buffers are based on a paper by Shu et al. (2019) which state that people are willing to walk approximately 250 m to a bike share facility. As there is no data, for bicycle parking it is assumed to be even lower, as parking bikes is typically done right at the destination.

The 250 meter buffer is used for bicycle renting. If there is a renting facility at the origin and the destination as 1 is awarded. As with parking a zero score is used if the criteria cannot be met.

3.5.11 Drinking water

As cycling is a physical activity, it is assumed to increase cycling quality if there is a *drinking water* location along the route. Drinking water locations are places where the city provides drinkable water for free. This is included in the model by buffering the route 50 and 100 m and awarding 1 and 0.5 respectively, if there are spots found. As with parking spots, the number of spots is irrelevant.

3.5.12 Intersections

Another influencing parameter is the quality and quantity of *intersections*. However, the factors influencing the quality of an intersection depend on a multitude of varieties, such as the direction a cyclist comes from, the direction he has to go, the size of the road, traffic lights, etc. To properly evaluate all these parameters is out of scope for this framework, as it would heavily influence the performance. Because of that, the intersections are rated using a simplified approach based on the road types connected to an intersection, the existence of intersection specific infrastructure, namely traffic lights and crossings, and the existence of bus or tram lines. Thanks to this reduction of complexity, many intersections can be ignored, respectively awarded a score of 0. This is the case for all intersections that are only connected to residential roads, paths, and tracks. The assumption is that, generally, these intersections are not improved by implementing traffic lights or crossings as they limit the flow of cyclists. Also, the number of cars is generally low in these areas.

The opposite is true for intersections with larger roads, like artery or connecting roads (tagged *primary*, *secondary*, *tertiary*). Due to the higher traffic flow, crossing them unprotected is not safe. The same applies to intersections in which tram or bus traffic enters. As a result, the rating for these intersections is as follows:

- If traffic lights → score = 1
- If crossing → score = 0.5
- If tram or bus line → score = -1
- If no infrastructure → score = -1

3.5.13 Comparative measures

The Brouter routing engine offers a wide range of profiles. In total, each origin–destination pairs produce three different routes based on different profiles. These routes are then compared and scored. The first different profile is *shortest path* which does exactly what the name suggests: It finds the shortest route to cycle compared by length. Within the cycling literature, a allowed detour rate of 20 % is used (Pritchard & Frøyen, 2019). Therefore, differences below this threshold are awarded zero. Higher differences are greater punished by -0.25; -0.5; -1 for detours of <50 %; <100 %; >100 %.

The same principal is used with the second profile "car", which calculates the fastest path for cars between origin and destination. Here, instead of the length of the trip, the time to travel is compared. This offers a better comparison as cars are to a lesser degree influenced by length relative to speed. Compared to the shortest path, the route can be faster than the cars. In these cases, a score of 1 is awarded. The rating for longer cycling routes is the same as before.

3.6 Implementation details

Within the case study, the inner workings of the framework are explained, and an example city is detailed. Yet, as its goal is intended to be used on a broader scale, namely all large cities in Germany, implementation details like performance, scalability, and ease of use must be addressed. These issues are addressed by implementing a microservice architecture as advised by the best practises from the literature.

3.6.1 Containerisation with Docker

Docker is an open-source software that uses containers to isolate software applications. Containers are made possible by process isolation and virtualization capabilities that is built into the Linux kernel. These capabilities enable multiple applications to share the resources of a single instance of the host operating system. An example are control groups that allocate resources, or namespaces that restrict a processes access or visibility into other resources or areas of the system. This is how hypervisors enable multiple virtual machines to share the CPU, memory, and other resources of a single hardware server. Therefore, containers offer all the functionality and benefits of virtual machines, including application isolation, cost-effective scalability, and disposability.

However, container provide additional other advantages. The typical container is smaller both in physical disk size and in processing requirements. This is because a container only includes the operational system processes that are required by the container. This allows users to run more containers concurrently than full featured virtual machines. Furthermore, contained applications allow applications to be written once and then run anywhere. This also includes Windows via the Windows Subsystem for Linux (WSL) and macOS. Lastly, the start time for a new container is around a single second[22].

A prevailing trend in software engineering is the shift from monolithic applications to smaller finer grained applications, that only deal with one single aspect of the application. This separation of concerns relies heavily on communication over the network using, for example, RPC or RestAPIs. As docker containers are, by nature, isolated, the docker compose has been developed on top of the docker runtime. This allows running multi-container applications. These applications are configured by a single file, whereas details like network access, data preservability, and replication are configured. Docker compose

[22] Docker Documentation (2022. Available at https://docs.docker.com (10.10.2022)

as such runs only on a single machine. Still, the switch from a single machine to nationwide compute cluster based on containerization is facilitated by tools like Docker Swarm or Kubernetes.

The docker technology stack relies on two different parts: images and containers. An image is a zipped archive of the actual application with all underlying technology.

To achieve this in a simple way, images can be built on top of each other. The basis is either a bare Linux-based operation system or another image. After this, all necessary programs can be integrated step by step, including the application code. Finally, the last line specific the command that is executed each time the container gets started. Figure 19 depicts the Docker file used to build the image for the osm2pg image.

```
# The base image we pull.
# We use this because all required gdal libs are already installed.
FROM postgis/postgis

# Update the package lib
RUN apt-get update -y
# Upgrade all packages
RUN apt-get upgrade -y

# Install nessary software
RUN apt install osm2pgsql -y

# Change working directory
WORKDIR "/var/www/osm2postgis"

# Command to be run when container starts.
CMD [ "/bin/bash" ]
```

Figure 18 Exemplary Docker file as used for osm2pgsql

After this setup, process images can easily be shared, distributed, and run anywhere. Images for common applications can also be shared on the internet, on pages like Docker Hub[23]. To run an application the docker daemon starts the images which then become separated images. An image can be run multiple times concurrently.

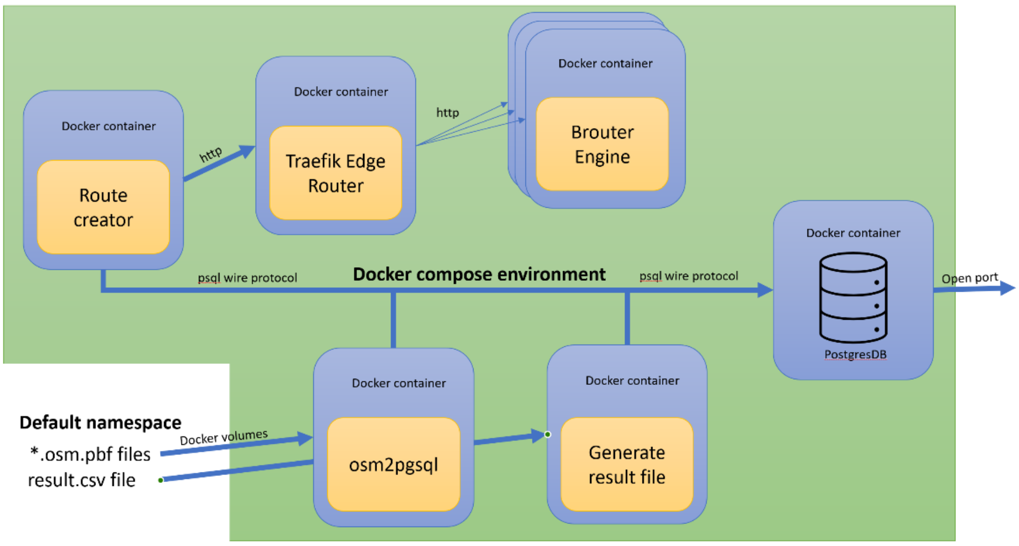


Figure 19 Implemented microservice architecture

[23] Docker Hub (2022): Docker Hub Container Image Library | App Containerization. Available at https://hub.docker.com (10.10.2022)

The implementation build for this work divides the application into six logical separated parts. Two of those are pulled from Docker Hub; the rest is individually designed. The six separate parts of this application are:

- PostGIS: a pulled image running a PostgreSQL database with the PostGIS extension enabled.
- Traefik: a pulled image running Traefik. Traefik is an open-source edge router that is used as a load balancer.
- Brouter: a build image running the Brouter routing engine on top of the ubuntu operation system.
- Go_generate_routes: a build image, running the route generation application written in go, running on top of the alpine operation system.
- Go_result: a build image, writing the final scores for to a .csv file written in go, running on top of the alpine operation system.
- Osm2pg: a build image that inserts the downloaded geofabrik.de extracts into the Postgres database running on the Debian operating system.

This separation aligns with modern microservice architectures that follow the Unix principles. According to this principle, each part should do only one thing but in a good way. Additionally, features, like the route building and the routing engine, are implemented in different programming languages which would require a message bus, when ran in the same application.

3.6.2 Performance

The performance of the model is measured by the time it takes to execute the complete flowchart in Figure 11. The two major time consumers are the Brouter engine (70% of time), and the data inserting via osm2pgsql (30% of time). Both software packages are already optimised for speed; improving them further is out of scope of this paper. However, the former one offers room for improvement, as the application runs on a single thread. Furthermore, it does not pose a problem to generate routes in parallel. This is opposed to the data insertion which cannot be parallelised.

The goal of parallelising the route generation explains the implementation details of the route generation application and the need for a load balancer. Besides, the underlying docker compose offers built-in functions that make this easily implementable. Within the docker-compose config file, a single line determines how many instances of an image should be started as a container.

Within a default docker compose application, all containers share the same network. One container can reach another by using the name specified in the config file. For example, the Brouter engine API is available to other containers under *http://brouter:8080.* The docker runtime makes these namespaces available under the name of the container. Yet, if the runtime spins up more than one container from the same image, the container name changes to brouter_1, brouter_2, … and so does the available namespace. A basic solution would be to hardcode the links into the *go_generate_routes* image, but this limits the ability to modify the software. The number of instances that can be spun up would be predetermined for all users. To solve this, a router is included. This router listens to requests at a certain port and forwards them. In this application the Traefik proxy can detect how many instances of Brouter engines are running and forwards requests accordingly. Traefik is an edge router, written in go, that provides easy integration within docker compose[24].

```yaml
# Version of docker compose used
version: "3.9"
# List of all used services
services:


  traefik: # Name of the image / container
    # Prebuild image version 2.8 pulled from hub.docker.com
    image: "traefik:v2.8"
    # Parameters specifiying behaviour
    command:
      - "--api.insecure=true"
      - "--providers.docker=true"
      - "--providers.docker.exposedbydefault=false"
    # Ports open outside the docker application
    ports:
      - "80:80" # Routes to brouter
      - "8080:8080" # Used to monitor load
    # Allows access to socket.
    volumes:
      - "/var/run/docker.sock:/var/run/docker.sock:ro"


  brouter: # Name of the image
    build: brouter/. # Refers to Dockerfile in subfolder brouter
    # Used to share .rd5 files with Windows File System
    volumes:
      - ./brouter/segments4:/var/www/brouter/misc/segments4
    # Parameters specifiying behaviour
    labels:
    - "traefik.enable=true"
    - "traefik.http.routers.brouter1.rule=Path(`/brouter`)"
    # Specifies number of instances.        You, vor 1 Sekunde • Uncommit
    deploy:
      replicas: 3
```

Figure 20 Docker-compose file excerpt

[24] Traefik Labs (2022): traefik proxy. The Cloud Native Application Proxy. Available at https://traefik.io/ (10.10.2022)

The *go_generate_routes* image always sends requests to the proxy and is unaware and uninfluenced by the total number of engines. This service is written in go a language with special features to facilitate parallelism[25]. This is archived by goroutines, which are light-weight processes that communicate over channels. Goroutines are needed because if the server sends one request to the routing engine, it does not block until the route is returned, which would leave all other instances of the routing engine idle. This can be implemented in other languages as well but requires some hoops to jump through. Additionally, a Postgres driver library for go offers the use of connection pooling. This allows for the reusage of database connections and the parallel inserting of routes. Establishing a connection is a heavy workload, so this further improves the performance. Lastly, go is a statically typed, compiled language which makes it fast compared to other obvious contenders, like Python. A proof of work was implemented in python which took twice as long.

The *go_results* container is also implemented in go. However, it is not performance-critical, as the time to execute is spend within the Postgres database.

3.6.3 Automation

To allow easy access, the complete workflow (except file download) for a single city can be invoked by executing a provided *.bat* script (windows only), with the only requirement that docker and docker compose are installed. A single python script is also included that generates data for all cities within a provided file. Furthermore, the software is written as such that parameters like the number of routes, the rate between special locations and normal buildings, the maximum route length, and the minimum route length can be changed in a central config file. The code is also made available at the GitHub repository[26] with an unrestricted MIT License.

3.7 Calculating the final scores

All presented parameters need to be combined to get the two searched metrics. Due to their different impacts on cycling quality, all variables are weighted. This is based on surveys pulled from literature, like, for example, Grigore et al. (2019), and assumptions for cases where no literature was found. Each variable is weighted with a score between 1 and 10, and the sum of these products is divided by the sum of weights. As this only serves the purpose of a concept model, no deeper sensitivity analysis is provided.

