

# Generative Chatbots zur Unterstützung bei der BIM-Anwendung

## Generative Chatbots for support in BIM application

Fabian Edenhofner, RWTH Aachen University, ICoM | Institut für Baumanagement, Digitales Bauen und Robotik im Bauwesen, Jülicher Straße 209d 52070 Aachen, [edenhofner@icom.rwth-aachen.de](mailto:edenhofner@icom.rwth-aachen.de)

### Kurzfassung

Der Einsatz von generativen Chatbots birgt ein Potential zur Anwendung in vielerlei alltäglichen Situationen und hat bereits seinen Einzug in Lehre und multiple Geschäftsfelder gefunden. Mit Blick auf die Architecture, Engineering & Construction (Abk. AEC) Branche zeigen neuartige Softwareprodukte sowie zahlreiche Beiträge in gängigen Social Media Portalen die vielversprechenden Möglichkeiten der Nutzung jener Technologien im BIM-spezifischen Zusammenhang. Der vorliegende Beitrag widmet sich der Frage, inwieweit die Anwendung von generativen Chatbots im BIM-spezifischen Kontext eine unterstützende Funktion bei der Umsetzung charakteristischer BIM-Anwendungsszenarien birgt. Neben einer Darstellung des Stands der Technik und Praxis wird die Fragestellung anhand einer Fallstudie zur Informationsextraktion aus einem BIM-Modell beleuchtet.

### Abstract

The use of generative chatbots has a potential for application in many everyday situations and has already found its way into teaching and multiple business areas. With regard to the Architecture, Engineering & Construction (AEC) industry, new software products as well as numerous contributions in popular social media portals show the promising possibilities of using these technologies in a BIM-specific context. This paper addresses the question to what extent the use of generative chatbots in a BIM-specific context has a supporting function in the implementation of characteristic BIM application scenarios. In addition to a presentation of the state of the technology and practice, the question is illuminated by means of a case study on information extraction from a BIM model.

## 1 Einleitung

### 1.1 Hintergrund

Mit dem Aufkommen einer neuen Generation an generativen Chatbots wie beispielsweise ChatGPT, Bing Chat oder Google Bard sowie ihren umfassenden Möglichkeiten stehen jene Tools im Jahr 2023 im medialen Fokus. Sie ermöglichen es, textuelle Eingaben zu erkennen, zu verarbeiten und entsprechend darauf zu reagieren. Chatbots wie ChatGPT generieren hierbei Ausgaben, welche denen von natürlichen Menschen ähnlich sind und erlauben umfangreiche Konversationen [1]. Als beispielhafte Anwendungsszenarien ergeben sich neben der Möglichkeit zur Konversation das Beantworten von Fragen, eine Textzusammenfassung oder für die Verwendung zur Generierung von ausführbaren Programmiercode [1, 2]. Ihr Einsatz birgt hierbei ein Anwendungspotential in multiplen Bereichen wie etwa in der Lehre [Vgl. 3] oder in Geschäftsfeldern des Gesundheitswesens, Finanzbereichs oder Marketings [Vgl. 4].

In Hinblick auf die AEC Branche deuten vielversprechende Softwareprodukte [Vgl. 5, 6] und zahlreiche Posts in Social Media-Portalen [Vgl. 7–11] insbesondere die Anwendung im Kontext des *Building Information Modelings* (Abk.

BIM). Die Einsatzbereiche von generativen Chatbots lassen sich hierbei zumeist in die Kategorie Informationsextraktion aus BIM-Modellen sowie eine Kategorie mit der Funktion zur Unterstützung beim architektonischen Entwurfsprozess gliedern.

Die BIM-Methode wird als Synonym für die Digitalisierung sämtlicher Prozesse der Bauwelt angesehen [12]. Zudem wird die BIM-Methode in der Literatur als zentraler Bestandteil der digitalen Transformation der Bauwelt betrachtet [13]. Die Anwendung von generativen Chatbots kann hierbei in BIM-spezifischen Szenarien ein lebenszyklusübergreifendes Anwendungspotential bergen [14]. Hierdurch ergibt sich ein möglicher Aufschwung der Digitalisierung der Bauwelt und eine etwaige Reduktion der Hürden zur Technologieanwendung und -akzeptanz.

### 1.2 Ziel und Aufbau des Beitrags

Mit dem vorliegenden Beitrag soll aufgezeigt werden, inwieweit die Anwendung von generativen Chatbots im BIM-Kontext eine unterstützende Funktion bei der Umsetzung spezifischer BIM-Anwendungsszenarien birgt. Hierzu erfolgt zunächst eine oberflächliche Beleuchtung des technischen Hintergrundes von generativen Chatbots. Anschließend wird ein Überblick über bisherige Forschungsarbeiten sowie Ansätzen aus der Praxis gegeben.

DOI: 10.17185/dupublico/79111



Im Rahmen einer Fallstudie werden ferner zwei Strategien zur Verwendung von ChatGPT zur Informationsextraktion aus einem BIM-Modell dargestellt und analysiert. Hierdurch soll aufgezeigt werden, inwiefern ChatGPT verwendet werden kann, um Prozesse der benutzerspezifischen Informationsausgabe zu automatisieren. Darauf aufbauend wird dargestellt, welche potentiellen Herausforderungen bei der praktischen Umsetzung gegeben sind und ob sich ein Mehrwert für nutzende Personen ergeben kann. Im Fazit und Ausblick des Beitrags sind die zentralen Ergebnisse zusammengefasst. Zudem werden Limitierungen der entwickelten Ansätze und weiterer Untersuchungsbedarf beschrieben.

## 2 Zur Nutzung von generativen Chatbots im BIM-spezifischen Kontext

### 2.1 Technischer Hintergrund

*Generative Artificial Intelligence* beschreibt grundlegend KI-basierte Ansätze zur Entwicklung von Datensätzen. Die Ansätze bauen hierbei auf umfangreiche Trainingsdaten auf, wodurch sie in der Lage sind, menschenähnliche Ausgaben zu generieren. [15] Solche Ansätze gibt es für eine Vielzahl von Anwendungsbereichen. So können beispielsweise mit Hilfe des Tools Midjourney Bilder aufbauend auf Nutzereingaben generiert werden [16]. Konversationelle Ansätze bzw. generative Chatbots ermöglichen es hingegen, Nutzereingaben in natürlicher Sprache zu verstehen, sie zu verarbeiten sowie entsprechend darauf zu reagieren [14]. Jene genannten Ansätze sind im Allgemeinen dem Bereich des *Natural Language Processings* (Abk. NLP) zuzuordnen. NLP beschäftigt sich grundsätzlich damit, wie Computersysteme natürliche Sprache, beispielsweise in Form von Texten, interpretieren sowie verarbeiten können. Diesbezüglich kann NLP in die beiden Abschnitte *Natural Language Understanding* sowie *Natural Language Generation* unterteilt werden. Ersteres widmet sich den Ansätzen zum Verstehen einer Spracheingabe, wohingegen zweiteres der Generierung einer Sprachausgabe dient. [17]

Unter den generativen Chatbots stellt ChatGPT derzeit denjenigen Ansatz dar, der im Fokus der medialen Aufmerksamkeit steht. ChatGPT wurde Ende 2022 von OpenAI der breiten Öffentlichkeit kostenlos zur Verfügung gestellt. Die kostenlose Variante bedient sich dem von OpenAI eigens entwickelten GPT-3.5 Sprachmodell. [18]

Unter (großen) Sprachmodellen (engl. *Large Language Models*, Abk. LLM) versteht man Ansätze, welche auf neuronalen Netzen basieren und es ermöglichen, natürliche sprachliche Texte zu verstehen sowie zu kreieren [19].

