

# Toward Context-based Text-to-3D Scene Generation

Dissertation  
zur Erlangung des Doktorgrades  
der Naturwissenschaften

vorgelegt beim Fachbereich Informatik und Mathematik  
der Johann Wolfgang Goethe-Universität  
in Frankfurt am Main

von  
Alexander Henlein  
aus Frankfurt am Main

Frankfurt 2022  
(D 30)

vom Fachbereich Informatik und Mathematik der

Johann Wolfgang Goethe - Universität als Dissertation angenommen.

Dekan: Prof. Dr. Martin Möller

Gutachter: Prof. Dr. Alexander Mehler und Prof. Dr. Visvanathan Ramesh

Datum der Disputation: 24.03.2023



# License

©2022 by Alexander Henlein. All Rights Reserved.

This work is licensed under a Creative Commons  
“Attribution-NonCommercial 4.0 International” license.





# Abstract

People can describe spatial scenes with language and, vice versa, create images based on linguistic descriptions. However, current systems do not even come close to matching the complexity of humans when it comes to reconstructing a scene from a given text. Even the ever-advancing development of better and better Transformer-based models has not been able to achieve this so far. This task, the automatic generation of a 3D scene based on an input text, is called text-to-3D scene generation. The key challenge, and focus of this dissertation, now relate to the following topics:

- (a) Analyses of how well current language models understand spatial information, how static embeddings compare, and whether they can be improved by anaphora resolution.
- (b) Automated resource generation for context expansion and grounding that can help in the creation of realistic scenes.
- (c) Creation of a VR-based text-to-3D scene system that can be used as an annotation and active-learning environment, but can also be easily extended in a modular way with additional features to solve more contexts in the future.
- (d) Analyze existing practices and tools for digital and virtual teaching, learning, and collaboration, as well as the conditions and strategies in the context of VR.

In the first part of this work, we could show that static word embeddings do not benefit significantly from pronoun substitution. We explain this result by the loss of contextual information, the reduction in the relative occurrence of rare words, and the absence of pronouns to be substituted. But we were able to we have shown that both static and contextualizing language models appear to encode object knowledge, but require a sophisticated apparatus to retrieve it. The models themselves in combination with the measures differ greatly in terms of the amount of knowledge they allow to extract. Classifier-based variants perform significantly better than the unsupervised methods from bias research, but this is also due to overfitting. The resources generated for this evaluation are later also an important component of point three.

In the second part, we present AffordanceUPT, a modularization of UPT trained on the HICO-DET dataset, which we have extended with Gibsonian/telic annotations. We then show that AffordanceUPT can effectively make the Gibsonian/telic distinction and that the model learns other correlations in the data to make such distinctions (e.g., the presence of hands in the image) that have important implications for grounding images to language.

The third part first presents a VR project to support spatial annotation respectively IsoSpace. The direct spatial visualization and the immediate interaction with the 3D objects should make the labeling more intuitive and thus easier. The project will later be incorporated as part of the Semantic Scene Builder (SESB). The project itself in turn relies on the TEXT2SCENEVR presented here for generating spatial hypertext, which in turn is based on the VANNOTATOR. Finally, we introduce Semantic Scene Builder (SESB), a VR-based text-to-3D scene framework using Semantic Annotation Framework (SemAF) as a scheme for annotating semantic relations. It integrates a wide range of tools and resources by utilizing SemAF and UIMA as a unified data structure to generate 3D scenes from textual descriptions and also supports annotations. When evaluating SESB against another state-of-the-art tool, it was found that our approach not only performed better, but also allowed us to model a wider variety of scenes. The final part reviews existing practices and tools for digital and virtual teaching, learning, and collaboration, as well as the conditions and strategies needed to make the most of technological opportunities in the future.

# Zusammenfassung

Menschen können räumliche Szenen mit Sprache beschreiben und umgekehrt Bilder auf der Grundlage von sprachlichen Beschreibungen erzeugen. Aktuelle Systeme kommen jedoch nicht einmal annähernd an die Komplexität von Menschen heran, wenn es darum geht, eine Szene aus einem gegebenen Text zu rekonstruieren. Auch die immer weiter fortschreitende Entwicklung immer besserer Transformator-basierter Modelle konnte dies bisher nicht leisten. Diese Aufgabe, die automatische Generierung einer 3D-Szene auf der Grundlage eines Eingabetextes, wird text-to-3D scene-Generierung genannt. Die zentrale Herausforderung und der Schwerpunkt dieser Dissertation beziehen sich nun auf die folgenden Themen:

- (a) Analysen, wie gut aktuelle Sprachmodelle räumliche Informationen verstehen, wie statische Einbettungen im Vergleich dazu abschneiden und ob sie durch Anaphora-Auflösung verbessert werden können.
- (b) Automatisierte Ressourcengenerierung für Kontexterweiterung und Erdung, die bei der Erstellung realistischer Szenen helfen können.
- (c) Schaffung eines VR-basierten text-to-3D scene-Systems, das als Annotations- und Active-Learning-Umgebung verwendet werden kann, aber auch leicht auf modulare Weise mit zusätzlichen Funktionen erweitert werden kann, um in Zukunft weitere Kontexte zu lösen.
- (d) Analysieren Sie bestehende Praktiken und Werkzeuge für digitales und virtuelles Lehren, Lernen und Kollaboration sowie die Bedingungen und Strategien im Kontext von VR.

Im ersten Teil dieser Arbeit konnten wir zeigen, dass statische Worteinbettungen nicht wesentlich von der Pronomenersetzung profitieren. Wir erklären dieses Ergebnis durch den Verlust von Kontextinformationen, die Verringerung des relativen Vorkommens seltener Wörter und das Fehlen von Pronomen, die ersetzt werden müssen. Wir konnten jedoch zeigen, dass sowohl statische als auch kontextualisierende Sprachmodelle Objektwissen zu kodieren scheinen, aber einen ausgeklügelten Apparat benötigen, um es abzurufen. Die Modelle selbst in Kombination mit den Maßnahmen unterscheiden sich stark in Bezug auf die Menge des Wissens, das sie zu extrahieren erlauben. Klassifikatorbasierte Varianten schneiden deutlich besser ab als die unüberwachten Methoden aus der Bias-Forschung, was aber auch auf Overfitting zurückzuführen ist. Die für diese Bewertung generierten Ressourcen sind später auch ein wichtiger Bestandteil von Punkt drei.

Im zweiten Teil stellen wir AffordanceUPT vor, eine Modularisierung von UPT, die auf dem HICO-DET-Datensatz trainiert wurde, den wir mit Gibsonien/telischen Annotationen erweitert haben. Wir zeigen dann, dass AffordanceUPT effektiv die Gibsonian/telic-Unterscheidung treffen kann und dass das Modell andere Korrelationen in den Daten erlernt, um solche Unterscheidungen zu treffen (z.B. das Vorhandensein von Händen im Bild), die wichtige Implikationen für die Erdung von Bildern mit Sprache haben.

Im dritten Teil wird zunächst ein VR-Projekt zur Unterstützung der räumlichen Annotation bzw. IsoSpace vorgestellt. Durch die direkte räumliche Visualisierung und die unmittelbare Interaktion mit den 3D-Objekten soll die Beschriftung intuitiver und damit einfacher werden. Das Projekt wird später als Teil des Semantic Scene Builders (SESB) integriert. Das Projekt selbst stützt sich wiederum auf die hier vorgestellte TEXT2SCENEVR zur Erzeugung von räumlichem Hypertext, die wiederum auf der VANNOTATOR basiert. Schließlich stellen wir den Semantic Scene Builder (SESB) vor, ein VR-basiertes text-to-3D scene-Framework, das das Semantic Annotation Framework (SemAF) als Schema für die Annotation semantischer Beziehungen verwendet. Es integriert eine Vielzahl von Werkzeugen und Ressourcen, indem es SemAF und UIMA als einheitliche Datenstruktur nutzt, um 3D-Szenen aus textuellen Beschreibungen zu generieren und auch Annotationen zu unterstützen. Bei der Bewertung von SESB im Vergleich zu einem anderen hochmodernen Tool zeigte sich, dass unser Ansatz nicht nur besser abschnitt, sondern auch eine größere Vielfalt von Szenen modellieren konnte. Der letzte Teil gibt einen Überblick über bestehende Praktiken und Werkzeuge für digitales und virtuelles Lehren, Lernen und Zusammenarbeiten sowie über die Bedingungen und Strategien, die erforderlich sind, um die technologischen Möglichkeiten in Zukunft optimal zu nutzen.

# Acknowledgment

During the preparation of this dissertation I received a lot of support and help from all sides.

I would first like to thank my supervisor, Professor Dr. Alexander Mehler, whose expertise, continuous support and patience was invaluable during my study. Your knowledge and experience have encouraged me throughout all my academic research and beyond. I gratefully acknowledge the funding received towards my thesis from the Stiftung Polytechnische Gesellschaft (SPTG) Main-Campus-doctus scholarship. By this I mean not only financial support, but also intangible support in the form of seminars, lectures and contacts that I have been able to establish.

I would also like to thank my colleague Giuseppe Abrami for your unconditional help with technical and content-related questions.

Last but not least, I would like to thank my parents for their wise advice and listening ear. Without you, this project would never have been possible.





# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	1
1.2	Challenges . . . . .	3
1.3	Contributions . . . . .	4
1.4	Dissertation Structure . . . . .	6
<b>2</b>	<b>Related Work</b>	<b>9</b>
2.1	Semantic Annotation Framework . . . . .	9
2.1.1	IsoSpace . . . . .	9
2.1.2	VoxML . . . . .	10
2.1.3	Other Modules . . . . .	10
2.2	Texttechnology Lab Annotation Architecture . . . . .	11
2.2.1	TextAnnotator . . . . .	12
2.2.2	TextImager . . . . .	12
2.2.3	VAnnotatoR . . . . .	13
2.3	Language Models . . . . .	13
2.3.1	Static Wordembeddings . . . . .	13
2.3.2	Transformers . . . . .	14
2.4	Text-to-3D Scene . . . . .	16
2.4.1	WordsEye . . . . .	16
2.4.2	SceneSeer . . . . .	16
2.4.3	Language-driven synthesis of 3D scenes from scene databases . .	16
2.4.4	SceneFormer . . . . .	17
2.4.5	Related Tasks . . . . .	17
<b>3</b>	<b>On the Influence of Coreference Resolution on Word Embeddings in Lexical-semantic Evaluation Tasks</b>	<b>19</b>
3.1	Introduction . . . . .	19
3.2	Related Work . . . . .	21
3.3	Coreference Substitutions for Enhancing Word Embeddings . . . . .	22
3.3.1	Extending the informational scope of window-based embeddings	22
3.3.2	Extending the informational scope of dependency-based embed- dings . . . . .	23
3.4	Experiments . . . . .	24
3.4.1	Data Sets and Models . . . . .	24
3.4.2	Pre-processing . . . . .	24

3.5	Evaluation . . . . .	24
3.5.1	Word Similarity . . . . .	24
3.5.2	Instances versus Concepts . . . . .	25
3.5.3	Feature Analysis . . . . .	26
3.5.4	Corpus Size . . . . .	27
3.5.5	Explanation of the results . . . . .	28
3.5.6	Discussion . . . . .	29
3.6	Conclusion . . . . .	29
<b>4</b>	<b>Transfer of ISOSpace into a 3D Environment for Annotations and Applications</b>	<b>31</b>
4.1	Introduction . . . . .	31
4.2	Related Work . . . . .	32
4.3	Our Current Project . . . . .	34
4.4	Conclusion . . . . .	36
<b>5</b>	<b>Text2SceneVR: Generating Hypertexts with VAnnotatoR as a Pre-processing Step for Text2Scene Systems</b>	<b>37</b>
5.1	Introduction . . . . .	37
5.2	Related Work . . . . .	41
5.3	From VANNOTATOR to Generating and Annotating Virtual Rooms . . . . .	43
5.3.1	VANNOTATOR’s Core Functionality . . . . .	45
5.3.2	VANNOTATOR’s TEXT2SCENEVR . . . . .	47
5.4	TEXT2SCENEVR’s Annotation Model . . . . .	51
5.5	Evaluation . . . . .	52
5.6	Future Work . . . . .	54
5.7	Summary . . . . .	55
<b>6</b>	<b>Digital learning, teaching, and collaboration in a time of ubiquitous quarantine</b>	<b>57</b>
6.1	Introduction . . . . .	57
6.2	VR environments: requirements analysis . . . . .	60
6.3	Related Work . . . . .	64
6.3.1	VR Platforms . . . . .	65
6.3.2	VANNOTATOR . . . . .	67
6.4	Analysis . . . . .	71
6.5	Discussion . . . . .	74
6.6	Conclusion . . . . .	74
<b>7</b>	<b>What do Toothbrushes do in the Kitchen? How Transformers Think our World is Structured</b>	<b>77</b>
7.1	Introduction . . . . .	77
7.2	Related Work . . . . .	79

7.3	Datasets Used for Evaluation . . . . .	80
7.3.1	Spatial Containment . . . . .	80
7.3.2	Part-whole Relations . . . . .	80
7.3.3	Action-object Relations . . . . .	81
7.4	Approach . . . . .	81
7.4.1	Static Models . . . . .	81
7.4.2	Contextualized Models . . . . .	82
7.4.3	Similarity Measures . . . . .	82
7.4.4	Scoring Measures and Classifiers . . . . .	85
7.5	Experiments . . . . .	85
7.5.1	Model-related Observations . . . . .	87
7.5.2	Dataset-related Observations . . . . .	87
7.5.3	Relation Observation . . . . .	88
7.6	Discussion . . . . .	88
7.7	Conclusion . . . . .	89
<b>8</b>	<b>Grounding Human-Object Interaction to Affordance Behavior in Mul-</b>	
	<b>timodal Datasets</b> . . . . .	<b>97</b>
8.1	Introduction . . . . .	97
8.2	Related Work . . . . .	99
8.3	An Approach to Detecting Affordances . . . . .	100
8.3.1	Theory . . . . .	100
8.3.2	Annotation . . . . .	101
8.3.3	Models . . . . .	103
8.4	Evaluation & Analyses . . . . .	105
8.4.1	Evaluation of AffordanceUPT . . . . .	105
8.4.2	Evaluation of PoseContrast . . . . .	106
8.4.3	Analysis of AffordanceUPT Tokens . . . . .	110
8.4.4	Automated Habitat Annotation . . . . .	110
8.5	Discussion and Conclusions . . . . .	111
8.5.1	Future Work . . . . .	111
<b>9</b>	<b>Semantic Scene Builder: Towards a context sensitive Text-to-3D Scene</b>	
	<b>Framework</b> . . . . .	<b>119</b>
9.1	Introduction . . . . .	119
9.2	Related Work . . . . .	122
9.2.1	Text-to-3D Scene . . . . .	122
9.2.2	SemAF & IsoSpace . . . . .	123
9.3	Semantic Scene Builder . . . . .	125
9.3.1	Parsing . . . . .	126
9.3.2	Inference . . . . .	128
9.3.3	Generation . . . . .	129
9.3.4	Annotation . . . . .	129
9.4	Implementation . . . . .	129

## Contents

9.5	Evaluation . . . . .	130
9.6	Discussion & Future Work . . . . .	131
9.7	Conclusion . . . . .	132
<b>10</b>	<b>Conclusion</b>	<b>135</b>
10.1	Summary . . . . .	135
10.2	Future Work . . . . .	136
10.2.1	Spatial expectations . . . . .	136
10.2.2	Temporal expectations . . . . .	136
10.2.3	Cultural expectations . . . . .	137
10.2.4	Conceptual expectations . . . . .	137
10.2.5	Requirements-related expectations . . . . .	137
10.2.6	Process-related expectations . . . . .	137
10.2.7	Annotation . . . . .	138
10.2.8	Combination . . . . .	138
	<b>Appendix: Zusammenfassung</b>	<b>177</b>

# List of Figures

1.1	Areas of contextual expectations that are relevant for scene generation. . . . .	2
2.1	Highly simplified overview of the Texttechnology Lab Annotation Architecture. . . . .	12
2.2	Transformer architecture based of Figure 1 of Vaswani et al. (2017). The dotted red lines are residual connections (He et al., 2016). . . . .	14
3.1	Dependency trees of two consecutive sentences. . . . .	20
3.2	Decision Tree for classifying the error distribution on the H-Union dataset.	27
3.3	Error distribution with SVM. . . . .	28
4.1	IsoSpace annotation example. . . . .	32
4.2	Workflow for ISOSpace Annotation. . . . .	34
5.1	The software landscape into which VANNOTATOR is embedded. . . . .	44
5.2	VANNOTATOR resources window. . . . .	46
5.3	VANNOTATOR text window. . . . .	47
5.4	VANNOTATOR room creation. . . . .	48
5.5	VANNOTATOR wall creation. . . . .	49
5.6	VANNOTATOR doors and windows. . . . .	50
5.7	VANNOTATOR texturing. . . . .	50
5.8	VANNOTATOR evaluation annotation example. . . . .	51
5.9	VANNOTATOR control window. . . . .	52
5.10	TEXT2SCENEVR’s data model for the annotation of 3D scenes. . . . .	53
5.11	Example scene according to the example. . . . .	53
5.12	The results of the time measurement with 15 participants. . . . .	55
5.13	The average results of the UMUX test with 15 participants. . . . .	55
6.1	Example of a multimodal hypertext created with VANNOTATOR. . . . .	69
6.2	An example of a Networked Hierarchical Room. . . . .	70
6.3	Visualization of the Networked Hierarchical Room. . . . .	71
6.4	VANNOTATOR surface example. . . . .	72
7.1	Small relation evaluation of BERT-large after the method of Kurita et al. (2019). . . . .	88
7.2	Heatmap of source-object associations based on BERT-Large and the room dataset. . . . .	93
7.3	Association heatmap of BERT-Large on the part dataset. . . . .	94

## List of Figures

7.4	Association heatmap of BERT-Large on the verb dataset. . . . .	95
8.1	Example image context annotation. . . . .	102
8.2	AffordanceUPT evaluation regarding object types and training data size. . . . .	104
8.3	PoseContrast orientation predictions. . . . .	107
8.4	ObjectNet3D dataset mapped to main orientations. . . . .	108
8.5	AffordanceUPT token-pair visualization. . . . .	109
8.6	Habitats based on the 1 200 image annotations. . . . .	110
8.8	Example Object Orientation. . . . .	113
8.9	PoseContrast Object orientation determination considering image size and blur. . . . .	114
8.10	t-SNE visualization of ResNet features. . . . .	115
8.11	Examples from HICO-DET and ObjectNet3D. . . . .	115
8.12	UPT Object Unary Token Visualization. . . . .	116
8.13	UPT Person Unary Token Visualization. . . . .	116
8.14	HDBSCAN results. . . . .	117
8.15	Various error cases of AffordanceUPT based on HICO-DET annotations. . . . .	118
9.1	Areas of contextual expectations that are relevant for scene generation. . . . .	120
9.2	Example of an IsoSpace-based, user-supported text-to-3D scene result generated with SESB. . . . .	124
9.5	Generated examples scenes from the evaluation. . . . .	134
10.1	Adaptation of Figure 1.1 for future work. . . . .	136

# List of Tables

2.1	All IsoSpace related Entities and Links. . . . .	10
3.1	Example input (reduced) of Levy & Goldberg (2014). . . . .	21
3.2	Evaluation of different embedding types with different window sizes. . .	23
3.3	Results on the Instances and Concepts dataset (Boleda et al., 2017) with the Cbow model. . . . .	25
3.4	Results on the Instances and Concepts dataset (Boleda et al., 2017) with the Levy. . . . .	26
3.5	Average results for Cbow on different amounts of Wikipedia articles. . .	27
5.1	Matrix of arguments and relations. . . . .	39
5.2	Sentences exemplifying referential meaning, topological, and part-whole relations. . . . .	40
6.1	Evaluation table for VR tools. . . . .	63
6.2	Overview VR platforms. . . . .	65
6.3	Overview of the functionality of VR platforms. . . . .	73
7.1	All results of the static models. . . . .	85
7.2	All results of the contextual masked-language models. . . . .	86
7.3	All results of the contextual causal-language models. . . . .	86
7.4	Statistics generated from ScanNet using NYU categories. . . . .	90
7.5	A subset of part-whole relations extracted from <i>Online-Bildwörterbuch</i> . .	90
7.6	A subset of verb-object relations extracted from an updated version of HowToKB. . . . .	91
7.7	Model overview. . . . .	91
7.8	Templates for calculating scores regarding <i>Masked Language Models</i> and <i>Causal Language Models</i> . . . . .	92
7.9	Distance Correlation calculated on the word frequencies of Google Ngram. .	92
8.1	A small subset of text annotations. G stands for Gibsonian and T for telic. .	102
8.2	UPT Results on the Gibsonian/telic text annotated HicoDet Test dataset. .	105
8.3	PoseContrast results on the image annotated HicoDet dataset. . . . .	107
8.4	Selection of object orientation datasets with information about their size and domain coverage. . . . .	113
8.5	Calculated IAA between the image and text annotations. . . . .	114
8.6	Hyperparameter for AffordanceUPT. . . . .	117

*List of Tables*

9.1	The list of RCC8+ ( <i>Region Connection Calculus</i> ) relations (ISO, 2020). . .	123
9.2	Evaluation of IsoSpaceSpERT. . . . .	127
9.3	IsoSpaceSpERT Hyperparameter . . . . .	133



# 1 Introduction

## 1.1 Motivation

People can describe visual scenes with language and, conversely, create visual images based on linguistic descriptions, e.g., in their heads or on paper (Sadoski et al., 1990; Sadoski & Paivio, 2013). However, current systems do not even come close to the complexity of humans when it comes to reconstructing the scene from a given text (Hassani & Lee, 2016). On the other hand, it is not trivial for a human to create a scene manually using appropriate software, as this always requires a certain level of expertise (e.g. using Adobe Photoshop© or Blender) (Ma et al., 2018).

A technology that has become increasingly popular in recent years is virtual reality (VR)<sup>1</sup> This is not only because the technology is becoming more and more sophisticated, but also because it is becoming more and more affordable and thus no longer represents a major barrier to entry (Rodriguez, 2016; Zantua, 2017). The same applies to the development tools to be able to develop software for these devices.

The main motivation of this work was to combine these two concepts (VR and text-to-3D scene generation) to create a system that not only provides a direct tunnel to the described scene, but also allows these scenes to be created in an intuitive way (through language and grab/drop objects). This system will be presented later as Semantic Scene Builder (SESB, Chapter 9).

The possible practical applications of text-to-3D scene generation are manifold. From applications for digital learning (this point will be discussed in Chapter 6) to private home applications (e.g. setting up a virtual home office) to commercial applications (e.g. planning kitchen interior).

Last but not least, text-to-3D scene generation is a highly interesting scientific topic where many disciplines come together. Not only from computer science (like Natural Language Processing (NLP) and Computer Vision (CV)) but also from linguistics (Dennerlein, 2009) and psychology (Greene, 2013), for example.

The main focus of this work is on the contextual associations that are relevant in scene descriptions (Figure 1.1).

**Spatial expectations** refer to the spatial arrangement and relations of and between objects. These can be co-occurrence relations between objects (e.g., a computer keyboard is usually also a computer mouse), hierarchies between objects (e.g., a piece of cake is

---

<sup>1</sup>VR here stands for *fully-immersive virtual reality*, supported by hand tracking and head-mounted displays (Riva, 2006).

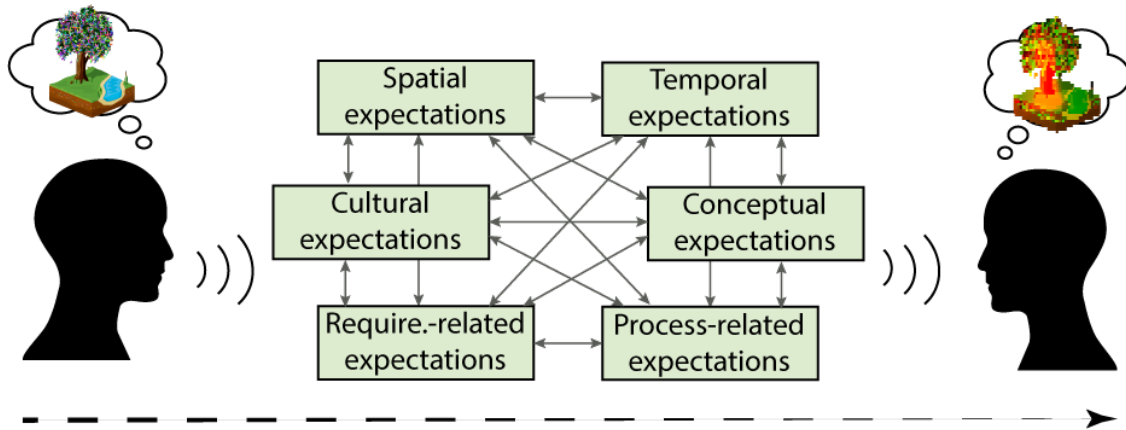


Figure 1.1: Areas of contextual expectations that are relevant for scene generation. These both local (within a text) and global (shared background between speaker and listener) can influence the scene being described. These can influence both locally (within a text) and globally (shared background of speaker and listener) the scene described by the speaker as well as the scene presented by the listener (Comparable to Pickering & Garrod (2004)).

usually on a plate and the plate is on a table, or pictures hanging on a wall), or spatial associations in general (e.g., in a kitchen there is usually also a stove).

**Temporal expectations** refer to temporal classifications. This can refer to historical classifications (e.g., a medieval kitchen compared to today, or radios in the 1960s compared to today), but also to individual time spans (e.g., you wear different clothes as a child than as an adult) or times of day (e.g., you eat different things in the morning than in the evening).

**Cultural expectations** is based on cultural differences, e.g. a typical German breakfast consists of different foods than a French breakfast. Or houses and rooms look different in Central Europe than in East Asia.

**Conceptual expectations** refers primarily to the affordances of objects (e.g., chairs are for sitting or cars for riding; Pustejovsky & Krishnaswamy, 2016).

**Requirements-related expectations** refer to the conditions that certain events or states require. These can be objects (e.g., to cook an omelet, you need eggs) but also other actions (e.g., to renovate a house includes painting the wall).

**Process-related expectations** expectations related to what has been processed so far or is likely to happen next.

This complexity is also reflected in the diversity of content of the publications written for this dissertation, most of which can be assigned to one of these contextual expectations. The following Section (Section 1.2 will now go further into the challenges of text-to-3D scene generation and then into the Contributions (Section 1.3) that this work has made to the various challenges.

## 1.2 Challenges

In recent years, work in text-to-3D scene generation has increasingly focused on generating more and more realistic scenes based on existing scene datasets. The actual language processing moved increasingly into the background and served only to add concrete expression conditions (parsable by predefined dependency rules) for scene generation (e.g. Ma et al., 2018; Chang et al., 2017b). Accordingly, from a linguistic perspective, many tasks remain unsolved (Hassani & Lee, 2016).

This already starts with the actual text processing. Thus, all relevant entities (which are relevant and which are not) in a text have to be recognized, coreferences have to be resolved, and finally, spatial relations and semantic roles have to be identified. This requires an extensive tool pipeline of not always homogeneous tools that must work together here. Most models are based on pre-trained word embeddings or language models trained only on text, and thus it is not clear how well they can capture spatial relationships.

Next, we need to resolve the various interconnected relationships and contexts (already addressed in the motivation from Section 1.1). Most of the implicit information that can be carried along in scene descriptions lacks the resources necessary to resolve it. This is also true for 3D scenes in general, so there are not enough resources for 3D scenes and associated text pairs. And for the few that do exist, the available descriptions are again very specific and leave little room for interpretation and context (e.g. Chang et al., 2015b).

The final challenge is to underlay the linguistic units with 3D objects and arrange them in a spatially meaningful way. Again, it may be necessary to insert additional objects into the scene to make it look more realistic.

In addition to the fact that the required data is not available in sufficient quantity, there is also no suitable annotation environment that supports the creation of this data. Most of the previous works are also not open source, so it is difficult to address the individual points and compare the results without directly developing and implementing a completely new text-to-3D scene system.

Finally, there is the question of whether such systems in VR are suitable for other applications besides the creation of simple scenes, such as digital education.

The following section will now discuss how this dissertation contributes to the various challenges.

## 1.3 Contributions

The following sections list the contributions of this dissertation. They are divided into four main categories and mapped as far as possible to the contextual expectations from Section 1.1 and Figure 1.1.