[25] Go Documentation (2020). Available at https://www.go.dev/ (17.10.2022)
[26] Wendel (2022) Available at https://github.com/timahawk/bikeability (17.10.2022)

3.7.1 Formula to calculate the segment scores

The segment score *ss* of segment *i* is based on the following parameters:

- Cycling infrastructure *ci*
- Speed limit *sl*
- Surface type *st*
- Number of trees *nt*
- Parks *pa*
- Forests *fo*
- Bus routes *br*
- Tram routes *tr*
- Permissions *pe*
- Parking *pa*
- Illumination *il*
- Straightness *sn*

These are combined to the following equation:

Equation 1 Individual segment score

$$ss_i = (ci * 10 + sl * 7 + st * 7 + nt * 3 + pa * 4 + fo * 4 + br * 6 + tr * 10 + pe * 1 + pa * 7 + il * 3 + sn * 2) / 64$$

3.7.2 Formula to calculate the route scores

The route score *rs of* route i is based on the following parameters:

- Segment score *ss*
- Bicycle parking *bp*
- Bicycle renting *br*
- Drinking water *dw*
- Intersections *is*
- Comparison of distance route and shortest route *sr*
- Comparison of cycle time and car time *ct*

The segment score for a route is calculated based on the sum of all segments weighted by their length divided by the length of the route. To clarify, the longer the segment, the greater the influence on the segment score.

Equation 2 Route score

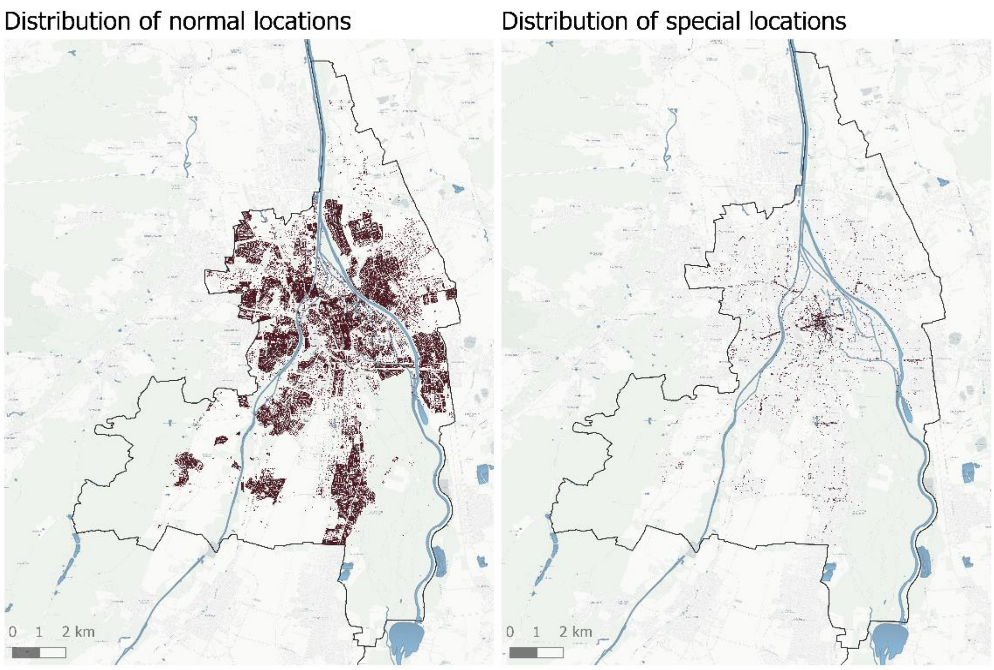
$$rs_i = (\sum_{seg} \frac{(seg_{ss} * seg_{len})}{r_{len}} * 10 + bp * 4 + br * 4 + dw * 2 + is * 5 + sr * 4 + ct * 4) / 28$$

The final city-wide score is calculated by averaging all route scores, independent of their length. The length of a route does not influence the final score, because that would give higher priority to locations that are far away.

3.8 The proof-of-concept model for Augsburg: results

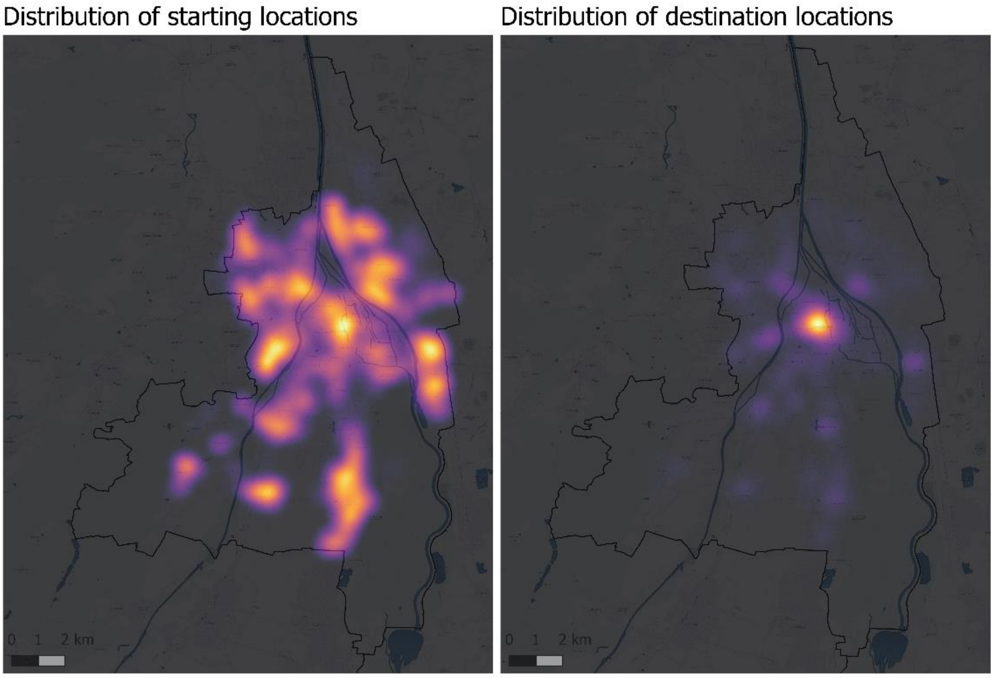
The following section will present various maps and statistics about the inputs and outputs of the framework.

3.8.1 Distribution of various input parameters



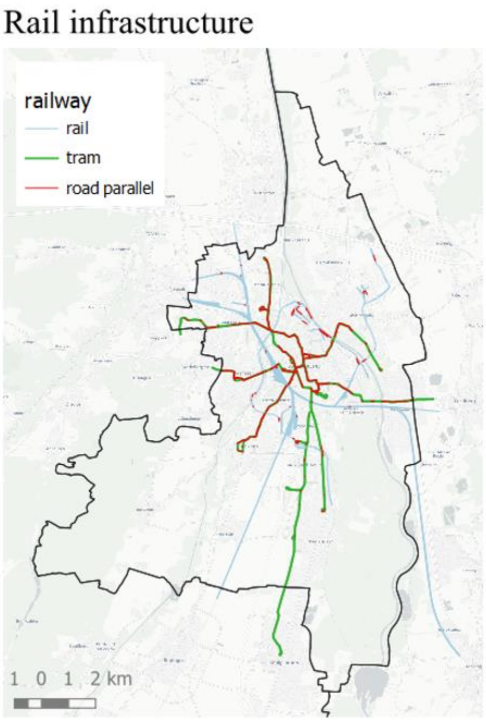
Map 1 Distribution of possible locations within the study area, split in buildings and offices, shops, and restaurants

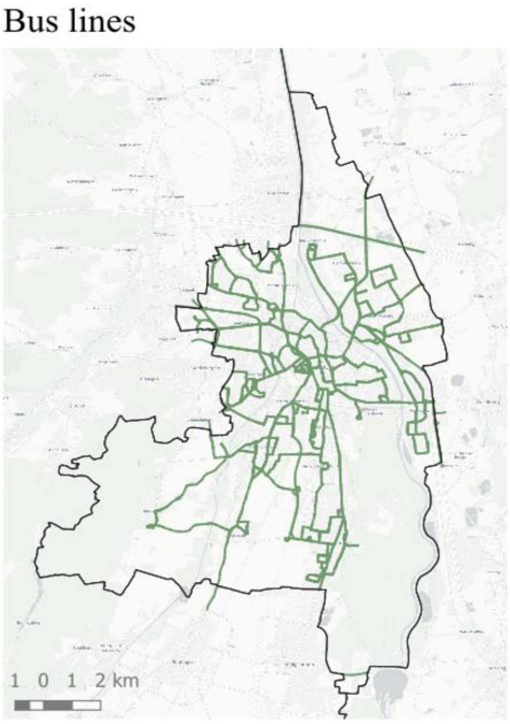
Map 1and Map 2 depict the distribution of the origins and destinations as well as the ones that were actually used. On the left side, Figure 1 displays the spatial distribution of all 3,887 locations which do have a special purpose. The right side depicts the centre point of



Map 2 Distribution of used starting and destination points within the study area

all 49,630 polygons tagged as building. Both sides can be used as destinations. Still, origins are always normal buildings. Map 2 shows the different distribution based on the semi-random approach introduced before. The origins are spread evenly across the study area, whereas the destinations are mainly grouped in the city centre and, to a lesser extent, in areas along larger roads that are sprinkled with shops.



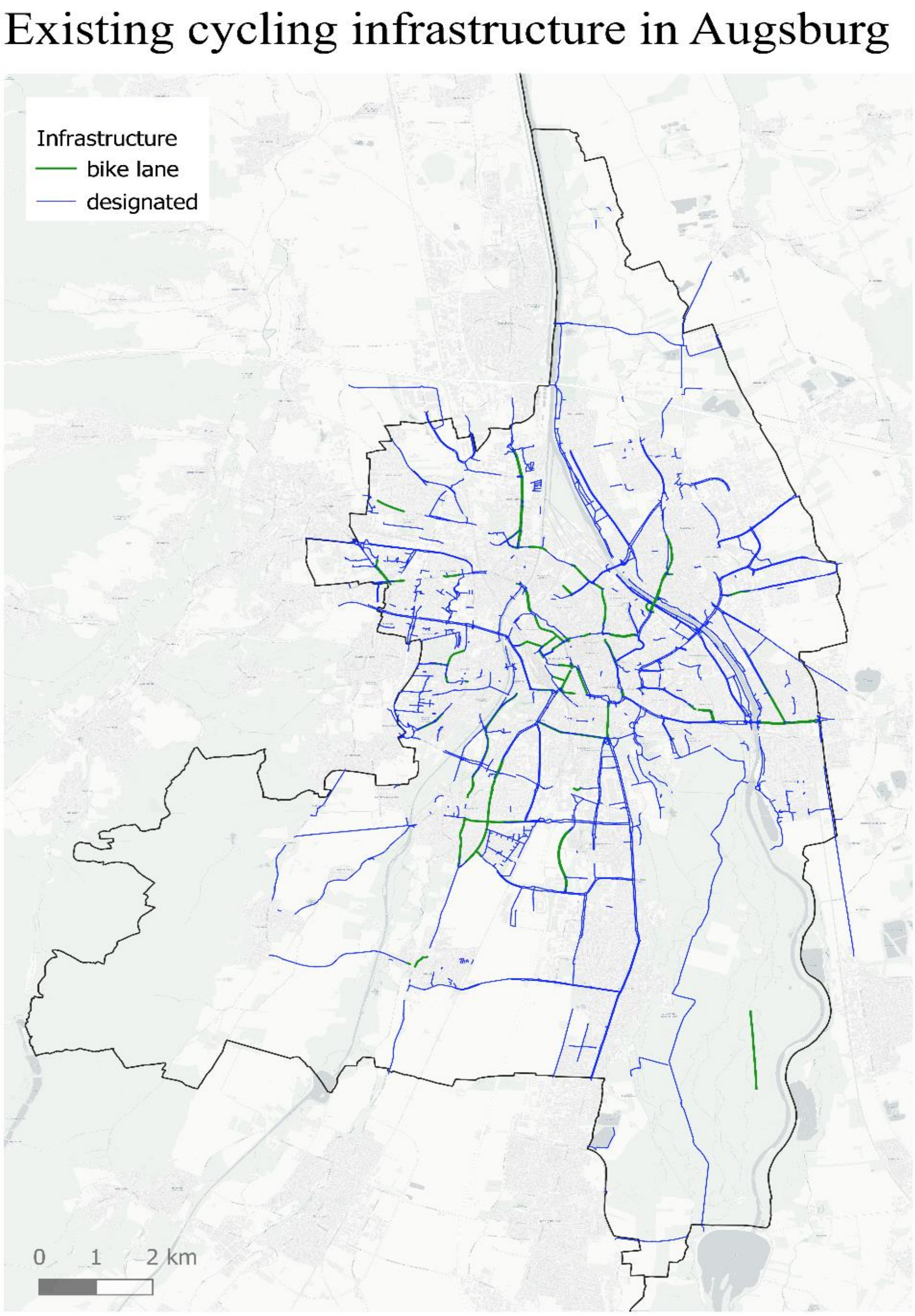


Map 3 Rail infrastructure in Augsburg **Map 4 Bus lines in Augsburg**

Map 3 shows the rail infrastructure in Augsburg. Light blue represents classic train tracks which typically do not run parallel to a road segment. Tram lines are depicted in green. Red areas symbolise trams that have been identified to be parallel to the road.

Map 4 depicts all bus lines. They are always on top of a road. The map, however, cannot show if there is an extra road only for busses.