ChatGPT baut auf einer *Generative Pre-Trained Transformer* (Abk. GPT) Architektur auf. Pre-Trained bezieht sich auf den Entwicklungsprozess solcher Sprachmodelle. Die Modelle werden zunächst auf großen textuellen

Datensätzen vortrainiert und eignen sich somit an, passende Wörter aufbauend auf den vorhergehenden Wörtern vorzuschlagen. [20] In einem darauffolgenden Prozess der Feinabstimmung (engl. *fine-tuning*) kann das vortrainierte Modell mit einem spezifischen, kleineren Datensatz entsprechend trainiert werden, um in einem individuellen Anwendungsszenario eine höhere Effizienz zu erreichen [21, 22]. Der Transformer stellt eine weitere grundlegende KI-Architekturkomponente dar. Transformer ermöglichen es, sequentielle Informationen effizienter zu verarbeiten und weitreichende Verbindungen in den Eingabedaten zu bilden [21].

Die Anwendung von generativen Chatbots und den zugrundeliegenden Sprachmodellen birgt jedoch ebenfalls Einschränkungen. Im Kontext des vorliegenden Beitrags ist insbesondere die Limitierung der Halluzination zu nennen. Unter Halluzination wird die Ausgabe falscher Inhalte verstanden [21]. Im Zuge eines spezifischen Anwendungsszenarios können sich Halluzinationen somit weitreichend auswirken.

### 2.2 Generative Chatbots und ihre Anwendung im BIM-spezifischen Kontext

Die Erhebung des Stands der Wissenschaft und der Praxis zur Anwendung von generativen Chatbots im BIM-spezifischen Kontext erfolgte auf Basis einer Literatur- sowie Praxisrecherche. Da es sich um ein junges Forschungsfeld handelt, wurden die recherchierten Ergebnisse aus wissenschaftlichen Publikationen bewusst um nicht-wissenschaftliche Suchergebnisse ergänzt. Der wissenschaftliche Status Quo wurde durch auf dem Markt verfügbare Ansätze sowie prototypische Implementierungen und Fallbeispiele aus Social Media Portalen und einer nicht-wissenschaftlichen Suchmaschine ergänzt. Der Fokus der Recherche lag auf der expliziten Anwendung von ChatGPT, generativen Chatbots sowie GPT Modellen in BIM-spezifischen Fallbeispielen.

Die Literaturrecherche fand auf den Datenbanken Science Direct sowie Google Scholar mit den Suchbegriffskombinationen *"building information modeling" OR BIM AND „language model“ OR "LLM" OR "GPT" OR "ChatGPT" OR "chatbot" OR "generative ai"* statt. Eine Eingrenzung der Suchergebnisse wurde durch die ausschließliche Betrachtung Englisch-sprachiger Literatur, einer open access bedingten Verfügbarkeit sowie anhand einer Eingrenzung durch den zeitlichen Betrachtungsraum von 2022 bis 2023 erreicht. Bei den betrachteten Ansätzen kann eine grundlegende Kategorisierung festgestellt werden. Es konnten Ansätze zur Informationsextraktion aus BIM-Modellen, Ansätze, die den architektonischen, BIM-basierten Entwurfsprozess unterstützen sowie eine Vermischung jener Kategorisierungen identifiziert werden.

Weiterführend lassen sich die identifizierten Ansätze hinsichtlich der Verwendung von generativen Chatbots für

die Entwicklung des jeweiligen Ansatzes oder des grundlegenden Aufbaus auf einem Chatbots bzw. seiner Technologie unterscheiden.

Zheng und Fischer [23] entwickelten im Rahmen ihrer Untersuchung einen GPT-basierten Prototypen zur effizienten Informationsextraktion aus BIM-Modellen. Das Ergebnis ihrer Untersuchung ist ein virtueller Assistent mitsamt BIM-Viewer, welcher es ermöglicht, natürliche Spracheingaben etwaiger Nutzenden zur Informationsextraktion aus BIM-Modellen aufzugreifen und entsprechend durchzuführen. Das hierfür eigens entwickelte BIM-GPT Framework basiert u. a. auf der OpenAI API. Mit ihrem entwickelten Ansatz konnten sie das Potential von GPT-Ansätzen in Kombination mit BIM sowie die Machbarkeit der automatisierten Informationsextraktion aus BIM-Modellen mittels eines GPT-Modells aufzeigen. Eine fehlende Evaluierung der tatsächlichen Nutzbarkeit des Ansatzes sollte in weiterführenden Untersuchungen überprüft werden. [23] Lin [24] stellt in seiner Veröffentlichung einen entwickelten konversationellen Chatbot vor, welcher für Baustellen-Szenarien als explizite Verwendung durch Baustellenarbeitende angedacht ist. Mit dem Konversations-basierten Ansatz sollen etwaige Technikhürden bei der Informationsextraktion aus einem BIM-Modell reduziert werden. Der entwickelte Ansatz basiert auf dem bestehenden Chatbot JULIE und ermöglicht, Nutzerfragen zu verstehen, die notwendigen Informationen aus einem BIM-Modell, vorliegend als Datenbank, zu extrahieren und schlussendlich grafisch auszugeben. Im Fokus der Studie stand die Abfrage gezielter Informationen zu Bauteilabmessungen sowie zur Bestimmung des Geschossdeckenabstands. Die korrekte Abfrage gezielter Informationen konnte mit einer ausreichenden Genauigkeit anhand eines Fallbeispiels dargestellt werden. Schwächen des Ansatzes liegen laut Lin [24] in der Abfrage von Informationen, welche über Bauteilabmessungen hinaus gehen. Der Ansatz müsste entsprechend erweitert werden, um die Abfrage aus weiteren gängigen Datenformaten, wie \*.docx oder \*.pdf, zu inkludieren. [24]

Ko et al. [25] stellen Ansätze zur generativen KI-basierten BIM-Modellierung für den architektonischen Entwurfsprozess vor. Diesbezüglich verwenden sie zunächst ChatGPT zur Ausgabe der architektonischen Anforderungen des angedachten Entwurfs als Code für die jeweilige Modellierungssoftware-spezifische API. Der Code wird folglich über die visuellen Programmieroberflächen der jeweiligen Modellierungssoftwares ausgeführt. Ko et al. unternahmen zwei exemplarische Untersuchungen: Zum einen wurde ChatGPT in Verbindung mit der BIM-Autorensoftware Autodesk Revit und der visuellen Programmieroberfläche Dynamo verwendet. Zum anderen erfolgte die Untersuchung in Kombination mittels der 3D-Modellierungssoftware Rhino sowie deren visuelle Programmieroberfläche Grasshopper. Im nächsten Schritt wurde das bereits auf dem Markt verfügbare Software-Tool Veras zur KI-unterstützten Visualisierungserstellung genutzt. Veras

ermöglicht die Erstellung von 2D-Renderings aufbauend auf BIM-Modellen und nutzt einen generativen KI-Ansatz, welcher die natürlichen Spracheingaben der Nutzenden umsetzt und so beispielsweise den Hintergrund der Szenerie oder die Materialien gezielter Bauteile abändert. Anhand ihrer experimentellen Untersuchungen konnten Ko et al. feststellen, dass die vorgestellten Ansätze zu einem effizienteren sowie kreativeren architektonischen Entwurfsprozess beitragen können. Unklarheiten der Tauglichkeit des Ansatzes ergeben sich für denkbar komplexere Entwurfsideen. [25]