### Analysis & Evaluation of Language Models

Word embeddings like Word2Vec (Mikolov et al., 2013a) or transformer-based (Vaswani et al., 2017) language models like BERT (Devlin et al., 2019) are an essential part of any modern NLP application. As the popularity grows, so does the desire to improve the quality of these foundations. Although static embeddings are increasingly being replaced by context-based variants, they still have their place as their application is much faster, more resource-efficient and easier to interpret (Gupta & Jaggi, 2021). This is also where the basic idea of using context-based models to improve the static ones comes from (Gupta & Jaggi, 2021). We used BERT-based models to resolve coreferences, replacing pronouns with their proper names in the training data. We were able to show that all tested word embedding approaches did not significantly benefit from pronoun substitution and explained the results by saying that we ended up with exactly what we were trying to prevent with the approach: the loss of contextual information (Henlein & Mehler, 2020, Chapter 3).

Another important issue, as mentioned earlier, is that of interpretability. With regard to the generations of text-to-3D scene, the question arises to what extent transformer-based language models enable the extraction of knowledge about object relations. In other words, to what extent such knowledge is represented in these models? For this purpose, we used different approaches from bias research and analyzed static and dynamic models. In doing so, we were able to show that the models differ greatly in terms of the amount of knowledge they allow to extract. Similarity measures perform much worse than classifier-based approaches. And static models perform almost as well as contextualized models - in some cases even better. The considered relations were: object relations ( $X$  occurs in  $Y$ ;  $X$  consists of  $Z$ ; action  $A$  involves the use of  $X$ ) and thus refer to these contextual expectations: *spatial expectation*, *conceptual expectations* and *requirement-related expectations* (Henlein & Mehler, 2022, Chapter 7).

### Grounding of Human-Object Interactions

Tightly linked to conceptual expectations is the grounding of human-object interactions. It deals with the question for which purposes/interactions objects are created and which conditions they have to fulfill for these actions to be possible. This is especially true for the orientation of objects, e.g., a spoon is handled differently when it is used to eat soup (horizontal) – when it is used to mix coffee with milk (vertical) – or if the spoon is only held (called *habitat*; Pustejovsky & Krishnaswamy, 2016). For this distinction, we annotated the HICO-DET dataset (Chao et al., 2018) with Gibsonian and telic (Pustejovsky, 2013) affordances and then trained our model (called *AffordanceUPT*) on this

dataset. For this model, we demonstrated that it is effective in discriminating between Gibsonian and telic affordances in images and that our model learns other correlations in the data to make such discriminations (e.g., the presence of hands in the image), although orientation recognition remains difficult, and thus habitat detection (Henlein et al., 2023a, Chapter 8).

## **SESB**

While the developments in text-to-image generation have recently shown huge progress due to the huge amount of available data, this amount of data is still unthinkable for 3D scene generation (see Section 2.4). The first work consisted of a concept for a VR environment for the annotation of ISO Space, a markup language for the annotation of spatial structures in texts (see Section 2.1.1). However, the annotation of IsoSpace itself is relatively complex, so the VR environment should not only support the annotator itself but also partially automate the process (Henlein et al., 2020, Chapter 4).

Alongside Text2SceneVR was developed, which allows the creation of spatial hyper-texts in VR. Supported functions were the free placement of objects and creation and texturing of walls and rooms (Abrami et al., 2020a, Chapter 5).

Both project ideas were then combined as Semantic Scene Builder (SESB), a modular text-to-3D scene generation system based on SemAF as the underlying data structure. Via VR, the user can make changes to the generated scenes at any time or create them himself from scratch. We also evaluated the modules built into SESB against a modern open-source method for text-to-scene (the only one publicly available) and found that our approach not only performed better but could also model a wider variety of scenes. SESB benefits from two self-generated resources that allow the model to resolve room names (spatial expectations) and required objects for actions performed by humans (requirement-related expectations). Furthermore, based on the different context expectations, we have shown which weaknesses the current systems still have and in which directions it is essential to further develop in the future so that the systems become even more realistic (Henlein et al., 2023b, Chapter 9).

## **VR as a Tool for Digital Learning**

The last point is about the concrete application of VR-based systems for digital learning and teaching purposes. A point that has become more important, especially in recent years, due to quarantine and corona regulations. Based on the work of Fowler (2015); Mikropoulos & Natsis (2011) and Mayes & Fowler (1999), we derived several requirements for educational VR applications and analyze a variety of current programs to see to what extent these are met. We were able to show that the possibilities offered by VR are far from exhausted, as most applications only try to emulate reality instead of expanding it. In addition to the existing tools, we show the possibilities in the field of virtual and three-dimensional teaching and learning environments using the example of VANNOTATOR (Henlein et al., 2021, Chapter 6).

## 1.4 Dissertation Structure

This dissertation is structured as follows. Chapter 3 - 9 are the published papers, which were made in the context of this dissertation.

**Chapter 1** provides an introduction to the topic (Section 1.1) discussed here, including challenges (Section 1.2) and contributions (Section 1.3) to the topic addressed by this dissertation.

**Chapter 2** presents related work that is built upon later in this dissertation or that is substantively related to this work. Specifically, the following areas are highlighted: The Semantic Annotation Framework (SemAF) and, as part of it, especially IsoSpace as an annotation scheme for the annotation of (spatial) semantics, which will later serve as a foundation for SeSB (Section 2.1). The architecture of TEXTIMAGER and the associated entire infrastructure of SeSB, consisting of TEXTIMAGER, TEXTANNOTATOR and VANNOTATOR (Section 2.2). The functionality of transformer models and associated language models such as BERT are the cornerstone of current state-of-the-art NLP methods (Section 2.3). And last but not least other text-to-3D scene generation models and their approaches (Section 2.4).

**Chapter 3** investigates the impact of coreference resolution as a preprocessing step for static word embeddings. Various downstream tasks serve as evaluation criteria for this purpose.

**Chapter 4** presents an initial concept of how a VR annotation environment can be used to annotate IsoSpace and text-to-3D scene data, and the benefits of this VR environment.

**Chapter 5** presents Text2SceneVR, a VR tool based on VANNOTATOR for creating spatial hypertexts for training future text-to-3D scene systems.

**Chapter 6** provides a detailed overview of existing practices and tools for digital and virtual teaching, learning, and collaboration, as well as the necessary requirements and strategies to make the most of technological opportunities in the future, with a focus on solutions and strategies for three-dimensional, virtual environments and applications.

**Chapter 7** evaluates the extent to which transformer-based language models allow us to extract knowledge about object relations (*X comes in Y*; *X consists of Z*; *action A involves the use of X*). For this purpose, we use approaches from BIAS research and compare the results with static embeddings.

**Chapter 8** introduces AffordanceUPT, a modular adaptation of UPT, for the classification of Gibsonian and telic affordances. To this end, the HICO-DET dataset has been extended accordingly, and interesting features of AffordanceUPT are highlighted that may be of interest for grounding affordances in the future.

**Chapter 9** finally introduces Semantic Scene Builder (SESB), a VR-based text-to-3D scene framework using SemAF and UIMA to integrate a variety of tools and resources.

**Chapter 10** summarizes the content of this dissertation (Section 10.1) and provides an outlook for future work (Section 10.2).





## 2 Related Work

This chapter presents the main related work to which this thesis refers. For this purpose, the following works are presented in particular: The annotation scheme for the annotation of semantic and thus also spatial structures in texts (Semantic Annotation Framework, Section 2.1). A framework for natural language annotation and processing (TextImager, Section 2.2). State-of-the-art language models that are currently used in almost all *Natural Language Processing* applications (Transformers, Section 2.3). And finally, the most important text-to-3D scene generation publications in recent years (Section 2.4).

### 2.1 Semantic Annotation Framework

The *Semantic Annotation Framework* (SemAF) is published under ISO/TC 37/SC 4/WG 2 Semantic Annotation. It was developed with the goal of creating a unified, and thus mutually compatible, framework to represent different levels of linguistic semantics (Ide & Pustejovsky, 2017, Chapter 4). The individual modules range from *temporal* (Iso-TimeML; Pustejovsky et al. (2010); ISO (2012a)) and *spatial* (IsoSpace; Pustejovsky et al. (2011a); ISO (2020)) annotations to annotations of discourse referents (SemAF-DS; ISO (2014c) and measurable quantitative (MQI; ISO (2021)). Other modules are still under development, such as for *spatial semantics* (ISO, 2022b).

SemAF consists of two main components: *Entities*, which can be marked in the text, and *Links*, which represent the possible relations between the entities. Different modules now introduce different entities and links that represent different semantic information. Both have additional attributes that depend on the entity/link type. The most important modules for this work, are discussed in the next sections. An example annotation can be found in a later Chapter (Chapter 9, Figure 9.2).

#### 2.1.1 IsoSpace

The IsoSpace module is used to annotate spatial semantics. The focus is on the labeling of spatial entities and their spatial relationships to each other. The entities themselves are divided into *Locations*, *Paths* and regular *Spatial Entities*. The spatial relations are annotated via *Qualitative Spatial Links* (QSLinks) and *Orientation Links* (OLinks). QSLinks represent topological RCC8+ (*Region Connection Calculus*) relations (Randell et al., 1992). OLinks represent all other spatial relations (e.g. *behind*, *south*, *across*). Additionally, there are *Measure Links* (MLinks) to be able to map concrete spatial dimensions. And *Movement Links* (MoveLinks) represent spatial and intrinsic movements or changes. For

Type	Name	Example	Short Description
Entity	SpatialEntity	he, dog, cup	regular spatial entities
	Place	Frankfurt, village	geographic or administrative location
	Path	street, river, coast	location consisting of a sequence of locations
	EventPath		triggered by <i>Motions</i> to describe the path of that motion
	SpatialRelation	in, on, west, to	signalwords for spatial relations or movements.
	Motion	swim, aged	An event that changes an object extrinsically or intrinsically.
	NonMotion	drink, live	broad term for all kinds of events that can take place anywhere
	Measure	5m, 20 kg	spatial dimension
Link	QSLink	EC, PO, TPP	RCC8+ relation between two entities
	OLink		spatial relation between two entities (from a relative viewpoint)
	MoveLink		connects a mover with an EventPath
	MLink		connects a Measure with an Entity

Table 2.1: All IsoSpace related Entities and Links.

a complete list of entities and links provided by IsoSpace, see Table 2.1.

### 2.1.2 VoxML

The SemAf VoxML module (ISO, 2022a) itself is still under development, so this section only discusses the VoxML paper published so far (Pustejovsky & Krishnaswamy, 2016).

Unlike the other SemAF modules, VoxML is not an additional annotation layer for texts, but a description language for 3D objects. The goal is to describe semantic knowledge about these objects, including attributes, events and habitats. Such knowledge is essential for tasks like text-to-3D scene generation where the positioning and orientation of objects is an essential part of making a scene look natural (cf. Biederman et al., 1982; Boyce & Pollatsek, 1992; Lauer et al., 2020). Unfortunately, there are not enough 3D objects with VoxML annotations yet, so their inclusion in text-to-3D scene is not yet useful.

### 2.1.3 Other Modules

In the following, a few more (but not all) modules of SemAF will be presented, which will play a role in the upcoming parts of this work.

**IsoTimeML** IsoTimeML (Pustejovsky et al., 2010; ISO, 2012a) is an revised and adapted version of TimeML (Pustejovsky et al., 2005a,b) to be compatible with SemAF standard. IsoTimeML is for annotating events, times and their temporal relationships in texts and therefore essential for understanding temporal relationships and sequences in scene descriptions.

**Semantic roles (SemAF-SR)** SemAF-SR (ISO, 2014b) supports the annotation of semantic roles (Palmer et al., 2005). This is important for scene descriptions in order to make it comprehensible who performs which actions and to assign objects to these actions.

**Reference annotation (RAF)** RAF (ISO, 2019) is used for the annotation of e.g. coreferences (Radford, 2004, p. 332) and thus allows the resolution of multiple names of the same entity.

**Measurable quantitative information (MQI)** MQI (ISO, 2021) allows annotating quantification and quantitative information. Since the model itself is comparatively new and the basic functionality is already available in IsoSpace, it will not be discussed further in the rest of this paper.

## 2.2 Texttechnology Lab Annotation Architecture

In the following section, the main components of the Texttechnology Lab Annotation Architecture and thus of SESB are presented. These main components are:

1. **TEXTANNOTATOR** (Abrami et al., 2019a, 2020c, 2021) as annotation web application, as well as REST application to serves as an interface between the documents and all other applications (Section 2.2.1).
2. **TEXTIMAGER** (Hemati et al., 2016) for automatic processing and preprocessing of text data via a lot of integrated machine learning tools (Section 2.2.2).
3. **VANNOTATOR** (Mehler et al., 2018; Spiekermann et al., 2018; Abrami et al., 2020a) as VR annotation environment implemented in Unity3D<sup>1</sup> (Section 2.2.3).

A complete overview of the entire current architecture can be found in the work of Abrami et al. (2021). An overview of how the individual components are connected is shown in Figure 2.1.

---

<sup>1</sup><https://unity.com/>

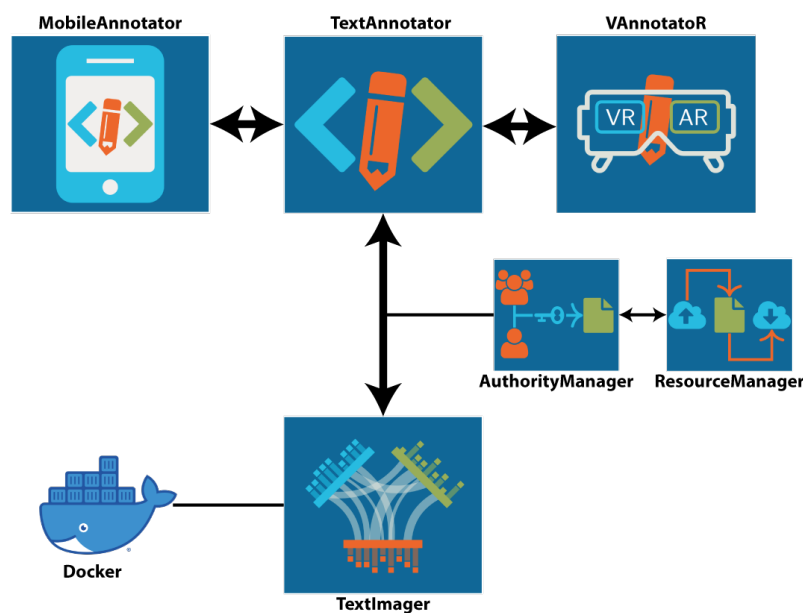


Figure 2.1: Highly simplified overview of the Texttechnology Lab Annotation Architecture.

## 2.2.1 TextAnnotator

TEXTANNOTATOR is an annotation framework that enables collaborative and simultaneous annotations based on UIMA (Ferrucci & Lally, 2004). The UIMA documents themselves are managed and stored within a MongoDB (Abrami & Mehler, 2018). TEXTANNOTATOR offers both a WebSocket application including various annotation tools for different applications, as well as a REST interface for connecting further tools, such as VANNOTATOR or MOBILEANNOTATOR (Adeberg, 2020). Among others, the following annotators have been implemented in TEXTANNOTATOR (Abrami et al., 2021):

- **QuickAnnotator** for annotating *Multi Word Expressions* and *Named Entities*.
- **PropAnnotator** for annotating *Semantic Roles* and *Word Sense Disambiguation*.
- **DepAnnotator** for annotating *Dependency Relations* based on different Tagsets.
- **SemAFAnnotator** (de Reichenfeld, 2022) for annotating *IsoSpace* and *3D scenes* from a top-down view (Compatible with SeSB).

## 2.2.2 TextImager

TEXTIMAGER is used to preprocess new documents and then save them directly in the UIMA format required by TEXTANNOTATOR where possible errors can also be corrected. In the TEXTIMAGER itself, a large number of tools are implemented for an equally large number of languages. These range from standard syntax parsers such as tokenizer, part-of-speech tagging, and dependency parsing to content analysis such as word sense dis-

ambiguation (Uslu et al., 2018a) and DDC classification (Uslu et al., 2018c). In addition, the TI has several visualization tools that allow you to visually compare a variety of different documents at different levels, e.g. Text2Voronoi (Mehler et al., 2016a) or LitViz (Uslu et al., 2018b).

### 2.2.3 VAnnotatoR

VANNOTATOR is a VR- and UIMA-based annotation tool implemented in Unity3D. The tool itself allows a variety of applications, e.g. the creation of multimodal hypertexts (Mehler et al., 2018) regular textual annotations (Spiekermann et al., 2018) or the interaction with historical information (Abrami et al., 2020b). The interaction with the environment takes place via a head-mounted display and the interaction via the corresponding VR controllers.

## 2.3 Language Models

In this section, various word embedding methods and language models are presented, with transformer-based language models in particular currently representing the state of the art in language processing.

### 2.3.1 Static Wordembeddings

Probably the best-known method to create static word embeddings on large amounts of data is Word2Vec (Mikolov et al., 2013a,b). The goal is to find vector representations for words that contain both syntactic and semantic information. The basic idea itself was not new (cf. Rumelhart et al., 1985), but what made Word2Vec stand out was that it could be trained on large amounts of data very quickly. The basic architecture is relatively simple. A window of size  $n$  is moved over the text, and a neural network is trained to determine the middle word based on the surrounding words (CBOW) or, conversely, to determine the given words based on the middle word (Skip-gram). Mikolov et al. (2013a) noted that CBOW usually performs better, whereas Irsoy et al. (2021) evaluated that the performance of both variants is the same and the difference was only due to a bug in the implementation.

Besides Word2Vec there are many other methods to create static embeddings. The most important of these are:

- GloVe (Pennington et al., 2014), using global word co-occurrence relations.
- fastText (Mikolov et al., 2018; Bojanowski et al., 2017; Grave et al., 2018), including subword information.
- Levy (Levy & Goldberg, 2014), based on dependency relations instead of linear word sequences

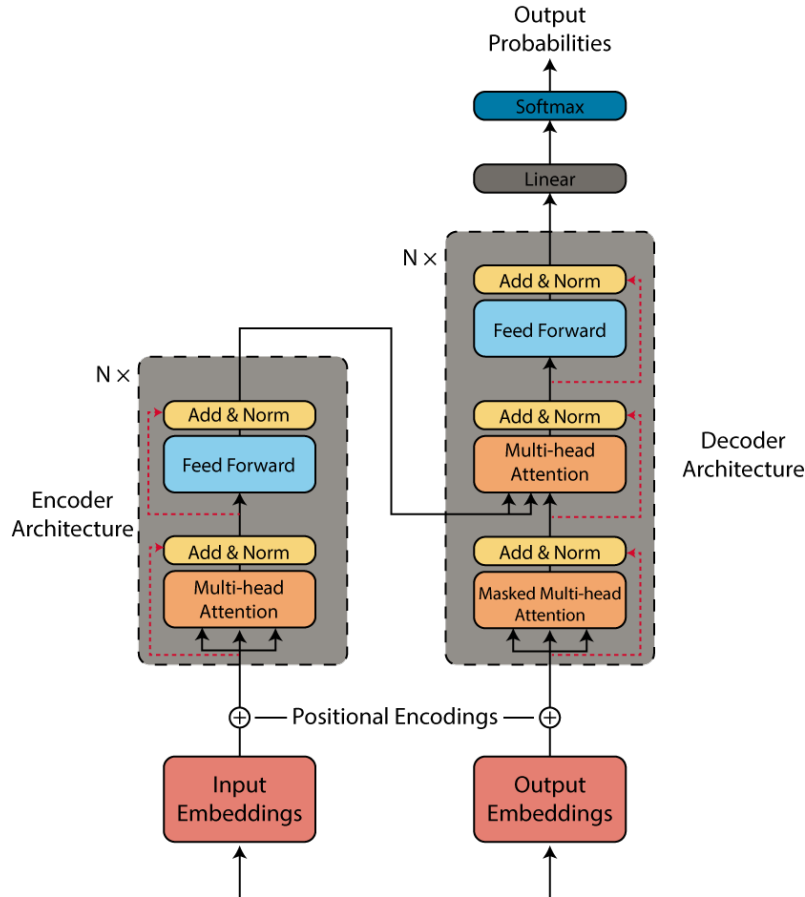


Figure 2.2: Transformer architecture based of Figure 1 of Vaswani et al. (2017). The dotted red lines are residual connections (He et al., 2016).

### 2.3.2 Transformers

Transformers were first introduced in the work of Vaswani et al. (2017). It was introduced as an encoder-decoder model for machine translation tasks. The idea was to avoid complex recurrent (Hochreiter & Schmidhuber, 1997) or convolution neural networks and instead rely on self-attention mechanics (Bahdanau et al., 2014) to allow more parallelization. The architecture is visualized in Figure 2.2. To be precise, “Scaled Dot-Product Attention” is used, whereby the input consists of so-called Queries  $Q$ , Keys  $K$ , and Values  $V$ .  $d_k$  describes the dimension of  $Q$  and  $K$  and is used as a scaling factor.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2.1)$$

Instead of applying the attention mechanisms only one time, the Transformer uses a so-called “Mutli-Head Attention System” where  $h$  defines the number of attention heads. In this way, the model can view information from different representation subspaces at different positions.

$$MultiHead(Q, K, V) = Concat(head_1, \dots, head_h)W^O \quad (2.2)$$

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V) \quad (2.3)$$

The decoder model now differs from the encoder model in two essential points. First, in the first layer, via masking of future positions, the model can only refer to information prior to  $i$  at any time  $i$ . Second, the decoder uses K and V of the decoder, which assists the decoder to focus on important parts in the input sequence.

To take into account the order of the words or their distance from each other, so-called “positional encodings” are used.

Based on this architecture, the following variants of language models have evolved:

**Encoder-based Language Models** Based on the Transformer-Encoder architecture, BERT (Bidirectional Encoder Representations from Transformers; Devlin et al., 2019) was developed. BERT itself is pre-trained on two tasks: “Masked Language Modeling” (MLM) where during training some of the words are masked and the model is supposed to predict these words based on the context, and “Next Sentence Prediction” (NSP) where the model is given two sentences and is asked to determine if the second sentences can come after the first sentences. As with Word2Vec, there are now countless variations, each with its own twist, to name just a few:

- RoBERTa (Liu et al., 2019b): They dropped NSP and instead used more training data and improved the masking algorithm.
- ALBERT (Lan et al., 2019): Instead of NSP, they predict the order of two sentences.
- ELECTRA (Clark et al., 2020): Instead of MLM, the tokens are replaced by another language model and the model itself should predict which tokens have been replaced.
- DistilBERT (Sanh et al., 2019): Created via “knowledge distillation” (Buciluă et al., 2006; Hinton et al., 2015) from a larger BERT model.

**Decoder-based Language Models** On the other hand, decoder-based language models were developed, such as GPT-2 (Radford et al., 2019). Instead of sentences with masked tokens, these models are trained with incomplete sentences, and the model is expected to predict the next token and therefore perform better on tasks like text generation. While encoder-based models almost always need to be tuned to the actual task after the fact, decoder-based models show much better performance on zero-shot and few-shot tasks (Brown et al., 2020; Wang et al., 2022; Scao et al., 2022).

Also, Transformer architectures are not limited to NLP applications but have been ported very successfully to other areas of computer science, such as Computer Vision (Khan et al., 2021; Han et al., 2022) or Finance (Ding et al., 2020).

## 2.4 Text-to-3D Scene

### 2.4.1 WordsEye

WordsEye(Coyne & Sproat, 2001)<sup>2</sup> was not the first, but the first successful text-to-3D scene system and is still developed today (e.g. Coyne et al., 2011; Ulinski et al., 2019)<sup>3</sup>. Even then, WordsEye supported objects, attributes, signs, poses, and spatial relationships, among others. Texts are processed using dependency parsing and PoS tagging, where nouns are recognized as 3D objects and relations are represented into a kind of frame semantics using dependency rules (Coyne et al., 2010). This is now supplemented by VigNet (Coyne et al., 2011), an extension of FrameNet (Baker et al., 1998), e.g. to ground the different meanings of prepositions.

### 2.4.2 SceneSeer

SceneSeer (Chang et al., 2017b, 2014a,b, 2015a) relies less on manually generated resources, but learns in advance from prebuilt scenes support priorities for objects, for example, that a piece of cake is on a plate and that plate in turn is on a table. The input text is processed via Stanford CoreNLP (Manning et al., 2014) and is based on similar rules as WordsEye. In addition, SceneSeer supports the subsequent modification of the created scenes with the help of text commands also within certain limits of Active Learning.

### 2.4.3 Language-driven synthesis of 3D scenes from scene databases

The work of Ma et al. (2018) is based even more on prebuilt 3D scenes. Again, the input text is processed via Stanford CoreNLP and transformed into an abstract scene layout via rule-based part-of-speech and dependency graph rules. The graph is then matched with existing scenes and the most similar scene is taken as a template. In addition to a self-taught model that provides additional support to parents, the following models are integrated:

1. a **co-occurrence model** for adding relevant objects based on co-occurrence probabilities (e.g., a mouse next to a keyboard).
2. a **pairwise model** for predicting the relative position between two objects (e.g., where to place a chair next to a table).
3. a **group model** for dealing with group relationships (e.g. “clean office table”).
4. and a **relative model** to handle conflicts between explicit relationships specified in the input dataset and implicit relationships specified by existing objects.

---

<sup>2</sup><http://www.wordseye.com/>

<sup>3</sup><https://wordseyeworld.com/>



### 2.4.4 SceneFormer

The focus of SceneFormer (Wang et al., 2021a) is the creation of realistic scenes (Scene Synthesis, see Section 2.4.5). It uses an end-to-end approach, combining multiple transformer-encoder models (Vaswani et al., 2017) to sequentially determine the object category, orientation, position, and dimension for each newly added object. However, they also present a text-conditional variant that only supports a maximum input of 3 sentences and 40 tokens. In addition, the models are trained only on living rooms and bedrooms using the SUNCG dataset (Song et al., 2017), which is no longer freely available for licensing reasons.

### 2.4.5 Related Tasks

#### Scene Synthesis

Scene Synthesis describes the task of creating realistic scenes. Most of the works in this direction focus specifically on indoor scenes (e.g. Li et al., 2019; Wang et al., 2021a; Ritchie et al., 2019; Huan et al., 2022) This is mostly not about the generation of random scenes but based on some form of input. This can be text-based, of course, but also based on RGB-D images (Huan et al., 2022), pre-made layouts (Wang et al., 2021a), user interactions (Zhang et al., 2021d), scene graphs (Dhamo et al., 2021) ect. (Zhang et al., 2019).

#### Text-to-Image

Related to text-to-3D scene generation is the generation of images. This task has also received much more attention lately (e.g. Tan et al., 2019; Ramesh et al., 2021, 2022; Saharia et al., 2022; Ding et al., 2022; Alayrac et al., 2022). These models benefit from the combination of advances in grounded language modeling (like CLIP; Radford et al., 2021) and the sheer amount of data that can be crawled from the Internet (cf. LAION-5b<sup>4</sup> which provides 5,85 billion image-text pairs).

#### Text-to-Shape Generation

An increasingly growing area of research is the creation of 3D models of objects based on text descriptions. Depending on the approach, these objects are created as point cloud (Yang et al., 2019; Achlioptas et al., 2018), voxel (Sanghi et al., 2022; Chen et al., 2018), mesh (Nash et al., 2020) or implicate representation (Chen & Zhang, 2019; Mescheder et al., 2019). Or alternatively get the most suitable object from an object database (Text-to-shape retrieval; Ruan et al., 2022).

---

<sup>4</sup><https://laion.ai/>

### **Text-to-Animation**

And finally, there is the variant of creating not static scenes from text descriptions, but entire animations. These applications are mostly very domain specific. This can be, for example, the generation of TV shows (Hayashi et al., 2014), emotional scenes (Hanser et al., 2009a,b) or human locomotions (Zhang et al., 2021c).

# 3 On the Influence of Coreference Resolution on Word Embeddings in Lexical-semantic Evaluation Tasks

Henlein, A. & Mehler, A. (2020). On the Influence of Coreference Resolution on Word Embeddings in Lexical-semantic Evaluation Tasks. In *Proceedings of The 12th Language Resources and Evaluation Conference* (pp. 27–33). Marseille, France: European Language Resources Association

## Abstract

Coreference resolution (CR) aims to find all spans of a text that refer to the same entity. The F1-Scores on these task have been greatly improved by new developed End2End-approaches (Lee et al., 2017) and transformer networks (Joshi et al., 2019). The inclusion of CR as a pre-processing step is expected to lead to improvements in downstream tasks. The paper examines this effect with respect to word embeddings. That is, we analyze the effects of CR on six different embedding methods and evaluate them in the context of seven lexical-semantic evaluation tasks and instantiation/hypernymy detection. Especially in the last task we hoped for a significant increase in performance. We show that all word embedding approaches do not benefit significantly from pronoun substitution. The measurable improvements are only marginal (around 0.5% in most test cases). We explain this result with the loss of contextual information, reduction of the relative occurrence of rare words and the lack of pronouns to be replaced.