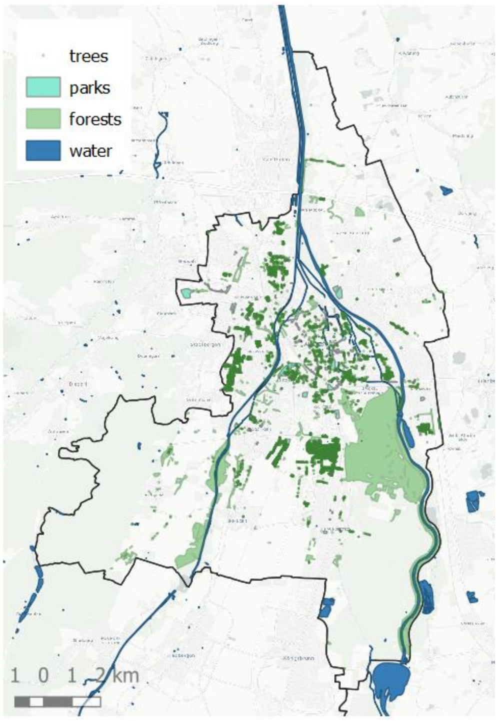
Map 5 shows the existing road-based cycling infrastructure in Augsburg. Blue lines represent designated cycle infrastructure, whereas green shows painted bicycle lanes. The total length of designated cycle lanes is approximately 300 km, that of the cycle lanes 35 km. The total network length, including paths, driveways, and alleys is 3,475 km.



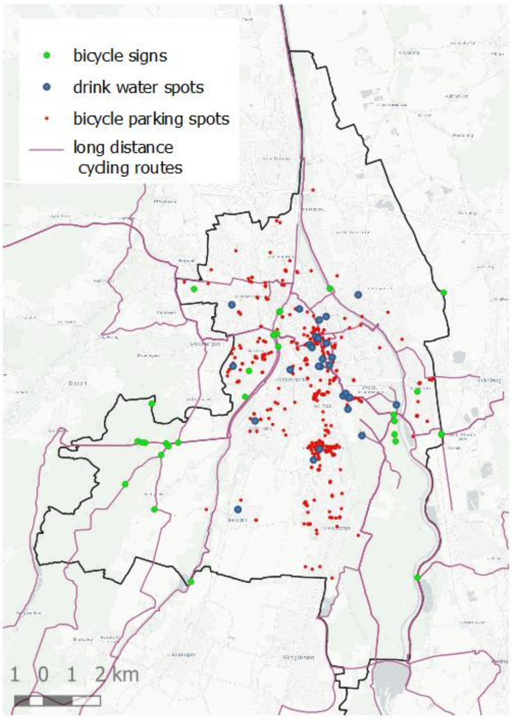
Map 5 Cycling infrastructure in Augsburg

Map 7 depicts all parameters related to natural features. This includes trees, parks, forests, and water thus, rivers, lakes, and basins. Trees are distributed irregularly through the city, which does not represent the real world. These clusters occur, because in these areas somebody tagged all available trees. This is the case for the University Campus and throughout most of the Universitätsviertel, as well as for new city parks, like the Sheridan Park. In other parks like, e.g., the Wittelsbacher Park, only one or two trees are tagged.

natural features



other influential parameters

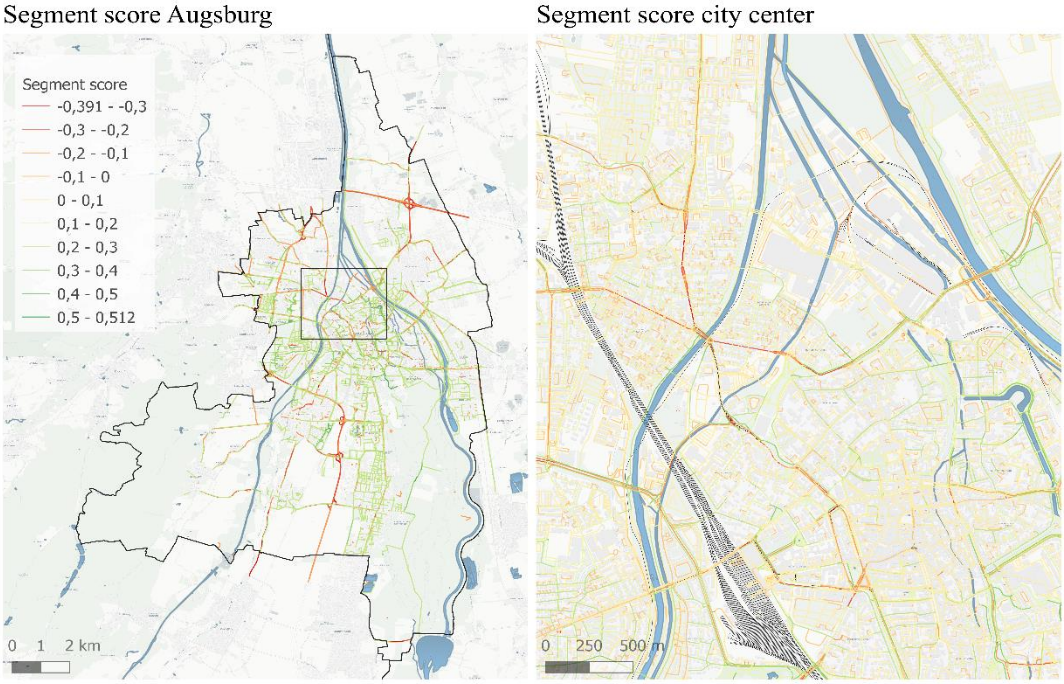


Map 7 Greenery in Augsburg

Map 6 Signs, routes, and drinking water spots

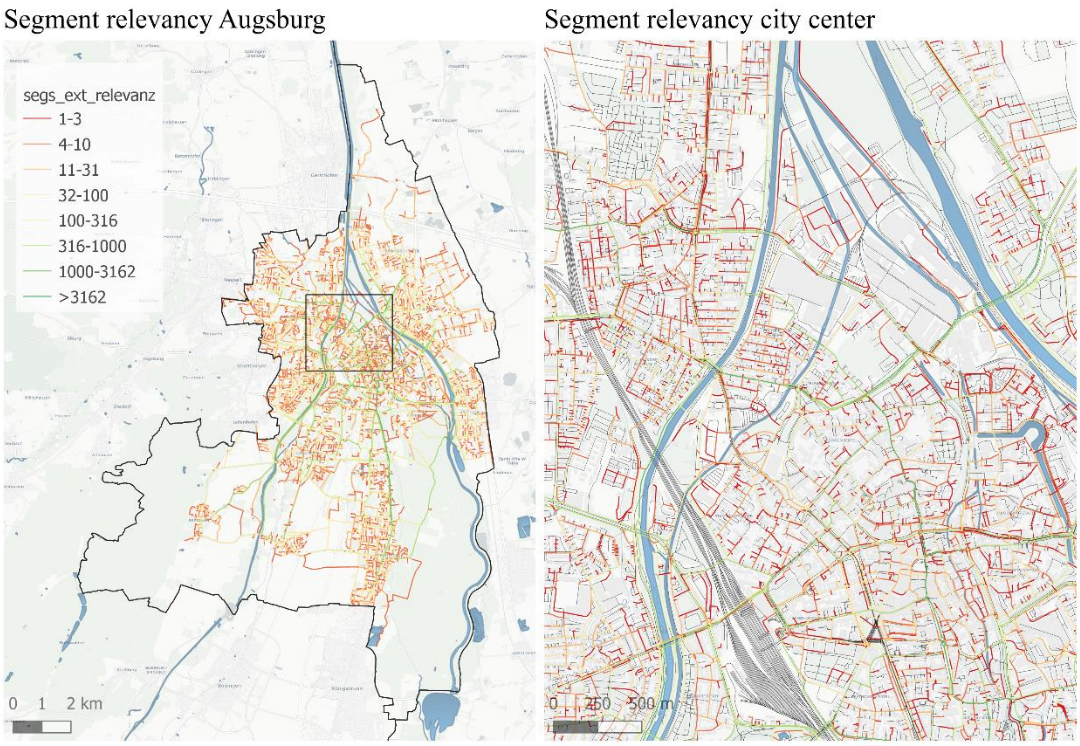
Map 6 displays four different parameters. As can be seen, most bicycle signs are tagged on top of long distance cycling routes. Water dispensers are mainly found in the city centre, near the Maximilianstraße. Equally to the distribution of trees is the distribution of bicycle parking spots. They are only tagged in limited areas. Additionally, there are some larger apartment blocks, where each entry has a tagged bicycle parking facility. At the next apartment block, there is not one single-tagged parking facility.

3.8.2 Distribution of the segment scores for Augsburg



Map 8 Individual segments categorised by score

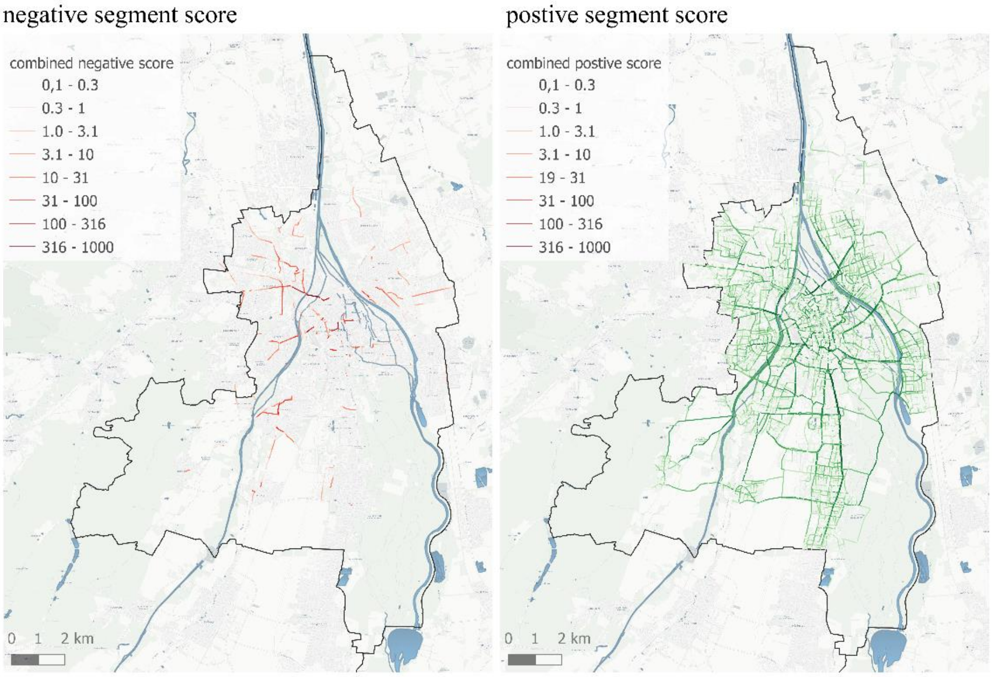
Map 8 shows the individual segments scores. On the left side, the complete study area can be seen. The right side is a narrowed section of the city centre and the southern parts of



Map 9 Individual segments categorised relevancy in road network

Oberhausen. To offer a cleaner visual, road sections with ratings between -0.1 and 0.1 are not shown in the larger map section.

The relevancy of each segment can be seen in Map 9. The described relevancy is the summed up number of routes that use this system. For clearer visualization, segments that are used by less than 3 routes are not shown in the larger section.



Map 10 Road segments weighted by relevancy and score

The combined score for each segment is shown in Map 10. It is the fusion of Map 8 and Map 9, obtained by the multiplication of both values. The result is presented in two maps. The left one presents segments, that have a negative score, whereas the right one presents positive values. Both are colour graded, using a logarithmic scale. As this works only with positive values, the negative scores are multiplied by minus 1. This also explains why the legend depicts positive values in both cases.

The previous maps showed the results for the individual road segments. Table 1 presents the summed up combined route score separated into each subcategory.