Zur Unterstützung des architektonischen Entwurfsprozesses eruierten Jang und Lee [26] die Verwendung von generativen KI-Ansätzen. Ihr Software-basierter Ansatz wurde gänzlich unter Verwendung von ChatGPT entwickelt und umfasst drei Komponenten. Zwei der Komponenten ermöglichen einerseits den Export eines BIM-Modells aus Autodesk Revit als \*.xml-Datei sowie den späteren Import der modifizierten \*.xml-Datei mit dem Ziel einer entsprechenden Anpassung des Modells. Zwischen den beiden Export- und Import-Komponenten befindet sich die Komponente zum generativen, KI-basierten interaktiven, architektonischen Design. Jang und Lee validierten ihre Komponenten in einem Fallbeispiel zur Planung von Wandaufbauten im Entwurfsprozess. Aufbauend auf im Revit-Modell hinterlegten Attributen zur örtlichen Platzierung der Bauteile konnte der Ansatz sinnvolle Vorschläge zur Wanddetaillierung aus einem vorab definierten Pool von Möglichkeiten ausgeben. Ihr entwickelter Ansatz bekräftigt die grundlegende Verwendung von generativen Chatbots im architektonischen Entwurfsprozess, weist jedoch aufgrund des gering skalierten Fallbeispiels eine bedingte Praxistauglichkeit auf. [26]

Der von Saka et al. [14] entwickelte Prototyp ermöglicht sowohl die Informationsextraktion aus einem BIM-Modell als auch die Unterstützung des architektonischen Entwurfsprozesses. Grundlegend gestattet das Tool eine auf ChatGPT-basierende Interaktion mit einem BIM-Modell. Im vorliegenden Fallbeispiel werden die Nutzenden im architektonischen Entwurfsprozess unterstützt, indem Vorschläge zur Wahl des optimalen Materials für ein spezifisches BIM-Objekt ausgegeben werden. ChatGPT wird in diesem Ansatz konkret verwendet, um die Nutzereingaben zu erkennen, die notwendigen Bauteil-bezogenen Informationen aus einer JSON-Datei des BIM-Modells zu extrahieren sowie Materialvorschläge aufbauend auf den Anforderungen der nutzenden Person auszugeben. Hierbei werden sowohl die Anforderungen aus der Nutzereingabe als auch die im BIM-Modell gegebenen Bedingungen beachtet. Weiterhin ermöglicht der Ansatz, das zu betrachtende Bauteil im integrierten BIM-Viewer des Tools farblich hervorzuheben. Dank des entwickelten Prototypen konnten Saka et al. [14] aufzeigen, dass ihr ChatGPT-basierter Ansatz eine Extraktion der geforderten Informationen aus dem BIM-Modell ermöglicht sowie umfangreiche und sinnvolle Vorschläge zu Materialien erzeugen

kann. Weiterführende Studien könnten sich insbesondere der Optimierung des ChatGPT Modells im Sinne einer Effizienzoptimierung und genauerer Nutzerantworten widmen. [14]

Die Praxisrecherche fand auf dem Suchportal Google sowie über die Social Media Plattformen YouTube und LinkedIn mit den Suchthermen *ChatGPT BIM*, *ChatGPT Revit*, *ChatGPT ifc*, *Text-to-BIM* statt. Bei den ausgewählten recherchierten Ergebnissen wird im Folgenden grundlegend zwischen auf dem Markt frei verfügbaren sowie etablierten Ansätzen und einzelnen prototypischen Implementierungen unterschieden. Weiterhin spiegelt sich die Einordnung der Ansätze nach der Anwendung zur Informationsextraktion sowie zur Unterstützung des architektonischen Entwurfsprozesses wider. Die identifizierten etablierten Ansätze beinhalten Chatbot-Konzepte, wohingegen im Rahmen der identifizierten prototypischen Implementierungen zumeist generative Chatbots zur eigentlichen Umsetzung des Ansatzes verwendet wurden. Letztere Ansätze konnten insbesondere in den Social Media Portalen identifiziert werden.

Im Kontext des zu erwerbenden Tools Hypar wird der Begriff Text-to-BIM verwendet. Es handelt sich hierbei um ein webbasiertes Tool mit einer interaktiven graphischen Oberfläche. Aufbauend auf Nutzereingaben können Bauwerksinformationsmodelle erzeugt werden kann. Hypar ermöglicht dies mittels eines NLP Ansatzes, welcher die Nutzereingaben verarbeitet und dies entsprechend als Modell umsetzt. Die Plattform eignet sich zur Erstellung von Gebäudeentwürfen und bietet weitere Möglichkeiten, beispielsweise zur Tageslichtsimulation, an. Darüber hinaus sind Schnittstellen zu gängigen Modellierungssoftwares sowie Exportmöglichkeiten, beispielsweise als \*.ifc-Datei, gegeben. [5, 27]

Das käufliche Tool Veras ermöglicht eine KI-unterstützte Visualisierung von BIM-Modellen und steht für gängige Modellierungssoftwares als Add-In zur Verfügung. Aufbauend auf einem BIM-Modell und spezifischen Nutzereingaben können mit Veras KI-basierte 2D-Visualisierungen erstellt werden. Veras generativer Ansatz verarbeitet die Nutzereingaben entsprechend und kann somit beispielsweise Hintergrundszenerien erschaffen oder Bauteile mit individuellen Materialien belegen. [6, 25] Konkrete Informationen zu den zugrundeliegenden technologischen Prinzipien konnten, neben der unspezifischen Nennung der Verwendung eines KI-Ansatzes [6], nicht identifiziert werden. Es ist jedoch davon auszugehen, dass die Nutzereingaben mittels eines NLP Ansatzes verarbeitet werden und gegebenenfalls ein weiterer KI-basierter Ansatz zur Ausgabe der 2D-Visualisierungen gegeben ist. Speckle ist eine open source Datenplattform für die AEC Branche und kann als Common Data Environment für BIM-Projekte verwendet werden. Es baut nicht auf dem gängigen IFC Standard auf. Stattdessen wird eine API zur Verfügung gestellt, sodass spezifische Schnittstellen zu

weiteren Anwendungssoftwares geschaffen werden können. Es existieren bereits vielzählige Schnittstellen mit gängigen Softwareprodukten der AEC Branche. [28] In einem Blogartikel stellt Speckle einen prototypischen ChatGPT-basierten, konversationellen Ansatz für seine Plattform vor. Dieser gestattet durch die Eingabe nutzerspezifischer Fragen eine Informationsausgabe aus ausgewählten BIM-Objekten. Zudem wird die Interaktion und Informationsextraktion aus angehängten \*.pdf-Dateien ermöglicht. [29]

Bei den betrachteten prototypischen Implementierungsansätzen aus den Social Media Portalen [Vgl. 10, 11, 30] handelt es sich um Szenarien der ChatGPT Nutzung im Kontext von Autodesk Revit und dessen visueller Programmoberfläche Dynamo. Die erfolgreich umgesetzten Beispiele widmen sich der BIM-Modell-Erstellung bzw. -Modifikation. ChatGPT wurde für die Erstellung entsprechender Python Codes verwendet, welche im nächsten Schritt über Dynamo implementiert wurden. [Vgl. 10, 11, 30]

Die identifizierten prototypischen Ansätze aus dem Social Media Portal LinkedIn [Vgl. 7–9] sowie dem Forum der OSArch Community [Vgl. 31] thematisieren die Verwendung von ChatGPT im Umgang mit \*.ifc-Dateien. ChatGPT wurde wiederum für die Erstellung von Programmiercodes verwendet. Die auf LinkedIn geposteten Ansätze demonstrieren insbesondere die Informationsextraktion aus \*.ifc-Dateien und eine entsprechende Nachbearbeitung [Vgl. 7–9]. Der Forumsbeitrag der OSArch Community beschreibt die grundlegende Machbarkeit der Geometriegenerierung und eine darauffolgende Speicherung als \*.ifc-Datei mittels eines Python-Skripts, welches auf Basis von ChatGPT-Eingaben entwickelt wurde [Vgl. 31].