## 3.1 Introduction

Many NLP systems use word embeddings as a fast to learn resource that captures important lexical information (Mikolov et al., 2013b). Once trained, embeddings can be used in many different tasks, like Coreference Resolution (Lee et al., 2018), Emotion Detection (Felbo et al., 2017), Biomedical Natural Language Processing (Wang et al., 2018), Image Caption Generation (Vinyals et al., 2015) or Text Classification (Uslu et al., 2019). Most of them rely on local information delimited by context windows or dependency parents to predict word relations (Levy & Goldberg, 2014). This approach encounters problems wherever semantic relationships have to be captured, which are expressed by

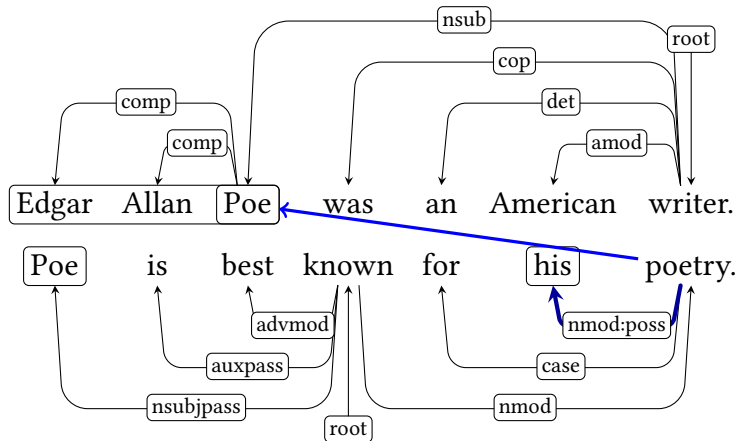


Figure 3.1: Dependency trees of two consecutive sentences. The blue arrow from *poetry* to *Poe* indicates the expanded context that is mediated by *his*.

coreference, as the following example illustrates:

“Edgar Allan Poe was an American writer. Poe is best known for his poetry.”

Based on a context window-based approach of a maximum of five right neighbors, we get data to examine the relationship of *Poe* and *writer* and of *his* and *poetry*. But the model is not informed about a relationship between *Poe* and *poetry* when using a too small window. Obviously, the detour via the use of overly large window sizes (which would capture wanted as well as unwanted co-occurrences) can be prevented by a coreference resolution which replaces *his* with *Poe*.

The mapping of different linguistic expressions to the same entity is called *Coreference Resolution* (CR) (Ponzetto & Poesio, 2009). Previous systems were computationally very intensive and required a large NLP pipeline to calculate the required features (Clark & Manning, 2015; Wiseman et al., 2016; Clark & Manning, 2016; Poesio et al., 2016). The currently most modern system (Lee et al., 2018; Joshi et al., 2020) does not need any of these features, therefore it is now possible to perform CR in a reasonable time. The resulting state-of-the-art score is 79.6% F1-Score for English. In this paper, we use CR as pre-processing step for training word embeddings, replace pronouns with their first mention, and evaluate the final word embeddings on different tasks. There are several approaches to evaluating word embeddings, which can be divided into *extrinsic* and *intrinsic* tasks. Extrinsic is the evaluation on downstream tasks such as POS tagging. Intrinsic evaluations explore word data about syntactic or semantic relations. The *Word Similarity* (WS) task, for example, evaluates how well the dot product of two word pairs correlates with the scores of human annotations (Jastrzebski et al., 2017). In this paper, we analyze the influence of resolving anaphoric relations on computing word embeddings by means of intrinsic approaches. As shown above, anaphoric relations are usually lost in training, although they manifest important relationships between words. Our experiments show that none of the embeddings analysed is improved by mention substitution – in any event, the improvements are only marginal.

Word	Context
poe	writer/nsub <sup>-1</sup> , (poetry/nmod:poss <sup>-1</sup> )
writer	poe/nsub
his	poetry/nmod:poss <sup>-1</sup>
poetry	his/nmod:poss, (poe/nmod:poss)

Table 3.1: Example input (reduced) of Levy & Goldberg (2014) based embeddings induced by the example of Figure 3.1. The additional contexts in parentheses are achieved with the help of CR.

We explain this result with the loss of contextual information, reduction of the relative occurrence of rare words and the lack of pronouns to be replaced. The paper is organized as follows: Section 3.2 gives a short overview of word embeddings and of CR. Then we present our approach to enhancing word embeddings based on CR in Section 3.3 The experimental setup is described in Section 3.4 The results in Section 3.5 A prospect to future work is presented in Section 3.6

## 3.2 Related Work

Pre-trained word embeddings (Mikolov et al., 2013b; Ling et al., 2015; Pennington et al., 2014; Levy & Goldberg, 2014; Komninos & Manandhar, 2016) are a standard component of most modern NLP architectures. However, most of these systems are based only on local word information, such as skip-grams (e.g. Mikolov et al. (2013b) or Ling et al. (2015)) or dependency relation-based windows (e.g. Levy & Goldberg (2014) or Komninos & Manandhar (2016)).

Only recently, new systems have been introduced which are trained on large contexts using LSTMs (Peters et al., 2018) or large neural attention systems (Transformers) based on more complex transfer-learning tasks (Devlin et al., 2019; Liu et al., 2019a) and are therefore not limited to local information – but at the price of additional computational complexity. At the same time, the list of proposals for new embedding methods that are pre-trained on ever larger corpora from more and more areas (genres, registers etc.) of more and more languages is constantly growing (Grave et al., 2018; Bojanowski et al., 2017; Radford et al., 2019). In recent years, the impact of various features such as POS-tags, subword information, semantic relations and in-domain data on word embeddings have been analyzed (Rezaeinia et al., 2017; Wendlandt et al., 2018; Bojanowski et al., 2017; Boleda et al., 2017; Gupta et al., 2017) and improved results have been obtained.

In this paper we complement this research and ask about the effects of CR on word embeddings. This is done by example of six methods of computing word embeddings: Cbow (Mikolov et al., 2013b), Skip (Mikolov et al., 2013b), Glove (Pennington et al., 2014), Wang (Ling et al., 2015), Levy (Levy & Goldberg, 2014) and Komninos (Komninos & Manandhar, 2016).

### 3.3 Coreference Substitutions for Enhancing Word Embeddings

In this section, we briefly introduce a formal apparatus to model coreference. Let

$$T = (w_1, \dots, w_i, \dots, w_n) \quad (3.1)$$

be a document with  $n$  tokens and words (lemmas)  $w = L(w_i)$  at position  $i$ . To avoid grammatical issues (especially morphological ones), we lemmatize all tokens in  $T$ .

A *mention*

$$m_{i:j} = (w_i, \dots, w_j) \quad (3.2)$$

is then defined as a continuous segment of tokens of  $T$ . Let

$$M = (m^1, \dots, m^p), p \leq n, \quad (3.3)$$

be the sequence of all mentions observed in  $T$ , sorted by occurrence. A mention  $m^i$  is said to be *antecedent* to a mention  $m^j$  if both are co-referent (and thus connected by a co-reference link) and if  $i < j$ . We denote this antecedence by  $m^i < m^j$ . Then we define the function

$$\text{first}(m^j) = \arg \min_{m^i \in \{m^i < m^j \mid i \in \{1, \dots, j-1\}\}} \{i\} \quad (3.4)$$

which returns the antecedent of  $m^j$  of lowest index and write  $m^i \ll m^j \Leftrightarrow \text{first}(m^j) = m^i$ .

#### 3.3.1 Extending the informational scope of window-based embeddings

Our approach to extending window-based embeddings by means of CR is the following: For all pronominal mentions  $m^i$ , for which  $\text{first}(m^i)$  is not pronominal, we replace:

$$m^i \leftarrow \text{first}(m^i) \quad (3.5)$$

This means that we replace each pronoun with its lowest index antecedent which in our case is represented by a corresponding lemma or multiword expression as shown in the following example:

$$\dots \text{his poetry.} \mapsto \dots \text{Edgar Allan Poe poetry.}$$

So far, our replacement procedure only considers pronouns. The reason is that we expect the greatest loss of information from not replacing them. In this way, we avoid problems that we would get if we replaced phrasal mentions (e.g. more complex noun phrases) with their phrasal antecedents.

### 3.3 Coreference Substitutions for Enhancing Word Embeddings

Type	WS	Average				MEN				WS353				SimLex999				RW				MTurk-287				Google				SemEval2012_2			
		c	p	h	ph	c	p	h	ph	c	p	h	ph	c	p	h	ph	c	p	h	ph	c	p	h	ph	c	p	h	ph				
Cbow	2	42.29	42.11	43.37	42.43	67.54	67.02	68.56	67.38	53.87	54.56	55.28	54.32	35.42	34.75	35.09	34.97	25.28	25.31	25.98	24.42	61.63	60.76	61.75	61.23	33.68	34.61	39.20	36.15	18.59	17.77	17.70	18.52
Cbow	5	44.64	44.47	45.80	44.94	70.45	70.21	71.24	70.71	58.24	57.42	58.61	58.52	37.42	37.18	37.78	37.57	24.36	24.32	24.78	24.35	61.60	60.64	62.21	61.89	41.63	43.08	46.37	43.94	18.79	18.45	19.61	17.62
Cbow	10	45.68	45.48	46.37	46.07	72.25	71.93	72.60	72.45	60.70	60.31	60.69	60.72	37.32	37.15	37.61	37.37	25.28	24.70	24.83	24.13	62.24	62.63	63.80	63.88	44.41	45.65	46.97	46.20	17.57	16.00	18.10	17.73
Skip	2	47.58	47.71	48.59	48.13	73.54	73.43	74.35	73.50	66.03	65.57	67.76	66.47	40.92	40.84	41.25	41.81	31.33	31.46	31.95	32.12	61.66	61.88	62.96	61.48	41.53	42.12	42.87	41.94	18.02	18.71	19.03	19.59
Skip	5	49.61	49.15	49.32	49.63	75.44	75.64	75.83	75.55	69.09	68.80	67.98	68.95	40.44	39.69	40.10	40.35	32.43	30.96	31.50	32.04	63.92	63.43	65.34	65.30	48.33	47.96	47.90	48.19	17.63	17.58	16.56	17.06
Skip	10	48.92	48.61	48.64	48.61	76.19	76.25	76.28	76.05	68.14	68.12	68.38	68.08	38.07	38.24	38.03	38.13	30.33	28.37	29.48	29.12	65.89	65.91	65.70	66.04	48.31	48.59	48.40	48.52	15.50	14.82	14.20	14.31
Glove	2	33.94	32.96	34.20	34.00	62.17	61.00	62.95	62.29	43.20	40.73	43.37	41.23	27.47	27.26	28.01	27.55	14.04	13.51	13.99	14.05	51.92	50.02	50.84	50.90	25.28	25.15	25.78	25.98	13.47	13.05	14.47	15.99
Glove	5	38.29	37.28	38.02	38.41	68.47	66.54	68.45	68.84	47.51	45.78	47.10	46.93	29.35	27.18	28.39	28.94	16.56	16.39	16.18	16.64	53.52	53.79	54.18	54.11	37.99	36.93	38.04	37.87	14.61	14.40	13.77	15.54
Glove	10	39.43	38.23	39.06	39.16	70.00	68.16	69.47	69.42	48.04	46.62	47.48	47.38	29.06	27.04	27.86	28.32	16.87	16.53	16.40	16.71	55.14	54.66	55.07	55.57	42.42	41.89	42.42	42.22	14.50	12.74	14.74	14.48
Wang	2	47.26	47.36	47.96	47.46	72.03	71.63	73.34	71.78	65.45	66.74	67.65	66.32	43.22	42.94	43.30	41.99	33.36	32.90	33.27	33.34	59.63	59.37	59.32	59.63	36.89	38.54	38.69	38.22	20.22	19.41	20.13	20.93
Wang	5	48.37	48.04	48.28	48.21	73.03	73.30	73.74	73.48	68.25	67.33	68.64	67.92	41.33	41.02	41.71	42.28	32.68	31.34	32.59	33.05	62.39	60.38	59.93	59.15	42.38	43.78	43.01	42.83	18.56	19.12	18.36	18.75
Wang	10	47.56	48.50	48.38	48.58	73.06	73.89	74.17	73.60	68.10	68.96	68.93	68.79	41.70	41.32	41.21	41.58	32.14	31.61	31.87	32.47	56.89	60.49	58.04	60.28	44.02	45.16	45.59	45.02	16.99	18.05	18.82	18.31
Levy		41.80	-	-	41.97	66.54	-	-	66.95	60.59	-	-	61.76	46.16	-	-	-	46.40	31.64	-	-	31.64	54.35	-	-	54.70	12.21	-	-	12.49	21.11	-	19.86
Komninos		47.45	-	-	47.26	72.68	-	-	72.50	62.84	-	-	62.68	42.09	-	-	41.29	33.73	-	-	33.46	61.00	-	-	60.80	38.84	-	-	40.28	20.97	-	19.78	

Table 3.2: Evaluation of different embedding types with different window sizes.  $c$  stands for the original dataset,  $p$  where we replaced only pronouns,  $h$  where we only replaced every mention with the mention head and  $ph$ , where we replaced only pronouns with the corresponding antecedent.

### 3.3.2 Extending the informational scope of dependency-based embeddings

For embeddings derived from dependency trees, we choose an approach that explores the underlying dependency relations. Let

$$D(w, T) = \{d(w_{i_1}), \dots, d(w_{i_k})\} \quad (3.6)$$

be the set of all parent tokens  $d(w_{i_h})$  to which the tokens  $w_{i_h}$ ,  $h = 1..k$ , of lemma  $w = L(w_{i_h})$  are directly dependent in text  $T$ . Conversely,

$$D^{-1}(w, T) = \{w_i \in T \mid L(d(w_i)) = w\} \quad (3.7)$$

is the set of all tokens that directly depend on some token of lemma  $w$  in  $T$ . A tabular representation of these sets derived from the text sample of Figure 3.1 is shown in Table 3.1. The procedure for extending the informational basis for computing dependency-based embeddings is now as follows: for each lemma for which there is a token that directly dominates a pronominal anaphoric mention, we add a dependency link from this token to the non-pronominal antecedent of lowest index of this pronoun. If this antecedent consists of several tokens, the root node of the corresponding dependency subtree is used as the target of the link. More formally: for each anaphoric pronoun  $w_k \in D^{-1}(w, T)$  depending on token  $d(w_k)$  of lemma  $w = L(d(w_k))$  such that there exists a mention  $w_k = m^j \in M$  (pronominal mentions are one-place), we extend the set of dependents  $D^{-1}(w, T)$  of  $w$  as follows:

$$\begin{aligned} \dot{D}^{-1}(w, T) &= D^{-1}(w, T) \cup \\ &\quad \{r(\text{tree}(m^i)) \mid \exists m^j \in M \\ &\quad \exists w_k \in D^{-1}(w, T): \\ &\quad w_k = m^j \wedge m^i \ll m^j\} \end{aligned} \quad (3.8)$$

where  $r(\text{tree}(m^i))$  denotes the root of the dependency subtree  $\text{tree}(m^i)$  spanned by mention  $m^i$ . A dependency tree showing an added link between *poetry* and *Poe* is exemplified

in Figure 3.1. The corresponding extended contexts are indicated by brackets in Table 3.1. By analogy to  $D^{-1}(w, T)$ , we extend  $D(w, T)$ , so that added links can be processed in both directions by means of the approach of Levy & Goldberg (2014). Note that we only consider anaphoric, but not cataphoric references which also allow for adding dependency links.

## 3.4 Experiments

### 3.4.1 Data Sets and Models

Our dataset used for training consists of the first paragraphs of 1 000 000 Wikipedia articles (effects of smaller datasets are analysed in Section 3.5.4) with almost 300 millions tokens, of which over 4 million (of almost 5.5 million) pronouns have been replaced or extended. The models used are the Skip and Cbow variant of Word2Vec (Mikolov et al., 2013b), Glove (Pennington et al., 2014) and Wang2Vec (Ling et al., 2015), Levy (Levy & Goldberg, 2014) and Komninos (Komninos & Manandhar, 2016). Word2Vec, Glove and Wang were trained with a fixed vocabulary of the 400.000 most commonly lemmatized tokens and Levy and Komninos with all lemmatized tokens that occurred at least 15 times in the data set. We trained all embeddings with a size of 300, standard parameters, window sizes of 2, 5 and 10, and 25 iterations.

### 3.4.2 Pre-processing

We used Spanbert-Base of Joshi et al. (2020) for coreference resolution. For the needed dependency features we used the AllenNLP’s (Gardner et al., 2018) implementation of Dozat & Manning (2017). For tokenization, lemmatization and POS tags, Spacy (Honninger & Montani, 2017) was used.

## 3.5 Evaluation

### 3.5.1 Word Similarity

The first analyses on the generated word vectors ran over various word similarity tasks. All results are listed in Table 3.2. For evaluation, we used the benchmark tool of Jastrzebski et al. (2017)<sup>1</sup> as it computes the accuracy for a lot of important Word Similarity and Analogy Tasks. We used: (MEN (Bruni et al., 2014), WS353 (Finkelstein et al., 2001), SimLex999 (Hill et al., 2015), RW (Luong et al., 2013), MTurk-287 (Radinsky et al., 2011), Google (Mikolov et al., 2013b), SemEval2012\_2 (Jurgens et al., 2012)). We compare the unmodified dataset (c-version) with a version, where we replaced pronouns with the complete antecedent (p-version, described in Section 3.3.1), replaced everything with the mention-head (h), and replaced only pronouns with the mention-head (ph-version,

---

<sup>1</sup><https://github.com/kudkudak/word-embeddings-benchmarks>



Ins/Hyp (Window)	Conc		Diff		DDSq	
	c	ph	c	ph	c	ph
I-NotInst (10)	81.09	81.09	78.97	80.48	80.73	<b>82.00</b>
I-Inverse (10)	98.79	98.34	99.09	99.24	<b>98.79</b>	99.24
I-I2I (10)	95.80	<b>96.40</b>	92.20	92.80	92.20	92.80
I-Union (10)	<b>84.94</b>	84.41	77.58	76.88	79.77	78.98
H-NotHyp (10)	55.16	55.53	54.32	54.23	72.84	<b>73.12</b>
H-Inverse (10)	81.75	79.56	<b>83.84</b>	82.01	83.76	81.92
H-C2C (10)	69.03	68.57	64.15	64.52	79.23	<b>79.78</b>
H-Union (10)	42.73	42.24	40.79	40.30	<b>52.98</b>	52.26
I-Union (2)	<b>86.25</b>	85.20	77.67	77.50	79.16	78.37
H-Union (2)	45.13	44.36	43.19	41.29	<b>53.61</b>	53.29

Table 3.3: Results on the Instances and Concepts dataset (Boleda et al., 2017) with the Cbow model.

described for dependency in Table 3.3.2). For most context window-based embeddings, the results based on the data set containing the co-reference do not differ markedly. It is noteworthy that the p-version is usually worse than the c-version. The observed reductions in the case of context window-based approaches can be explained by the effect of the loss of semantic contexts (see Section 3.5.5). The h- and ph-versions perform therefore better. We therefore only consider these versions in further analyses. But still, some embeddings have a tendency towards slightly better results (e.g. Cbow), while others tend to get a little worse (Wang2Vec). The best responding test data is by far Google, with an increasing of 5.52% with Cbow (2). The worst results were obtained on the RW and MTurk-287 data set. Intuitively, the results for coreference embeddings are better for small window sizes.

### 3.5.2 Instances versus Concepts

Next, we tested whether the vectors could better distinguish between *instances* or *concepts*. The embedding task including the test dataset was presented by Boleda et al. (2017). The data set consists of word pairs  $(x, y)$  where a linear classifier is used to decide whether  $x$  is an instance or a hyponym of  $y$ . As a negative example, the data set contains various error cases, like  $\text{swap}(x, y)$  (inverse). Further details can be found in Boleda et al. (2017). As in the original work, we trained a linear logistic regression classifier with the concatenation (Conc), the difference (Diff) and the squared difference (DDSq) of the vectors as input. We used scikit-learn (Pedregosa et al., 2011) for implementing this. The results for the Cbow model are listed in Table 3.3 and for the Levy model in Table 3.4. Again, the vectors do not seem to achieve any performance improvement. However, with regard to the Union dataset, it appears that the results have tended to get worse.

Ins/Hyp	Conc		Diff		DDSq	
	c	ph	c	ph	c	ph
I-NotInst	<b>82.96</b>	80.84	81.45	80.54	81.90	80.84
I-Inverse	99.55	99.70	99.70	<b>99.85</b>	99.70	<b>99.85</b>
I-I2I	98.40	<b>98.60</b>	96.01	95.61	96.01	95.61
I-Union	<b>87.16</b>	87.07	80.00	80.17	81.22	80.70
H-NotHyp	56.55	57.66	53.67	53.11	67.69	<b>68.52</b>
H-Inverse	84.37	84.02	85.76	<b>86.46</b>	85.59	86.11
H-C2C	72.45	72.54	66.30	66.02	<b>74.56</b>	73.92
H-Union	44.74	45.06	41.13	41.08	<b>50.25</b>	50.16

Table 3.4: Results on the Instances and Concepts dataset (Boleda et al., 2017) with the Levy.

### 3.5.3 Feature Analysis

To analyze the results, we took the classification results of the development and test dataset from the linear classifier of Section 3.5.2 to decide, which words were classified better or worse. With this information we trained a *Decision Tree* (DT) and a Support Vector Machine (SVM) to predict whether the classification of a word  $w$  is improved or worsened when taking into account the following features: 1. How often did we use  $w$  to replace a pronoun according to Section 3.3 (Replacer), 2. Log-frequency of  $w$  in the corpus (VocabC), 3. Frequency in the test set (inTest) and 4. Character count of  $w$  (WordLen). The generated DT for the Cbow model with window size 10 on the H-Union dataset is shown in Figure 3.2. One observation is that words that appear more frequently in the corpus become slightly better, whereas words that are already rare tend to get worse. But as soon as words occur too often, they tend to get worse again. It seems that the embeddings already contain all necessary neighborhood information in the case of high-frequency words. Rare words, on the other hand, become even rarer and therefore their vector representations are worsened. The strongest feature for the SVM was VocabC and the log of Replacer, so we trained a small version with only these two features to show their behaviour in a two-dimensional space (see Figure 3.3). The results are similar to those of DT. However, with the decision boundaries it is recognizable how the word frequencies correlate with the results. The words tend to get better if they are neither too frequent nor too rare in the training data. The same applies to the replacement. One possible explanation is that common words already cover all information. Rare words, on the other hand, are rarely referenced by anaphora and do not benefit from this procedure. It should be noted that this is not so easy to detect with smaller window sizes.

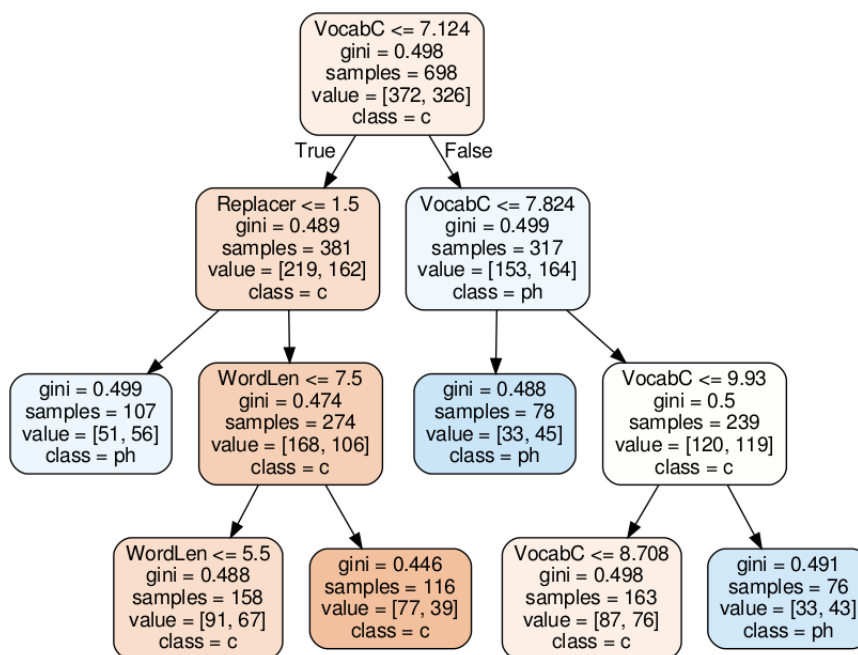


Figure 3.2: Decision Tree for classifying the error distribution on the H-Union dataset. Red nodes stand for word embeddings, which tend to get worse through pronoun substitution. Blue nodes tend to get better through pronoun substitution. Gini is a measure of the probability that a randomly selected element from the data will be misclassified. Value stands for the division of the samples into the two classes at this node.

Doc Count	Average			
	c	p	h	ph
100	4.41	5.05	5.00	5.15
1 000	14.92	15.54	15.35	<b>16.13</b>
10 000	<b>29.87</b>	29.79	28.80	29.67
100 000	38.56	38.69	<b>39.54</b>	39.01
1 000 000	43.35	43.54	<b>43.71</b>	43.58

Table 3.5: Average results (see Section 3.5.1) for Cbow with vector size 100 and window size 10 on different amounts of Wikipedia articles.

### 3.5.4 Corpus Size

We have also tested different corpus sizes, but have not found any significant effect for them either. The results are listed in Table 3.5. Doc Count is the randomly selected number of Wikipedia articles.

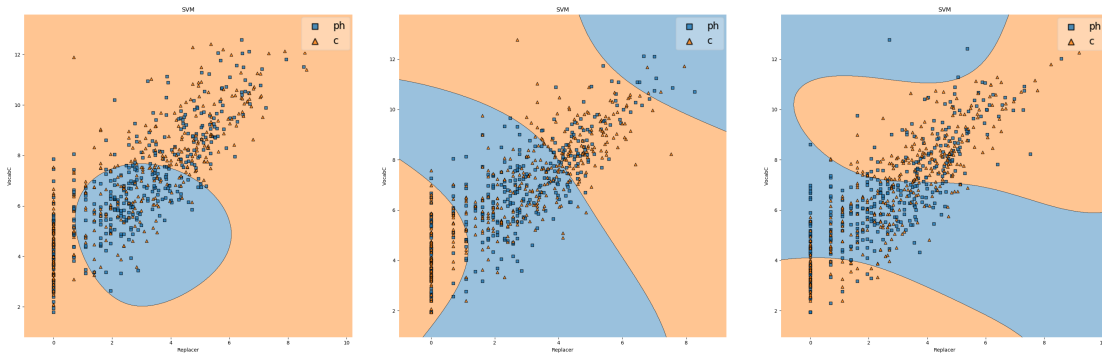


Figure 3.3: Error distribution of the SVM with  $\log(\text{Replacer})$  on the x-axis and VocabC on the y-axis. Cbow (left), Skip (middle), Glove (right) with window size 10. Red areas stand for word embeddings, which tend to get worse through pronoun substitution. Blue areas tend to get better through pronoun substitution. The decision-boundaries reveal, that words that are neither too frequent nor too rare in the corpus tend to produce better results if they are neither replaced too often nor too rarely.

### 3.5.5 Explanation of the results

#### Loss of Semantic Contexts

Windows-based embeddings achieve their quality by looking at which words appear together in observed windows. In the example

“ $[\text{Edgar Allan Poe}]_1$  was an American writer.  $[\text{Poe}]_1$  is best known for  $[\text{his}]_1$  poetry.”

↓

“ $[\text{Edgar Allan Poe}]_1$  was an American writer.  $[\text{Poe}]_1$  is best known for  $[\text{Edgar Allan Poe}]_1$  poetry.”

the distance between associated words (e.g. *writer* and *poetry*) increases so much by the substitution of *his* that the system is no longer informed about their association in this example. This effect is increased by the fact that we always replace pronouns with possibly longer mentions (experiment p). In this way, we amplify the effect that we originally wanted to avoid. The example also shows that substituting pronouns is not a trivial task and can distort the semantics of a sentence. The same may happen with syntax as shown in the example above.