Parameter	Value
num. routes	9473
permission	-0.001536632
surface	-0.037957546
max. speed	0.283213571
infrastructure	0.45748339
parallel rail	-0.086140882
parallel bus	-0.216575984
straightness	0.184974067
time route vs. car	-0.101961691
cycle route vs. shortest route	-0.011123094
long-distance cycling routes	0.268858856
bicycle parking	0.157705541
drinking water	0.411004981
segment straightness	0.974386312
trees	0.019519964
forest	0.012636701
park	0.067575932
parking	0
illumination	0.475807894
final score	0.18416173

Table 1 Summed weighted route score for Augsburg, split by parameter

3.9 Summary of the results for all analysed German cities

The following table presents an excerpt for all cities between 200,000 and 600,000 inhabitants in Germany. The scores present the five most influential parameters according to formula 2. Additionally, the number of routes that were successfully created is shown. The number can be subtracted from 10,000 to get the sum of failed route generations. The table is listed in descending order of the final score.

	num. routes	infrastructure	max. speed	surface	parallel rail	parking	final score
Bremen	9451	0.682075398	0.23915953	-0.020116463	-0.004763035	-0.00293434	0.256568824
Mannheim	9855	0.579804563	0.252631246	-0.017102098	-0.008726913	0	0.225958603
Münster	9496	0.613446622	0.190960748	-0.014487315	-0.000590568	0	0.221687229
Lübeck	9324	0.541887818	0.240868547	-0.015409942	-0.000486362	0	0.220494989
Karlsruhe	9461	0.371730079	0.330616076	-0.01041931	-0.012280512	-8.9532E-05	0.21972005
Freiburg	8994	0.514578268	0.347513923	-0.012674103	-0.015499883	-2.7609E-05	0.219171085
Kiel	9810	0.59647834	0.22454674	-0.038536306	-2.24564E-05	-0.00022094	0.211127756
Braunschweig	9632	0.56861067	0.172381456	-0.032322176	-0.00831153	0	0.203733535
Nürnberg	9870	0.463289397	0.305352539	-0.019232743	-0.015311181	-9.2816E-05	0.196807057
Rostock	9450	0.551395957	0.218888314	-0.027630125	-0.003788265	-1.2791E-05	0.195435176
Bonn	9799	0.448987002	0.315874197	-0.014599925	-0.013462732	0	0.190098562
Augsburg	9473	0.45748339	0.283213571	-0.037957546	-0.086140882	0	0.18416173
Magdeburg	9807	0.430697072	0.213965637	-0.026240841	-0.00937324	0	0.183615321
Essen	9838	0.314961452	0.388420752	-0.009360019	-0.02190147	-0.00013839	0.172520947
Mainz	9603	0.310841267	0.314485216	-0.009826434	-0.011892234	-0.00038261	0.17065245
Dortmund	9819	0.306343552	0.311066449	-0.044225027	-0.006675758	0	0.170644496
Kassel	9824	0.198765642	0.416474856	-0.035749124	-0.011074499	-6.3119E-05	0.159204779
Hannover	6725	0.421026057	0.165168029	-0.05561756	-0.004473402	0	0.153800509
Bielefeld	9816	0.270215915	0.237765695	-0.041353328	-0.023438975	0	0.145482083
Bochum	9879	0.192338287	0.334180534	-0.011204778	-0.039246407	-9.9509E-07	0.14349192
Oberhausen	9846	0.403347009	0.214296451	-0.069214413	-0.001886398	0	0.142934898
Dresden	9655	0.194349449	0.288597491	-0.017431517	-0.059736271	-0.00013526	0.139484537
Duisburg	9701	0.399450877	0.218447688	-0.077866381	-0.002112981	0	0.138639664
Aachen	9062	0.272451064	0.305452693	-0.053278636	-0.001224971	0	0.133615691
Erfurt	9615	0.247990818	0.20846591	-0.056201075	-0.025917925	0	0.131633021
Mönchengladbach	9768	0.310418025	0.243735641	-0.110193262	-0.001091977	-0.00058018	0.127517135
Halle	9757	0.24179458	0.198724408	-0.031033828	-0.044486371	0	0.127309502
Krefeld	9880	0.156470935	0.399665482	-0.064485644	-0.065118714	0	0.122726162
Gelsenkirchen	9781	0.162775648	0.239515023	-0.058414465	-0.023126131	0	0.114052667
Wiesbaden	9695	0.031368309	0.262628233	-0.037114875	-0.000733237	0	0.085664849
Chemnitz	9643	0.038843785	0.267569495	-0.030013852	-0.008693292	-0.00651708	0.082195534
Wuppertal	9711	0.02338001	0.259688874	-0.022545894	-0.000728644	-0.00016412	0.078968889

Table 2 Bikeability index for all German cites from 200,000 to 600,000 inhabitants

4 Discussion

The next section will discuss trade-off decisions that have influenced the framework. The advantages and limitations of each trade-off are discussed, including a reasoning for why they had to be made.

4.1 Explaining the implementation details

When designing a software architecture for a new project, engineers can choose between two major styles for their project, SOA and microservices. Each comes with certain advantages but also disadvantages. Whilst the technological challenges within the framework are limited, and don't necessarily require horizontal scaling as such, being able to do so is no disadvantage. Software architecture should never be neglected. Therefore, both major architectural styles, have been researched. In the end, it was decided to implement the framework using the microservice pattern. The reason for this is the easy scalability, the independence between services, and the technical freedom. By embracing the container technology, services can be scaled independently, which is already done with the routing engine container. As the Brouter is the cornerstone technology that makes the framework function, implementing SOA would also dictate further design processes. The database would be unaffected by this, but the route generator and *final result* container, which are currently implemented in go, would have to be implemented in Java; Java and the Java virtual machine are the technology that powers the engine. From a technological standpoint, implementing both services in java is possible. However, the language lacks useful features that were used such as goroutine or channels. Additionally, SOA is usually implemented on a company scale, instead of on a project scale. Further advantages are the ease of changing individual containers, like switching from Traefik to Nginx, that can be implemented in minutes. Finally, it is also a question of cost and reproducibly. Most of the technology powering microservices is free and open source. This includes Docker, docker-compose, as well as the images like Traefik or Postgres. SOA, in contrast, requires an Enterprise Service Bus, which is a very complex piece of software and only available for paying customers. Therefore, the framework cannot be easily implemented, and the results cannot be reproduced. This would not follow the open requirements and therefore fail one of the main previously stated goals.

One of the key requirements of the framework was performance, which was always an equal consideration when deciding on the design. This relates to both how parameters are implemented (see chapter Straightness) and how and which software components are used. Running on the author's setup, the average time it takes to process the complete workflow for a city is between 20 and 30 minutes. This does not include time to download the datasets or building and pulling images. The author's system is based around an AMD Ryzen 5 3600X 6-core processor, running 12 threads. Attached are 32 GB of DDR4 memory, running at 3000 MHz and a 500 GB SSD. The question that arises is if the focus on performance is even necessary. Running the model for all cities within the 200,000–600,000 inhabitant interval takes approximately 14 hours. It could be argued that letting the PC run for another 10 hours is worth it if the results are more accurate/better and the code is easier to understand. However, when it comes to scalability, this argument must be reconsidered: The framework aims to be used for all major cities throughout Germany and similar countries. For Germany alone, calculating the score for all cities with over 50,000 inhabitants adds 153 new cities to the list. This would either take 2.5 or 5 days. This is a significant difference and cannot be dismissed. At the same time, there are many ways to improve the performance that were not used, because it would limit clarity and quality. Even the presented possibility of spawning multiple routing engines was not pushed to the limit, as can be seen in Figure 21. The average CPU load is about 50 %. By deploying more Routing engine, this number could be easily increased. Even if more performance is necessary, the system is designed to be deployable into larger, possibly cloud-based, systems, which would improve performance drastically. This must be recognized as a key strength of the framework.

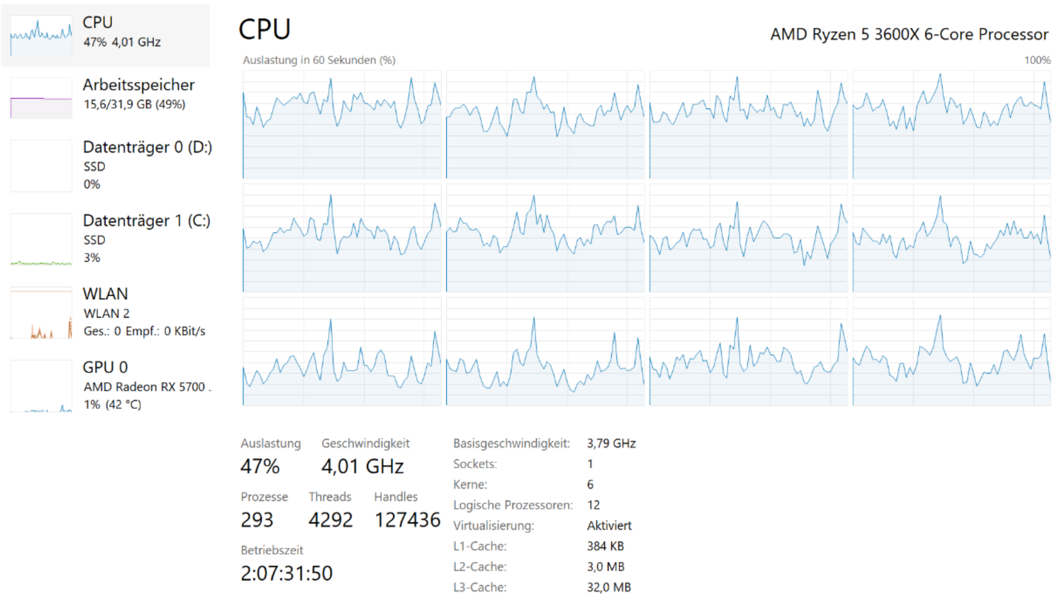


Figure 21 CPU usage during model run on a Ryzen5 3600X processor

Affecting scalability, there is, however, a design decision that should be changed from an implementation standpoint and probably as well from a bikeability index standpoint. The current implementation is based around **.osm.pbf* extracts from geofabrik.de. These come either at a state or region level and are therefore quite large (100-350mb). Each time a new

city gets implemented; the extract that contains that city is uploaded completely to the database. This means, in areas like the Ruhrgebiet, where one large city is next to another large city, the underlying data is inserted multiple times. Only the views and generated tables change. This decision was made because it eases the use for people that only want to use the framework for a single city. When evaluating multiple cities, the database can grow exponentially. During this work, the database grew to a notable 57 GB in disc size. After the framework run finishes, the typical table size is around 30 MB for the routes, and 200 MB for the road segments but 250 MB for lines, 100 MB for points, and 400 MB for polygons, which are 1) mostly outside the study area, and 2) probably even duplicated.
Related to that is a secondary question. Is it even necessary to physically safe all results to disc? Thanks to new tools, like pyrosm[27], the complete score calculation could be performed within python notebooks, powered by technologies like numpy[28], pandas[29], and geopandas[30]. The performance benefits would be limited to the final calculations, but as those tools operate entirely within memory, once the calculations are finished, the disc size would not be impacted. For the framework, the decision was made for a Database back-end, for the simple reasons, that data within Postgres can be visualised with tools like QGIS. Moreover, it allows to calculate modified city or segments scores for all cities in a fast and accessible manor thanks to the powerful query engine in Postgres and the stored data. Furthermore, thanks to the container architecture deleting all related data, it is as simple as deleting a container.

The last major implementation decision that needs to be discussed is the choice to use county borders to outline routes within. When it comes to an automated approach, there are two possibilities to find a relevant area of interest.
The first is to use the borders of a city. This has the benefits that they are a) predefined, b) align exactly with the **.osm.pbf* extracts, and c) represent only the area of interest. There is, however, a major drawback along with a smaller one. The big disadvantage of this approach is that surrounding villages are not included in the route-finding process. This leads to a lower number of trips along the edge of study area. This can be ignored for smaller villages in less urbanised areas. In the case of Augsburg this could be places like Derching or Stätzling. It is less tolerable for bordering satellite cities, like Stadtbergen or Gersthofen. They are already fused, so that people don’t even know in which city they are now. Still, they pose are relatively low population size. Where the decision needs to be

[27] Pyrosm (2022). Available at https://pyrosm.readthedocs.io/en/latest/ (14.10.2022)
[28] NumPy (2022). Available at https://numpy.org/ (14.10.2022)
[29] Pandas (2022): Python Data Analysis Library. Available at https://pandas.pydata.org/ (14.10.2022)
[30] GeoPandas (2022): Available at https://geopandas.org/en/stable/ (14.10.2022)

questioned outright is when looking at “twin-cities” like Mainz & Wiesbaden or Mannheim & Ludwigshafen (or probably the complete Ruhr region). There are no numbers available, but the bridges over the Rhine should be one of the most used roads in both cities. Therefore, a high emphasis should be put on these and surrounding roads. The presented approach has zero trips over any bridge and, as such, fails to model the feasible situation. The other extreme is a too large study area. The locations are to far apart that the routing engine can generate a allowed route (within the interval of 10 km straight line distance). Even for Augsburg, which is compact in its district size, villages like Bergheim or Wellenburg are included. In Map 9, it is visible that the road to Bergheim is used by many routes. Yet these roads are not relevant to measure bikeability in Augsburg. An extreme example for this would be the city of Hannover. Hannover and Aachen are two edge cases within the German district system, as their district area is bigger in size and includes more surrounding area. The “Region Hannover” (and also the Städteregion Aachen) is nearly 10 times larger than Augsburg with only one fourth of the population density. The influence of non-urban areas in the results is therefore much higher. On a sidenote, this explains the unexpected low number of trips in the area. The places were simply too far apart to be allowed into the result set.

The alternative would be to either buffer the complete district or an area around the city centre. In both cases, the complete workflow would have to be rewritten for all cities that border a state or a region. Be that as it may, the bigger challenge would be in determining the size of the buffer. If the decision would be to use the district, even by glancing at the borders of Augsburg, problems arise immediately. For the northern part of Augsburg this approach would make sense. However, buffering the south to include towns and villages, like Meringen or Oberottmarshausen, would just lead to the same problems that occurred in Hannover.

The second approach of using the city centre would result in the same problems. Also, there would be the added problem of finding the city centre. A 5 km buffer might work for Augsburg, but it could be too large for smaller cities, and too small for larger cities. Attaching the buffer size to the population size might be an interesting idea, but again, looking at Augsburg, it should be clear that cities are never perfectly round-shaped. The spread of a city is always influenced and sometimes limited by the physical environment. In the case of Augsburg, the city is laid out along the two dominate rivers Lech and Wertach, and the topology they formed. This led to a longer spread in the north–south direction and, consequently, a shorter east–west spread.

The last possible option would be drawing handcrafted borders for each study area. For Augsburg that could mean that Stadtbergen and Gersthofen are included, but Friedberg is excluded; the university district is included but Haunstetten is considered too far away. It should be clear that this is the exact opposite of an automated framework. Additionally, these custom polygons would also never be universally accepted but always be influenced by the opinions of the creator. For all the mentioned arguments, the framework uses the district borders, even though there are some cases where these borders provide unrealistic results.