### 3 Fallstudie

Mit Hilfe der vorliegenden Fallstudie werden weitere Möglichkeiten zur Verwendung von ChatGPT zur Informationsextraktion aus BIM-Modellen vorgestellt. Zentrales Augenmerk liegt auf der Fragestellung, inwiefern ChatGPT im BIM-spezifischen Kontext zur Automatisierung von Prozessen der benutzerspezifischen Informationsausgabe genutzt werden kann. Hinzukommend ergaben sich die folgenden Fragestellungen:

- Welche Herausforderungen ergeben sich bei der Verwendung von ChatGPT zur Automatisierung von Prozessen der Informationsextraktion im BIM-spezifischen Kontext?
- Ergibt sich ein Mehrwert für nutzende Personen bei der Umsetzung des genannten Szenarios?

### 3.1 Fallbeschreibung



**Bild 1** Ausgewähltes BIM-Modell der Fallstudie

Das BIM-Modell der Fallstudie (siehe Bild 1) ist ein einfaches Modell aus der universitären Lehre des ICoM und liegt als proprietäre \*.rvt-Datei der BIM-Autorensoftware Autodesk Revit 2024 sowie als \*.ifc-Datei des genormten Standards IFC vor.

In dem fiktiven Fallbeispiel sollen für eine Mengenermittlung automatisiert die Volumina der strukturellen Bauteile des Bauwerks ermittelt werden. Konkret soll je Bauteiltyp die Anzahl der Bauteile und deren jeweilig summiertes Gesamtvolumen als \*.csv-Datei ausgegeben werden.

Zur Anwendung der BIM-Autorensoftware Autodesk Revit sowie zur generellen BIM-Anwendung, Programmierung und zum ChatGPT-Gebrauch wird von basalen Kenntnissen ausgegangen. Aufgrund vielversprechender Ansätze zur Nutzung von generativen Chatbots im BIM-spezifischen Kontext auf gängigen Social Media Plattformen [Vgl. 7–11, 30] soll ChatGPT für die Umsetzung der angestrebten Informationsextraktion verwendet werden. Bei der Realisierung des Ansatzes sollen die folgenden Anforderungen beachtet werden:

- Es sollen Ansätze für die beiden zur Verfügung stehenden Dateien des BIM-Modells erörtert werden.
- Es soll eine kostenlose Alternative zur proprietären BIM-Autorensoftware eruiert werden.
- Die gesuchten Informationen sollen aus Objekten der Objekttypen Wände, Geschossdecke, Fundamente, Stütze und Unterzüge ausgelesen werden.
- Die Entwicklungszeit soll möglichst klein gehalten werden.
- Der Ansatz soll benutzerfreundlich und wiederverwendbar sein.

### 3.2 Design der Fallstudie

Aufbauend auf der Literatur- sowie Praxisrecherche konnte festgestellt werden, dass ChatGPT unter anderem ausführbare Codes für die visuelle Programmieroberfläche Dynamo [Vgl. 10, 11, 25, 30] sowie für das IFC Toolkit *IfcOpenShell* [Vgl. 31] erzeugen kann. In Kombination mit den festgelegten Anforderungen ergaben sich die folgenden zu erörternden Strategien zur Umsetzung der Fallstudie:

- Strategie 1: Verwendung der \*.rvt-Datei – ChatGPT wird genutzt, um in Autodesk Revit 2024

über die Revit API und die visuelle Programmieroberfläche Dynamo die notwendigen Informationen aus der \*.rvt-Datei des BIM-Modells auszulesen und als \*.csv-Datei auszugeben.

- Strategie 2: Verwendung der \*.ifc-Datei – ChatGPT wird genutzt, um über einen openBIM-basierten Ansatz die notwendigen Informationen aus der \*.ifc-Datei des BIM-Modells auszulesen und als \*.csv-Datei auszugeben. Hierfür wird die Open Source Library *IfcOpenShell* verwendet. Die prototypische Implementierung erfolgt in der Entwicklungsumgebung Microsoft Visual Studio Code.

Zum Einsatz kam ChatGPT mit dem GPT-3.5 Modell. Für die initialen Eingaben in ChatGPT wurden bewusst ähnliche Formulierungen getroffen, welche sich lediglich hinsichtlich der jeweiligen Tools und ihrer Dateiformate unterscheiden. Hierdurch sollen gleiche Startbedingungen geschaffen werden. Der folgende Absatz zeigt die initiale Eingabeaufforderung. Die unterschiedlichen Textstellen der beiden Strategien sind durch eckige Klammern ange deutet.:

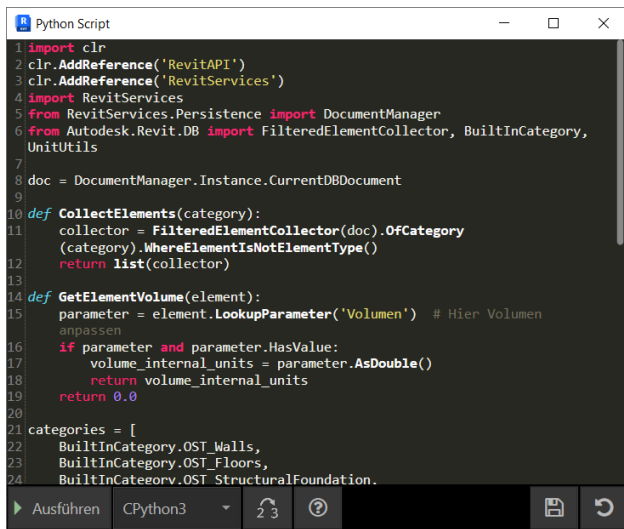
*Aus einem BIM-Modell, welches als [\*.rvt/\*.ifc]-Datei vorliegt, sollen die Volumina aller Wände, Geschossdecken, Fundamente, Stützen und Unterzüge ausgelesen werden. Konkret soll die Anzahl der Bauteile je Bauteiltyp und deren jeweilig summiertes Gesamtvolumen als \*.csv-Datei ausgegeben werden. Setze dieses Szenario [für Autodesk Revit] als Python Skript [zur Implementierung über Dynamo und] unter Verwendung [der Revit API/ IfcOpenShell] um.*

Um die zu erwartenden Ergebnisse der Ansätze hinsichtlich ihrer Richtigkeit zu überprüfen, wurden die Anzahl der Bauteile sowie die Summen der Volumina einerseits aus einer in Autodesk Revit 2024 automatisiert erstellen Materialliste sowie mittels händischer Überprüfung der eigentlichen \*.ifc-Datei entnommen.