#### Word Frequency

We were able to show that words that are neither too frequent nor too rare in the corpus tend to produce better results if they are neither replaced too often nor too rarely. In contrast, the use of frequent words to replace pronouns tend to noise out their already well-documented contextual information within the original corpus. And for rare words,

the additional context information gained by CR is not detailed enough to calculate better embeddings for them. However, it should be noted that the replacements have only led to a minimal increase in the volume of data.

### 3.5.6 Discussion

Our goal was not primarily to achieve the best results for the evaluation tasks we carried out, but to investigate the effects of coreference resolution on computing word embeddings. Actually, there is an effect, but only a small one. This finding indicates the need to further elaborate the interplay of pre-processing routines like coreference resolution and downstream tasks such as training word embeddings. With a more elaborated substitution function first:  $M \rightarrow M$  than the one implemented here better results might be achieved. An extension would be, for example, training with both sentences, the ones in which substitutions are made and the original ones. Replacing with (parts of) nominal phrases might distort the training as well. The use of only named entities could help with this problem, but would further reduce the amount of information obtained.

## 3.6 Conclusion

We experimented with improving word embeddings based on CR as a pre-processing step. We have shown that word embedding approaches do not tend to benefit significantly from pronoun substitution. The measurable improvements were only marginal, even though we could achieve strong improvements with Cbow on the Google dataset. In future work, we want to analyze the effect of linking all mentions of the same reference chain with each other (completely connected graph). In addition, we want to find out which dependency edges contribute to the information gain by training corresponding classifiers.



# 4 Transfer of ISOSpace into a 3D Environment for Annotations and Applications

Henlein, A., Abrami, G., Kett, A., & Mehler, A. (2020). Transfer of isospace into a 3d environment for annotations and applications. In *16th Joint ACL - ISO Workshop on Interoperable Semantic Annotation PROCEEDINGS* (pp. 32–35). Marseille: European Language Resources Association

## Abstract

People’s visual perception is very pronounced and therefore it is usually no problem for them to describe the space around them in words. Conversely, people also have no problems imagining a concept of a described space. In recent years many efforts have been made to develop a linguistic scheme for spatial and spatial-temporal relations. However, the systems have not really caught on so far, which in our opinion is due to the complex models on which they are based and the lack of available training data and automated taggers. In this paper we describe a project to support spatial annotation, which could facilitate annotation by its many functions, but also enrich it with many more information. This is to be achieved by an extension by means of a VR environment, with which spatial relations can be better visualized and connected with real objects. And we want to use the available data to develop a new state-of-the-art tagger and thus lay the foundation for future systems such as improved text understanding for text-to-3D scene Generation.

## 4.1 Introduction

Humans have a strong spatial perception. This is reflected not only in how well people can adapt to new spatial environments, but also in their language (Haun et al., 2011).

In recent years there have been increased efforts to create a linguistic model for these spatial references. This led to new linguistic models, like ISOSpace (ISO, 2014a) and SceneML (Gaizauskas & Alrashid, 2019) and new tasks, such as Spatial Role Labeling (Kordjamshidi et al., 2010) or SpaceEval (Pustejovsky et al., 2015). Nevertheless, these annotation schemes have not really been able to establish themselves in applications so far. This could be due to the models’ complexity, the availability of annotated training data

and the lack of automated taggers. There were indeed approaches to apply such models to image descriptions (Pustejovsky & Yocum, 2014), but to our knowledge there were no efforts to transfer the corresponding annotation schemes into three-dimensionality. For the latter, the language model would be particularly interesting, for example, to reconstruct scenes from speech and text three-dimensionally.

In this paper we present our project plan on a 3D VR framework that addresses the problems mentioned above and offers a direct application. In Section 4.2 we describe the models and systems we refer to in our project, and in Section 4.3 we explain how we build on these models to create a framework that supports both annotation and application of these language models.

His [room]<sub>p1</sub>, a proper [room]<sub>p1</sub> for a human being, only somewhat too small, lay quietly [between]<sub>ss1</sub> the four well-known [walls]<sub>se1</sub>. [Above]<sub>ss2</sub> the [table]<sub>se2</sub>, [on]<sub>ss3</sub> which an unpacked collection of [sample cloth goods]<sub>se3</sub> was spread out, hung the [picture]<sub>se4</sub> which he had [cut out]<sub>m1</sub> of an illustrated [magazine]<sub>se6</sub> a little while ago and [set in]<sub>m2</sub> a pretty gilt [frame]<sub>se7</sub>.

QSLINK(p1, se1, ss1, between)  
 QSLINK(se3, se2, ss3, EC)  
 OLINK(se3, se2, ss3, above)  
 OLINK(se4, se2, ss2, above)  
 MOVELINK(m1, se4, se6, se4)  
 MOVELINK(m2, se4, se4, se7)

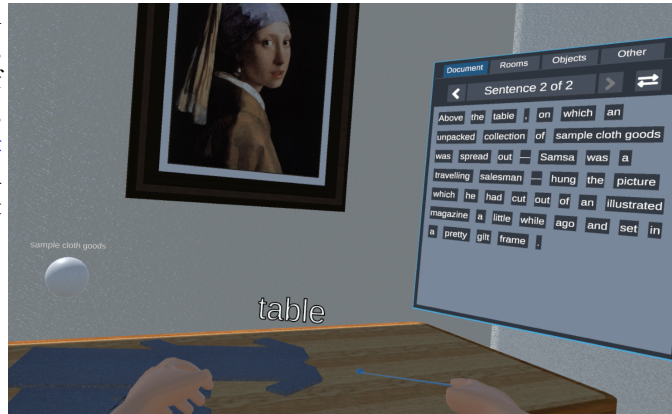


Figure 4.1: IsoSpace annotation example. On the left side a (simplified) annotation of an abridged section of Kafka's: The Metamorphosis according to the ISOSpace (2014) scheme. On the right side a 3D representation. Each entity in the text is linked to the corresponding 3D object from ShapeNetSem and we linked the two clothing to one object group. The relationship between the table and the room is not explicitly mentioned, but is implied by the placement of the table in the room.

*p*: place, *se*: spatial entity, *ss*: spatial signal, *m*: move event.

QS/OLINK(*figure*, *ground*, *signal*, *relation*). MOVELINK(*move*, *mover*, *source*, *goal*).

## 4.2 Related Work

In recent years, much work has been spent on the development of linguistic models for the semantic understanding of language. The largest of these is probably the Semantic Annotation Framework (SemAF), published under ISO/TC 37/SC 4/WG 2 Semantic Annotation. This consists of individual modules that relate to specific semantic units and are



compatible with each other (Ide & Pustejovsky, 2017, Chapter 4). The most widespread model of SemAF is ISOTimeML (Pustejovsky et al., 2010; ISO, 2012a), a scheme for the annotation of time and time dependencies of events based on TimeML (Pustejovsky et al., 2005a). Such dependencies are important for text understanding, because without them text contents can hardly be fully understood (Ide & Pustejovsky, 2017, p. 942). There is also a model that focuses more on spatial and spatial-temporal structures, the ISOSpace (Pustejovsky et al., 2011a; ISO, 2014a). The focus is on spatial and spatial-temporal relations between (spatial) entities and the connection via motion events. Spatial Entities are marked and connected to each other via different spatial connections. QSLinks (Qualitative Spatial Links) are for topological relations, OLinks (Orientation Links) for non-topological relations and MoveLinks for movements of entities in space. This scheme was the basis of SpaceEval (Pustejovsky et al., 2015) and was successfully applied to image descriptions to differentiate between content and structural statements (Pustejovsky & Yocum, 2014).

ISOSpace in particular is being further improved (ISO, 2020) and serves as a basis for more specialized models, such as SceneML (Gaizauskas & Alrashid, 2019) for scene descriptions. In addition, SemAF contains schemata such as Semantic Roles (ISO, 2014b), Dialog Acts (ISO, 2012b) and other modules are under development, e.g. QuantML (Bunt et al., 2018).

As the requirements for the annotation of text contexts are constantly changing, flexible and dynamic annotation environments are required to enable the efficient annotation of complex situations. This challenge is addressed by TEXTANNOTATOR (Abrami et al., 2019a), a browser-based and therefore platform-independent annotation tool for collaborative multi-modal annotation of texts. Using TEXTANNOTATOR, NER annotations can be created in texts in a short execution time as well as the annotation of rhetorical (Helfrich et al., 2018), time, propositional and even argument structures can be graphically visualised and executed. Furthermore, texts can be linked to ontological resources (e.g. Wikipedia, Wikidata, Wiktionary) and the annotations are managed in different annotation views based on user and group-based permissions (Gleim et al., 2012). As a result, TEXTANNOTATOR is capable of creating a real-time calculation of an inter-annotator agreement based on classes defined in the annotation task (Abrami et al., 2020c).

Since humans are spatially anchored not only in their actions and perception but also in their linguistic behavior (Bateman, 2010; Bateman et al., 2010), this led to new efforts to spatially translate annotations by means of virtual reality. One of these projects is VANNOTATOR (Spiekermann et al., 2018), a system for the annotation of linguistic and multi-modal information units, implemented in Unity3D<sup>1</sup>. VANNOTATOR is a platform for use in various scenarios such as visualization and interaction with historical information (Abrami et al., 2020b) or the annotation of texts and the linking of texts and images with 3D objects (Mehler et al., 2018). Since VANNOTATOR integrates TEXTANNOTATOR and thus makes the annotation spectrum of the latter available in VR, annotations in VANNOTATOR can be performed collaboratively (in workgroups) as well as simultaneously.

---

<sup>1</sup><https://unity.com/>

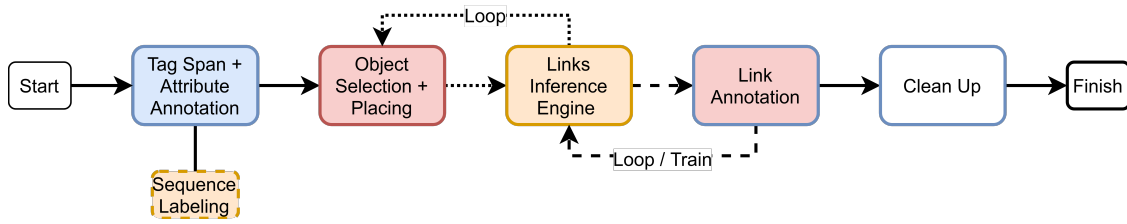


Figure 4.2: Workflow for ISOSpace Annotation. Blue borders stand for the original annotation steps (Pustejovsky et al., 2015). Red filled for VR support and orange for machine learning support. Span tagging can be supported with a sequence labeling system. And the link inference engine learns through annotations.

### 4.3 Our Current Project

ISOSpace is a very expressive model, but its complexity makes it difficult to use it as a basis for annotation. Work is not made easier when 3D information is annotated on a 2D surface. This becomes particularly clear in the annotation of spatial relations between entities, where, e.g., in the case of SpaceEval data, the inter-annotator agreement was only 33% for QSLinks and 39% (Pustejovsky et al., 2015) for OLinks. These are hardly values that guarantee high data quality. Here an extended visualization, as our project aims at, could significantly support these annotation tasks.

To this end, our aim is to integrate ISOSpace and other SemAF models such as ISO-TimeML into TEXTANNOTATOR. Since TEXTANNOTATOR is based on UIMA (*Unstructured Information Management Applications*) (Ferrucci & Lally, 2004), its annotation schemes are defined as UIMA TYPE SYSTEM DESCRIPTORS (TSD). Before the ISO models can be used in UIMA, they have to be transferred to TSD. This is the first step towards collaborative annotation in a visually supporting interface. The annotation can then be enriched by TEXTANNOTATOR embedded into VANNOTATOR. This enables spatial annotations with a 3D interface in VR. In addition, spatial entities can be directly linked to 3D objects via a large number of categorized objects from ShapeNet (Chang et al., 2015a), the slightly deeper annotated objects from ShapeNetSem (Savva et al., 2015), objects annotated using VoxML notation (Pustejovsky & Krishnaswamy, 2016) (under development) or via abstract representations (as exemplified in Figure 4.1). Simply by placing the objects in space, conclusions can be drawn about the relationships between them (and thus also about QSLinks and OLinks) because the information bandwidth of annotation acts in VR is much larger than with pure text annotation. For example, if a book is placed on the desk in VR, the corresponding QSLink and OLink can be set automatically with their relevant attributes. Such concrete pictorial representations are not always unambiguous, but in conjunction with the corresponding sentence, classifiers can be trained to solve this (Hürlimann & Bos, 2016). This can also be extended to MoveLinks, which are set automatically when, for example, the book is carried through the room and placed on a shelf. Or the annotator can follow a direction described in the text in the VR environment. Such actions are much more natural and easier for humans to perform than abstract annotations in a 2D display. Missing links can thus be more easily

identified and in some cases automatically predicted and attributed, e.g., by examining transitive relations. Such support has also been successfully applied to the annotation of the TimeML standard (Setzer et al., 2005; Verhagen et al., 2006; Verhagen, 2007). The underlying workflow is shown in Figure 4.2.

A central challenge will be the underspecification of scene descriptions. Related issues concern descriptions containing negations. Though we do not yet have a solution to solve the problems involved, we assume that by combining spatial experience in VR with annotation services provided by annotators, for example, underspecified reference relations can be annotated by exploring additional information with regard to the annotators' positions in relation to referred objects. In examples such as "There is no book on the table" a corresponding book object can be highlighted to indicate the negation (as done, e.g., in WordsEye (Coyne & Sproat, 2001)). In the case of underspecified relations, as expressed in examples of the sort of "The pencil is next to the book", there is the possibility of assigning relative or variable positions to objects (so that they take up tipping states in the visualization).

The next step is the stepwise extension of our annotation system by further (e.g. ISO-TimeML) and future (e.g. QuantML (Bunt et al., 2018)) SemAF modules. In this way we create a multi-modal, virtualized annotation system capable of mapping text to abstract or concrete spatial representations of a very broad complexity.

The available ISOSpace data will then be used to develop and train taggers that automatically perform or largely support this annotation. The taggers can support annotators with annotation suggestions, which the annotators then only have to accept or minimally correct.

TEXTANNOTATOR is already actively used for annotating historical text data in the BIOfid project<sup>2</sup>. These annotations (Ahmed et al., 2019) will be extended in the near future to include ISOSpace, ISOTimeML, SemAF-SR and probably also QuantML.

Such in-depth annotations could form the still missing basis for text-to-3D scene systems (Coyne & Sproat, 2001), which in turn should be able to provide a much deeper understanding of spatial language than previous systems that focus primarily on key words (e.g. (Chang et al., 2017b; Ma et al., 2018)). Application areas could be, for example: Reconstructing events from multiple texts (based on Twitter, news reports, etc.), visualizing descriptions of accidents (Johansson et al., 2005) or crime scenes or 3D visualizations of text content to clarify certain relations (e.g. intersections of biographical life paths). This could also help to identify weaknesses of the ISOSpace model, such as missing information relevant for spatial annotation. A problem that could occur is that RCC (Region Connection Calculus) (Randell et al., 1992) for representing topological relations of regions is not sufficient to represent 3D spaces. One reason is that it does not refer to a specific dimension (Renz, 2002).

---

<sup>2</sup><https://www.biofid.de/en/>

## 4.4 Conclusion

We argued that ISOSpace, despite its expressiveness, has not yet reached the application density that is essential to provide training data for tools for automatically annotating spatial language. To fill this gap, we plan to integrate ISOSpace into VANNO-TATOR to enable 3D annotations of spatial language. This will also include other SemAF models in order to ultimately provide the data basis for the creation of text-to-3D scene systems.

# 5 Text2SceneVR: Generating Hypertexts with VAnnotatoR as a Pre-processing Step for Text2Scene Systems

Abrami, G., Henlein, A., Kett, A., & Mehler, A. (2020a). Text2SceneVR: Generating hypertexts with vannotator as a pre-processing step for text2scene systems. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media*, HT '20 (pp. 177–186). New York, NY, USA: Association for Computing Machinery

## Abstract

The automatic generation of digital scenes from texts is a central task of computer science. This task requires a kind of text comprehension, the automation of which is tied to the availability of sufficiently large, diverse and deeply annotated data, which is freely available. This paper introduces TEXT2SCENEVR, a system that addresses this bottleneck problem by allowing its users to create a sort of spatial hypertexts in *Virtual Reality* (VR). We describe TEXT2SCENEVR's data model, its user interface and a number of problems related to the implicitness of natural language in the manifestation of spatial relations that TEXT2SCENEVR aims to address while trying to remain language independent. Finally, we present a user study with which we evaluated TEXT2SCENEVR.

## 5.1 Introduction

Human information processing is strongly spatial, not only in perception but also in language (Lakoff, 1987). It is not a problem for us, for example, to describe scenes or mentally reconstruct scenes from linguistic descriptions. Take the following text (Sample 1): “*During my last conference, I stayed in a beautiful hotel room with a red sofa, dark blue curtains and a breathtaking view of the old town, which was offered to me through my window beside the desk.*” Imagining the scene described by this sentence is no problem for us, while computers still have fundamental difficulties in visualizing even elementary aspects of it. In particular, relations that are not directly mentioned, but are part of general knowledge, for example, are particularly difficult for computers to process (e.g. the fact that the curtains being mentioned are probably attached to the window).

Although texts can be processed with various linguistic tools (e.g. Qi et al., 2020; Manning et al., 2014; Akbik et al., 2018; Hemati et al., 2016; Gardner et al., 2018), entities be recognized and mapped to 3D objects (e.g. Coyne & Sproat, 2001; Ma et al., 2018; Chang et al., 2017b), we are still very far from understanding such texts on a similar level as humans.

In this paper we introduce TEXT2SCENEVR, an open hypermedia system for generating a special type of spatial hypertext, which aims to generate data for the training of Text2Scene systems (Coyne & Sproat, 2001). The spatial data available so far usually have no textual description (e.g. SUNCG (Song et al., 2017)), and if they do, they are rather sparse and only connect complete rooms with generic statements (e.g. Stanford Text2Scene (Chang et al., 2014b)). There is no assignment between component objects and the corresponding text sections, which would provide additional information for training. In this way, the basic problem of such text understanding systems, namely the lack of sufficiently large, deeply annotated and openly accessible data, is addressed. This data bottleneck problem currently prevents the effective development of systems that automatically map texts to computer-based scene representations. By transferring this annotation problem to virtual reality and thus associating it with 3D, spatial hypertext, we benefit from the extended annotation possibilities offered by such systems (e.g. Spiekermann et al., 2018). Our approach is to specifically address the problem of the implicitness of natural language: for this purpose, we allow users to extend input texts with sentences and text segments which, from their point of view, are related to the input (by being entailed by it) but not explicitly mentioned. Annotation with TEXT2SCENEVR then means to connect segments of the input text with virtual objects or their spatial relations and to do the very same with entailed descriptions.

The resulting hypertexts can than be used to train systems that are ideally able to do this themselves.

To generate such systems, we distinguish the following relations:

1. **Object recognition:** The first task concerns the identification of described objects. This requires entity recognition methods that recognize successive descriptions of the same objects, their attributes and relations, in texts.
2. **Referential meaning relations:** In order to recognize objects correctly one has to understand the meanings of their linguistic descriptions. This relates to explicit as well as implicit descriptions. Explicit descriptions are usually manifested by definite noun phrases (e.g. *the red sofa*). Implicit descriptions concern under-specified, possibly contradictory, vague or otherwise informationally uncertain descriptions. When referring, for example, to a hotel room it can be implicitly assumed that a bed is likely contained in it. But this does not need to be mentioned in its description. In any event, it is expected that the result of an automatic processing of scene descriptions itself is not under-specified or too low in content. With TEXT2SCENEVR we introduce a tool for generating annotation data for training systems that automatically interpret under-specified scene descriptions.
3. **Part-whole-relations:** A related challenge concerns implicit descriptions of part-

Relation		Argument of relation (object)	
		explicit	implicit
Referential meaning	explicit	$OR_1$	$OR_2$
	implicit	$OR_3$	$OR_4$
Part-whole	explicit	$OH_1$	$OH_2$
	implicit	$OH_3$	$OH_4$
Topological	explicit	$OM_1$	$OM_2$
	implicit	$OM_3$	$OM_4$

Table 5.1: Matrix of arguments and relations.

whole-relations of objects. Depending on the level of detail of the scene description, it may be necessary to additionally refer to components of objects and the materials of which they consist. In the above text, the visualization of the bed may include, for example, references to mattresses, sheets, pillows, quilts, covers, etc.

4. **Topological relations:** Beyond part-whole-relations we have to consider that objects are topologically arranged - and again, this may not be explicitly expressed (as in “*The printer is placed besides the PC*” – left or right?). This also refers to spatial distance, perspectivation, scaling and contextual relations (such as the left-right distinction). The following sentence illustrates how scale descriptions can be underspecified: *The big ant is on the little elephant.* (Kamp, 1975) We can assume that the elephant is significantly larger than the ant. That is, the attribute is scaled relative to the referenced entity. But also the pronoun “on” is ambiguous: in the sentence “*The ant is on the airplane*”, *on* means “inside” and not “on top”.

Based on these preliminaries, we distinguish relations (rows in Table 5.1) and their arguments (columns in Table 5.1) in order to specify 16 sub-tasks of Text2Scene systems. Given the example “*I leafed through my newspaper in my garden in the shade of a tree while leaning on its root*”, Table 5.2 lists sentences that are either explicitly or implicitly entailed by this sample and thereby exemplify the cases distinguished by Table 5.1. Since these cases are usually mixed, it is very difficult to correctly identify object relations expressed in texts and to convert them into scenic representations. This task of automatically generating scenes based on text descriptions is approached under the name *Tex2Scene*. Only humans are currently capable of solving this task. In order to train Text2Scene systems appropriately, we need both: sufficiently deep and accurate annotations of texts like Sample 1, but also annotations of the same quality of sentences and texts entailed by such samples in order to get a better grip on the problem of underspecified space and object descriptions. TEXT2SCENEVR is dedicated to this task.

Different disciplines have varying views on how to define a *scene*. In linguistics, the “narrated space” is often divided hierarchically. These range from the lowest level of the “spatial framework” in which the current action takes place to the highest level of the “narrative universe” (Dennerlein, 2009, chap. 2.3), which describes “the world (in the spatio-temporal sense of the term) presented as actual by the text, plus all the counterfac-

Type	Example
$OR_1$	“I <sub>o</sub> have <sub>r</sub> a garden <sub>o</sub> .”
$OR_2$	“I lean <sub>r</sub> with my <b>back</b> <sub>o</sub> against the root <sub>o</sub> .”
$OR_3$	“I <sub>o</sub> <b>read</b> <sub>r</sub> the newspaper <sub>o</sub> .”
$OR_4$	“The <b>weather</b> <sub>o</sub> is <sub>r</sub> <b>sunny</b> <sub>o</sub> .”
$OH_1$	“The root <sub>o</sub> is part_of <sub>r</sub> the tree <sub>o</sub> .”
$OH_2$	“The tree <sub>o</sub> has <sub>r</sub> <b>branches</b> <sub>o</sub> .”
$OH_3$	“The newspaper <sub>o</sub> <b>consists_of</b> <sub>r</sub> pages <sub>o</sub> .”
$OH_4$	“ <b>Leaves</b> <sub>o</sub> hang <sub>r</sub> from the <b>branches</b> <sub>o</sub> of the tree.”
$OM_1$	“I <sub>o</sub> am in <sub>r</sub> my garden <sub>o</sub> .”
$OM_2$	“The <b>grass</b> <sub>o</sub> beneath me is in <sub>r</sub> the shadow <sub>o</sub> of the tree.”
$OM_3$	“The newspaper <sub>o</sub> is <b>in_front_of</b> <sub>r</sub> me <sub>o</sub> .”
$OM_4$	“The <b>sun</b> <sub>o</sub> is <b>behind</b> <sub>r</sub> the tree <sub>o</sub> .”

Table 5.2: Sentences exemplifying referential meaning, topological, and part-whole relations as distinguished by Table 5.1. All these sentences are entailed by Sample (1). Mentions of objects and relations are identified by  $o$  and  $r$ , with implicit mentions in bold.

tual worlds constructed by characters as beliefs, wishes, fears, speculations, hypothetical thinking, dreams, and fantasies (Ryan, 2012, chap. 2.1e)”.

For psychology, a scene is much more object-bound. Thus, in experiments, a scene is often understood as a set of all objects included by the corresponding scene (e.g. Greene, 2013), or of all objects that are perceived (e.g. Vö et al., 2019) in the scene. Existing scene synthesis systems interpret objects in a similar way to psychologists and describe scenes as a series of objects in space arranged in a certain way (Zhang et al., 2019). The creation of scenes from text are usually realized in three steps: *preprocessing/parsing*, *optimization/inference* and *generation* (Zhang et al., 2019; Chang et al., 2017b). The formalisation is as follows (Chang et al., 2017b):

$$P(s|u) = P(t|u)P(t'|t)P(s|t') \quad (5.1)$$

where  $u$  is the original utterance used to generate the scene  $s$ ,  $t$  is the original scene template and  $t'$  is the optimized one.  $P(t|u)$  is therefore the parsing phase in which the template is generated from the input,  $P(t'|t)$  the interference phase in which the template is optimized and  $P(s|t')$  the generation phase in which the final scene is generated from the template. The recognition of objects and their direct relationships can be processed directly in the parsing phase through extensive NLP pre-processing. Usually the problems arise from the implicit relationships, which can be resolved in the *Inference phase*. However, interpretations of *meaning representations of objects*, *part-whole relations* and *contiguity relations* mostly depend on the respective context and the availability of general knowledge. Many systems try to solve the corresponding bottleneck problem by using knowledge bases like WordNet (Miller, 1995) or ConceptNet (Speer et al., 2017)



(e.g. *WordsEye* (Coyne & Sproat, 2001) or *SceneMaker* (Hanser et al., 2009a, 2010)); alternatively they use data-driven methods (e.g. *SceneSeer* (Chang et al., 2017b) or Ma et al. (2018)).

It becomes clear that Tex2Scene is a highly challenging task that requires certain conditions for its implementation: A high degree of machine learning is necessary, which in turn requires the availability of appropriate, deeply annotated training data that contain a variety of under-specified representations of spatial relationships and explicate these as far as possible. Our approach is to generate these training data in the form of spatial hypertexts. Since there is currently no patent solution for this scenario, we extended VANNOTATOR (Spiekermann et al., 2018; Abrami et al., 2019b) – an open hypermedia system for the visualisation and annotation of graph structures for the representation of natural language texts. More specifically, we added the functionality of visualizing and annotating spatial hypertexts in *Virtual Reality* (VR). Moreover, the relations described in Table 5.1 are processed succesively and beside our focus on the generation of training data for machine learning, the interaction of users with objects can be included in later stages.

This paper describes an extension of VANNOTATOR as a system for generating spatial hypertexts, the underlying annotation model, its exemplification and evaluation, the so-called **TEXT2SCENEVR** and is structured as follows: Section 5.2 provides an overview of related work. Section 5.3 describes the annotation environment, the architecture, and the dataset used for evaluation, while the annotation model is described in Section 5.4. Afterwards, Section 5.5 presents the evaluation of VANNOTATOR and Section 5.6 outlines future work. The paper is summarized in Section 5.7.

## 5.2 Related Work

There are some projects that focus on spatial hypertexts but the aspect of three-dimensionality is not considered in our knowledge. For this reason this overview is basically functional and we refer to comprehensive overview articles. Therefore a well overview of these mostly older projects and the observation of the absence of a common vocabulary regarding spatial hypertexts, see (Bernstein, 2011).

The understanding of spatial hypertexts used to consist more in the visualization of these graph structures. Accordingly, the origins were rather browser-based procedures with the goal of visualizing the underlying networks (Thüring et al., 1991; Marshall & Shipman III., 1995). An example for this is the tool *VIKI* (Marshall et al., 1994). The system supports the reader by its visual representation through the spatial usability of relative nodes as well as the writer by means of a visual language. In the following years, the system was continuously further-developed (Marshall & Shipman III., 1995, 1997).

The *Visual Knowledge Builder* (VKB) (Shipman III. et al., 2001) considered itself a “second generation” of spatial hypertexts. The focus of this system was on long-term cooperation and linking through the introduction of processing histories. Since then, many other applications have been developed and enhanced according to the Spatial Hypertext principle, e.g. Wikis (Solís & Ali, 2008), visualization of relevant content (Roßner

et al., 2019), use of a document store for data storage (Rubart, 2019) or interpretation of spatial ambiguity (Francisco-Revilla & Shipman, 2005).