4.2 Data quality and temporal relevance

Nevertheless, unsatisfying is a term that should not be associated with data quality. To offer a meaningful result, a model needs adequate and extensive base data. Added to the immense importance of OpenStreetMap data in research as well as in the private sector, many studies have been conducted to quantify this. However, as the underlying data pool spans worldwide, labelling OSM data as simply good or bad is of no use. Researchers like Borkowska and Pokonieczny (2022), Brückner et al. (2021) or Yeboah et al. (2021) conclude that data quality and data availability vary. The argue that the more urban and industrialised an area is, the better the quality and availability. This is also influenced by socioeconomic factors, like gross domestic product, unemployment, and proportion of academics. The results fit well into the proposed framework as all areas are large cities. Additionally, these assessments are highly influenced by what type of data they are comparing to a ground truth. For larger features, like buildings, roads, and rail, the data is nearly prefect. Brückner et al. (2021) found areas in which shops were 100 % accurately tagged. For smaller features, this is only true to a lesser extent. It aligns with the question of what a data availability of 100 % for OpenStreetMap even means. Based on the unrestricted and modular key-value tagging system, there are thousands of physical features that could be inserted into the dataset. To clarify, a simple playground can be regarded. Expecting a complete OSM dataset, there should be a polygon representing the size and location of the playground. Yet, there are over 40 different types of devices or installations, subdivided into three categories, that can be tagged within the area[31]. Is the OSM dataset 100 % accurate if the sandpit or the swing are not tagged? If the climbing wall is identified as a climbing frame in the OSM dataset, does that mean the data quality is 100 %?

Reflecting on the framework, the question is simpler. Are all parameters included within the model 100 % accurate and 100 % available? The answer must clearly be no. This is easily proved by Map 7 Greenery in Augsburg. There are vast spatial disbalances between

[31] OSM Contributor (2022). Available at https://wiki.openstreetmap.org/wiki/Key:playground (17.10.2022)

locations even within city districts. To solve this, the model is built as such that if data is unavailable for a road segment, it or the city score are not negatively influenced when data like the number of trees or number of parking spots is missing. However, there is an exception to this approach: bike infrastructure. This parameter is of such importance that the score is punishing for nonexistence, but the general availability is good enough to rectify this.

By comparing the cycling infrastructure as provided via the geoportal (Geoportal Augsburg, 2022) from Augsburg, the general route network seems identical. Sadly, the Tiefbauamt Abt. Verkehrsplanung was not able to provide the necessary base data for a deeper analysis. The data provided overlaps both within the location and mostly within the category. The data from Augsburg offers the additional category "Mitfahren im Verkehr" which is a not tagged in OSM and even subdivided into speed limits of the road, and the special category "unkomfortabler Belag" (uncomfortable surface). To summarise, the data availability and quality for this important factor is very good within the study area. This is also the case for all underlying data, namely roads, buildings, and rail infrastructure. Basically, everything that is represented by more than a point, or attached to something else as a tag is complete. Smaller features like drinking water spots or bicycle parking and attached tags, like permissions, illuminations, or surface type are not.

The OSM dataset still offers another advantage namely temporal relevance. This is related to how quickly changes in the real world are reflected in the data set provided. Again, this is only observed for cycling infrastructure. Here it can be shown that the OSM dataset is the fastest when comparing it with the official website and Google Maps. During the pandemic, many cities have implemented pop-up cycling infrastructure throughout the city. This is now coming to Augsburg, with a two-year delay. Pop-up cycle paths are mostly painted bike lanes that are created without an extended planning process. The space for this was created by reducing the number of car lanes. An example can be seen in Neuburger Straße in Lechhausen, Augsburg. Neuburger Straße used to be a four-lane artery road (two lanes in each direction), connecting the city centre and the industrial quarter of Augsburg East. Now, the road is a single lane plus a marked bike lane. The bike lane is therefore very wide, except at the intersection Neuburger Straße–Klausstraße, because cars get an extra turn lane. The changes were implemented around July 21, 2022 by the city council (Pressemitteilungen Fahrradstadt, 2022). One and a half months later, these changes could be found solely in OpenStreetMap. This is not to say it took OSM that time to update the data but the author to notice and look up the situation. In Google Maps, however, the change of the road type was not updated at that time.

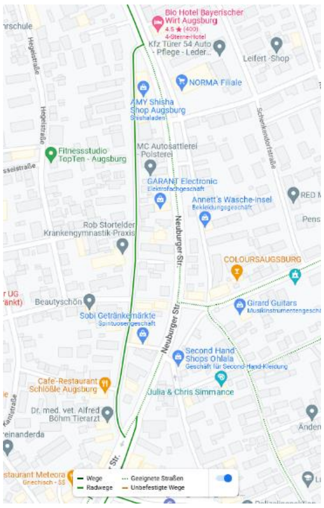


Figure 24 Screenshot of Google Maps

Figure 23 Screenshot of Cyclosm

Figure 22 Screenshot of Geoportal Augsburg

Figures 22[32], 23[33] and 24[34] depict the same road on September 9, 2022. The relevant street is the large north–south bound road in the middle of the images. Google Maps labels the dotted green line as suitable for cyclists. The Geoportal does not even mark any cycling infrastructure there. Except in the southern part, it is labelled as drive along the traffic, at 50 km/h. Only the Cyclosm map, which is based on the OSM data, depicts the situation as it really is – two painted cycle lanes.

4.3 Brouter – Google Maps

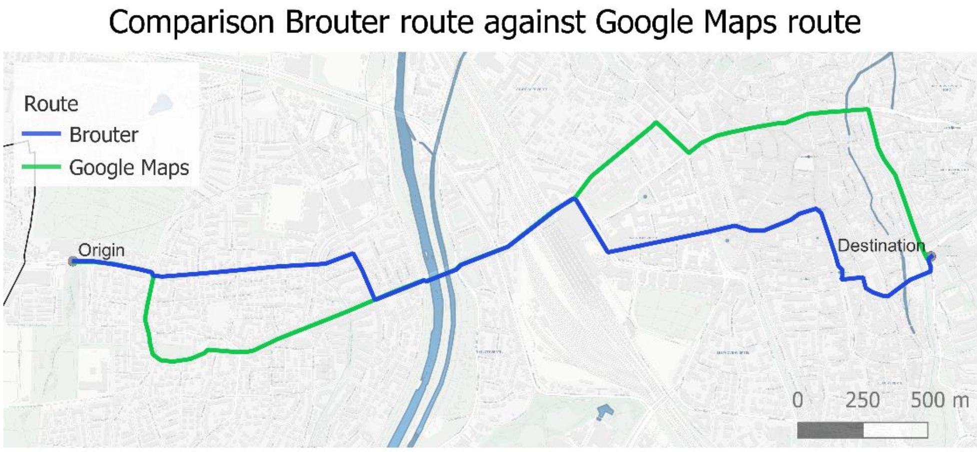
As there are differences in the underlying data sources, there are also difference when it comes to routing algorithms. The herein employed Brouter engine is not well known throughout Germany, and most referenced within the limited group of gravel bike cyclists. The average consumer immediately associates routing with either Google Maps or Apple Maps. The real market share for these companies is kept hidden as a trade secret. There are, however, some valid assumptions that were made throughout the years. These indicate that Google Maps has a market share between 70% and 80%[35] [36].

A cornerstone of this framework is that people either cycle like a routing app tells them,

---

[32] Google Maps (2022): Available at https://www.google.com/maps (09.09.2022)
[33] Cyclosm (2022): Available at https://www.cyclosm.org (09.09.2022)
[34] Geoportal Augsburg (2022): Available at https:// https://geoportal.augsburg.de (09.09.2022)
[35]Beirne, Justin (2021): How Many People Use Google Maps Compared to Apple Maps? Available at https://www.justinobeirne.com/how-many-people-use-google-maps-compared-to-apple-maps (11.10.2022)
[36] The Manifest (2018): The Popularity of Google Maps: Trends in Navigation Apps in 2018. Available at http://themanifest.com/app-development/trends-navigation-apps (11.10.2022)

or that a routing app can reproduce the routes people take. Naturally, this cannot always be the case because every person is different and so are their routing choices. However, this rational can be argued for, based on three main points. Firstly, the general spatial awareness is declining due to tools like Google Maps (Ishikawa et al., 2008; Willis et al., 2009). This leads to more people needing an app to navigate. Secondly, even though apps use only a certain number of parameters, thanks to long standing research in navigation behaviour, these overlap with the routes people use. Lastly, the vast data input that google has needs be mentioned. Due the widespread adoption of Android phones and its market share within the mapping segment, Google has access to a vast data pool of how people travel. Assumed, Person A wants to visit Person B using their bike. They agree on a meeting time of 6 pm. Person A knows exactly where Person B lives and how to get there but is just missing the information on how long it will take to get there. So, Person A opens their phone, enters the origin and destination, and receives the information from Google that they reach the destination within 10 minutes. Person A calculates their travel time with 10 min but chooses



Map 11 Comparison of a Brouter route with generated Google Maps route.

the known route, disregarding the one proposed by Google Maps. Google, as a result, has the following information: a start-destination pair, a proposed route and the chosen route. If scenarios like these happen regularly, why should Google not adapt future road predictions to the roads that people really cycle? Not being provided with the best possible route results in more and more people that are defecting to competitors. Google is a billion dollar company which has the technology and motivation to implement this. People using their services, including Google Maps, is their core business. Again, this is speculative, as Google wants to stay a billion dollar company and does not share trade secrets freely. Based on the points above, using the Brouter engine is only a valid choice if they propose routes that are nearly identical to those by Google Maps. This was evaluated using 100 origin–destination pairs and querying a provided API for each service. The resulting routes were overlayed, and the percentage of overlap was measured. The result is the mean overlap of those 100 routes. A comparative map between the two routes is provided as an example. The route originates in Pfersee, an area outside the city centre in the east of the Wertach. The destination is the *Citygalerie Augsburg,* a shopping mall residing in the middle of the town, located at a square which also hosts large office buildings and a school. The trip is 3.6 km long as the crow flies and represents a typical shopping or commuting trip. The discrepancy in this example is above average. The average similarity between two maps is about 74.8%. The route is around 500 m shorter and uses more cycle infrastructure. Google Maps seems to propose the same route that is optimised for cars. The likely reason, at least for the divergence in Pfersee (east of the river), is that Maps applies car rules to cyclists, whereas Brouter does not. The four-way intersection at the eastern divergence is labelled, so that cars coming from the east can only go south (the main direction of the road, and the tramline on top of it). For pedestrians and cyclists, the city set up a traffic light. Even though the traffic light is in Google Maps, it is probably not incorporated into the cycling algorithm. Whilst Brouter objectively offers the better route, it could be argued that people actually cycle how Maps routes them, as this street and its destinations are better known than the small side streets that Brouter engine uses.

The numbers of times that *likely* and *probably* have been mentioned within the last chapter is exactly why this framework relies on open source software, and the adoption of open source should be pushed. The complete routing decision of Google is a black box, but if a person wanted to check the decision at a certain intersection with the Brouter, the codebase would be available. Furthermore, if desired, the codebase could also be changed to replicate the behaviour of Google Maps.

4.4 Reasons for excluding certain parameters

As extracted from the literature exogenous variables also influence the average cyclist. These include, but are not limited to, the weather, the climate, and the topology of an area. For both parameters, open data provided by government agencies is available. Still, the framework does not include them in the final score. Elevation, however, is included in the pathing algorithm. This decision is based on three key points. First and foremost, these are variables that cannot be changed. Yes, it is true that humans changed the climate and, as such, the weather. This is one of the reasons why more people should use bikes. Yet, locally changing the climate so that a city can have higher bikeability is nonsense. On this level, there is also the argument that climate change has improved bikeability because, at least in Germany, we have milder winters and less rain. Regarding the topology, there is also very little that cities can do. The rare exception is Trondheim, Norway, where a special bike lift

Name	Score
Karlsruhe	3,07
Münster	3,17
Freiburg	3,35
Kiel	3,54
Bremen	3,57
Hannover	3,67
Braunschweig	3,72
Leipzig	3,85
Mannheim	3,90
Bielefeld	3,92
Wiesbaden	3,92
Rostock	3,94
Mainz	3,96
Augsburg	3,97
Oberhausen	4,01
Dresden	4,02
Kassel	4,08
Düsseldorf	4,12
Nürnberg	4,15
Bonn	4,17
Chemnitz	4,18
Bochum	4,20
Halle	4,21
Erfurt	4,22
Essen	4,22
Aachen	4,23
Wuppertal	4,24
Gelsenkirchen	4,26
Lübeck	4,30
Magdeburg	4,32
Dortmund	4,35
Köln	4,37
Krefeld	4,39
Mönchengladbach	4,42
Duisburg	4,47

Table 3 ADFC Fahrradklima Index[38]

was built to support cyclists on a 150 m climb[37]. Furthermore, these datasets are large. To get the elevation for Augsburg alone, one would have to download and extract a file of approximately 1 GB. This would greatly influence the performance. Additionally, the elevation datasets have a resolution of 25 x 25 m. This would make the deduction for small road segments useless. The last points which apply only to the weathers is that it is a highly temporal characteristic. As previously discussed, the used rout time is not a factor. Compared to the averages would also not help the model, as the average within Germany is evenly distributed.