### 3.2 Umsetzung der Fallstudie

Grundlegend lässt sich ein Python Code in Dynamo über einen Python Script Knoten (siehe Bild 2) eingeben sowie ausführen. Die Verwendung von benutzerspezifischen Python Knoten in Dynamo anstelle der generell zur Verfügung stehenden Dynamo Knoten und Verbindungen ermöglicht es, geforderte Tasks in einer reduzierten, grafisch übersichtlicheren Form darzustellen sowie auf umfangreichere Möglichkeiten bei der Programmierung zurückzugreifen. Hierbei können beispielsweise die Funktionen der Revit API aufgegriffen werden. Python ist in Dynamo als IronPython implementiert und ermöglicht somit eine reibungslose Interaktion mit dem zugrundeliegenden .NET-Framework. Aufgrund der Tatsache, dass IronPython ein open source Projekt ist, kann es vorkommen, dass die Implementierung nicht auf dem Stand der jeweils aktuellen Python Version ist. Hierdurch können gegebenenfalls

externe Bibliotheken aufgrund fehlender Kompatibilitäten nicht eingebunden werden. [32]



```

1 import clr
2 clr.AddReference('RevitAPI')
3 clr.AddReference('RevitServices')
4 import RevitServices
5 from RevitServices.Persistence import DocumentManager
6 from Autodesk.Revit.DB import FilteredElementCollector, BuiltInCategory,
  UnitUtils
7
8 doc = DocumentManager.Instance.CurrentDBDocument
9
10 def CollectElements(category):
11     collector = FilteredElementCollector(doc).OfCategory
12     (category).WhereElementIsNotElementType()
13     return list(collector)
14
15 def GetElementVolume(element):
16     parameter = element.LookupParameter('Volumen') # Hier Volumen
17     anpassen
18     if parameter and parameter.HasValue:
19         volume_internal_units = parameter.AsDouble()
20         return volume_internal_units
21     return 0.0
22
23 categories = [
24     BuiltInCategory.OST_Walls,
25     BuiltInCategory.OST_Floors,
26     BuiltInCategory.OST_StructuralFoundation,
27 ]
  
```

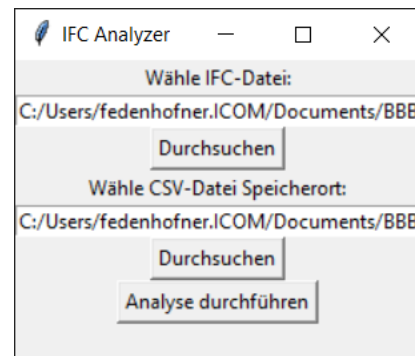
**Bild 2** Python Script Knoten mit entwickeltem Code

Ergebnis der initialen Eingabe des Starttextes in ChatGPT war ein erstes Python Skript zur Ausführung in Dynamo mitsamt beschreibenden Anmerkungen. In einer regen Kommunikation [33] mit dem Chatbot konnte innerhalb von 70 Minuten ein brauchbarer Ansatz erarbeitet werden, welcher den Anforderungen genügt. Schwierigkeiten bei der Umsetzung ergaben sich aus der Fehleranfälligkeit von ChatGPT bei der Wahl korrekter Revit API Funktionen. Durch die interaktive Kommunikation mit dem Chatbot können jedoch die Fehlermeldungen aus Dynamo zurückgespielt werden und der Code wird entsprechend angepasst. Ein wertvoller dem YouTube-Video [30, 5:50] entnommener Hinweis zum fälschlich von ChatGPT vorgeschlagenen Zugriff auf das aktuelle Revit Dokument wurde entsprechend beachtet und umgesetzt. Weitere Schwierigkeiten entstanden bei der Ermittlung der Volumina. Während zunächst das Volumen aufgrund der Wahl eines falschen Attributes nicht ausgelesen werden konnte, kam es im nächsten Schritt zu einer scheinbar falschen Ergebnisausgabe. Die vermeintlich in Kubikmeter ausgegeben summierten Volumina wichen vom tatsächlich zu erwartenden Ergebnis ab. Die Ursache des Problems lag in den internen Einheiten der Revit API. Diese greift auf den imperialen Fuß zur Längenbeschreibung zurück [34]. Demnach werden Volumina standardmäßig in Kubikfuß ermittelt. Nach einer entsprechenden Konvertierung der Ergebnisse konnten die geforderten Informationen in der korrekten Einheit als \*.csv-Datei ausgegeben werden.

Bei der zweiten, zu untersuchenden Strategie der Fallstudie wurde per ChatGPT ein Python Code generiert, welcher über die kostenlose Entwicklungsumgebung Microsoft Visual Studio Code implementiert wurde. Für den Python-basierten Umgang mit dem Datenstandard IFC wurde gezielt die open source Bibliothek *IfcOpenShell* ausgewählt. Jene Bibliothek ermöglicht dank zahlreicher

Funktionen unter anderem die Kreation und Modifikation von \*.ifc-Dateien sowie eine gezielte Informationsextraktion [35].

Nach Eingabe der initialen Aufforderung in ChatGPT ergab sich binnen Minuten ein erster brauchbarer Ansatz. Ein finaler Stand, welcher die Anforderungen bedient, konnte nach 80 Minuten Bearbeitungszeit erstellt werden [36]. Während des ersten Vorschlages mussten die Dateipfade zum Import sowie Export händisch eingegeben werden. Auf eine zusätzliche Anweisung zur Wahl einer interaktiven Dateieingabe sowie -ablage kreierte ChatGPT mit Hilfe der *tkinter* Bibliothek eine grafische Oberfläche für das Skript (siehe Bild 3).



**Bild 3** Grafische Bedienoberfläche des Python Skriptes

Maßgebende Probleme dieser Umsetzung ergaben sich bei der Entnahme der Volumina aus der \*.ifc-Datei. Trotz wiederholter Versuche konnte ohne eine gezielte Steuerung des Dialogs nicht das tatsächlich gesuchte Netto-Volumen der einzelnen Bauteile entnommen werden. Generell liegt das Volumen eines Bauteils als Objekt der Klasse *IfcQuantityVolume* vor. *IfcQuantityVolume* besitzt konkrete Attribute zur Beschreibung der jeweiligen Instanz. Unter dem Attribut *Name* wird festgelegt, ob es sich beispielsweise um das *NetVolume* (Netto-Volumen) oder *GrossVolume* (Brutto-Volumen) handelt. Weiterhin ist *IfcQuantityVolume* in einem Mengen-bezogenen Eigenschaftensatz *IfcElementQuantity* verortet. Ein Eigenschaftensatz wie *IfcElementQuantity* ist wiederum einem konkreten *IfcObject* mit der Verbindung *IfcRelDefinesByProperties* zugeordnet. [37]

Erst durch eine konkrete Eingabe jener semantischen Verbindungen in den Chatdialog konnte ein Python Code generiert werden, welcher den Anforderungen genügt. Schlussendlich konnte auf Nachfrage hin eine Anweisung ausgegeben werden, um das Skript mittels der Python Library *pyinstaller* als eigenständig ausführbare Datei umzusetzen.