Related work also concerns VR-based systems. Since it is impossible to list all relevant VR projects, we highlight selected projects which focus on dynamic virtual environments. 3D visualizations of software code that enable immersive “flights” by users, where classes are represented as virtual solar systems or as cities is described by (Oberhauser & Lecon, 2017). Likewise Kett et al. (Kett et al., 2018) introduce *resources2city*, a system for visualizing and interacting with file systems represented as cities, while (Wolf et al., 2017) examine the effects of virtualized architectural structures on users. In addition, (Nguyen et al., 2017) introduce *Vremiere*, a video editing tool designed to break the boundaries of 2D applications by processing and visualizing in 3D environments. There is also a range of earlier projects that address information management and retrieval using 3D environments such as (Card et al., 1991, 1996; Benford et al., 1997). All these tools have in common that they allow for rich object-related annotations in VR making use of spatial metaphors for information modeling. Our task will be to add the generation of training data for the automatic recognition of spatial structures to this area.

Applications in VR are also available in different fields of application such as medicine (Kuehn, 2018)), psychiatry (Benbouriche et al., 2014) and learning (Sampaio et al., 2013; Naranjo et al., 2017). The first uses VR to develop immersive therapies for patients with post-traumatic stress disorders. The second investigates the potential of VR in forensic psychiatry. Thirdly, (Naranjo et al., 2017) describe the use of 3D environments in teaching autistic children. Although there are many projects of this kind, there is no system that allows to generate training data especially for Text2Scene systems. *TEXT2SCENEVR* is being developed to fill this gap.

A second field of related work concerns semiotic analyses of VR, which are rather rare (see (Barricelli et al., 2016) for a review of this literature). We concentrate on the few articles that focus on an operative, at least classificatory concept of semiotics. (Marini et al., 2012) provide a semiotic analysis from the point of view of pragmatics and especially rhetoric. (Barricelli et al., 2016) extend this approach by considering syntactic, semantic and pragmatic aspects of classifying VR systems. In doing so, they focus primarily on visual, iconic signs. (Barricelli et al., 2018) use this classification in a user study of eight VR systems. In contrast to these approaches, we start with an analysis of linguistic signs to enter the field of indexical (hyperlinks) and iconic signs (3D simulations).

Furthermore, there are numerous works focusing on the recognition of objects in texts and their transformation into three-dimensional representations. The first successful system was *WordsEye* (Coyne & Sproat, 2001). This has been further developed until today and is one of the linguistically most flexible Text2Scene systems (Hassani & Lee, 2016) because of the resulting resources like *VigNet* (Coyne et al., 2011) and *Spatial-Net* (Ulinski et al., 2019). *WordsEye* is largely based on manually annotated rules for processing input texts.

The *StanfordText2Scene* (Chang et al., 2015b, 2014a,b) project is based on the *Stanford-NLP* pipeline (Manning et al., 2014) and therefore includes a wide range of pre-processing tools. Since the placement of objects is based on statistically learned spatial knowledge and the system enables interaction with the user, *StanfordText2Scene* learns from user

behavior, improving both the selection and placement of objects. In more recent works, the focus continues to be on the realistic representation of rooms and groups of objects, rather than on language analysis (e.g. (Ma et al., 2018)). In general, most Text2Scene systems lack sufficient linguistic pre-processing or the ability to post-correct generated scenes or rooms manually (Hassani & Lee, 2016).

There are efforts to learn from human corrections (e.g. Chang et al., 2014a) or map certain linguistic expressions to spatial references, such as *VigNet* (Coyne et al., 2011) (an extension of *FrameNet* (Baker et al., 1998)) or *SpatialNet* (Ulinski et al., 2019), but these only refer to individual linguistic phenomena and the data is not publicly available. In the meantime, efficient systems have been developed that map linguistic descriptions to images (Zitnick et al., 2013; Tan et al., 2019) or generate descriptions for images (Vinyals et al., 2015). But even these try to avoid the problems mentioned above by using ever larger neural end-to-end systems, which require even larger large data sets, such as *COCO* (Lin et al., 2014) or *Conceptual Captations* (Sharma et al., 2018). However, *these datasets are not yet available for 3D*. More specifically, in the present context, end-to-end learning means that the entire model is differentiable so that it can therefore be trained via gradient descent. Since these models often consist of millions of parameters that are trained via training data, correspondingly large amounts of data are required (Glas-machers, 2017). Note that end-to-end learning has established itself as a state-of-the-art method in many NLP areas such as coreference resolution (Lee et al., 2017) or speech recognition (Hannun et al., 2014).

## 5.3 From VANNOTATOR to Generating and Annotating Virtual Rooms

For the generation of spatial hypertext it is necessary to learn topological as well as part-whole relations from texts. For this, the implementation of a system for the generation spatial hypertexts has already been the object of previous work (Mehler et al., 2018), the so-called VANNOTATOR (Spiekermann et al., 2018). VANNOTATOR allows for creating, visualizing and interacting with multimedia data (texts, images, segments of texts and images, geo-coordinates, video and audio files, URLs (by means of virtual browsers) and 3D models of objects and especially of (virtual reconstructions of) buildings. For this purpose, 3D glasses (*HTC Vive*<sup>1</sup> and *Oculus Rift*<sup>2</sup>) are used as VR devices, while Google's *ARCore*<sup>3</sup> is used as a platform for *Augmented Reality* (AR) devices (Mehler et al., 2018). In addition to visualization and interaction with objects, annotation, i.e. the explicit relation of signs and objects, is an essential feature of VANNOTATOR. This functionality depends on the type of the object: texts and images can be segmented with VANNOTATOR, for example, links are processed with its virtual browser and video files are processed using a virtual viewer. Beyond that, VANNOTATOR includes a variety of methods for the

---

<sup>1</sup>[https://www.vive.com/de/product/#vive\\_series](https://www.vive.com/de/product/#vive_series)

<sup>2</sup><https://www.oculus.com/rift/>

<sup>3</sup><https://developers.google.com/ar>



interaction with 3D content:

- **Highlighting:** links between objects can be highlighted by the user to get an overview or to create reminder marks.
- **Looking ahead:** remote objects linked to an object can be visualized with a preview function, especially if they are out of sight in virtual space. This preview serves as a preparation step for what we call teleportation.
- **Teleportation:** in order to bridge the spatial distances between objects, portals can be created which visualize a preview of the target and, when used (selected or entered), provide a virtual transportation to the remote object (Mehler et al., 2018).

The automatic generation of scenes in VR based on text descriptions makes it necessary to train suitable machine-learning models, which in turn are bound to the sufficient availability of training data. To this end, we implemented and tested an annotation model (see Section 5.4) which is based on the core technology of VANNOTATOR. The annotation model is exemplified by virtual rooms. Thus, the present paper describes the extension of VANNOTATOR for generating and annotating virtual rooms as a means to generate annotation data for training Text2Scene systems.

#### 5.3.1 VANNOTATOR's Core Functionality

VANNOTATOR<sup>4</sup> was developed as a virtual research platform for the visualization, annotation and processing of multimedia content. VANNOTATOR processes a wide range of content objects: this includes (segments of) texts, images, videos and audio streams as well as 3D representation of buildings or places (Mehler et al., 2018). Figure 5.1 shows the software landscape in which VANNOTATOR is embedded (c.f. Spiekermann et al., 2018; Abrami et al., 2019b; Kett et al., 2018; Kühn et al., 2020). By integrating TEXTANNOTATOR (Abrami et al., 2019a), a platform-independent annotation tool, VANNOTATOR allows for annotating texts on various levels of text structuring (Kett, 2020). Amongst other things, this includes anaphoric relations, propositional structures, argument structures and rhetorical text structures (Abrami et al., 2019a). TEXTANNOTATOR operates on texts using the UIMA (Götz & Suhre, 2004; Ferrucci et al., 2009) format. In this way, external NLP tools can be easily integrated and, conversely, the output of TEXTANNOTATOR can be exchanged interoperably (Ide & Suderman, 2009). In fact, any UIMA document that is serialized and interchangeable via XML can be processed with TEXTANNOTATOR in this way. Thanks to the additional integration of TEXTIMAGER (Hemati et al., 2016), VANNOTATOR dispenses with the need to manually annotate documents virtually in raw format on all levels. That is, a wide spectrum of language levels is automatically pre-processed and annotated using the NLP pipeline (including tools for tokenization, named entity recognition, relation extraction, semantic role labeling, etc.) of TEXTIMAGER. In addition, by means of DUCC (Challenger et al., 2016)<sup>5</sup>, TEXTIMAGER allows for

---

<sup>4</sup>For videos introducing into VANNOTATOR see <https://tinyurl.com/w4jctvv>

<sup>5</sup>Distributed UIMA Cluster Computing

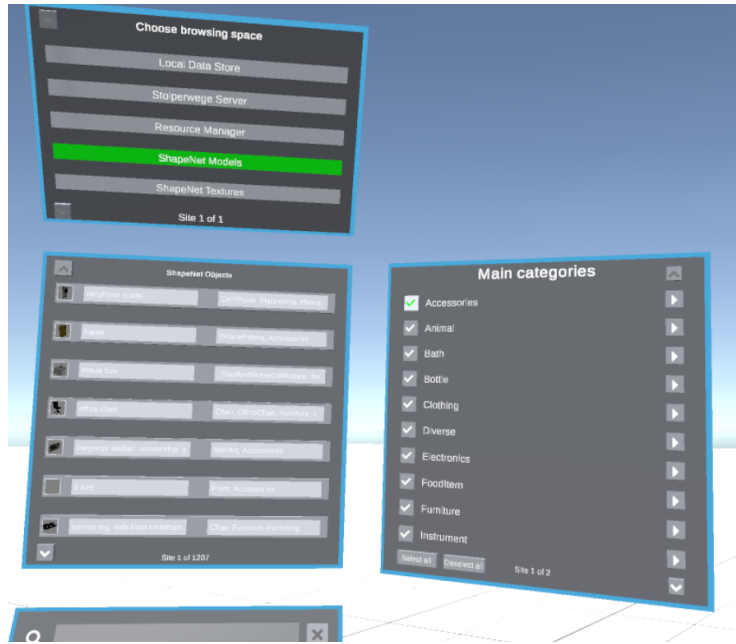


Figure 5.2: VANNOTATOR resources window. VANNOTATOR gives access to various external resources, whereby its users can choose between different sources: resources can be selected from the local computer, the *Stolperwege* server (Mehler et al., 2017), the *ResourceManager* (Gleim et al., 2012), or from *ShapeNetSem* (Savva et al., 2015).

processing large amounts of text data in a horizontally and vertically distributed server landscape. TEXTIMAGER generates UIMA documents, which are managed by the so-called *ResourceManager* and the *UIMA Database Interface* (UIMA-DI) (Abrami & Mehler, 2018). UIMA-DI is a database solution for the document-based approach of UIMA and enables the real-time use of UIMA documents for annotation processes. Annotations of UIMA documents are defined by means of annotation schemes, that is, so-called *UIMA Type System Descriptors*. With the help of *ResourceManager* (Gleim et al., 2012) UIMA documents can be given user and group related access rights. In addition to these documents, VANNOTATOR can process a number of other resources, as shown in Figure 5.2.

The communication between VANNOTATOR and TEXTANNOTATOR takes place via a web socket. This 1-to-1 connection of both tools enables direct interaction between different users without time-consuming requests for changes (Abrami et al., 2020c). TEXTANNOTATOR allows the simultaneous annotation of the same text by several users (Abrami et al., 2020c). To this end, views are generated so that texts can be annotated by different users in a collaborative manner or logically and contextually separated from each other. And since the views are provided with access rights, a very flexible use is guaranteed. In addition, the real-time evaluation of different views of the same documents in terms of the inter-annotator agreement allows their selection for training machine learning systems from a quality perspective (Abrami et al., 2020c).

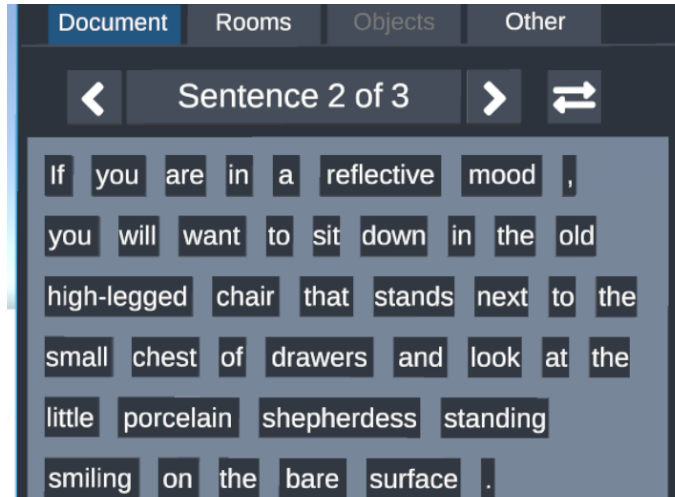


Figure 5.3: VANNOTATOR text window. Text pre-processed by TEXTIMAGER is made accessible to VANNOTATOR via TEXTANNOTATOR and displayed in an annotation box. Only one sentence is displayed at a time; however, users can switch between the sentences.

### 5.3.2 VANNOTATOR'S TEXT2SCENEVR

Up to this point already existing features of VANNOTATOR were described. The following enhancements of VANNOTATOR is about creating spatial structures based on their textual descriptions (see Table 5.1) to arrive at annotation data for training Text2Scene systems, each annotation task begins with an input text as illustrated in Figure 5.3: the text is pre-processed by TEXTIMAGER, loaded via TEXTANNOTATOR and visualized in an annotation box of VANNOTATOR, in which words are separated on the token level. Within the box, tokens can be merged to map multi-word expressions and to relate them to spatial objects created by the user (see Section 5.5). After a connection to TEXTANNOTATOR has been established, a spatial hypertext can be generated from the input text with VANNOTATOR's so-called TEXT2SCENEVR. For this purpose, references to the 3D objects created by the user and their contents must be generated from the text and its segments. In addition, textual relations manifested in the text must be mapped to corresponding spatial relations; in other words: spatial configurations must be created that correspond to these textual relations (see Section 5.1). Finally, the user may generate additional sentences that are entailed by corresponding sentences of the input text from his point of view and process them according to the same procedure. In this way, the relational spectrum described in Section 5.1 is mapped in such a way that the respective input text and its user-dependent text extensions are interwoven with the user-generated object space and its spatial arrangement. This is what we call a spatial hypertext in VR. To generate such hypertexts, the following operations are available for users:

1. **Creating rooms:** the first step for creating a spatial hypertext in VR is to create a room. For this purpose there is a menu item in the annotation box that allows to draw the room's dimensions on the floor as a grid (Figure 5.4). The dimensions

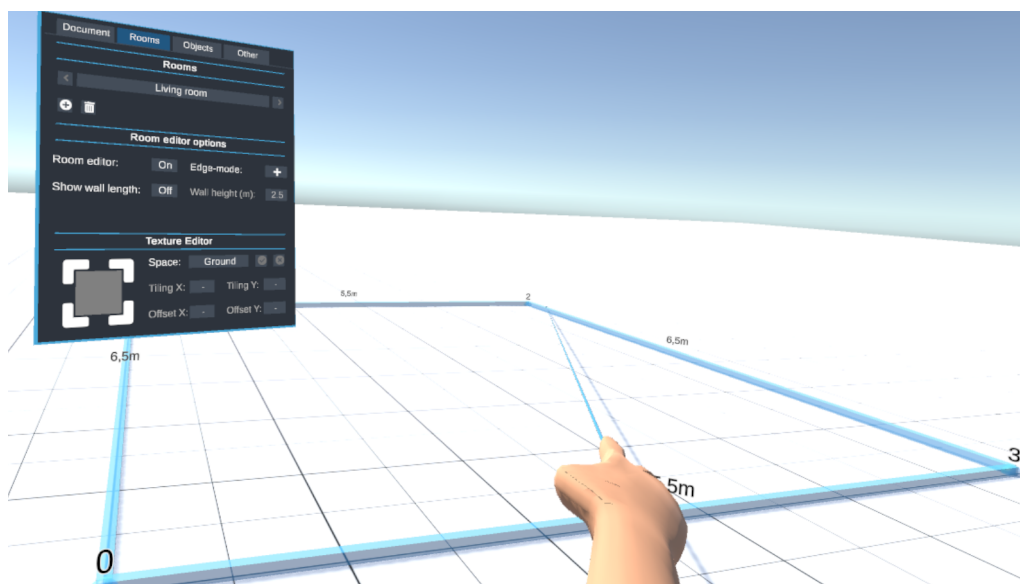


Figure 5.4: VANNOTATOR room creation. Virtual rooms are created by first defining their dimensions.

between the corners of the room, which need not be square, are then visualized. The grid spacing can be freely configured, so that a flexible design of rooms is possible. After the outlines of a new room have been defined, it can be configured in detail (Figure 5.5).

2. **Creating windows, doors and use textures:** the rooms can be equipped with doors and windows (Figure 5.6) and also textured (Figure 5.7). The rooms can be placed and arranged as desired in the virtual environment. It is possible to arrange them next to each other, to connect them and to form room ensembles (Figure 5.7).
3. **Object placement:** further functions include the selection and configuration of room contents and their spatial arrangement. As shown in Figure 5.8, objects as provided by *ShapeNetSem* (Savva et al., 2015) can be placed anywhere in the virtual environment. Besides positioning, objects can be scaled, rotated and clustered into organizational groups.

We distinguish four usage settings (see Figure 5.9):

- **Document:** the *document tab* provides access to the text to be annotated (see Figure 5.3) and allows for switching between its sentences, creating multi-token units, and linking text segments to 3D objects (see Section 5.5).
- **Rooms:** with the *room tab* users can create new rooms (Figure 5.4), edit existing rooms, dimensionalize them and apply suitable textures to walls/floors/ceilings.
- **Objects:** With the *object tab*, 3D objects (furniture, artifacts etc.) can be created to fill the rooms or existing objects can be modified (see Figure 5.9). In addition,



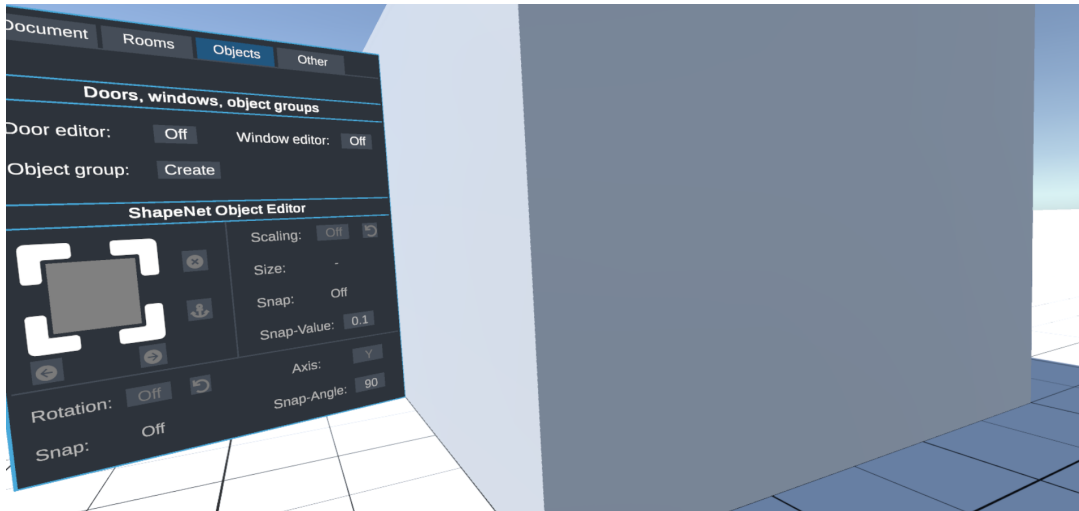


Figure 5.5: VANNOTATOR wall creation. After defining the corners of a room, its walls are created, whose height depends on the user settings.

doors and windows can be created and placed. Furthermore, objects selected from ShapeNetSem can be placed within the rooms, scaled, rotated or adapted to existing surfaces (see Figure 5.8).

- **Other** The *other tab* offers additional settings such as size settings for the grid.

An important step in generating spatial hypertexts is the specification of objects, which are explicitly or implicitly mentioned in the texts, and their placement as contents of the previously generated rooms (see *object tab*). To this end, a wide range of 3D objects and textures are available. Objects are taken from ShapeNetSem (Savva et al., 2015), a sub-project of ShapeNet (Chang et al., 2015a), which includes more than 12 000 3D objects from 270 categories. Each of these objects is annotated with semantic features such as scaling, orientation, estimated weight and volume. Textures are taken from 3dtextures.me<sup>6</sup>. About 700 textures from 45 main categories and 200 subcategories are available. VANNOTATOR thus has a large number of degrees of freedom for creating rooms and their contents<sup>7</sup>.

TEXT2SCENEVR is currently being used by students as part of a practical course at the Goethe University Frankfurt. Until now, twelve paragraphs have been annotated and virtual rooms have been created with TEXT2SCENEVR, which forms a corpus of annotated rooms. In addition, two students separately used Kafka's "The Metamorphosis" as a basis for modeling and annotating the corresponding apartment with an average of 45 objects (without walls, windows and doors).

<sup>6</sup><https://3dtextures.me/>

<sup>7</sup>For the demonstration of the use of VANNOTATOR for the creation of spatial hypertexts see our YouTube videos (<https://tinyurl.com/y87wtveq>).

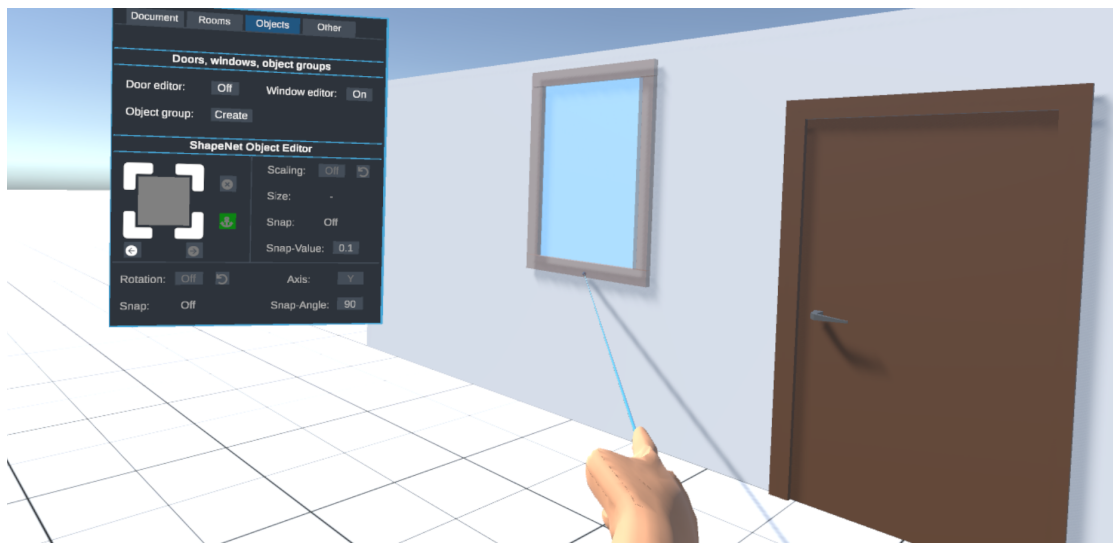


Figure 5.6: VANNOTATOR doors and windows. Freely configurable doors and windows are positioned on the walls of a room. If two rooms are next to each other, it is possible to create a passage between them.



Figure 5.7: VANNOTATOR texturing. Virtual rooms can be provided with textures that reflect information contained in the underlying text. All textures are taken from *3dtextures.me*.

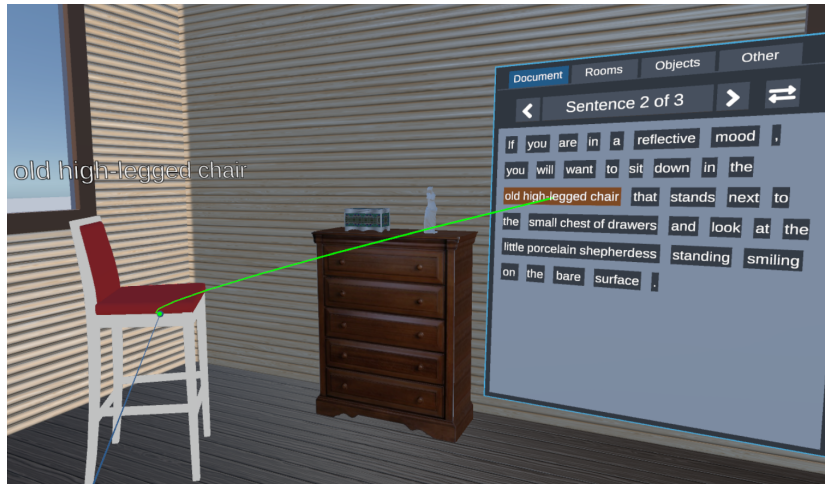


Figure 5.8: VANNOTATOR evaluation annotation example. A look at the annotation from our evaluation. The room described in the input text was created, textured, objects were placed and positioned and the multi-word unit “old high-legged chair” was linked to the chair-object (annotation). The representation of the chair does not fully correspond to the object mentioned in the text; this is because a corresponding object does not exist in ShapeNetSem. The blue line visualizes the pointing gesture (Kühn et al., 2020) used to create the annotation; the green line visualizes the annotation of the chair in the room by a segment of the input text.

## 5.4 TEXT2SCENEVR’s Annotation Model

In order to map spatial objects to linguistic expressions, a data model is required that is flexible, extensible and interoperable by using known formats. For this purpose, we developed a data model, which is largely based on UIMA type system descriptors. The data model is shown in Figure 5.10. Though we implemented this model by means of ShapeNetSem, it can be extended to include related object models as generated, for example, with *VoxML* (Pustejovsky & Krishnaswamy, 2016).

Our data model requires that annotations are selected in the input text and anchored with a so-called *RoomObject*. Our model does not require that an object specified in this way is a concrete object that can be mapped to ShapeNetSem; rather, it can also be an abstract object that is composed of several sub-objects. In any event, the entire scene described by the input text itself is considered a *RoomObject* (e.g. a kitchen scene).

The walls of a room are saved as a sorted list of nodes and assigned to the corresponding room object as attributes, as shown in Figure 5.4. This approach enables not only the hierarchical structuring of the scene representation, but also the linking of text segments with groups of objects. This regards, for example, the modeling of quantifiers (e.g. all glasses) or of expressions that denote groups of objects (e.g. seating group). Finally, room objects that are not mentioned directly in the input text are classified as parts of object groups (partly with reference to decoration purposes) and assigned to the overall

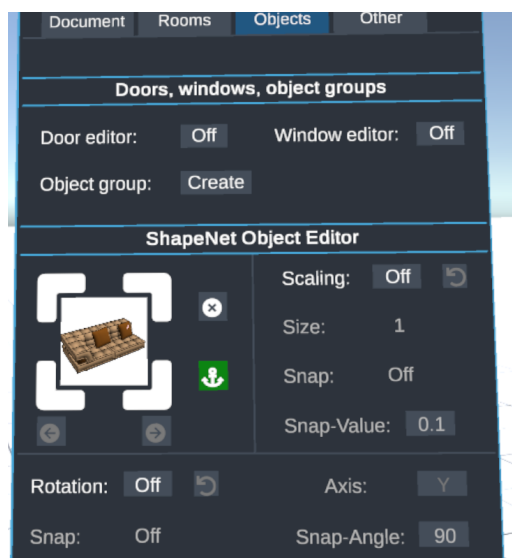


Figure 5.9: VANNOTATOR control window. VANNOTATOR uses a control window for visualizing and annotating spatial structures. It allows for accessing and modifying the input text (*document tab*), the properties of rooms (*room tab*) and objects (*object tab*). With the help of the object tab, doors, windows and other objects are created, scaled or rotated.

text.

A representation of the examples from Table 5.2 is shown in Figure 5.11. Implicit referential meaning is resolved by the selection of objects, implicit part-whole relations by child links and implicit topological relations by the spatial placement of objects.

## 5.5 Evaluation

We conducted a user study to evaluate TEXT2SCENEVR. Starting from a text sample, the task of the test persons was to create a room, select and place objects within this room and to annotate the objects by assigning them to corresponding text segments (see Figure 5.8 for a snapshot of this evaluation task). In addition, a UMUX test (Finstad, 2010) (*Usability Metric for User Experience*) was carried out following the annotation task. The UMUX test included the following questions, which had to be answered on a scale of 1-7, where 1 means that one strongly disagrees and 7 that one strongly agrees:

1. Using VANNOTATOR to annotate spatial structures is a frustrating experience.
2. The functions in VANNOTATOR, to annotate spatial structures, meets my requirements.
3. VANNOTATOR is easy to use.
4. The creation of spatial structures in 3D environments are easy to perform with VANNOTATOR.