4.5 Comparing city-wide results with other frameworks

The major goal of this work is to create a technology that can be used to calculate bikeability in a comparable and reproducible way. To evaluate the basic idea, a proof-of-concept model is included. The default values included in there are deducted from the executed literature review on bikeability. Whereas this approach does not meet all scientific requirements, it should still give a fair approximation of the real world. To support this claim, the citywide score will be compared with other indexes that rate bikeability on different parameters and with different approaches.

The first other rating comes from the *Allgemeine Deutsche Fahrradclub*[38]. The ADFC Bicycle Climate Test takes place regularly. It features questions relating the cycling climate in a city and municipality, and evaluates the local situation as based on average people. The ADFC Cycling Climate Test is the largest survey on the cycling climate worldwide and was funded by the Federal Ministry of Transport and Digital

[37] Xie, Jenny (2014): This Bike Elevator Makes Steep Hills a Little More Manageable. Available at https://www.bloomberg.com/news/articles/2014-04-03/this-bike-elevator-makes-steep-hills-a-little-more-manageable (13.10.2022)

[38] ADFC (2022): Fahrradklima-Test Ergebnisse. Available at https://fahrradklima-test.adfc.de/ergebnisse (13.10.2022)

Infrastructure as part of the National Cycling Plan. The survey asks around 30 questions about cycling. Questions include whether cycling is fun or stressful, whether cycle lanes are kept clear of parking violations, whether people feel that they are taken seriously as road users, and whether cycling feels safe for families with children (ADFC, 2022). The following table represents the results from the ADFC survey for 2020. The table is a combination of two tables that were split by the ADFC. The lower a score, the better the cycling quality. By comparing the top five with the proof-of-concept model, three out of five cites appear in both schemata. These are Münster (4.), Bremen (1.) and Karlsruhe (5.). When looking at the top ten, the similarities become even higher (8 out of 10). When looking at the lower rated scores, they are also similar. As they are both very different approaches to measure a similar metric the number of 1on1 matches is low. Because of that the focus of the comparison should be on general intervals. As there are large similarities the performance for the model is good. A notable exception is Hannover. This city is rated high in the ADFC ranking and low when using the model. This can be explained by the study area. The other large discrepancy is Lübeck, which is 4 in the model but in the bottom third of the ADFC list. A possible explanation would be a high impact of either exogenous variables or soft measurements, like crime and safety.

Another score which rates cites on a global scale is the Copenhagenize index[39]. The index rates cities based on 13 parameters, divided into three groups. The parameters include:

- bicycle infrastructure
- bicycle facilities
- traffic calming
- gender split
- modal share for bikes
- modal share increase over the last 10 years
- indicators of safety
- image of the bicycle
- cargo bikes
- advocacy
- politics

[39] CopenhagenizeIndex.eu (2022): the Copenhagenize Index. Available at https://copenhagenizeindex.eu/about/methodology (13.10.2022)

- bike share
- urban planning

In their index, the highest rated German city is Bremen. This aligns perfectly with the scores that the default model produced.

Order	City
1	Münster
2	Bremen
3	Hannover
4	Hamburg
5	Leipzig
6	Nürnberg
7	Berlin
8	Köln
9	Dresden
10	Frankfurt

Table 4 Luko scores for Germany

The last Germany wide rating available is generated by the insurance startup *luko*[40]. They employ 16 different parameters, including four that are related to safety and crime. The extract for German cities can be found in Table 4. Comparing both scores, there is an overlap between the top 2 cities. Most of the other cities in the score are not calculated, because the city was not within the tested interval. Sadly, *luko.com* does not provide any data regarding the worst cities in Germany. The same is true for the Copenhagenize Index that also only offers data for the top 20 cities worldwide.

4.6 Comparing individual segment results with Augsburg residents

The worst road segment within the study area is, without a close contender, Ulmer Straße between Wertachbrücke and Oberhausen Bahnhof. The street is characterised by its surplus of traffic, parking spots on both sides, 11 connection roads along a length of 600 m, a tram line running in both directions, and a bustling sidewalk thanks to many small shops. Combined with the absence of cycling infrastructure and no suitable reroutes (because it is necessary to cross the bridge to get to/from the city centre), it should be obvious that this street has no high bikeability. When asking a group (13) of bike messengers working for the local bike delivery service what the worst street to cycle in Augsburg is, this stretch of road was always in the top three. While this survey has no scientific background, the model

[40] luko.com (2022): Global Bicycle Cities Index 2022. Available at https://de.luko.eu/en/advice/guide/bike-index/ (13.10.2022)

Figure 25 Ulmer Straße, the worst road to cycle in Augsburg.

can be positively credited for that. The second worst road that the model found is the Augsburger Straße between Pfersee and the Hauptbahnhof. This street can be characterised exactly like the Ulmer Straße only with fewer shops. Therefore, the same logic can be applied. Related to the little survey with the bike messengers, the worst street that is not found by the model is the Hoher Weg– Frauentorstraße in the west of the Augsburger Dom. The problem with this street is, that is also a tram track. Moreover, it is an old, cobbled road in a very bad condition, where the cyclists go downhill through a curvy section. Especially in wet conditions this road is very unsafe, due to slippery tram tracks and cobblestone. The framework fails to recognize this road, because the condition is not properly tracked, and the weather does not influence the results.

5 Conclusion and future work

The goal of this thesis was to present a framework that can be used to calculate bikeability for cities. The requirements to the framework can be categorised into three groups and are evaluated afterwards.

The first goal was that the model fulfils the requirements that are set for *open science*. Therefore, the model needs to be reproducible and accessible. Thanks to its use of *open source* and *open data*, this goal was fulfilled. All data, code, and further information can now be found in the authors personal GitHub[41]. Furthermore, the project is attached to the generous MIT License, which grants everybody permission to change and share the framework. Like that, the repository is set up to allow everybody to collaborate and evolve the framework.

The second goal was to offer good performance and high configurability, so that adaptations to models can be implemented fast and easily. This goal has also been reached by implementing a microservice-based architecture with Docker containers as a foundation. The framework can also be tweaked easily by using the supplied config file.

The last goal was to generate comparable bikeability indexes for complete cites and individual streets. These indexes should be influenced by how relevant they really are within the road network. The relevance is calculated by using a semi-random approach that focuses on cycling trips to special destinations like shops or offices. The actual routes were generated using the Brouter engine, as it offers easy configurability, is developed for cyclists, and also incorporates cycling into the route-finding decision. As bikeability is such a broad concept affected by a multitude of factors, developing a fundamental model proved to be out of scope for this work. Still, the supplied proof-of-concept model that is based on published scientific findings showed great potential, when compared with other indexes like the ADFC Fahrradklima. Therefore, this last goal is also achieved but leaves room for further development.

However, there are things that should be addressed in the future. Firstly, the implementation of the study area should be addressed, or made to be configurable, so that users of the framework can decide if they want to use the city district, a bounding box polygon, or a custom shape. Related to that is the implementation of a second mode which uses more than one *.osm.pbf data import. This would also require the import container to be rewritten, so that the imported data can be used for more than one city.

Secondly, it would be interesting to implement *time* as a parameter into the model.

[41] Wendel (2022): Available at https://github.com/Timahawk/bikeability (Release v1.0.0; 21.10.2022) (21.10.2022)

Implementing a daily scale would affect permissions for certain roads, like pedestrian roads, and the amount of traffic. Artery roads can have an uneven distribution over the hourly traffic. Time could also be implemented on a seasonal scale, so that the bikeability is affected by heat, snow, or rain. Especially, concerning the winter, the framework could implement a parameter *cleared*, reflecting whether snow gets removed on highly frequented streets.

This also aligns with a third adaption, the implementation of more parameters. As mentioned above, most of the available parameters are implemented, but if a user has access to further data, it should be possible to integrate them into the model without much effort. This plug and play feature is currently not included. This would also increase the ease to use the framework for other modes of transport like e-scooter or inline skating, or for a more specific group in the cycling community like cargo bikes.

To summarise, this thesis presents a highly configurable framework that can be used to assess the bikeability of cities in a performant and comparable manner. The framework complies with the ideas of open source, open data, and open science. The framework could be the foundation for a comparable, citywide bikeability index that not only generates a score for each city but also offers insights into individual road segments and how they affect the cycling quality. To validate these claims, the framework comes attached with a proof-of-concept model, based on existing literature, that when compared with other indices and local expert knowledge, seems promising.

6 References

Adams, M. A., Todd, M., Kurka, J., Conway, T. L., Cain, K. L., Frank, L. D., & Sallis, J. F. (2015). Patterns of Walkability, Transit, and Recreation Environment for Physical Activity. *American Journal of Preventive Medicine*, *49*(6), 878–887. https://doi.org/10.1016/j.amepre.2015.05.024

ADFC. (2022, October 13). *Fahrradklima-Test Ergebnisse*. https://fahrradklima-test.adfc.de/ergebnisse

Alveano-Aguerrebere, I., Javier Ayvar-Campos, F., Farvid, M., & Lusk, A. (2017). Bicycle Facilities That Address Safety, Crime, and Economic Development: Perceptions from Morelia, Mexico. *International Journal of Environmental Research and Public Health*, *15*(1). https://doi.org/10.3390/ijerph15010001

Amt für Statistik und Stadtforschung, Stadt Augsburg. (2021). *Kurzmitteilung aus Statistik und Stadtforschung*. https://www.augsburg.de/fileadmin/user_upload/buergerservice_rathaus/rathaus/statisiken_und_geodaten/statistiken/kurzmitteilungen/KM_2021_05_Straftaten_und_Tatverdaechtige_in_Augsburg_2020.pdf

Amt für Verkehrsmanagement Stadt Köln. (2022). *Verkehrsaufkommen Stadt Köln: Open.NRW CKAN*. https://ckan.open.nrw.de/nl/dataset/verkehrsaufkommen-stadt-koln-k

Augsburg Stadt. (2022). *Strukturatlas der Stadt Augsburg 2022.*

Ayachi, F. S., Dorey, J., & Guastavino, C. (2015). Identifying factors of bicycle comfort: An online survey with enthusiast cyclists. *Applied Ergonomics*, *46 Pt A*, 124–136. https://doi.org/10.1016/j.apergo.2014.07.010

Ben-Akiva, M [Moshe], & Bierlaire, M [Michel]. (1999). Discrete Choice Methods and their Applications to Short Term Travel Decisions. In F. S. Hillier & R. W. Hall (Eds.), *International Series in Operations Research & Management Science. Handbook of Transportation Science* (Vol. 23, pp. 5–33). Springer US. https://doi.org/10.1007/978-1-4615-5203-1_2

Bergström, A., & Magnusson, R. (2003). Potential of transferring car trips to bicycle during winter. *Transportation Research Part a: Policy and Practice*, *37*(8), 649–666. https://doi.org/10.1016/S0965-8564(03)00012-0

Berke, E. M., Koepsell, T. D., Moudon, A. V., Hoskins, R. E., & Larson, E. B. (2007). Association of the built environment with physical activity and obesity in older persons. *American Journal of Public Health*, *97*(3), 486–492. https://doi.org/10.2105/AJPH.2006.085837

Bikeability. (2022, June 17). *Bikeability: Cycle training for everyone delivering better and safer training*. https://www.bikeability.org.uk/

Blondiau, T., van Zeebroeck, B., & Haubold, H. (2016). Economic Benefits of Increased Cycling. *Transportation Research Procedia*, *14*, 2306–2313. https://doi.org/10.1016/j.trpro.2016.05.247

Bohannon, R. W. (1997). Comfortable and maximum walking speed of adults aged 20-79 years: Reference values and determinants. *Age and Ageing*, *26*(1), 15–19. https://doi.org/10.1093/ageing/26.1.15

Borkowska, S., & Pokonieczny, K. (2022). Analysis of OpenStreetMap Data Quality for Selected Counties in Poland in Terms of Sustainable Development. *Sustainability*, *14*(7), 3728. https://doi.org/10.3390/su14073728

Botma, H. (1995). Method to determine level of service for bicycle paths and pedestrian-bicycle paths. *Transportation Research Record 1502*. https://www.safetylit.org/citations/index.php?fuseaction=citations.viewdetails&citationids=citjournalarticle_687912_38

Boufous, S., Hatfield, J., & Grzebieta, R. (2018). The impact of environmental factors on cycling speed on shared paths. *Accident; Analysis and Prevention*, *110*, 171–176. https://doi.org/10.1016/j.aap.2017.09.017

Broach, J., Dill, J., & Gliebe, J. (2012). Where do cyclists ride? A route choice model developed with revealed preference GPS data. *Transportation Research Part a: Policy and Practice*, *46*(10), 1730–1740. https://doi.org/10.1016/j.tra.2012.07.005

Brückner, J., Schott, M., Zipf, A., & Lautenbach, S. (2021). Assessing shop completeness in OpenStreetMap for two federal states in Germany. *AGILE: GIScience Series*, *2*, 1–7. https://doi.org/10.5194/agile-giss-2-20-2021

Buehler, R., & Dill, J. (2016). Bikeway Networks: A Review of Effects on Cycling. *Transport Reviews*, *36*(1), 9–27. https://doi.org/10.1080/01441647.2015.1069908

Bundesministerium für Digitales und Verkehr. (2022). *Mobilithek*. https://mobilithek.info/imprint

Straßenverkehrs-Ordnung, March 6, 2013.

§ 12a Gesetz zur Förderung der elektronischen Verwaltung, June 13, 2017.