Die erzeugten Python Skripte der Strategien sowie die entsprechenden Chatdialoge sind unter dem folgenden Link einzusehen:

[https://github.com/fjeden/BBB\\_Beitrag\\_Edenhofner\\_2023](https://github.com/fjeden/BBB_Beitrag_Edenhofner_2023)

### 3.3 Diskussion der Fallstudie

Bei der Umsetzung der beiden Strategien konnten in einem subjektiv empfunden akzeptablen Zeitrahmen von 70-80 Minuten zwei funktionierende Prototypen für die zur Verfügung stehenden Dateien des BIM-Modells entwickelt werden. Die Prototypen ermöglichen es, die gesuchten Informationen korrekt aus den geforderten Objekten auszulesen und als \*.csv-Datei auszugeben. Die Wiederverwendbarkeit beider Ansätze ist grundsätzlich gegeben. Das Ergebnis der Strategie 1 kann als \*.dyn-Datei weitergegeben werden und prinzipiell in anderen Projekten wiederverwendet werden. Mit dem Outcome der Strategie 2 kann sowohl das Python Skript über eine Entwicklungsumgebung als auch das kreierte ausführbare Programm ausgeführt werden. Letzteres stellt im Vergleich zu den anderen entwickelten Ansätzen eine äußerst benutzerfreundliche Option dar und kann hierdurch zu einer Reduktion etwaiger Vorbehalte bei der Anwendung der entwickelten Ansätze beitragen. Weiterhin erfüllt das Ergebnis der Strategie 2 das Kriterium einer kostenlosen Alternative zur proprietären BIM-Autorensoftware Autodesk Revit 2024. Die in der Fallstudie umgesetzten Strategien stellen denkbare Ansätze zur Verwendung von ChatGPT im BIM-spezifischen Kontext dar und bekräftigen das in der Literatur- und Praxisrecherche aufgezeigte Potential zu ebenjener Verwendung. Mit den entwickelten Prototypen werden Möglichkeiten aufgezeigt, inwiefern ChatGPT im BIM-spezifischen Kontext genutzt werden kann, um Prozesse der benutzerspezifischen Informationsausgabe zu automatisieren.

Die Umsetzung beider Strategien barg Herausforderungen, welche in einem praxisnahen Szenario von den Nutzenden zu beachten sind.

Die Hürden der Strategie 1 lagen vor allem in den von ChatGPT fälschlich vorgeschlagenen Revit API Funktionen sowie der Eigenheit von Autodesk Revit, einige interne Einheiten im imperialen System anzusetzen. Anhand der interaktiven Kommunikation mit dem Chatbot können etwaige Fehler bezüglich fälschlich angesetzter Funktionen zurückgespielt werden. Jedoch führte dies bei der Umsetzung unzuverlässig zum gewünschten Ergebnis. Des Weiteren ist unklar, ob ein brauchbarer Ansatz ohne den Hinweis des ermittelten YouTube Videos [30, 5:50] hätte entwickelt werden können. Die Problematik der imperialen Einheiten konnte aufgrund einer initial subjektiven Vermutung sowie einer weiterführenden Recherche aufgeklärt werden. Die aufgetretenen Hürden der Strategie 1 zeigen, dass bei der Umsetzung ähnlicher Szenarien ein grundlegendes Verständnis der Revit API von Vorteil ist. Weiterhin sind fundamentale Programmierkenntnisse von Vorteil, um die entwickelten Codes verstehen sowie etwaige Fehler beurteilen zu können. Kenntnisse zur Anwendung von Autodesk Revit und Dynamo sind obligatorisch.

Die aufgetretenen Hürden bei der Umsetzung der zweiten Strategie lagen maßgeblich in der wiederholt falschen

Wahl der korrekten IfcOpenShell Funktionen zur Extraktion der gesuchten Informationen. Aufbauend auf einer eigenständigen Recherche des IFC Standards sowie einer gezielten Steuerung des Chatdialogs konnte das gewünschte Ergebnis erreicht werden. Ein Grundverständnis des objektorientierten IFC Standards ist von Vorteil. Die Entwicklung eines ähnlichen Ansatzes bedarf sowohl Programmierkenntnisse als auch Kompetenzen zum Umgang mit einer Entwicklungsumgebung wie Microsoft Visual Studio Code. Letzteres birgt die Chance, den zu entwickelnden Ansatz mittels Debugging zu überprüfen und ermöglicht somit, Fehler im vorgeschlagenen Code leichter zu finden.

Bei dem vorliegenden Szenario der Informationsextraktion der Bauteilvolumina handelt es sich um eine simple Informationsabfrage zum Zweck der Mengenbestimmung. Die Umsetzung solcher Mengenbestimmungen kann von bestehenden Ansätzen wie Materiallisten in Autodesk Revit oder BIM-fähige AVA-Softwares entsprechend erfolgen. Es wurde ein reduziertes Szenario gewählt, um die grundlegende Machbarkeit aufzuzeigen sowie etwaige Herausforderungen und Chancen zu identifizieren. Limitationen der vorgestellten Möglichkeiten ergeben sich aus der geringen Komplexität des Fallbeispiels. In weiterführenden Untersuchungen sollten praxisnähere sowie komplizierte Abfragen zur Informationsextraktion ausgewählt werden. Eine grundlegende Praxistauglichkeit zur Verwendung von ChatGPT, um Prozesse der benutzerspezifischen Informationsausgabe aus BIM-Modellen zu automatisieren, konnte anhand der vorliegenden Fallstudie aufgezeigt werden. Bei der Umsetzung komplizierterer Szenarien der Informationsausgabe ist ein Basiswissen der zugrundeliegenden Ansätze unabdingbar.

Mehrwerte ergeben sich bei der Umsetzung ähnlicher Szenarien zum einen durch die Möglichkeit, benutzerspezifische Informationsausgaben zu automatisieren, ohne hierfür tiefgreifende Anwendungs- bzw. Programmierkenntnisse besitzen zu müssen. Zum anderen kann ein Ansatz resultieren, welcher den benutzerspezifischen Anforderungen genügt und wiederverwendet werden kann. Ferner bietet die Realisierung eines solchen Szenarios die Chance, sich umfangreich mit der BIM-Methode zu beschäftigen und einen wertvollen Wissenszuwachs zu generieren.

Bei der Wahl des geeigneten Ansatzes sollten die gegebenen Rahmenbedingungen beachtet werden. Darunter fallen beispielsweise die vorliegenden Software-Tools und Dateien sowie der vorhandene Wissensstand.

## 4 Fazit und Ausblick

Der vorliegende Beitrag zeigt Anwendungsmöglichkeiten von generativen Chatbots im BIM-spezifischen Kontext auf. Mit einer einleitenden Literatur- und Praxisrecherche konnte ein Einblick in den Stand der Wissenschaft sowie der Praxis gegeben werden. Die Ergebnisse aus

wissenschaftlichen Publikationen wurden hierbei um nicht-wissenschaftliche Suchergebnisse erweitert, um den wissenschaftlichen Status Quo durch auf dem Markt verfügbare Ansätze sowie prototypische Implementierungen und Fallbeispiele zu ergänzen. Die vorgestellten Ansätze wurden folgendermaßen differenziert: Es konnten einerseits Ansätze zur Informationsextraktion aus BIM-Modellen identifiziert werden. Andererseits wurden Ansätze erfasst, die den architektonischen, BIM-basierten Entwurfsprozess unterstützen. Darüber hinaus konnte eine Mischkategorie erkannt werden. Zu unterscheiden gilt außerdem, ob generative Chatbots für die Entwicklung des jeweiligen Ansatzes verwendet wurden oder ob in dem Ansatz Komponenten eines Chatbots integriert wurden.

Für die ausgewählten Umsetzungsstrategien der BIM-basierten Informationsausgaben konnten zwei funktionsfähige Prototypen entwickelt werden.