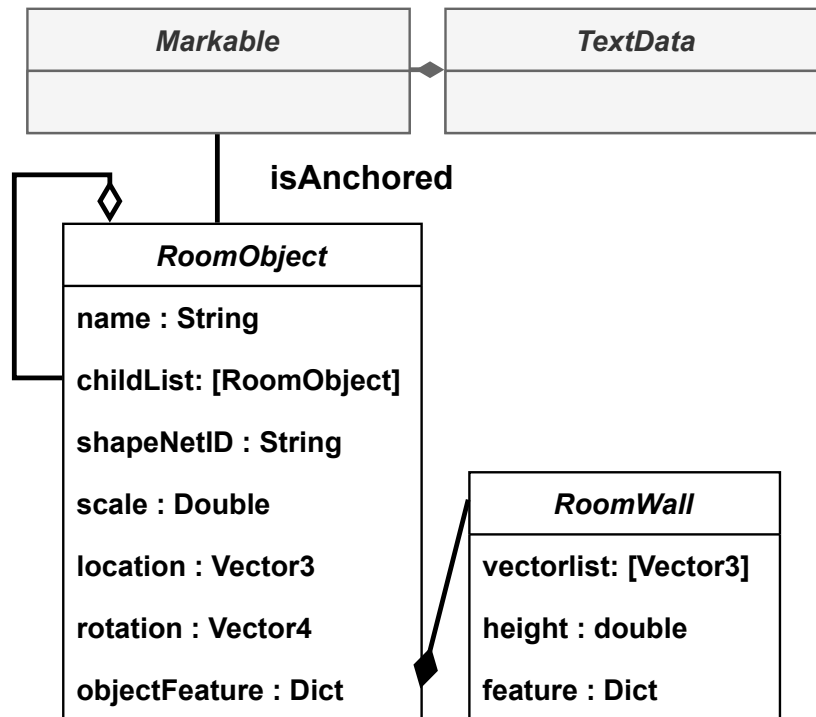


Figure 5.10: TEXT2SCENEVR's data model for the annotation of 3D scenes.

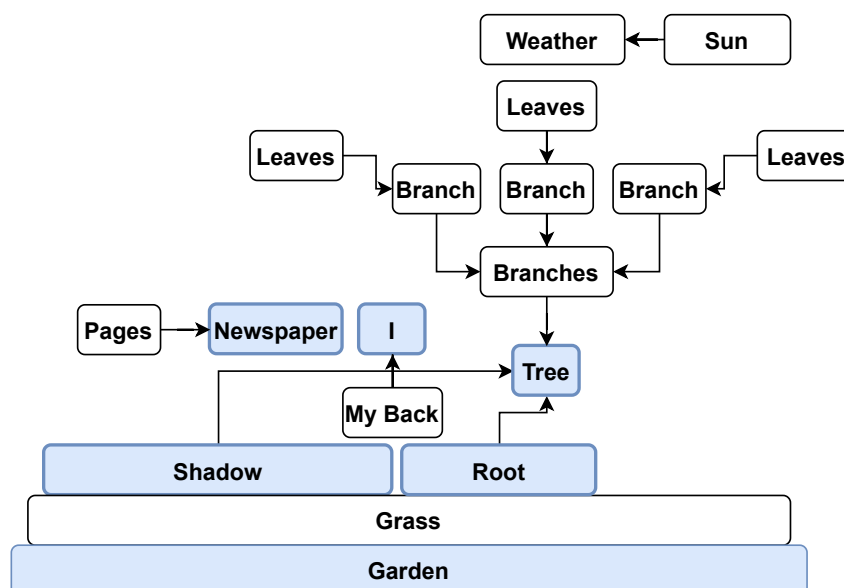


Figure 5.11: Example scene according to the example in Table 5.2. Arrows mark the child/part of relationship and the objects linked to the text are blue. The other objects are derived and the spatial arrangement is based on a 3D representation.

5. The annotations of text to spatial structures in VANNO-TATOR are easy to perform.

Each participant created an individually designed room with partly different objects placed in it. This is due to the fact that each test person imagines the room somehow differently, while the text sample is somehow under-specified with regard to all spatial details. This relates, for example, to the choice of objects, their positioning and design. For this reason, a comparison between the individual annotations is only possible to a limited degree.

Therefore, an analysis was performed to compare how much time the annotators needed per object to recognize them in the text, select them from the database and afterwards placing them in the room. Because individual participants have created the rooms in various degrees of detail, only the concrete object creation and placement process is used for comparison. The results are shown in Figure 5.12, where the participants needed on average 3.2 minutes to create and place an object. Unfortunately two participants could not be evaluated because of problems with motion sickness. With state-of-the-art VR hardware and alternative movement options, this problem could be solved in the future. The participants spent most of their time looking for suitable objects in the database. Since the database did not contain a suitable 3D object for all object descriptions, sometimes long searches were the result and the next suitable object was selected. This shows that the selection process for objects must be optimized. For example, proposals can be generated based on selected tokens in a text and their textual contexts. But as the UMUX test results of Figure 5.13 show, the participants were mostly satisfied with the usability of VANNO-TATOR. The greatest frustration was caused by getting used to the controls and unfamiliarity with VR. But the participants got used to them after annotating 1-2 objects. This also explains the slightly worse results for Question 3. On the other hand, Question 4 and 5, and thus the focus of our tool, were rated best, which speaks for its handling.

## 5.6 Future Work

There is a growing need in computational linguistics to extract spatial and temporal relations from texts (Pustejovsky et al., 2011b). This led to linguistic schemes such as *ISOSpace* (ISO, 2014a; Pustejovsky et al., 2011a), which serve to model spatial relations of the referents of linguistic expressions. We aim to map the spatial model of VANNO-TATOR directly to *ISOSpace*. This will facilitate the recognition and learning of links between expressions and spatial relations. Further, by using *SemAF-ISO* (Semantic Annotation Framework) (Ide & Pustejovsky, 2017, chap. 4.2), we plan to map our model to *ISOTimeML* (ISO, 2012a) in order to annotate temporal structures and to connect them with spatial annotations. Furthermore, our model currently only allows the annotation of entire objects or groups of objects by linking them to text segments. To overcome this bottleneck, we plan to integrate *PartNet* (Mo et al., 2019). This will make it possible to annotate components of objects.

A transformation of *ShapeNetSem* into *VoxML* (Pustejovsky & Krishnaswamy, 2016) is desirable as soon as its development has progressed sufficiently and thus far more objects

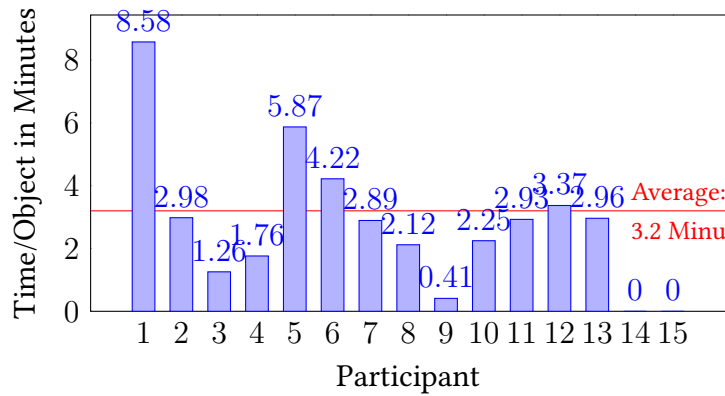


Figure 5.12: The results of the time measurement with 15 participants of which two participants could not be analysed. On average it took each of the 13 participants 3.2 minutes to place each object. Time includes: recognizing objects in the text, searching for suitable objects from the database and finally placing them in the room.

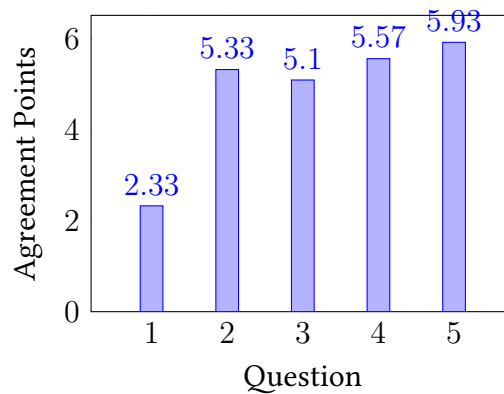


Figure 5.13: The average results of the UMUX test with 15 participants.

are available than up to now. *VoxML* is a modeling language that represents semantic knowledge about 3D objects and links it to representations of actions. A final task will be to speed up the process of selecting 3D objects by recommending candidate objects as soon as a token or multi-word expression is selected in the input text.

## 5.7 Summary

We introduced `TEXT2SCENEVR`, a `VANNOTATOR`-based tool for generating spatial hypertexts that can be used as training data for `Text2Scene` systems. It uses `TEXTANNOTATOR` to link texts with spatial objects. The resulting hypertexts can be used to train `Text2Scene` systems that automatically generate virtual scenes from textual descriptions. In this way we expect to make a significant contribution to solving the bottleneck prob-

lem of Text2Scene systems regarding the lack of training data. Based on the analysis of textual manifestations of spatial content, we distinguished three types of object relations and gave examples of when these are explicitly or implicitly expressed linguistically. In this way we distinguished 12 tasks for Text2Scene systems, which we address with TEXT2SCENEVR. By using existing databases such as ShapeNetSem and 3dtextures, TEXT2SCENEVR already achieves a high degree of freedom in modeling scene descriptions. In an evaluation, we successfully tested the usability of TEXT2SCENEVR. Future work will deal with the integration of ISOSpace, ISOTimeML and PartNet to increase the expressiveness of TEXT2SCENEVR by far.

TEXT2SCENEVR will be published in GitHub (<https://github.com/texttechnologylab/VAnnotatoR>) under the AGPL license.



# 6 Digital learning, teaching, and collaboration in a time of ubiquitous quarantine

Henlein, A., Abrami, G., Kett, A., Spiekermann, C., & Mehler, A. (2021). Digital learning, teaching and collaboration in an era of ubiquitous quarantine. In L. Daniela & A. Visvizin (Eds.), *Remote Learning in Times of Pandemic - Issues, Implications and Best Practice* chapter 3. Thames, Oxfordshire, England, UK: Routledge

## Abstract

Circumstances surrounding the COVID-19 pandemic have serious implications for a multitude of areas of life. Alongside a decrease in the state of health of a considerable number of people, this global crisis also shows that society – both civil and professional, regardless of the sector - is now facing new technological challenges. Furthermore, due to the extensive quarantine measures and the associated closure of educational institutions, a considerable number of deficits have become apparent in the educational sector, which is particularly evident in communication, collaboration, and teaching. These circumstances show above all that in the fields of digital and non-stationary learning, teaching, and collaboration, there is an enormous amount of untapped potential, which - with regard to the existing tools and methods – is far from being explored. This chapter provides an in-depth review of existing practices and tools for digital and virtual teaching, learning, and collaboration, as well as the necessary conditions and strategies to make the best use of technological opportunities in the future. Turning to the future, this chapter focuses on solutions and strategies for three-dimensional, virtual environments and applications. In addition to existing tools, we demonstrate the possibilities in the field of virtual and three-dimensional teaching and learning environments by the example of the so-called VANNOTATOR.

## 6.1 Introduction

The global outbreak of the COVID-19 pandemic has left countries, companies, and people unprepared, and if nothing now changes technologically at the global level, the process that accompanies it could be a pattern for further future pandemics (Afelt et al., 2018; Frutos et al., 2020). The restrictions resulting from such a pandemic, e.g. contact restric-

tions and quarantine (Hellewell et al., 2020), represent a major challenge for companies and people in general, as it is more difficult to coordinate in teams in the home office or maintain social contacts. Quarantine also has significant psychological effects on people as reflected by traumatic stress symptoms or depressions, which can last even longer after quarantine (Brooks et al., 2020). Psychological effects can continue even years after a pandemic (Lam et al., 2009). Suggestions to mitigate the consequences of the quarantine are to “[r]educe the boredom and improve the communication” (Brooks et al., 2020, p. 918). For this reason, the first apps and chatbots have already been developed to support people during this time (Ouerhani et al., 2020).

*How can these and related problems in quarantine periods be countered?* The fast and extensive lockdown made it necessary, especially for companies and educational institutions, to reuse existing applications. In this context, priority was given to the dissemination and distribution of data. Since there are many different existing technical “solutions”, this has resulted in an extremely heterogeneous approach, even within the same organizations. Video conferencing systems such as Zoom, Vidyo, Jitsi, Blue Button, Skype, and Microsoft Teams as well as Dropbox, ownCloud, iCloud, Google Drive, OneDrive, and other data distribution services were used. In terms of data security, flexibility, homogeneity, data protection, and functionalities supporting collaboration, these platforms have different capabilities. *But does their use offer a suitable solution to the problems outlined above?* Evidently, none of these tools is equipped with the most promising technology for overcoming isolation and supporting cooperation under the given circumstances: *virtual Reality (VR)*. VR systems can be a solution for the problems under consideration because they support a much greater variety and range of interactions. Thus, if social isolation, lack of communication, and cooperation due to quarantines or lockdowns are to be prevented, VR systems could be starting points for new technologies that help dampen negative consequences of pandemics. Of course, working and communicating in VR is not per se a substitute for social interaction in the real world, but it can offer additional or alternative possibilities to traditional communication channels:

- According to Gigante (1993, p. 3), VR conveys the illusion of participating in a synthetic environment rather than the observation of such an environment from an external perspective.
- VR enables many possibly remote users sharing the same virtual place (Gigante, 1993, p. 14). In the words of Gigante: “VR can improve the quality of life for workers in hazardous or uncomfortable environments and may eventually impact on the whole of society” (Gigante, 1993, p. 14).

The perception and interaction with these synthetic environments are often supported by appropriate tools, such as head-mounted 3D displays and hand tracking systems. Even though the technology has been around for a relatively long time (with the first known head-mounted display developed by Sutherland (1968)), only in recent years has the technology reached broad segments of the population. This is due to the development of affordable consumer market devices with VR headsets such as Oculus Rift and HTC Vive. Modern smartphones can also be used as VR headsets with the help of simple

cardboard tools such as Google Cardboard (Fabola et al., 2015). Moreover, *Educational Virtual Environments* (EVEs) were developed early on (e.g. Psootka (1995)) and are now actively used in many areas, such as Medicine (Li et al., 2017), Tourism (Guttentag, 2010), or Video Games (Bozgeyikli et al., 2016). Not only VR equipment has become more popular but also tools for developing suitable applications for these technologies, thanks to systems like *Unity3D* and *Unreal Engine* (Martín-Gutiérrez et al., 2017; Indraprastha & Shinozaki, 2009).

The potential of VR to alleviate the problems associated with epidemics has already been recognized by companies like Ford, which has constructed its latest racing car entirely in VR (Foote, 2020). Another example is the *6th International Conference of the Immersive Learning Research Network* (iLRN 2020) which ran completely in VR over VIRBELA and ALTSPEACEVR: conferences of this sort are not only virtual but also take place in VR. The potential of VR has not only been recognized by manufacturers and conference organizers but also museums are now offering virtual tours (e.g. Arts & Culture from Google) through historical places (e.g. *VersaillesVR*); furthermore, concerts (Stirling, 2019) can now be visited in VR while music festivals (like *Wacken World Wide*) are streamed with mixed-reality enhancements.

In general, there are several concepts of VR, which differ in their capabilities and requirements (Riva, 2006; Martín-Gutiérrez et al., 2017; Abrami et al., 2020b) when it comes to supporting such applications:

- **(Fully) immersive VR** is probably the best-known variant in which the real world is replaced by an artificial one with the help of head-mounted displays. The user is able to move in such artificial worlds and interact with them via controllers or by means of hand and body movement tracking (Riva, 2006). Since this is the most widespread variant, we will focus on it in this chapter. So when speaking about VR, we mean immersive VR.
- **Semi-immersive VR** is created by projecting virtual environments onto real environments (Martín-Gutiérrez et al., 2017). The most prominent example is given by *CAVE* applications, in which the virtual environment is projected onto the surrounding wall (Riva, 2006).
- **Non-immersive VR** means the traditional methods of representing a virtual environment, e.g. via a monitor. Sometimes non-interactive head-mounted devices such as *Google Cardboard* or *Samsung Gear VR* are included (Riva, 2006; Abrami et al., 2020b).
- **In Augment Reality (AR)**, the real world is not hidden but enriched with additional information. This is usually done by means of smartphone apps (Butchart, 2011). Examples are mobile games (*Pokémon GO* or *Ingress*), applications to support home furnishings (*IKEA Mobile App*), or virtual MakeUp applications (*L'Oréal Makeup App*).
- **In Mixed Reality (MR)**, the real world and the virtual world are merged. The collective term for all three approaches (AR, MR, VR) is *XR*. The best-known example

of MR devices is *Microsoft HoloLens* and *Magic Leap One*.

Although the latter terms are well defined in technical terms, this is not true for the underlying concepts of perception and interaction; this should be considered in future work. In this chapter, we address the question of what practices and tools are needed for digital and virtual teaching, learning, and collaboration. We will also ask about the conditions and strategies necessary to make the most of technological opportunities in the future. First, we analyze what types of (fully immersive) VR learning environments exist and what requirements they should meet. From this, we derive a basic functionality that these tools should provide in order to best meet these requirements (Section 6.2). For this purpose, we present existing VR systems from different application contexts (Section 6.3) and evaluate them with respect to the previously derived set of requirements (Section 6.4). Finally, the significance of the results is discussed (Section 6.5) and a conclusion is drawn based on the current status in order to better predict future developments (Section 6.6).

## 6.2 VR environments: requirements analysis

The potential of VR for teaching was investigated very early on. In this context, Psotka (1995) investigates immersion as a component of VR. The advantages of VR were already recognized then and what steps would be necessary to ensure its implementation. For example, teaching staff would have to be trained at an early stage and digital libraries would have to open up for VR offers. Obviously, we are still far from implementing these and related steps.

In addition to VR, there has been a lot of research on 3D *Virtual Learning Environments* (VLE) over the last 20 years (e.g. Dalgarno & Lee, 2010; Mikropoulos & Natsis, 2011; Girard et al., 2013). Dalgarno & Lee (2010) present a model for 3D VLEs and discuss their benefits for learning. Their proposal consists of two components: *Representation Fidelity* and *Learning Interaction*. VLE-related fidelity can be achieved through (visually and behaviourally) realistic environments, user avatars, and spatial audio sources that allow the learner to interact with the environment, either through direct actions or by adjusting parameters and scripts. The resulting *Sense of Presence*, *Construction of Identity*, and *Co-presence* with other participants lead to the following learning benefits (Dalgarno & Lee, 2010):

- Development of enhanced spatial knowledge representation.
- Facilitation of (otherwise impractical or impossible) experimental learning tasks.
- Increasing intrinsic motivation and engagement.
- Improved transfer of knowledge and skills.
- More effective collaborative learning.

Not all these points are equally important. This is shown in studies in which spatial audio rarely plays a role and even less often haptic feedback, as the corresponding tools are expensive and inaccessible and therefore unsuitable for regular applications (Martín-Gutiérrez et al., 2017). Thus the visual part and with VR especially the immersion come to the fore. In any case, the question remains whether *fidelity* and *interactivity* are sufficient for a good teaching environment.

Fowler (2015) follows this critical stance and shows that in the works mentioned, the technical perspective predominates while the pedagogical one is almost completely left out. To address this deficit, Fowler (2015) extends the framework of Mikropoulos & Natsis (2011) by means of the pedagogical approach of Mayes & Fowler (1999). They refer to the notion of *immersion* to link both worlds. According to Dalgarno & Lee (2010) and Hedberg & Alexander (1994), immersion results from the interplay between *fidelity* and *interaction*. In contrast to this, another concept of immersion arises from the interplay of different pedagogical concepts, that is *conceptualisation*, *construction* and *dialogue* (named *coursware* by Mayes & Fowler, 1999). *Conceptualization* (or *primary coursware*) refers to the presentation of the concept to be learned to the learner, for example through textbooks, presentations or models. *Construction* (or *secondary coursware*) is the more detailed examination of the topic by the learner, where the learner controls the information and receives feedback. In the *Dialogue Phase* (or *tertiary coursware*) the learner tests his acquired knowledge in dialogue or discursive interaction with other learners, e.g. in tutorials.

Both models (the pedagogical model by Fowler (2015) and the technical model by Mikropoulos & Natsis (2011)) are combined to meet the *Intended Learning Outcomes* (ILO), which are “what learners are expected to know, understand and be able to do by the end of the learning experience (Biggs, 2011)” (Fowler, 2015, p. 417). This is achieved through a so-called *Design for Learning*. For this purpose, the *Learning Requirements* and *Task Affordances* must be recognized and thus the learning specifications be defined (which is anything but trivial) so that as a final product the *Learning Outcomes* are achieved. At the same time, Fowler (2015) fears that the systems will only imitate old patterns instead of developing new, better methods that are now within reach with VR, and that much more research is needed in this direction. At the same time it has to be considered that the requirements analysis depends on the area of application (see e.g. Abrami et al. (2020b) for special requirements in the field of historical education).

Before analyzing and comparing VR systems, this section describes the requirements for a suitable teaching and collaboration environment based on VR and the technical functions required for its implementation. To keep the learning concepts as general as possible and not to focus on specific topics, the aim is to define a basic functionality for all systems. This should allow for evaluating different systems in different application contexts. VR has many advantages, e.g. as a result of “*learning by doing*”, which makes it possible to simulate teaching situations without having to resort to expensive materials or exposing people to risk, as is the case with surgery training (Gurusamy et al., 2009). According to Allcoat & von Mühlénen (2018), many other positive effects can be observed compared to classical learning with textbooks or videos, like an enhanced mood and better test results. *But which points contribute to a better learning experience*

*in general and how can they be implemented in VR?*

### **Representation Fidelity**

Dalgarno & Lee (2010) discuss arguments in support of *representational fidelity*. This concerns, for example, the representational realistic and smooth representation of environments and the coherent reconstruction of object behaviour, both visually (e.g. by designing light effects and textures) and physically. Smoothness is particularly important since lagging behind can lead to motion sickness (Akizuki et al., 2005). On the other hand, realistic representations are application specific: it is certainly important for medical simulations, but one can also imagine abstract applications that simulate what-if situations beyond what is realistic. We leave both criteria out of our evaluation: *representational realism* because of its context dependency and *smoothness*, because it concerns rather hardware limitations. Other criteria concern *user representation*, *spatial audio*, and *feedback*. Avatars (concerning the criterion of user representation) and spatial audio facilitate immersion and interaction with other people. We refer to this scenario by the **avatar requirement** and the **spatial audio requirement**. Kinetic or haptic feedback, on the other hand, is not yet mature as a technology and is therefore not discussed further in this chapter. What is important, however, is how freely users can move in VR. This can range from fixed (but changeable) seating positions through free movements in space to the simulation of flights. We refer to this requirement scenario by means of the **movement requirement**.

### **Learner Interaction**

If different users work together, they should be able to communicate with each other. The most intuitive way is voice chat, as speech is better suited to conveying emotions and moods (Fussell, 2002) than text messages (Hancock et al., 2007). But avatars also enable communication through facial expressions (Osgood, 1966) and hand movements (McNeill, 2016). This scenario is subsumed under the notion of the communication requirement.

Learner interaction means not only the interaction between individual users (regarding the so-called **multi-user requirement**), but also with the virtual environment. One of the most important prerequisites for creating a good teaching atmosphere is that the virtualized environment meets the teaching requirements. There is a lot of work to be done on the topic of what an ideal learning environment should look like, especially since it depends on the application context (Land & Jonassen, 2012; Moreno & Mayer, 2007; Moore et al., 2011; Lage et al., 2000; Fraser & Goh, 2003; Abrami et al., 2019a). For example, different requirements must be met for primary school children than for vocational training. To ensure an ideal teaching and learning environment for all application scenarios, users must be given the opportunity to design their own learning environment (**world-building requirement**). Depending on the scope of the underlying software, it may not be sufficient for more specific learning purposes. It should therefore always be possible to change or extend it. This can be achieved through *application programming*

Requirement	●	○	+	++
Avatar	no	exists	customisable	image reconstruction
Spatial audio	no	some objects / background	avatars	'every' object
Communication	no	writing	speech	hand and face
Multi-user	no (1)			
Worldbuilding	no	extern	intern	intern (dynamic)
Adaptability	no	changeable object attributes	API	Open Source
Permission-orientation	no	predefined classes	individual permissions	group-based permission
User & Group management	no	= 2	restricted group management	group management
Content sharing	no	data sharing	screen sharing	interactivity
Information organisation	no	2D	3D	+ linking
Feedback	no	user feedback (emotes)	metrics-based	real-time
Multi-view	no	–	–	yes
NoVR	no	–	abstract	Full 3D
Data protection	no	–	server	E2E
Platform independence	one system	important systems	most systems	+mobile
Movement	fixed seat position	moving	moving + teleport	flying

Table 6.1: Evaluation table for Section 6.3 based on the requirements of Section 6.2.

Legend: ● (not fulfilled), ○ (partly fulfilled), + (well fulfilled), ++ (completely fulfilled).

*interfaces* (APIs), or better, by having all codes open source (**adaptability** requirement). Users can be distinguished according to their roles (teacher, student, audience, etc.). This should be reflected in their user rights. For example, presenters should be able to mute other users or give them permission to speak while conference participants should not be able to change settings (**permission-orientation** requirement). This includes grouping participants so that they can work in small groups or hold their own sessions for example (**user & group management** requirement) (Table 6.1).

### Conceptualisation

To exchange ideas, introduce or present concepts, approaches, methods, or theories, there should be the possibility of oral communication. However, additional methods of *content sharing* should also be available (**content sharing** requirement). Examples include the sharing of repositories or individual documents, the ability to hold one's own presentations, drawing on whiteboards, collaborative writing, or even collaborative 3D modelling.

### Construction

In the construction phase, it is important that learners have the opportunity to gather and structure relevant knowledge (**information organization** requirement). Furthermore, the learning process should be supported by appropriate evaluation metrics (**feedback** requirement). Since different users can edit the same documents at the same time, an im-

plementation of multi-view functionality is required (**multi-view** requirement). Finally, it is desirable that the construction process is supported by active learning (Settles, 2009) or reinforcement learning (Sutton & Barto, 2018) components so that recurring processes are automated to support the learner. This is a means for *Machine Learning* (ML) to enter educational technologies in a way that supports interpretability of ML results.

### Dialogue

In the dialogue phase, all the requirements mentioned so far come together to enable a lively exchange among learners. This concerns not only direct communication but also the common usability of content sharing and group management functions.

### Other

Last but not least, care must be taken to ensure that no one is excluded who does not have suitable VR equipment. Therefore, the whole system should also run on standard desktops (**NoVR** requirement). Following Abrami et al. (2020b), we add two more requirements: the **data protection** and the **platform independence** requirement. Users must always have full control over their data. This is important for institutional and private users. And given the rapid development of VR systems, different users of different systems should be able to work together seamlessly. To ensure this, standards have been developed (e.g. *OpenVR* and *OpenXR*) that should be met.

## 6.3 Related Work

We now review VR systems that are designed for collaboration, learning, and social exchange. Further, we evaluate them regarding the requirements analysis of the previous section. We consider most of the tools described in Harfouche & Nakhle (2020) and by Lang (2020). We divide these tools into three main application areas (taken from Lang (2020)):

1. Social VR Platforms: Tools for group events and activities.
2. Education & Training: Tools for education, teaching and larger presentations.
3. Team Collaboration & Presentation: Tools for presentations, discussions and productivity.

A list of all tools considered here including their technical specifications is given in Table 6.2. Table 6.3 shows the evaluations according to the requirements analysis of Section 6.2(a more detailed breakdown of our rating system is given in Table 6.1). It should be noted that all these systems are still under development so that they can be expanded with new functions and old ones can be discontinued. We will leave out the first two requirements, representational realism and smoothness (see Section 6.2, *Representation Fidelity*). If a system does not run smoothly on the software side, we will nevertheless mention this.




