Cao, X., Mokhtarian, P. L., & Handy, S. L. (2009). No Particular Place to Go. *Environment and Behavior*, *41*(2), 233–257. https://doi.org/10.1177/0013916507310318

Cavill, N., Kahlmeier, S., Rutter, H., Racioppi, F., & Oja, P [Pekka] (2008). Economic analyses of transport infrastructure and policies including health effects related to cycling and walking: A systematic review. *Transport Policy*, *15*(5), 291–304. https://doi.org/10.1016/j.tranpol.2008.11.001

Cerny, T., Donahoo, M. J., & Pechanec, J. (2017). Disambiguation and Comparison of SOA, Microservices and Self-Contained Systems. In *Proceedings of the International Conference on Research in Adaptive and Convergent Systems.* ACM. https://doi.org/10.1145/3129676.3129682

Chen, L. (2015). Continuous Delivery: Huge Benefits, but Challenges Too. *IEEE Software*, *32*(2), 50–54. https://doi.org/10.1109/ms.2015.27

Chen, Y., Zhang, Y., Coffman, D., & Mi, Z. (2022). An environmental benefit analysis of bike sharing in New York City. *Cities*, *121*, 103475. https://doi.org/10.1016/j.cities.2021.103475

Clements, P., Garlan, D., Little, R., Nord, R., & Stafford, J. (2003). Documenting software architectures: views and beyond. In *25th International Conference on Software Engineering, 2003. Proceedings.* IEEE. https://doi.org/10.1109/icse.2003.1201264

Conway, M. (1968). *How do committees invent*. https://hashingit.com/elements/research-resources/1968-04-committees.pdf

Curtis, C., Renne, J. L., & Bertolini, L. (2009). *Transit oriented development: Making it happen*. *Transport and mobility*. Ashgate.

Deutschlandatlas. (2022). *Straftaten*. https://www.deutschlandatlas.bund.de/DE/Karten/Unsere-Sicherheit/190-Straftaten.html

Di Francesco, P., Malavolta, I., & Lago, P. (2017). Research on Architecting Microservices: Trends, Focus, and Potential for Industrial Adoption. In *2017 IEEE International Conference on Software Architecture (ICSA).* IEEE. https://doi.org/10.1109/icsa.2017.24

Dovey, K., & Pafka, E. (2020). What is walkability? The urban DMA. *Urban Studies*, *57*(1), 93–108. https://doi.org/10.1177/0042098018819727

Duvall, P. M., Matyas, S., & Glover, A. (2007). *Continuous Integration: Improving Software Quality and Reducing Risk*. Pearson Education.

Ellis, G., Hunter, R., Tully, M. A., Donnelly, M., Kelleher, L., & Kee, F. (2016). Connectivity and physical activity: using footpath networks to measure the walkability of built environments. *Environment and Planning B: Planning and Design*, *43*(1), 130–151. https://doi.org/10.1177/0265813515610672

Evans, E., & Fowler, M. (2019). *Domain-driven design: Tackling complexity in the heart of software*. Addison-Wesley.

Fernández-Heredia, Á., Monzón, A., & Jara-Díaz, S. (2014). Understanding cyclists' perceptions, keys for a successful bicycle promotion. *Transportation Research Part a: Policy and Practice*, *63*, 1–11. https://doi.org/10.1016/j.tra.2014.02.013

Fishman, E., Washington, S., & Haworth, N. (2013). Bike Share: A Synthesis of the Literature. *Transport Reviews*, *33*(2), 148–165. https://doi.org/10.1080/01441647.2013.775612

Fitzgerald (2006). The Transformation of Open Source Software. *MIS Quarterly*, *30*(3), 587. https://doi.org/10.2307/25148740

Fonseca, F., Ribeiro, P. J. G., Conticelli, E., Jabbari, M., Papageorgiou, G., Tondelli, S., & Ramos, R. A. R. (2022). Built environment attributes and their influence on walkability. *International Journal of Sustainable Transportation*, *16*(7), 660–679. https://doi.org/10.1080/15568318.2021.1914793

Fosgerau, M., Frejinger, E [Emma], & Karlstrom, A. (2013). A link based network route choice model with unrestricted choice set. *Transportation Research Part B: Methodological*, *56*, 70–80. https://doi.org/10.1016/j.trb.2013.07.012

Fowler, M. (2014, March 25). *Microservices: a definition of this new architectural term*. https://martinfowler.com/articles/microservices.html

Frejinger, E [E.], Bierlaire, M [M.], & Ben-Akiva, M [M.] (2009). Sampling of alternatives for route choice modeling. *Transportation Research Part B: Methodological*, *43*(10), 984–994. https://doi.org/10.1016/j.trb.2009.03.001

Gebhardt, L., Ehrenberger, S., Wolf, C., & Cyganski, R. (2022). Can shared E-scooters reduce CO2 emissions by substituting car trips in Germany? *Transportation Research Part D: Transport and Environment*, *109*, 103328. https://doi.org/10.1016/j.trd.2022.103328

Geoportal Augsburg. (2022, October 11). *Fahrradkarte.* Tiefbauamt Abt. Verkehrsplanung. https://geoportal.augsburg.de/WebDaten/synserver?project=fahrradkarte&language=de&client=auto

Goel, R., Goodman, A., Aldred, R., Nakamura, R., Tatah, L., Garcia, L. M. T., Zapata-Diomedi, B., Sa, T. H. de, Tiwari, G., Nazelle, A. de, Tainio, M., Buehler, R., Götschi, T., & Woodcock, J. (2022). Cycling behaviour in 17 countries across 6 continents: Levels of cycling, who cycles, for what purpose, and how far? *Transport Reviews*, *42*(1), 58–81. https://doi.org/10.1080/01441647.2021.1915898

Gong, H., Chen, C., Bialostozky, E., & Lawson, C. T. (2012). A GPS/GIS method for travel mode detection in New York City. *Computers, Environment and Urban Systems*, *36*(2), 131–139. https://doi.org/10.1016/j.compenvurbsys.2011.05.003

Grigore, E., Garrick, N., Fuhrer, R., & Axhausen, I. K. W. (2019). Bikeability in Basel. *Transportation Research Record*, *2673*(6), 607–617. https://doi.org/10.1177/0361198119839982

Habibian, M., & Hosseinzadeh, A. (2018). Walkability index across trip purposes. *Sustainable Cities and Society*, *42*, 216–225. https://doi.org/10.1016/j.scs.2018.07.005

Harvey, F., Krizek, K. J., & Collins, R. (2008). Using GPS Data to Assess Bicycle Commuter Route Choice. *Transportation Research Board 87th Annual MeetingTransportation Research Board*(08-2951). https://trid.trb.org/view/848852

Heorhiadi, V., Rajagopalan, S., Jamjoom, H., Reiter, M. K., & Sekar, V. (2016). Gremlin: Systematic Resilience Testing of Microservices. In *2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS)*. IEEE. https://doi.org/10.1109/icdcs.2016.11

Heron, M. J., Hanson, V. L., & Ricketts, I. (2013). Open Source and Accessibility: Advantages and Limitations. *Journal of Interaction Science*, *1*(1), 2. https://doi.org/10.1186/2194-0827-1-2

Hood, J., Sall, E., & Charlton, B. (2011). A GPS-based bicycle route choice model for San Francisco, California. *Transportation Letters*, *3*(1), 63–75. https://doi.org/10.3328/TL.2011.03.01.63-75

Hoogendoorn, S., & Daamen, W. (2016). Bicycle Headway Modeling and Its Applications. *Transportation Research Record*, *2587*(1), 34–40. https://doi.org/10.3141/2587-05

Hudde, A. (2022). The unequal cycling boom in Germany. *Journal of Transport Geography*, *98*, 103244. https://doi.org/10.1016/j.jtrangeo.2021.103244

Hull, A., & O’Holleran, C. (2014). Bicycle infrastructure: can good design encourage cycling? *Urban, Planning and Transport Research*, *2*(1), 369–406. https://doi.org/10.1080/21650020.2014.955210

Hummer, J. E., Rouphail, N. M., Toole, J. L., Patten, R. S., Schneider, R. J., Green, J. S., Hughes, R. G., & Fain, S. J. (2006). *Evaluation of Safety, Design, and Operation of Shared-Use Paths—Final Report* (FHWA-HRT-05-137). https://trid.trb.org/view/790403

Hunter, W. W., Srinivasan, R., Thomas, L., Martell, C. A., & Seiderman, C. B. (2011). Evaluation of Shared Lane Markings in Cambridge, Massachusetts. *Transportation Research Record*, *2247*(1), 72–80. https://doi.org/10.3141/2247-09

Ishikawa, T., Fujiwara, H., Imai, O., & Okabe, A. (2008). Wayfinding with a GPS-based mobile navigation system: A comparison with maps and direct experience. *Journal of Environmental Psychology*, *28*(1), 74–82. https://doi.org/10.1016/j.jenvp.2007.09.002

Jensen, S. U. (2007). Pedestrian and Bicyclist Level of Service on Roadway Segments. *Transportation Research Record*, *2031*(1), 43–51. https://doi.org/10.3141/2031-06

Johan de Hartog, J., Boogaard, H., Nijland, H., & Hoek, G. (2010). Do the health benefits of cycling outweigh the risks? *Environmental Health Perspectives*, *118*(8), 1109–1116. https://doi.org/10.1289/ehp.0901747

Johnson, R. (2012, June 18). *Do Components Exist?* http://wiki.c2.com/?RalphJohnson

Kang, L., Xiong, Y., & Mannering, F. L. (2013). Statistical analysis of pedestrian perceptions of sidewalk level of service in the presence of bicycles. *Transportation Research Part a: Policy and Practice*, *53*, 10–21. https://doi.org/10.1016/j.tra.2013.05.002

Koohsari, M. J., Owen, N., Cerin, E., Giles-Corti, B., & Sugiyama, T. (2016). Walkability and walking for transport: Characterizing the built environment using space syntax. *International Journal of Behavioral Nutrition and Physical Activity*, *13*(1), 121. https://doi.org/10.1186/s12966-016-0448-9

Krenn, P. J., Oja, P [Pekka], & Titze, S [Sylvia] (2015). Development of a Bikeability Index to Assess the Bicycle-Friendliness of Urban Environments. *Open Journal of Civil Engineering*, *05*(04), 451–459. https://doi.org/10.4236/ojce.2015.54045

Krizek, K. J., Handy, S. L., & Forsyth, A. (2009). Explaining Changes in Walking and Bicycling Behavior: Challenges for Transportation Research. *Environment and Planning B: Planning and Design*, *36*(4), 725–740. https://doi.org/10.1068/b34023

Lee, S., Lee, C., Nam, J. W., Abbey-Lambertz, M., & Mendoza, J. (2020). School Walkability Index: Application of Environmental Audit Tool and GIS. *Journal of Transport & Health*, *18*. https://doi.org/10.1016/j.jth.2020.100880

Lowry, M. B., Callister, D., Gresham, M., & Moore, B. (2012). Assessment of Communitywide Bikeability with Bicycle Level of Service. *Transportation Research Record*, *2314*(1), 41–48. https://doi.org/10.3141/2314-06

Lu, M., Blokpoel, R., & Joueiai, M. (2018). Enhancement of safety and comfort of cyclists at intersections. *IET Intelligent Transport Systems*, *12*(6), 527–532. https://doi.org/10.1049/iet-its.2017.0250

Massey, D. B. (2005). *For space*. https://www.torrossa.com/gs/resourceproxy?an=4912000&publisher=fz7200

Menghini, G., Carrasco, N., Schüssler, N., & Axhausen, K. W [K. W.] (2010). Route choice of cyclists in Zurich. *Transportation Research Part a: Policy and Practice*, *44*(9), 754–765. https://doi.org/10.1016/j.tra.2010.07.008

Mertens, L., van Dyck, D., Ghekiere, A., Bourdeaudhuij, I. de, Deforche, B., van de Weghe, N., & van Cauwenberg, J. (2016). Which environmental factors most strongly influence a street's appeal for bicycle transport among adults? A conjoint study using manipulated photographs. *International Journal of Health Geographics*, *15*(1), 31. https://doi.org/10.1186/s12942-016-0058-4

Montesi, F., & Weber, J. (2016). *Circuit Breakers, Discovery, and API Gateways in Microservices.*

Moran, M. R., Rodríguez, D. A., & Corburn, J. (2018). Examining the role of trip destination and neighborhood attributes in shaping environmental influences on children’s route choice. *Transportation Research Part D: Transport and Environment*, *65*, 63–81. https://doi.org/10.1016/j.trd.2018.08.001

Moudon, A. V., & Lee, C. (2003). Walking and bicycling: An evaluation of environmental audit instruments. *American Journal of Health Promotion : AJHP*, *18*(1), 21–37. https://doi.org/10.4278/0890-1171-18.1.21

Moudon, A. V., Lee, C., Cheadle, A. D., Collier, C. W., Johnson, D., Schmid, T. L., & Weather, R. D. (2005). Cycling and the built environment, a US perspective. *Transportation Research Part D: Transport and Environment*, *10*(3), 245–261. https://doi.org/10.1016/j.trd.2005.04.001

Muhs, C. D., & Clifton, K. J. (2016). Do characteristics of walkable environments support bicycling? Toward a definition of bicycle-supported development. *Journal of Transport and Land Use.* Advance online publication. https://doi.org/10.5198/jtlu.2015.727

Nair, S., Javkar, K., Wu, J., & Frias-Martinez, V. (2019). Understanding Cycling Trip Purpose and Route Choice Using GPS Traces and Open Data. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, *3*(1), 1–26. https://doi.org/10.1145/3314407

Nankervis, M. (1999). The effect of weather and climate on bicycle commuting. *Transportation Research Part a: Policy and Practice*, *33*(6), 417–431. https://doi.org/10.1016/S0965-8564(98)00022-6

Navin, F. P. D. (1994). Bicycle traffic flow characteristics: experimental results and comparisons. *ITE Journal 64*. https://www.safetylit.org/citations/index.php?fuseaction=citations.viewdetails&citationids=citjournalarticle_241140_38

Nielsen, T. A. S., & Skov-Petersen, H. (2018). Bikeability – Urban structures supporting cycling. Effects of local, urban and regional scale urban form factors on cycling from home and workplace locations in Denmark. *Journal of Transport Geography*, *69*, 36–44. https://doi.org/10.1016/j.jtrangeo.2018.04.015

Nyman, L. M., Mikkonen, T., Lindman, J., & Fougère, M. (2011). Forking: The Invisible Hand of Sustainability in Open Source Software. *Proceedings of SOS 2011: Towards Sustainable Open Source Towards Sustainable Open Source*. https://helda.helsinki.fi/handle/10138/157663

Oasis, S. O. (2005). Reference Model TC. *OASIS Reference Model for Service Oriented Architecture*, *1*.