Die erfolgreiche Umsetzung der gewählten Strategien zeigt das Potential zur Verwendung von ChatGPT im BIM-spezifischen Kontext. Herausforderungen bei der Umsetzung ergaben sich allem voran durch von ChatGPT falsch angesetzten Funktionen. Der interaktive Chatdialog ermöglicht es jedoch, Fehler zurückzuspielen und neue Lösungsvarianten zu generieren. Dennoch ist für eine erfolgreiche Umsetzung ähnlicher Szenarien ein Basiswissen der zu verwendenden Technologien empfehlenswert. Ein konkreter Mehrwert ergibt sich folglich dadurch, dass Prozesse automatisiert werden können, ohne tiefgreifende Anwendungs- bzw. Programmierkenntnisse besitzen zu müssen. Somit kann ein nutzbarer und wiederverwendbarer Ansatz entwickelt sowie umfangreiches Wissen zur BIM-Methode generiert werden.

In weiteren Untersuchungen sollten komplexere Fallbeispiele gewählt werden, um die Praxistauglichkeit von generativen Chatbots im BIM-spezifischen Kontext umfassender zu validieren. Ebenso sollten verschiedene Chatbot-Ansätze mit gegebenenfalls neueren Sprachmodellen eruiert werden. Nicht zu missachten bei der Entwicklung weiterer Chatbot-basierter Ansätze sind potentiell auftretende Halluzinationen bei der Chatbot-Ausgabe. In Abhängigkeit der Tragweite eines denkbar realistischen Anwendungsszenarios der AEC Branche können mögliche Halluzinationen schwerwiegenden Auswirkungen haben. Es empfiehlt sich daher, Entwicklungen umfangreich zu validieren, bevor sie ihre Anwendung in der Praxis finden.

## 5 Literatur

- [1] M. Abdullah, A. Madain und Y. Jararweh, „ChatGPT: Fundamentals, Applications and Social Impacts“ in *2022 Ninth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, Milan, Italy, 2022, S. 1–8, doi: 10.1109/SNAMS58071.2022.10062688.
- [2] S. Jalil, S. Rafi, T. D. LaToza, K. Moran und W. Lam, „ChatGPT and Software Testing Education: Promises & Perils“ in *2023 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, Dublin, Ireland, 2023, S. 4130–4137, doi: 10.1109/ICSTW58534.2023.00078.
- [3] E. Kasneci *et al.*, „ChatGPT for good? On opportunities and challenges of large language models for education“, *Learning and Individual Differences*, Jg. 103, S. 102274, 2023, doi: 10.1016/j.lindif.2023.102274.
- [4] A.Shaji George, A.S.Hovan George und A.S.Gabrio Martin, „A Review of ChatGPT AI's Impact on Several Business Sectors“, 2023, doi: 10.5281/ZENODO.7644359.
- [5] Hypar Inc, *HYPAR*. [Online]. Verfügbar unter: <https://hypar.io/> (Zugriff am: 18. August 2023).
- [6] Evolve Lab, *VERAS*. [Online]. Verfügbar unter: <https://www.evolveai.io/veras> (Zugriff am: 18. August 2023).
- [7] I. Fahdah. [Online]. Verfügbar unter: [https://www.linkedin.com/feed/update/urn:li:activity:7079789208801882113?updateEntityUrn=urn%3Ali%3Afs\\_updateV2%3A%28urn%3Ali%3Aactivity%3A7079789208801882113%2CFEED\\_DETAIL%2CEMPTY%2CDEFAULT%2Cfalse%29](https://www.linkedin.com/feed/update/urn:li:activity:7079789208801882113?updateEntityUrn=urn%3Ali%3Afs_updateV2%3A%28urn%3Ali%3Aactivity%3A7079789208801882113%2CFEED_DETAIL%2CEMPTY%2CDEFAULT%2Cfalse%29) (Zugriff am: 18. August 2023).
- [8] D. Garbuzenko. [Online]. Verfügbar unter: [https://www.linkedin.com/feed/update/urn:li:activity:7052553883205726209?updateEntityUrn=urn%3Ali%3Afs\\_updateV2%3A%28urn%3Ali%3Aactivity%3A7052553883205726209%2CFEED\\_DETAIL%2CEMPTY%2CDEFAULT%2Cfalse%29](https://www.linkedin.com/feed/update/urn:li:activity:7052553883205726209?updateEntityUrn=urn%3Ali%3Afs_updateV2%3A%28urn%3Ali%3Aactivity%3A7052553883205726209%2CFEED_DETAIL%2CEMPTY%2CDEFAULT%2Cfalse%29) (Zugriff am: 18. August 2023).
- [9] V. Scopece. [Online]. Verfügbar unter: [https://www.linkedin.com/feed/update/urn:li:activity:7019619105217347584?updateEntityUrn=urn%3Ali%3Afs\\_updateV2%3A%28urn%3Ali%3Aactivity%3A7019619105217347584%2CFEED\\_DETAIL%2CEMPTY%2CDEFAULT%2Cfalse%29](https://www.linkedin.com/feed/update/urn:li:activity:7019619105217347584?updateEntityUrn=urn%3Ali%3Afs_updateV2%3A%28urn%3Ali%3Aactivity%3A7019619105217347584%2CFEED_DETAIL%2CEMPTY%2CDEFAULT%2Cfalse%29) (Zugriff am: 18. August 2023).
- [10] The BIM Coordinator, *Revit 2024 and Python | Streamlining Model Setup with ChatGPT | Building Smarter, Faster Models*. [Online]. Verfügbar unter: <https://www.youtube.com/watch?v=ZCU70AbFD24> (Zugriff am: 18. August 2023).
- [11] thisendup76, *Chat GPT - success story*. [Online]. Verfügbar unter: [https://www.reddit.com/r/bim/comments/10zssra/chat\\_gpt\\_success\\_story/](https://www.reddit.com/r/bim/comments/10zssra/chat_gpt_success_story/) (Zugriff am: 18. August 2023).
- [12] Bundesministerium für Verkehr und digitale Infrastruktur, „BIM4INFRA2020\_AP4\_Teil8“, 2019. [Online]. Verfügbar unter: [https://bim4infra.de/wp-content/uploads/2019/07/BIM4INFRA2020\\_AP4\\_Teil8.pdf](https://bim4infra.de/wp-content/uploads/2019/07/BIM4INFRA2020_AP4_Teil8.pdf)