VR Platform	Domain	Pricing	Supported OS	Supported VR	Website
AltSpaceVR	Social VR Platform	Free		HTC Vive, Oculus, WMR	altvr.com
Bigscreen	Virtual Desktop	Free		HTC Vive, Oculus, WMR, Valve Index	bigscreenvr.com
Engage	Education & Training	Free, \$	  	HTC Vive/Cosmos, Oculus, WMR, Valve Index	engagevr.io
FrameVR	Collaboration & Education	Free		Browser	framevr.io
Immersed	Collaboration & Virtual Office	Free, \$	  	Oculus Quest, Go	immersedvr.com
Mozilla Hubs	Social VR Platform	Free		Browser	hubs.mozilla.com
Rumii	Collaboration & Virtual Office	Free	  	HTC Vive, Oculus, WMR, Valve Index	www.dogheadsimulations.com/rumii
Softspace	Collaboration & Virtual Office	Free		HTC Vive, Oculus, WMR	www.soft.space
Spatial	Collaboration & Virtual Office	Free, \$	 	Oculus Quest, HoloLens, Magic Leap	spatial.io
vSpatial	Collaboration & Virtual Office	Free	 	HTC Vive, Oculus, WMR, Valve Index	www.vspatial.com
Wonda VR	Education & Training	Free, \$	 	HTC Vive, Oculus, WMR	www.wondavr.com

Table 6.2: Overview VR platforms: the systems were selected from Harfouche & Nakhle (2020) and *roadtovr*. Legend:  (Windows),  (Linux),  (Android),  (Mac),  (Browser), \$ (paid feature).

### 6.3.1 VR Platforms

**ALTSPEACEVR** offers a virtual “meeting space”. The platform hosts daily events on various topics: from stand-up comedy, language or specialist courses, self-help groups to entire conferences. An available SDK allows for customizing virtual environments according to the users’ needs. Compared to other tools, the room size of up to 70 people is noteworthy. Private rooms and worlds can unfortunately only be created and modified with the SDK and not directly in VR. The focus is on social exchange, but no documents can be exchanged or collaboratively edited.

**BIGSCREEN** is a virtual desktop environment for watching movies or playing video games. However, its screen-sharing function also makes it possible to work on joint projects in virtual offices. The room size is rather limited with 12 persons. One can choose from a set of predefined environments, but these cannot be changed afterwards.

**ENGAGE** describes itself as an education and training platform which also supports meetings and events. The platform regularly hosts events and offers additional features for selling live training courses. In terms of pure functionality and presentation, ENGAGE is probably the most advanced tool due to its strong educational and training focus. Avatars can be generated from images. And not only visual media can be placed in the

room, but also e.g. sounds and the existing 3D objects have animations. The legal system is more extensive than in most other systems; lectures can be recorded and exchanged internally, and there are various functions for “controlling” participants, such as “putting everyone in their seats” or “collecting them at specific points”. Due to the extensive functionalities of ENGAGE, other education-related VR systems are also based on it, e.g. *VictoryXR*.

**FRAMEVR** runs completely in the browser and therefore works on all browser-compatible devices (including desktop VR headsets). A special feature is that different frames can be created in parallel, between which users can switch back and forth, but still see each other.

**IMMERSED** provides a virtual workbench to increase the user’s productivity. It also allows for collaborating with other users via telepresence, screen sharing, and whiteboards.

**MOZILLA HUBS** is browser-based like FRAMEVR and the only application considered here that is *Open Source*. Objects, GIFs, videos, and images can be dynamically loaded into the given scene so that one can interact with them. With the additional online tool, *Spoke* rooms and environments can be created to interact and cooperate with users. However, the authorization system is very limited and only differentiates between users and administrators.

**RUMII** focuses more on presentations and thus on education and training. API support has not yet been released, but is planned for the future. However, RUMII offers many possibilities to design rooms, from abstract 3D objects to 3D drawings and concrete objects.

**SOFTSPACE** offers an empty, white space that can be filled with multimodal content that can be grouped into cubes or frames. Moving in SOFTSPACE does not work with a thumbstick as with the tools mentioned so far, but by “grabbing the space” with the hands and pulling or pushing off the corresponding grip point. In addition, the size of the avatar representing the user can be changed so that the virtual space can be spatially structured on the micro and macro levels. Finally, SOFTSPACE also allows for viewing rooms ‘from outside’.

**SPATIAL** is characterized by its AR support (based e.g. on HoloLens), but the range of supported VR headsets is somewhat limited. SPATIAL aims to provide a collaborative XR work environment supported by screen and media sharing and by loading self-constructed 3D objects.

**vSPATIAL** is a virtual work or desktop environment. is a virtual work or desktop environment. The virtual desktops and files can be shared for collaboration, team formation, and presentations.

**WONDAVR** has a greater focus on situation training and live tours. Different scenes can be logically linked together to simulate different situations. Triggers can be used to trigger them or to start other events. The integration of quizzes and scorecards makes it possible to react to feedback. Most of the functionality is implemented via the browser while the live implementation runs via VR.

**Honorable Mentions:** There are many such tools of the just mentioned kind for related purposes, whose discussion would go beyond the scope of this chapter. This refers to tools with an industrial focus that are not freely available or for which no free trial version is available so that they cannot be tested (e.g. *GLUE*, *MEETINVR* or *MEETINGROOM*). Other tools that we have excluded are those with a strong technical focus with respect to medicine (e.g. *OXFORD MEDICAL SIMULATION* or *ACADICUS*) or engineering (e.g. *NVIDIA HOLODECK*). These tools are often fee-based, and we did not have the domain background to evaluate them. The third category, which we excluded, concerns tools with a primarily focus in gaming or Second Life. This includes programmes like *VRCHAT* or *SOMNIUM SPACE*. The latter is particularly interesting because of its approach based on block chains (Swan, 2015). However, this is primarily used for a virtual marketplace to sell virtual products, such as in-game properties.

### 6.3.2 VANNOTATOR

A tool, now considered in detail, is the so-called *VANNOTATOR* (Spiekermann et al., 2018) which is developed in Unity3D. *VANNOTATOR* is designed as a framework for the visualization of and interaction with virtual 3D environments. Within these environments, multimodal content such as texts, images, videos, audios, websites and 3D reconstructions of (e.g. historical) buildings can be visualized and annotated (Mehler et al., 2018). By using VR headsets, *VANNOTATOR* allows for navigating in fully immersive virtual environments so that virtual learning environments are created (Abrami et al., 2020b). This fully immersive capability allows users to move freely in the virtual space. In this way, the **Movement** requirement is met (see Table 6.3).

Besides the visualization and interaction with multimodal objects, their annotation (e.g. regarding sign/object relations) is an essential feature of *VANNOTATOR*. The type and range of the annotation functionalities depend on the type of the object: texts and images can be segmented with *VANNOTATOR*, e.g. links are edited with a virtual browser and video files with a virtual viewer. Beyond that, *VANNOTATOR* includes a variety of methods for the interaction with 3D content:

1. **Highlighting:** relations between objects can be highlighted to provide overviews.
2. **Look ahead:** distant objects connected to an item can be visualized with a preview

function, especially if they are out of sight in virtual space. This preview serves as a preparatory step for virtual “teleportations” (see Figure 6.1).

3. **Teleportation:** To bridge spatial distances between remote objects, portals can be created that display a preview of the target object and, when used (selected or entered), perform a virtual transport to this target (Mehler et al., 2018).
4. **Virtual surfaces:** to use virtual environments flexibly, it is not only important to be able to arrange multimodal objects and interact with them. What is additionally required is the ability to enrich objects with content, annotate them or link them to content. For this purpose VANNOTATOR uses virtual surfaces (Figure 6.4), which can be inserted into virtual rooms analogous to portals or attached to existing objects (e.g. room walls).
5. **Virtual boxes:** The processing of complex information and the division of tasks involved makes the usability of nestable (vertical structuring) and spatially separated information processing sequences (horizontal structuring) necessary for mastering learning and teaching processes. To meet this requirement, VANNOTATOR extends the concept of virtual surfaces by enabling the creation of *virtual boxes*. These boxes are containers of multimodal objects that are created and modified by the user at runtime. Boxes form virtual portals to separated virtual rooms, that is, to rooms within rooms, between which users move to perform room-specific tasks (Figure 6.2, 6.3 and 6.4). In this way, hierarchical structures of tasks can be mapped recursively and corresponding task completion responsibilities can be assigned (*divide-and-conquer approach*). Since portals and teleportations can be used within this structure to additionally link arbitrary subrooms, a so-called generalized tree (a network with a kernel hierarchical structure; cf. Mehler 2009) is created. This creates a very powerful format for the representation of tasks, the objects and tools affected by them, and of the corresponding responsibilities of users.

These functions serve to make virtual environments more flexible for different learning scenarios. VANNOTATOR goes beyond simulations of virtual classrooms by incorporating methods for the spatial organization and structuring of knowledge. Using Unity3D, it enables multiple users to jointly interact with 2D and 3D content.

Furthermore, by integrating external infrastructures, such as TEXTANNOTATOR (Abrami et al., 2020c), it enables the simultaneous as well as collaborative annotation of learning objects and thus fulfills the **multi-user and content sharing** requirement (see Table 6.3). By using TEXTANNOTATOR, which is based on *UIMA* (Ferrucci & Lally, 2004) for data description and storage, it is possible to annotate objects across platforms and technologies (Unity3D, browser, mobile app) (**platform independence** requirement). And since VANNOTATOR is open source, it also meets the **adaptability** requirement. Since complex annotation processes involve several annotators, VANNOTATOR enables the simultaneous and collaborative processing of data: users, represented as virtual avatars (**avatar** requirement), can communicate (via voice chat; Abrami et al. 2020b) and interact with each other (**communication** requirement).

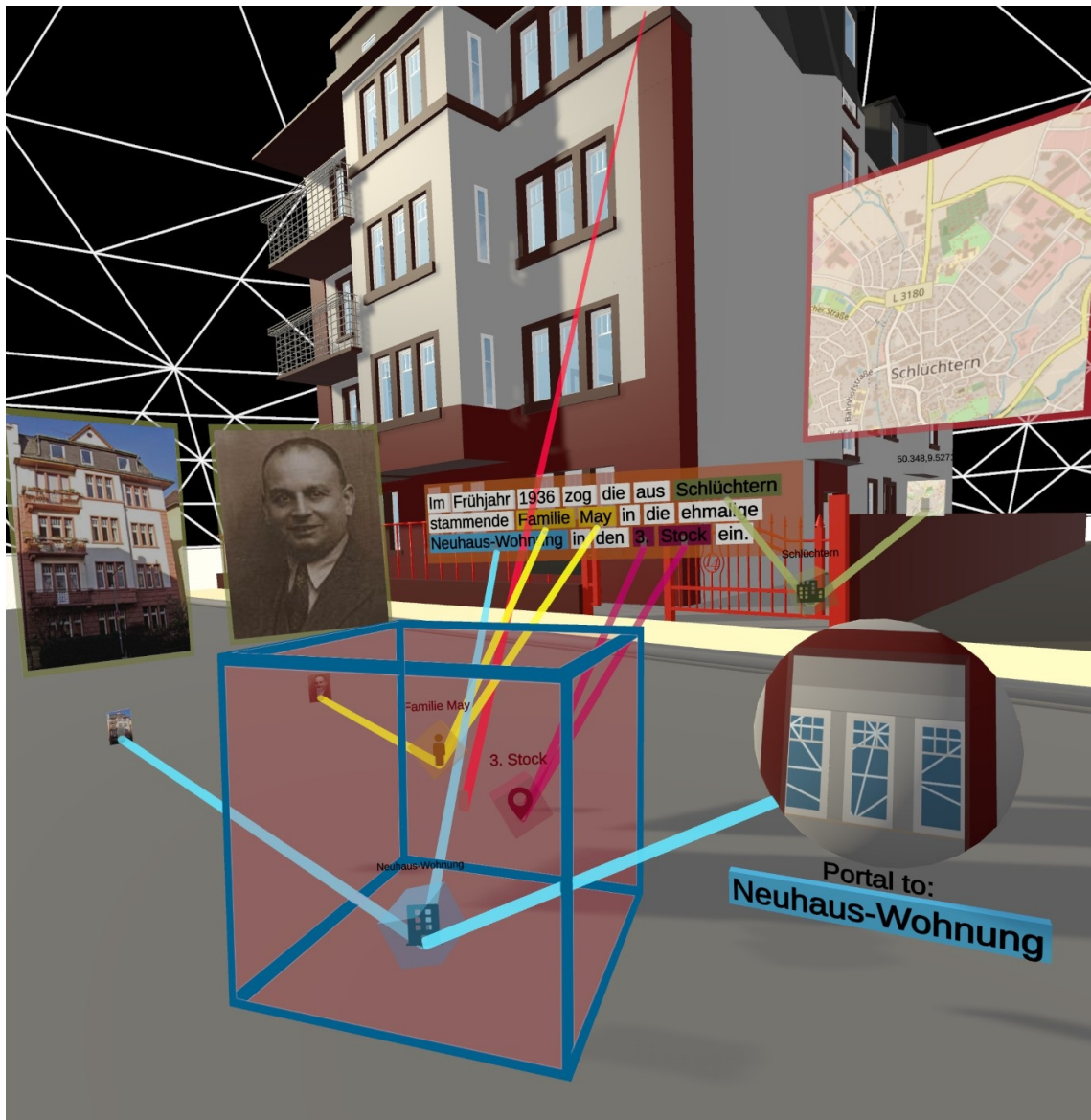


Figure 6.1: Example of a multimodal hypertext created with VANNOTATOR (Abrami et al., 2019c). It shows a reconstructed building in the background and multimodal annotated objects in the foreground as well as a portal (bottom left) which shows a preview of the apartment in the building. By entering or activating the portal, the user reaches the displayed destination (Abrami et al., 2020b).

VANNOTATOR shares all features of TEXTANNOTATOR for annotation, task management, evaluation (**feature feedback** requirement), sharing and editing resources. Objects, object relations and annotations are organized in annotation views that are assigned access rights with respect to (groups of) users using eHUMANITIES DESKTOP (Gleim et al., 2012) so that the requirements of **permission-orientation**, **user & group management** and **multi-view** are met. Furthermore, the requirement **multi-view** is realized

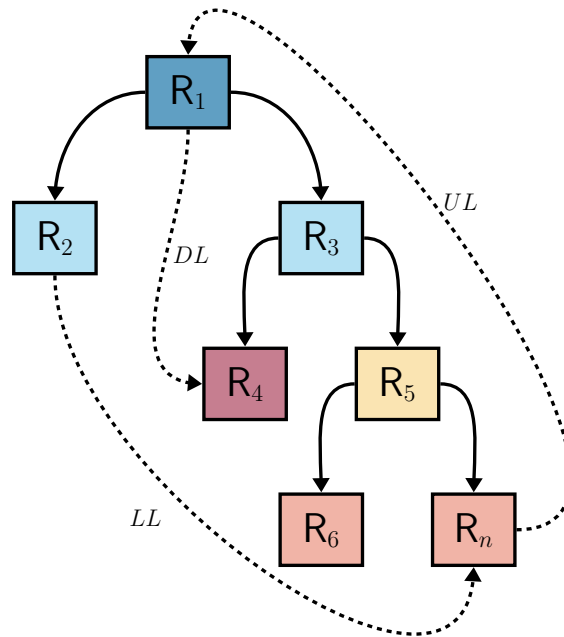


Figure 6.2: An example of a *Networked Hierarchical Room* (NHR) visualized as a *Generalized Tree* (GT) (cf. Mehler, 2009). NHR is a format of room formation as supported by VANNOTATOR, whose data model is based on GTs. Dashed lines depict network-forming relations (i.e. hyperlinks between different rooms connected by portals). We distinguish three types of such relations and corresponding links: *Down Links* (DL), *Up Links* (UL) and *Lateral Links* (LL). Straight lines depict inclusion relations (in the sense that dominating nodes contain dominated ones). Colors of vertices are selected by analogy to the colors of rooms (boxes) in Figure 6.3.

by the combined use of eHUMANITIES DESKTOP and TEXTANNOTATOR (Abrami et al., 2020c). To be usable in different scenarios, the so-called ENVIRONMENTBUILDER of VANNOTATOR is provided, which allows for creating virtual environments, the positioning of objects, their nesting, and linkage (Abrami et al., 2020a) (**information organization and worldbuilding** requirement). In addition, databases of 3D objects can be searched to select objects, e.g. furniture (Abrami et al., 2020b). This results in generating virtual rooms as spatial multimodal hypertexts that support collaboration and interactive learning. An example of such a hypertext generated with the help of VANNOTATOR is shown in Figure 6.1.

Although the requirements of Section 6.3 are currently not fully met by VANNOTATOR, its extensions indicate a potential for further development (Kett et al., 2018; Kett, 2020; Kühn et al., 2020).

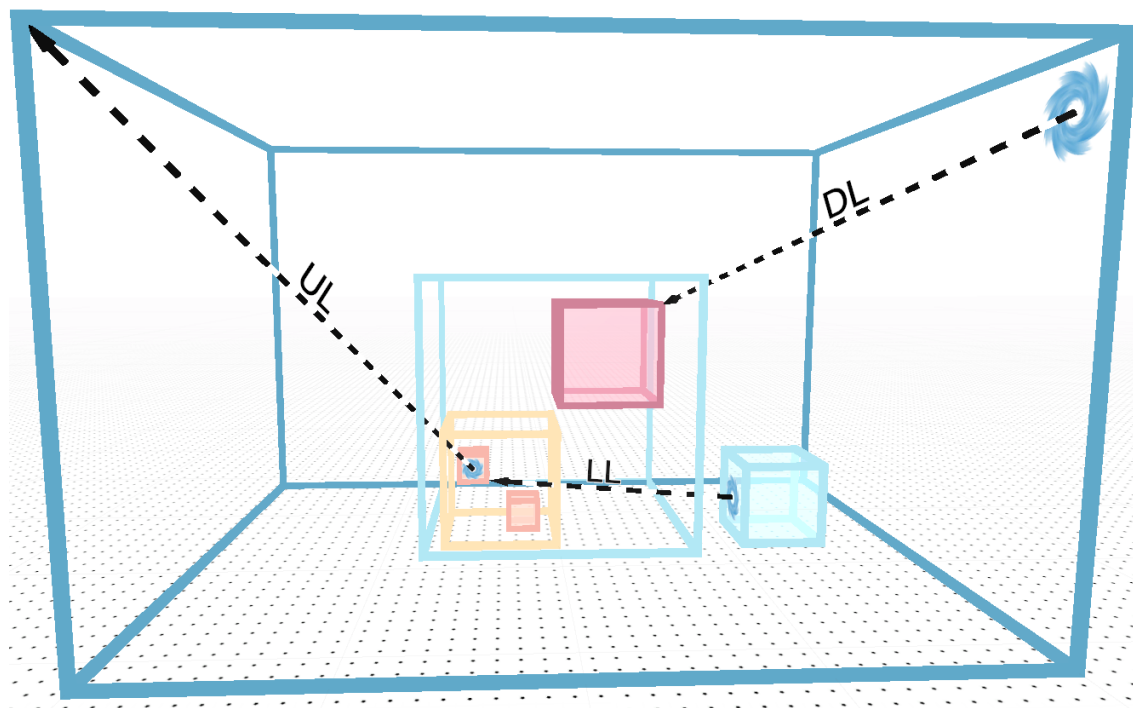


Figure 6.3: Visualization of the Networked Hierarchical Room depicted as a generalized tree in Figure 6.2. Link anchors (possibly starting from portals) are represented by circular symbols.

## 6.4 Analysis

We now take a look at the tools of Section 6.3.1 using the evaluation criteria of Section 6.2. These tools are all under development, some of them only in beta stage, so that their functional range may change in the near future. It is also possible that some of their functions are hidden behind a payment wall without being documented online. Nevertheless, a trend can be deduced in which direction the programs (will) go. The provision of avatars applies uniformly to all tools. Except from *SOFTSPACE* and *VANNOTATOR*, the avatars can be edited; in *ENGAGE* and *SPATIAL*, they can even be generated from images. The same applies to spatial audio sources, which are usually supported by the 3D engines. Furthermore, speech is a standard for communication between users. Many tools support hand movements and gestures, such as mouth movements displayed using avatars. The tools differ with regard to the maximum number of supported users. Not all tools make it clear how many users are supported; the number may depend on the number of supported VR headsets or may be limited by the payment model. In general, however, social platforms support significantly more users than systems that focus on collaborative office. Differences also concern worldbuilding. Many tools allow for placing and moving objects directly in VR. Others outsource this functionality to the browser or to additional APIs. Some tools only offer a pre-selection of environments. Only few tools are adaptable and open source, including *MOZILLA HUBS* and *VANNOTATOR*.



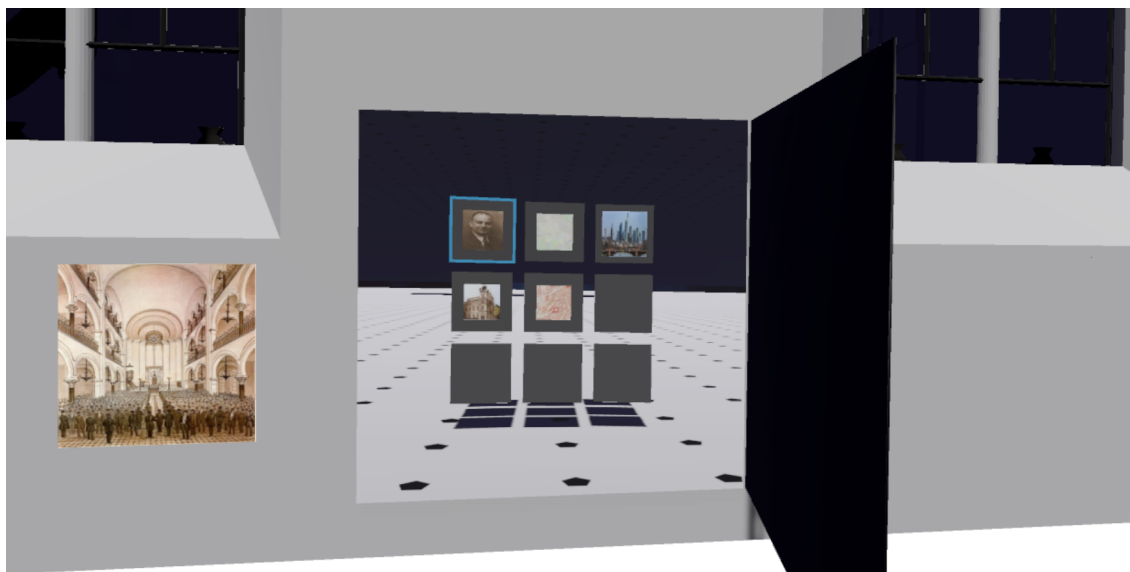


Figure 6.4: VANNOTATOR surface example. The example shows a building on which VANNOTATOR has detected surfaces. These surfaces can be provided with arbitrary multimodal objects to transmit corresponding information. In this example an image is projected into the left area of the picture. The middle area shows a virtual box, which can contain any number of additional objects; virtual boxes represent virtual rooms in virtual environments, which can be arranged recursively (Abrami et al., 2020b) and networked with each other to generate generalized trees of virtual rooms (Mehler, 2009), as depicted in Figure 6.2 and Figure 6.3.

toR. With few exceptions, the ability to manage permissions and groups is limited in most tools. In general, only admins and participants (or guests) are distinguished without allowing fine-tuning group assignments. Most tools support screen sharing features that are visible to anyone using 2D screens in VR. In most cases, there are also whiteboards that can be written with a virtual pen. Sometimes, one can create 3D drawings (e.g. in RUMI). But very few tools support document sharing. And when they do, the choice of supported file formats is very limited. The (spatial) organization of information (beyond what is provided by browsers) works with most tools. SOFTSPACE also allows for grouping information objects into cubes and frames. But only VANNOTATOR supports the linkage of content units. Feedback is underrepresented due to the purposes of most tools, especially in the case of virtual social media platforms. The focus of WONDAVR on situation training includes the implementation of questionnaires and scorecards to give feedback to trainers. VANNOTATOR, on the other hand, uses a multi-view approach to determine the *Inter Annotator Agreement* among annotators; this is done to compare them or assess them against a gold standard. FRAMEVR also implements this multi-view approach, but on a room rather than document basis. Users are located in different spatial environments while remaining visible to each other.

About half of the tools have a NoVR mode with desktop plug-ins or browser sup-



VR Platform	Avatar	Spatial audio	Communication	(max) Multi-user	Worldbuilding	Adaptability	Permission-orientation	Group management	Content sharing	Information organisation	Feedback	Multi-view	NoVR	Data protection	Platform independence	Movement
AltSpaceVR	+	++	++	70	○	+	+	+	++/○	●	○	●	++	?	+	++
Bigscreen	+	++	++	4-12	●	●	○	○	++/○	○	○	●	●	peer-to-peer	+	●
Engage	++	++	++	50	++	●	+	○	++	+	○	●	++	?	++	+
FrameVR	+	++	+	~15-20	(○)	●	○	○	++	+	●	++	++	?	++	+
Immersed	+	+	+	?	●	●	●	●	++/○	●	●	●	●	?	○	●
Mozilla Hubs	+	++	++	50	++	++	+	○	++/○	+	○	●	++	data-encryption	++	++
Rumii	+	+	+	5-40	++	(+)	○	○	++	+	○	●	++	?	++	+
Softspace	○	+	+	12	++	○	○	○	++/+	(++)	●	●	●	?	+	(++)
Spatial	++	++	++	30	++	●	●	●	++	+	○	●	●	?	○	+
vSpatial	+	○	+	?	●	●	●	●	(++)	+	●	●	●	?	○	●
Wonda VR	+	++	+	15	○	●	+	○	(++)	●	+	●	++?	?	+	+
VAnnotatoR	○	+	++	2+	++	++	++	++	○	++	++	++	+	🔒→👤/👥	++	+

Table 6.3: Overview of the functionality of VR platforms: the list of evaluation criteria is shown in Table 6.1. A slash (/) indicates that the right part is excluded (e.g. Content sharing ++/○: Screen sharing but no Data sharing). Brackets indicate that the function is partially but not completely fulfilled.

Legend: 🔒→👤/👥 (user and group based permissions), ● (not fulfilled), ○ (partly fulfilled), + (well fulfilled), ++ (completely fulfilled), ? (not clear).

port while offering full functionality. Sometimes, webcam recordings are used instead of avatars for user representation. Data security is a big challenge for the tools. It is largely unknown how the user data is transmitted, e.g. whether it is encrypted. When information on data security is given, this usually only applies to one aspect, such as user data. However, data security affects many more aspects, such as telemetry, room configurations, uploaded files, voice, and chat data. How this data is handled is rarely known, and if it is, then often only by a simple reference to the EU *General Data Protection Regulation*. In times of big data (Bertino & Ferrari, 2018; Zhang, 2018), this topic is widely discussed and should not be underestimated. Most tools support a variety of VR headsets. Only a few, like IMMERSED, have chosen one or two headsets (mostly Oculus Quest because of its stand-alone feature). AR, and thus HoloLens, is only supported by SPATIAL. Movement usually works as expected. Some office applications that claim to simulate workstations have no movement functions (e.g. BIGSCREEN, vSPATIAL), while others allow free vertical movement (flying) (e.g. MOZILLA HUBS, ENGAGE). Interesting is SOFTSPACE, where one grabs into the space and pushes or pulls away from the corresponding point with arm movements; this works well after a short period of getting used to it.

## 6.5 Discussion

Once a system has been tried out, it is easy to recognize that the premonition of Fowler (2015) has come true and that much effort has been put into “emulate[ing] current practices” but not “to innovate new, pedagogically sound practices” (p. 416). A closer look at the analyses from Section 6.4 shows that important functional foundations for general pedagogical approaches are generally missing. This is particularly noticeable in the *construction phase*, whose basic functions such as information organization, feedback, and multi-views are hardly or not at all available. Instead, there is a focus on spatial fidelity. There are always exceptions, such as VANNOTATOR, whose multimodal hypertext approach covers some pedagogical aspects, or SOFTSPACE with its more artistic approach, but here too, functions are missing or limited, such as an avatar editor. A direct comparison of all tools is not always fair since some of them address different target groups and not all functions are equally relevant to them. Furthermore, our evaluation does not extend to subjective judgements regarding, for example, the intuitiveness of the user interface, the visual appearance of the graphics, control elements, or more specific functionalities (e.g. opening multiple desktop windows). In any event, the tools tend to offer many related functions.