Oja, P [P.], Titze, S [S.], Bauman, A., Geus, B. de, Krenn, P., Reger-Nash, B., & Kohlberger, T. (2011). Health benefits of cycling: A systematic review. *Scandinavian Journal of Medicine & Science in Sports*, *21*(4), 496–509. https://doi.org/10.1111/j.1600-0838.2011.01299.x

The Open Group. (2011). *SOA Source Book: Edition 7*. http://www.opengroup.org/soa/source-book/soa/index.htm

Open Knowledge Foundation. (2022, February 23). *The Open Definition: Defining Open in Open Data, Open Content and Open Knowledge*. http://opendefinition.org/

Open Source Initiative. (2007, March 22). *The Open Source Definition*. https://opensource.org/osd

OpenStreetMap Authors. (2022). *OpenStreetMap about*. https://www.openstreetmap.org/about

Orban, S. (2015). *Why You Should Run What You Build: Enterprise DevOps:*. https://aws.amazon.com/de/blogs/enterprise-strategy/enterprise-devops-why-you-should-run-what-you-build/

osm2pgsql. (2022, September 30). *osm2pgsql*. https://osm2pgsql.org/

Padeiro, M., Louro, A., & Da Costa, N. M. (2019). Transit-oriented development and gentrification: A systematic review. *Transport Reviews*, *39*(6), 733–754. https://doi.org/10.1080/01441647.2019.1649316

Parkin, J., & Rotheram, J. (2010). Design speeds and acceleration characteristics of bicycle traffic for use in planning, design and appraisal. *Transport Policy*, *17*(5), 335–341. https://doi.org/10.1016/j.tranpol.2010.03.001

Parkin, J., Wardman, M., & Page, M. (2007). Estimation of the determinants of bicycle mode share for the journey to work using census data. *Transportation*, *35*(1), 93–109. https://doi.org/10.1007/s11116-007-9137-5

PostGIS. (2022, September 14). *PostGIS*. https://postgis.net/

Prato, C. G. (2009). Route choice modeling: Past, present and future research directions. *Journal of Choice Modelling*, *2*(1), 65–100. https://doi.org/10.1016/s1755-5345(13)70005-8

Pressemitteilungen Fahrradstadt. (2022, October 11). *Neuburger Straße bekommt neue Radfahrstreifen*. https://www.augsburg.de/aktuelles-aus-der-stadt/detail/neuburger-strasse-bekommt-neue-radfahrstreifen

Pritchard, R., & Frøyen, Y. (2019). Location, location, relocation: How the relocation of offices from suburbs to the inner city impacts commuting on foot and by bike. *European Transport Research Review*, *11*(1), 1–20. https://doi.org/10.1186/s12544-019-0348-6

Rademacher, F., Sorgalla, J., & Sachweh, S. (2018). Challenges of Domain-Driven Microservice Design: A Model-Driven Perspective. *IEEE Software*, *35*(3), 36–43. https://doi.org/10.1109/ms.2018.2141028

Rieser-Schüssler, N., Balmer, M., & Axhausen, K. W [Kay W.] (2013). Route choice sets for very high-resolution data. *Transportmetrica a: Transport Science*, *9*(9), 825–845. https://doi.org/10.1080/18128602.2012.671383

Robinson, O., Tamayo, I., Castro, M. de, Valentin, A., Giorgis-Allemand, L., Hjertager Krog, N., Marit Aasvang, G., Ambros, A., Ballester, F., Bird, P., Chatzi, L., Cirach, M., Dėdelė, A., Donaire-Gonzalez, D., Gražuleviciene, R., Iakovidis, M., Ibarluzea, J., Kampouri, M., Lepeule, J., . . . Basagaña, X. (2018). The Urban Exposome during Pregnancy and Its Socioeconomic Determinants. *Environmental Health Perspectives*, *126*(7), 77005. https://doi.org/10.1289/EHP2862

Rosen, M., Lublinsky, B., Smith, K. T., & Balcer, M. J. (2012). *Applied SOA: Service-Oriented Architecture and Design Strategies*. John Wiley & Sons.

Saelens, B. E., Sallis, J. F., & Frank, L. D. (2003). Environmental correlates of walking and cycling: Findings from the transportation, urban design, and planning literatures. *Annals of Behavioral Medicine : A Publication of the Society of Behavioral Medicine*, *25*(2), 80–91. https://doi.org/10.1207/S15324796ABM2502_03

Savan, B., Cohlmeyer, E., & Ledsham, T. (2017). Integrated strategies to accelerate the adoption of cycling for transportation. *Transportation Research Part F: Traffic Psychology and Behaviour*, *46*, 236–249. https://doi.org/10.1016/j.trf.2017.03.002

Schleinitz, K., Petzoldt, T., Franke-Bartholdt, L., Krems, J., & Gehlert, T. (2017). The German Naturalistic Cycling Study – Comparing cycling speed of riders of different e-bikes and conventional bicycles. *Safety Science*, *92*, 290–297. https://doi.org/10.1016/j.ssci.2015.07.027

Schmid-Querg, J., Keler, A., & Grigoropoulos, G. (2021). The Munich Bikeability Index: A Practical Approach for Measuring Urban Bikeability. *Sustainability*, *13*(1), 428. https://doi.org/10.3390/su13010428

Schmidt, M.-T., Hutchison, B., Lambros, P., & Phippen, R. (2005). The Enterprise Service Bus: Making service-oriented architecture real. *IBM Systems Journal*, *44*(4), 781–797. https://doi.org/10.1147/sj.444.0781

Sener, I. N., Eluru, N., & Bhat, C. R. (2009). An analysis of bicycle route choice preferences in Texas, US. *Transportation*, *36*(5), 511–539. https://doi.org/10.1007/s11116-009-9201-4

Shaw, C., Russell, M., Keall, M., MacBride-Stewart, S., Wild, K., Reeves, D., Bentley, R., & Woodward, A. (2020). Beyond the bicycle: Seeing the context of the gender gap in cycling. *Journal of Transport & Health*, *18*, 100871. https://doi.org/10.1016/j.jth.2020.100871

Shu, S., Bian, Y., Rong, J., & Xu, D. (2019). Determining the exact location of a public bicycle station-The optimal distance between the building entrance/exit and the station. *PLOS ONE*, *14*(2), e0212478. https://doi.org/10.1371/journal.pone.0212478

Sprott, D., & Wilkes, L. (2006). Understanding Service-Oriented Architecture. In J. McGovern & P. Herzum (Eds.), *Enterprise service oriented architectures: Concepts, challenges, recommendations* (pp. 1–48). Springer. https://doi.org/10.1007/1-4020-3705-8_1

Stevenson, M., Thompson, J., Sá, T. H. de, Ewing, R., Mohan, D., McClure, R., Roberts, I., Tiwari, G., Giles-Corti, B., Sun, X., Wallace, M., & Woodcock, J. (2016). Land use, transport, and population health: Estimating the health benefits of compact cities. *The Lancet*, *388*(10062), 2925–2935. https://doi.org/10.1016/s0140-6736(16)30067-8

Stinson, M. A., & Bhat, C. R. (2003). Commuter Bicyclist Route Choice: Analysis Using a Stated Preference Survey. *Transportation Research Record*, *1828*(1), 107–115. https://doi.org/10.3141/1828-13

Storper, M., & Venables, A. J. (2004). Buzz: face-to-face contact and the urban economy. *Journal of Economic Geography*, *4*(4), 351–370. https://doi.org/10.1093/jnlecg/lbh027

Swiss Association of Road and Traffic Professionals. (2017). *SN 640 252: Knoten Führung des Veloverkehrs*. Zürich, Switzerland.

Taleai, M., & Yameqani, A. S. (2018). Integration of GIS, remote sensing and Multi-Criteria Evaluation tools in the search for healthy walking paths. *KSCE Journal of Civil Engineering*, *22*(1), 279–291. https://doi.org/10.1007/s12205-017-2538-x

Vargo, J., Stone, B., & Glanz, K. (2012). Google walkability: A new tool for local planning and public health research? *Journal of Physical Activity & Health*, *9*(5), 689–697. https://doi.org/10.1123/jpah.9.5.689

Williams, G. C., Borghese, M. M., & Janssen, I. (2018). Neighborhood walkability and objectively measured active transportation among 10–13 year olds. *Journal of Transport & Health*, *8*, 202–209. https://doi.org/10.1016/j.jth.2017.12.006

Willis, K. S., Hölscher, C., Wilbertz, G., & Li, C. (2009). A comparison of spatial knowledge acquisition with maps and mobile maps. *Computers, Environment and*

Urban Systems, *33*(2), 100–110. https://doi.org/10.1016/j.compenvurbsys.2009.01.004

Winters, M., Brauer, M., Setton, E. M., & Teschke, K. (2013). Mapping bikeability: a spatial tool to support sustainable travel. *Environment and Planning B: Planning and Design*, *40*(5), 865–883. https://doi.org/10.1068/b38185

Wu, J., Hong, L., & Frias-Martinez, V. (2018). Predicting Perceived Cycling Safety Levels Using Open and Crowdsourced Data. In *2018 IEEE International Conference on Big Data (Big Data).* IEEE. https://doi.org/10.1109/bigdata.2018.8622291

Wu, Z., Wang, W., Hu, X., de Zhao, & Li, Z. (2015). Evaluating the Impacts of Pavement Damage on Bicycle Traffic Flow on Exclusive Bicycle Paths. *Transportation Research Board 94th Annual MeetingTransportation Research Board*(15-2566). https://trid.trb.org/view/1337690

Xiao, Z., Wijegunaratne, I., & Qiang, X. (2016). Reflections on SOA and Microservices. In *2016 4th International Conference on Enterprise Systems (ES).* IEEE. https://doi.org/10.1109/es.2016.14

Yeboah, G., Porto de Albuquerque, J., Troilo, R., Tregonning, G., Perera, S., Ahmed, Syed A. K. Shifat, Ajisola, M., Alam, O., Aujla, N., Azam, S. I., Azeem, K., Bakibinga, P., Chen, Y.-F., Choudhury, N. N., Diggle, P. J., Fayehun, O., Gill, P., Griffiths, F., Harris, B., . . . Yusuf, R. (2021). Analysis of OpenStreetMap Data Quality at Different Stages of a Participatory Mapping Process: Evidence from Slums in Africa and Asia. *ISPRS International Journal of Geo-Information*, *10*(4), 265. https://doi.org/10.3390/ijgi10040265

Yin, L. (2017). Street level urban design qualities for walkability: Combining 2D and 3D GIS measures. *Computers, Environment and Urban Systems*, *64*, 288–296. https://doi.org/10.1016/j.compenvurbsys.2017.04.001

Yuan, C., Sun, Y., Lv, J., & Lusk, A. C. (2017). Cycle Tracks and Parking Environments in China: Learning from College Students at Peking University. *International Journal of Environmental Research and Public Health*, *14*(8). https://doi.org/10.3390/ijerph14080930

Yuan, Y., Daamen, W., Goñi-Ros, B., & Hoogendoorn, S. (2018). Investigating cyclist interaction behavior through a controlled laboratory experiment. *Journal of Transport and Land Use*, *11*(1). https://doi.org/10.5198/jtlu.2018.1155

Zhao, J., Guo, C., Zhang, R., Guo, D., & Palmer, M. (2019). Impacts of weather on cycling and walking on twin trails in Seattle. *Transportation Research Part D: Transport and Environment*, *77*, 573–588. https://doi.org/10.1016/j.trd.2019.09.022

Zimmermann, M., Mai, T., & Frejinger, E [Emma] (2017). Bike route choice modeling using GPS data without choice sets of paths. *Transportation Research Part C: Emerging Technologies*, *75*, 183–196. https://doi.org/10.1016/j.trc.2016.12.009

Zuiderwijk, A., Janssen, M., Poulis, K., & van de Kaa, G. (2015). Open data for competitive advantage: insights from open data use by companies. *Proceedings of the 16th Annual International Conference on Digital Government Research*, 79–88.