- [13] Bundesinstitut für Bau-, Stadt- und Raumforschung (BBSR) im Bundesamt für Bauwesen, *Beitrag der Digitalisierung zur Produktivität in der Baubranche*, 2019. Aufl. Bonn, 2019.
- [14] A. Saka *et al.*, „GPT Models in Construction Industry: Opportunities, Limitations, and a Use Case Validation“, 30. Mai 2023. [Online]. Verfügbar unter: <https://arxiv.org/pdf/2305.18997>.
- [15] F. García-Peñalvo und A. Vázquez-Ingelmo, „What Do We Mean by GenAI? A Systematic Mapping of The Evolution, Trends, and Techniques Involved in Generative AI“, *IJIMAI*, In Press, In Press, S. 1, 2023, doi: 10.9781/ijimai.2023.07.006.
- [16] Midjourney, *Midjourney*. [Online]. Verfügbar unter: <https://www.midjourney.com/home/> (Zugriff am: 3. September 2023).
- [17] D. Khurana, A. Koli, K. Khatter und S. Singh, „Natural language processing: state of the art, current trends and challenges“ (eng), *Multimedia tools and applications*, Jg. 82, Nr. 3, S. 3713–3744, 2023, doi: 10.1007/s11042-022-13428-4.
- [18] OpenAI, *Introducing ChatGPT*. [Online]. Verfügbar unter: <https://openai.com/blog/chatgpt> (Zugriff am: 3. September 2023).
- [19] Y. Shen *et al.*, „ChatGPT and Other Large Language Models Are Double-edged Swords“ (eng), *Radiology*, Jg. 307, Nr. 2, e230163, 2023, doi: 10.1148/radiol.230163.
- [20] S. S. Gill und R. Kaur, „ChatGPT: Vision and challenges“, *Internet of Things and Cyber-Physical Systems*, Jg. 3, S. 262–271, 2023, doi: 10.1016/j.iotcps.2023.05.004.
- [21] D. Ghosh, D. Soni, S. Gopikanna, N. A Nambiar, Y. S. Apoorva und S. Kumarasamy, „Interactive Design Automation Framework Using Large Language Model and Past Data“, International Conference on Cement and Building Concrete for a sustainable and Resilient infrastructure, NIT, Warangal, India, März 2023. [Online]. Verfügbar unter: [https://www.researchgate.net/publication/370427479\\_Interactive\\_Design\\_Automation\\_Framework\\_Using\\_Large\\_Language\\_Model\\_and\\_Past\\_Data](https://www.researchgate.net/publication/370427479_Interactive_Design_Automation_Framework_Using_Large_Language_Model_and_Past_Data). Zugriff am: 3. September 2023.
- [22] Brady D Lund, „A Brief Review of ChatGPT: Its Value and the Underlying GPT Technology“, 2023. [Online]. Verfügbar unter: [https://www.researchgate.net/profile/Brady-Lund/publication/366809571\\_A\\_Brief\\_Review\\_of\\_ChatGPT\\_Its\\_Value\\_and\\_the\\_Underlying\\_GPT\\_Technology/links/64986a2b8de7ed28ba56a253/A-Brief-Review-of-ChatGPT-Its-Value-and-the-Underlying-GPT-Technology.pdf](https://www.researchgate.net/profile/Brady-Lund/publication/366809571_A_Brief_Review_of_ChatGPT_Its_Value_and_the_Underlying_GPT_Technology/links/64986a2b8de7ed28ba56a253/A-Brief-Review-of-ChatGPT-Its-Value-and-the-Underlying-GPT-Technology.pdf). Zugriff am: 5. September 2023.
- [23] J. Zheng und M. Fischer, „BIM-GPT: a Prompt-Based Virtual Assistant Framework for BIM Information Retrieval“, 2023.
- [24] W. Y. Lin, „Prototyping a Chatbot for Site Managers Using Building Information Modeling (BIM) and Natural Language Understanding (NLU) Techniques“ (eng), *Sensors (Basel, Switzerland)*, Jg. 23, Nr. 6, 2023, doi: 10.3390/s23062942.
- [25] J. Ko, J. Ajibefun und W. Yan, „Experiments on Generative AI-Powered Parametric Modeling and BIM for Architectural Design“, 1. Aug. 2023. [Online]. Verfügbar unter: <https://arxiv.org/pdf/2308.00227>.
- [26] S. Jang und G. Lee, „Interactive Design by Integrating a Large Pre-Trained Language Model and Building Information Modeling“, 25. Juni 2023. [Online]. Verfügbar unter: <https://arxiv.org/pdf/2306.14165>.
- [27] M. Day, *Hypar: text-to-BIM and beyond*. [Online]. Verfügbar unter: <https://aecmag.com/ai/hypar-text-to-bim-and-beyond/> (Zugriff am: 18. August 2023).
- [28] P. Poinet, D. Stefanescu und E. Papadonikolaki, „Collaborative Workflows and Version Control Through Open-Source and Distributed Common Data Environment“ in *Lecture Notes in Civil Engineering, Proceedings of the 18th International Conference on Computing in Civil and Building Engineering*, E. Toledo Santos und S. Scheer, Hg., Cham: Springer International Publishing, 2021, S. 228–247, doi: 10.1007/978-3-030-51295-8\_18.
- [29] J. Broughton, *AI-Powered Conversations with Speckle*. [Online]. Verfügbar unter: <https://speckle.systems/blog/ai-powered-conversations-with-speckle/> (Zugriff am: 18. August 2023).
- [30] Design Tech Unraveled, *How to use Chat GPT to Delete All Walls in a Revit Model (with Python)*. [Online]. Verfügbar unter: <https://www.youtube.com/watch?v=vaHmuX1htbU> (Zugriff am: 18. August 2023).
- [31] Martin156131, *[IfcOpenShell-Python] How to create a cone in an ifc file? i asked chatgpt*. [Online]. Verfügbar unter: <https://community.osarch.org/discussion/1330/ifc-openshell-python-how-to-create-a-cone-in-an-ifc-file-i-asked-chatgpt> (Zugriff am: 18. August 2023).
- [32] O. Green, *Dynamo Python Primer*. [Online]. Verfügbar unter: <https://dynamopythonprimer.gitbook.io/dynamopython-primer/> (Zugriff am: 19. August 2023).
- [33] ChatGPT, *Dialog zu Strategie 1*. [Online]. Verfügbar unter: <https://chat.openai.com/share/a0f56009-941f-46bc-968a-70fccb28352>.
- [34] Autodesk Inc., *Autodesk Revit 2023 Help*. [Online]. Verfügbar unter:

<https://help.autodesk.com/view/RVT/2023/ENU/>  
(Zugriff am: 19. August 2023).

- [35] IfcOpenShell, *IfcOpenShell: The open source IFC toolkit and geometry engine*. [Online]. Verfügbar unter: <https://ifcopenshell.org/index.html> (Zugriff am: 20. August 2023).
- [36] ChatGPT, *Dialog zu Strategie 2*. [Online]. Verfügbar unter: <https://chat.openai.com/share/275d8cbd-02c9-40c9-89df-540fb31b70ea> (Zugriff am: 26. August 2023).
- [37] buildingSMART International Ltd., *4.2.3 Quantity Sets*. [Online]. Verfügbar unter: [https://standards.buildingsmart.org/IFC/DEV/IFC4\\_2/FINAL/HTML/schema/templates/quantity-sets.htm](https://standards.buildingsmart.org/IFC/DEV/IFC4_2/FINAL/HTML/schema/templates/quantity-sets.htm) (Zugriff am: 20. August 2023).

# DuEPublico

Duisburg-Essen Publications online

UNIVERSITÄT  
DUISBURG  
ESSEN

*Offen im Denken*

ub | universitäts  
bibliothek

Dieser Text wird via DuEPublico, dem Dokumenten- und Publikationsserver der Universität Duisburg-Essen, zur Verfügung gestellt. Die hier veröffentlichte Version der E-Publikation kann von einer eventuell ebenfalls veröffentlichten Verlagsversion abweichen.

**DOI:** 10.17185/duepublico/79111

**URN:** urn:nbn:de:hbz:465-20231017-143134-3

In: Tagungsband zum 32. BBB-Assistent:innentreffen 2023: 04.10.2023 - 06.10.2023,  
Universität Duisburg-Essen.



Dieses Werk kann unter einer Creative Commons Namensnennung - Nicht kommerziell 4.0 Lizenz (CC BY-NC 4.0) genutzt werden.