It should also be clear that VR environments are no substitute for real meetings or can currently be. It is therefore certainly a very, very long way to virtual schools. For example, a significant portion of regular school education would suffer when using VR learning platforms: the school break on the playground, where the pupils are more active than during the rest of the day (Dessing et al., 2013). Such activities increase student performance (Loucaides et al., 2009) and would be missed in purely VR-based teaching. So-called Re-Energizer are also used outside school (e.g. in professional seminars and workshops) (Chlup & Collins, 2010). And finally, many other non-verbal communication channels (Mehrabian, 1972) cannot yet be implemented in VR. This ranges from detailed facial expressions to body movements and posture, such as shoulder position.

## 6.6 Conclusion

In this article, new challenges for the use of virtual learning environments are highlighted by the current Covid-19 pandemic and the resulting quarantine regulations. For this purpose, technical evaluation criteria for VLEs were defined, which were derived from the work of Dalgarno & Lee (2010) and Fowler (2015), in order to analyze current VR systems with regard to their functionality, to identify conceptual problems and to specify development perspectives for further VLEs. We have developed these criteria in such a way that they cover all the pedagogical and technical aspects of the preparatory work, but are still fundamental enough to be applied to a large number of VLEs with different focuses and fields of application. As a result, the weaknesses of 12 VLEs were identified and a perspective for future functional enhancements was formulated. However, the results of this analysis show that most systems aim to simulate familiar learning environments such as lecture halls, offices, seminars, or classrooms without

developing new pedagogical approaches that could be implemented in VR. Hardly any attention is paid to the construction phase. Supporting functions would be the placement, linking, and grouping of multimodal content and user-specific edits and views. One reason for this strategy may be that VR technology has only been made available to a broader target group in recent years and therefore concrete applications have only recently been developed. Furthermore, from the point of view of commercial marketing, it is much more interesting to offer a product where customers initially feel more comfortable with familiar visualizations, although alternative concepts might be more appropriate. Independent of VR, there are many other approaches for technology-based learning that should be considered and could be integrated into VR (see e.g. Visvizi et al., 2019; Daniela et al., 2018). For example, analytical methods could be used to evaluate users and thus improve the learning environment and feedback (Visvizi et al., 2020; Sedrakyan et al., 2020).

According to these results, our future work will consider the following tasks: first, a more in-depth analysis based on more specific applications and requirements is required. Furthermore, only general concepts for VLEs have been considered, but situations outside of VLEs must also be analyzed. These can be networking situations at conferences, chance encounters in corridors or streets, or the short exchange during lunch breaks. As far as we know, these types of short and spontaneous learning or exchange environments in connection with VR have not been considered in any way so far. Last but not least, an in-depth analysis of the language used by the different actors through the different communication channels would be very interesting. Depending on the channel (live, audio, video, VR, etc.) this could be very different, which allows conclusions about optimal communication conditions.

## Appendix

### Technical References

System	URL
Academicus	<a href="https://academicus.com/">https://academicus.com/</a>
AltSpaceVR events	<a href="https://account.altvr.com/events/featured/">https://account.altvr.com/events/featured/</a>
EU General Data Protection Regulation	<a href="https://gdpr.eu/">https://gdpr.eu/</a>
Glue	<a href="https://glue.work/">https://glue.work/</a>
Ikea Mobile App	<a href="https://www.ikea.com/ca/en/customer-service/mobile-apps/">https://www.ikea.com/ca/en/customer-service/mobile-apps/</a>
Ingress	<a href="https://www.ingress.com/">https://www.ingress.com/</a>
MeetingRoom	<a href="https://meetingroom.io/">https://meetingroom.io/</a>
MeetinVR	<a href="https://meetinvr.com/">https://meetinvr.com/</a>
Nvidia Holodeck	<a href="https://www.nvidia.com/en-us/design-visualization/technologies/holodeck/">https://www.nvidia.com/en-us/design-visualization/technologies/holodeck/</a>
OpenVR	<a href="https://github.com/ValveSoftware/openvr/">https://github.com/ValveSoftware/openvr/</a>
OpenXR	<a href="https://www.khronos.org/openxr/">https://www.khronos.org/openxr/</a>
Oxford Medical Simulation	<a href="http://oxfordmedicalsimulation.com/">http://oxfordmedicalsimulation.com/</a>
PokemonGo	<a href="https://pokemongolive.com/de/">https://pokemongolive.com/de/</a>
SomniumSpace	<a href="https://somniumspace.com/">https://somniumspace.com/</a>
VictoryXR	<a href="https://www.victoryxr.com/">https://www.victoryxr.com/</a>
Virbela	<a href="https://www.virbela.com/">https://www.virbela.com/</a>
VRChat	<a href="https://www.vrchat.com/">https://www.vrchat.com/</a>

# 7 What do Toothbrushes do in the Kitchen? How Transformers Think our World is Structured

Henlein, A. & Mehler, A. (2022). What do toothbrushes do in the kitchen? how transformers think our world is structured. In *Proceedings of the 2022 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2022)*. accepted

## Abstract

Transformer-based models are now predominant in NLP. They outperform approaches based on static models in many respects. This success has in turn prompted research that reveals a number of biases in the language models generated by transformers. In this paper we utilize this research on biases to investigate to what extent transformer-based language models allow for extracting knowledge about object relations (*X occurs in Y; X consists of Z; action A involves using X*). To this end, we compare contextualized models with their static counterparts. We make this comparison dependent on the application of a number of similarity measures and classifiers. Our results are threefold: Firstly, we show that the models combined with the different similarity measures differ greatly in terms of the amount of knowledge they allow for extracting. Secondly, our results suggest that similarity measures perform much worse than classifier-based approaches. Thirdly, we show that, surprisingly, static models perform almost as well as contextualized models – in some cases even better.

## 7.1 Introduction

Few models have recently influenced NLP as much as transformers (Vaswani et al., 2017). Hardly any new NLP system today is introduced without a transformer-based model such as BERT (Devlin et al., 2019) or GPT (Radford et al., 2019). As a result, static models such as word2vec (Mikolov et al., 2013b) are increasingly being substituted. Nevertheless, transformers are still far from being fully understood. Thus, research studies are being conducted to find out how they work and what properties the language models they generate have.

During training, transformers seem to capture both syntactic and semantic features

(Rogers et al., 2020). For example, dependency trees can be reconstructed from trained attention heads (Clark et al., 2019), syntactic trees can be reconstructed from word encodings (Hewitt & Manning, 2019), and these encodings can be clustered into representations of word senses (Reif et al., 2019). BERT also seems to encode information about entity types and semantic roles (Tenney et al., 2019). For an overview of this research see Rogers et al. (2020).

Since BERT and other transformers are trained on various data crawled from the internet, they are sensitive to biases (Caliskan et al., 2017; May et al., 2019; Bender et al., 2021). In practice, instead of reproducing negative biases, they are expected to allow for the derivation of statements, such as that toothbrushes are spatially associated with bathrooms rather than living rooms. In this line of thinking, approaches such as the popularization of knowledge graphs can be located (Yao et al., 2019; Petroni et al., 2019; Heinzerling & Inui, 2021). Our paper is situated in this context. More specifically, we examine the extent to which knowledge about spatial objects and their relations is implicitly encoded in these models. Since the underlying texts are rather implicit regarding such information, it can be assumed that the object relations derivable from transformers are weakly encoded (cf. Landau & Jackendoff, 1993; Hayward & Tarr, 1995). Reading, for example, the sentence:

*“After getting up, I ate an apple”*

one may assume that the narrator got up from his bed in the bedroom, went to the kitchen, took an apple, washed it in the sink, and finally ate it. The apple is also likely to have been peeled and cut. Equally, however, nothing is said in the sentence about a bedroom or a kitchen. Nevertheless, it is a well known approach to explore the usage regularities of words, currently most efficiently represented by neural networks, as a source for knowledge extraction (see, e.g. Zhang et al., 2017; Bouraoui et al., 2020; Shin et al., 2020; Petroni et al., 2019).

In this work, we use a number of methods to identify biases in contextualized models and ask to what extent they can be used to extract object-based knowledge from these models. To this end, we consider three relations:

1. *Spatial containment of (source) objects in (target) rooms*: e.g. a fridge probably belongs in a kitchen, but not in a living room;
2. *Parts (source) in relation to composite objects (target)*: e.g. a refrigerator compartment is probably a part of a fridge;
3. *Objects (source) in relation to actions (target) that involve them*: e.g. reading involves something being read, e.g., a book.

Regarding these relations, we examine a set of pre-trained contextualized and static word representation models. This is done to answer the question to what extent they allow the extraction of instances of these relations when trained on very large datasets. We focus on rather common terms (*kitchen, to read* etc.) as part of the general language.

It is assumed that (static or contextualized) models implicitly represent such relations, so that it is possible to identify probable targets starting from certain sources. That is, for a word like *fridge* (source), we expect it to be semantically more strongly associated with *kitchen* (target) than with words naming other rooms, since fridges are more likely to be found in kitchens than in other rooms, and that certain word representation models reflect this association. We also assume that this association is asymmetric and exists to a lesser extent from target to source (cf. Tversky & Gati (2004)).

The paper is organized as follows: Related work is reported in Section 7.2. The datasets we use are represented in Section 7.3 and our method in Section 7.4. Our experiments are presented in Section 7.5 and discussed in Section 7.6. Section 7.7 provides a conclusion. All used data, scripts and results are open source on GitHub<sup>1</sup>.

## 7.2 Related Work

Biases in NLP models are not a new problem that appeared with BERT, but affect almost all models trained on language datasets (Caliskan et al., 2017). As such, there are methods for measuring social biases in static models such as word2vec (Mikolov et al., 2013b). One of the best known approaches is WEAT (Caliskan et al., 2017). Here, two groups of concepts are compared with two groups of attributes based on the difference between the sums of their cosine similarities (see Section Section 7.4). This approach already points to a methodological premise that also guides our work: Relations of entities are tentatively determined by similarity analyses of vectorial word representations.

However, a direct comparison of word vectors is not possible with contextualized methods such as BERT, where the vector representation of a word varies with the context of its occurrence (cf. Ethayarajh, 2019). Efforts to transfer the cosine-based approach from static to contextualized models have not been able to recreate similar performances (May et al., 2019). Therefore, new approaches have been developed based on the specifics of contextualized models. For example, BERT is trained using masked language modeling, where the model estimates the probability of masked words in sentences (Devlin et al., 2019). The probability distribution for a masked word in a given context can then be used as information to characterize candidate words (Kurita et al., 2019). Section 7.4.3 describes this approach in more detail. An alternative approach is to examine the interpretability of models (Belinkov & Glass, 2019; Jiang et al., 2020; Petroni et al., 2019, 2020; Bommasani et al., 2020; Hupkes et al., 2020), which goes beyond the scope of this paper. In any event, both approaches share the same basic ideas, e.g., the probability prediction of mask tokens (cf. Kurita et al., 2019; Belinkov & Glass, 2019).

Work has also been done on how BERT represents information about spatial objects. For example, BERT has problems with certain object properties (e.g. *cheap* or *cute*) or implicit visual properties that are rarely expressed (Da & Kasai, 2019). Problems are also encountered with extracting numerical commonsense knowledge, such as the typical number of tires on a car or the feet on a bird (Lin et al., 2020). More than that, the models seem to allow for extracting some object knowledge, but not with respect to

<sup>1</sup><https://github.com/texttechnologylab/SpatialAssociationsInLM>

properties based on their affordance (e.g. objects through which one can see are transparent (Forbes et al., 2019)). Even though these results seem to question the use of BERT and its competitors for knowledge extraction, these models still perform better in downstream tasks than their static competitors (Devlin et al., 2019; Liu et al., 2019a; Brown et al., 2020; Da & Kasai, 2019). Bouraoui et al. (2020) compared these models using different datasets and lexical relations. These include relations similar to those examined here (e.g. a pot is usually found in a kitchen), but beyond the level of detail achieved in our study.

What will become increasingly important is the so-called grounding of language models (Merrill et al., 2021): Here, the models are trained not only on increasingly large text data, but also, for example, on images thus enabling better “understanding” of spatial relations (Sileo, 2021; Li et al., 2020). In this paper, we focus on models without grounding.

## 7.3 Datasets Used for Evaluation

### 7.3.1 Spatial Containment

The *NYU Depth V2 Dataset* (Silberman et al., 2012) consists of video sequences of numerous indoor scenes. It features 464 labeled scenes using a rich category set. We use this dataset as a basis for evaluating the probability of occurrence of objects in rooms (e.g. kitchen, living room, etc.). That is, we estimate the conditional probability  $P(r | o)$  of a room  $r$  (target) given an object  $o$  (source). In this way, we aim to measure the strength of an object’s association with a particular room as reflecting the corresponding spatial containment relation. At the same time, we want to filter out objects such as *window* that are evenly distributed among the rooms studied here. In our experiments, we consider the ten most frequently mentioned objects in NYU to associate with the five most frequently mentioned spaces. This data is shown in the Table 7.4 (appendix).

The advantage of NYU over other scene datasets such as 3D-Front (Fu et al., 2021a) is that it deals with real spaces and not artificially created ones. In addition, NYU’s object category set is relatively fine-grained (we counted 895 different object names) and uses colloquial terms. This is in contrast to, for example, SUNCG (Song et al., 2017) (with categories like “slot machine with chair”, “range hood with cabinet”, “food processor”) and ShapeNetCore (Chang et al., 2015a) with only 55 object categories or COCO (Lin et al., 2014) with 80 object categories. This makes NYU more suitable for our task of evaluating word representation models as resources for knowledge extraction starting from general language.

### 7.3.2 Part-whole Relations

We use a subset of the object descriptions from *Online-Bildwörterbuch*<sup>2</sup>. This resource describes very fine-grained part-whole relations of objects expressed by colloquial names,

<sup>2</sup><http://www.bildwoerterbuch.com/en/home>



in contrast to, e.g., PartNet (Mo et al., 2019) where one finds labels such as *seat single surface* or *arm near vertical bar*. The list of objects from *Online-Bildwörterbuch* used in our study and their subdivisions is shown in Table 7.5. The selected objects were chosen by hand, provided that the chosen examples are general enough and the subdivision is sufficiently fine.

### 7.3.3 Action-object Relations

To study entities as typical objects of certain actions, we derive a dataset from HowToKB (Chu et al., 2017) which is based on WikiHow<sup>3</sup>. In HowToKB, task frames, temporal sequences of subtasks, and attributes for involved objects were extracted from WikiHow articles. Some changes were made to the knowledge database, including a newly crawled version of WikiHow. In addition, the pre-processing tools have been updated and partially extended (see Table 7.6).

#### Related Datasets

For evaluating static models, there are datasets and approaches to measuring lexical relations, such as DiffVec (Vylomova et al., 2016), BATS (Gladkova et al., 2016) or BLiMP (Warstadt et al., 2020). Although these datasets are also used to evaluate BERT (Bouraoui et al., 2020), they represent only an unstructured subset of the data we used and are thus not appropriate for our study.

## 7.4 Approach

We now present the static and contextualized models used in our study. Table 7.7 in the appendix lists these models and their sources. We also specify the measures used to compute word associations as a source of knowledge extraction, and describe how to use classifiers as an alternative to them.

### 7.4.1 Static Models

Probably the best known static model is word2vec (Mikolov et al., 2013b). Its CBOW variant is trained to predict words in the context of their surrounding words. The word representations trained in this way partially encode semantic relations (Mikolov et al., 2013b), making them a suitable candidate for comparison with the corresponding information values of contextualized word representations. In addition to word2vec, we consider GloVe (Pennington et al., 2014), Levy (Levy & Goldberg, 2014), fastText (Mikolov et al., 2018) and a static BERT model (Gupta & Jaggi, 2021). Unlike window-based approaches to static embeddings, Levy embeddings are trained on dependency trees.

---

<sup>3</sup><https://www.wikihow.com/>

## 7.4.2 Contextualized Models

Unlike static models, the vector representations of (sub-)word (units) in contextualized models depend on the context in which they occur so that tokens of the same type may each be represented differently in different contexts. All contextual models we evaluate here are pre-trained and come from the *huggingface models repository*<sup>4</sup>. We evaluate two types of contextualized models:

**Masked Language Models (MLM)** are trained to reconstruct randomly masked words in input sequences. We experiment with BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019b), ELECTRA (Clark et al., 2020) and ALBERT (Lan et al., 2019). The models differ in training, training data, and model size. BERT is trained using masked language modeling and next sentence prediction. RoBERTa omits the second task, but uses much more training data. Two models are trained for ELECTRA: one on masked language modeling (generator) and a second one that recognizes just these replaced tokens (discriminator). Since many of our evaluations need mask tokens, we only use the generator model for the evaluations. Finally, ALBERT is trained to predict the order of pairs of consecutive text segments in addition to masked language modeling.

**Causal Language Models (CLM)** are trained to predict the next word for a given input text. From this class we experiment with GPT-2 (Radford et al., 2019), GPT-Neo (Gao et al., 2021; Black et al., 2021) and GPT-J (Wang & Komatsuzaki, 2021). GPT-Neo and GPT-J are re-implementations of GPT-3 (Brown et al., 2020) where GPT-J was trained on a significantly larger data set named *The Pile* (Gao et al., 2021) (cf. Table 7.7 in the appendix).

## 7.4.3 Similarity Measures

To compute similarities of word associations based on the models studied here, we make use of research on biases in such models. These approaches calculate biases between two groups of concepts with respect to candidate groups of attributes. To this end, associations are evaluated by computing the similarities of vector representations of concepts and attributes. We adopt this approach to investigate our research question. However, as we consider knowledge extraction starting from source words (e.g. *toaster*, *shower*) in relation to target words (e.g. *kitchen*, *bathroom*), we modify it as described below.

### Cosine and Correlation Measures

Based on the human implicit association test (Greenwald et al., 1998), WEAT (Caliskan et al., 2017) is originally designed to compare the association between two sets of concepts ( $X$  and  $Y$ ) and two sets of attributes ( $A$  and  $B$ ). The degree of bias is calculated as

---

<sup>4</sup><https://huggingface.co/models>

follows:

$$s(X, Y, A, B) = \sum_{x \in X} s(x, A, B) - \sum_{y \in Y} s(y, A, B) \quad (7.1)$$

$$s(w, A, B) = \sum_{a \in A} \cos(w, a) - \sum_{b \in B} \cos(w, b) \quad (7.2)$$

Since we are considering source words in relation to target words, we use the following variant:

$$s(X, A) = \frac{1}{|X||A|} \sum_{x \in X} \sum_{a \in A} \cos(x, a) \quad (7.3)$$

For contextualized models, we adopt the approach of May et al. (2019), that is, we generate sentences such as “This is a {x}.” or “A {x} is here”. All templates used in our study are listed in the appendix Table 7.8. However, instead of using the BERT token [CLS] (the default token at the beginning of an input sequence, which often serves as the default representation of the entire sequence), we use the maximum of the vector representations of all subwords of the expression. This approach is suitable for models like RoBERTa that do not use the [CLS] token for training, or the GPT models that do not have this token at all. In addition, we also achieved slightly better results on regular BERT models using this approach. We explain this with the fact that our focus is actually only on single tokens and that the vector representation of the [CLS] token often focuses only on a few dimensions (Zhou et al., 2019). Our approach results in a set of contextualized representations for each source and target word, which are then compared using Equation 7.3. We were able to obtain better results in our experiments with this representation than with those generated via the [CLS] token. For static models, if there is no vector representation for a potential multiword expressions (MWE)<sup>5</sup>, the average of the vectors of their components is used. This representation yielded the largest bias in the work of Azarpanah & Farhadloo (2021). For the static models, we also experimented with *distance correlation* (Székely et al., 2007), *Pearson correlation* (Benesty et al., 2009), *Spearman correlation* (Kokoska & Zwillinger, 2000), *Kendall’s tau* (Kendall, 1938) and *Mahalanobis distance* (Mahalanobis, 1936) – cf. Torregrossa et al. (2020); Azarpanah & Farhadloo (2021) – of the word vectors. Due to space limitations, only the values of the distance correlation and Kendall’s tau are shown (see Table 7.1); the other correlation measures behave similarly. Moreover, the values for these measures tend to perform worse for contextualized models. This observation is consistent with findings of Azarpanah & Farhadloo (2021) where the Mahalanobis distance measure performed worst.

### Increased Log Probability

The cosine measure has shown to be problematic for assessing bias in contextualized models such as BERT (May et al., 2019; Kurita et al., 2019). Kurita et al. (2019) have

<sup>5</sup>Word2Vec contains vectors for MWE’s.

therefore developed a new approach for models trained using masked language modeling. They weight the probability of a target word in a simple sentence template, assuming that an attribute is given or not:

$$\text{score}(\text{target}, \text{attribute}) = \log \frac{P([\text{MASK}] = [\text{target}] \mid [\text{MASK}] \text{ is a } [\text{attribute}])}{P([\text{MASK}_1] = [\text{target}] \mid [\text{MASK}_1] \text{ is a } [\text{MASK}_2])}$$

Experiments show that the values of this measure correlate significantly better with human biases.

Since this measure is based on the context sensitivity of models, it cannot be applied to static models. For contextualized models, we use the probability of the last token (e.g. *curtain* in the case of *shower curtain*) for source-forming MWEs and the first token (e.g. *living* in the case of *living room*) for target-forming MWEs. We also performed experiments with multiple masks, one for each of the components of a MWE. However, this did not produce better results. We adapt this approach for causal language models as follows: Instead of a complete sentence, we use incomplete sentence templates such as “A(n) {object} is usually in the ...” or “In the {room} is usually a/an ...”. The model should then predict the next token. Instead of masking the seed word, a neutral equivalent is used for calculation:

$$\begin{array}{c} A(n) \{object\} \text{ is usually in the ...} \\ \Downarrow \\ \text{This is usually in the ....} \end{array}$$

Instead of performing the analysis in only one direction, we determine the score for both the target and the source given the other.

### Classifier-based Measures

In addition to the previously described measures, we experiment with classifiers. To this end, we train three classifiers on the model representations of the source word to determine the associated target word as a class label (e.g. predict *kitchen*, given the vector of *frying pan*). We generate the set of source word representations  $X$  in the same way as in the case of the cosine measure (see Section 7.4.3) and average them before classification:

$$\text{target} = \text{Classifier} \left( \frac{1}{|X|} \sum_{\vec{x} \in X} \vec{x} \right)$$

The training runs on a leave-one-out cross-validation repeated 100 times. The target vector was then generated from the counted predicted classes (see Figure 7.2b in Appendix). We trained a  $k$ -nearest neighbors classifier with  $k = 5$  (KNN), an SVM with a linear kernel and a feed-forward network (FFN). A small hyperparameter optimization was performed for the FFN, which resulted in the following parameters: Adam Optimizer (Kingma & Ba, 2014) with a learning rate of 0.01 over 100 epochs and one hidden layer of size 100 and ReLU as activation function.

	Word2Vec						GloVe						Levy						fastText						static-BERT						
	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	
room	bathroom	0.37	0.37	0.37	0.39	0.62	0.82	0.38	0.39	0.38	0.57	0.93	0.93	0.39	0.40	0.39	0.14	0.34	0.37	0.53	0.53	0.52	0.73	0.67	0.90	0.54	0.50	0.50	0.25	0.66	0.70
	bedroom	0.20	0.20	0.20	0.13	0.49	0.70	0.31	0.29	0.30	0.28	0.66	0.45	0.21	0.21	0.21	0.10	0.25	0.11	0.30	0.31	0.32	0.26	0.44	0.59	0.28	0.27	0.27	0.35	0.33	0.35
	kitchen	0.35	0.34	0.35	0.20	0.55	0.53	0.37	0.40	0.41	0.52	0.65	0.81	0.17	0.17	0.18	0.09	0.32	0.30	0.38	0.36	0.34	0.41	0.66	0.76	0.40	0.41	0.41	0.45	0.53	0.68
	living room	0.23	0.23	0.24	0.06	0.33	0.35	0.30	0.27	0.28	0.10	0.49	0.51	0.24	0.24	0.23	0.40	0.16	0.25	0.25	0.26	0.24	0.09	0.36	0.60	0.19	0.19	0.19	0.00	0.10	0.46
	office	0.28	0.28	0.26	0.51	0.51	0.55	0.14	0.31	0.35	0.51	0.59	0.64	0.25	0.27	0.28	0.40	0.36	0.25	0.25	0.30	0.33	0.45	0.32	0.63	0.40	0.44	0.45	0.10	0.21	0.32
	CONC	0.23	0.23	0.23	0.22	0.50	0.60	0.27	0.31	0.32	0.37	0.67	0.67	0.16	0.15	0.15	0.15	0.11	0.23	0.30	0.31	0.31	0.40	0.45	0.70	0.31	0.31	0.31	0.18	0.39	0.48
thing	bed	0.41	0.41	0.40	0.64	0.56	0.56	0.38	0.51	0.51	0.56	0.76	0.84	-	-	-	-	-	0.42	0.51	0.52	0.69	0.61	0.67	0.47	0.48	0.46	0.16	0.59	0.54	
	dishwasher	0.19	0.23	0.23	0.06	0.37	0.27	0.33	0.32	0.30	0.03	0.19	0.32	-	-	-	-	-	0.35	0.33	0.33	0.06	0.13	0.23	0.17	0.17	0.17	0.13	0.28	0.31	
	door	0.12	0.11	0.11	0.54	0.75	0.75	0.19	0.23	0.22	0.48	0.81	0.85	-	-	-	-	-	0.25	0.27	0.24	0.36	0.55	0.84	0.24	0.25	0.25	0.36	0.73	0.67	
	mortise lock	0.15	0.16	0.16	0.16	0.50	0.54	0.22	0.26	0.28	0.45	0.74	0.68	-	-	-	-	-	0.11	0.17	0.20	0.68	0.55	0.68	0.20	0.21	0.21	0.14	0.49	0.47	
	refrigerator	0.44	0.46	0.46	0.51	0.47	0.52	0.53	0.57	0.56	0.55	0.55	0.66	-	-	-	-	-	0.54	0.58	0.58	0.28	0.40	0.55	0.50	0.50	0.50	0.56	0.56	0.53	
	toilet	0.28	0.28	0.28	0.01	0.49	0.55	0.33	0.33	0.32	0.31	0.63	0.60	-	-	-	-	-	0.37	0.34	0.33	0.55	0.50	0.72	0.24	0.23	0.23	0.34	0.57	0.58	
CONC	0.25	0.27	0.26	0.28	0.52	0.53	0.30	0.34	0.34	0.39	0.60	0.65	-	-	-	-	-	0.28	0.33	0.33	0.35	0.43	0.61	0.29	0.29	0.29	0.23	0.54	0.52		
verb	eat	0.79	0.79	0.77	0.89	0.89	0.89	0.77	0.86	0.80	0.89	0.89	0.92	0.46	0.45	0.45	0.66	0.87	0.87	0.73	0.80	0.79	0.69	0.89	0.89	0.83	0.84	0.83	0.61	0.89	0.87*
	listen to	0.54	0.64	0.56	0.21	0.38	0.46	0.59	0.70	0.65	0.06	0.53	0.49	0.28	0.22	0.23	0.20	0.38	0.52	0.42	0.53	0.63	0.21	0.42	0.40	0.54	0.56	0.53	0.00	0.39	0.50
	play	0.64	0.69	0.64	0.60	0.66	0.60	0.65	0.80	0.73	0.43	0.45	0.45	0.44	0.45	0.43	0.41	0.50	0.57	0.63	0.69	0.68	0.28	0.66	0.66	0.56	0.56	0.54	0.00	0.49	0.63*
	read	0.43	0.52	0.48	0.38	0.59	0.61	0.51	0.60	0.59	0.48	0.53	0.50	0.31	0.31	0.31	0.49	0.31	0.50	0.54	0.56	0.59	0.42	0.50	0.59	0.48	0.52	0.48	0.00	0.31	0.47
	wash with	0.53	0.54	0.53	0.48	0.61	0.63	0.48	0.57	0.53	0.66	0.66	0.62	0.37	0.34	0.35	0.41	0.66	0.62	0.45	0.51	0.49	0.67	0.66	0.66	0.39	0.40	0.40	0.11	0.55	0.61
	wear	0.76	0.78	0.76	0.88	0.84	0.88	0.80	0.87	0.84	0.88	0.83	0.85	0.56	0.52	0.50	0.82	0.85	0.85	0.77	0.80	0.79	0.59	0.93	0.92	0.78	0.82	0.80	0.72	0.81	0.84
CONC	0.58	0.60	0.57	0.56	0.64	0.67	0.59	0.68	0.65	0.55	0.65	0.65	0.34	0.32	0.31	0.46	0.59	0.65	0.51	0.58	0.58	0.43	0.66	0.68	0.54	0.55	0.54	0.15	0.56	0.65	

Table 7.1: All results of the static models. cos: Cosine Measure, dist: Distance Correlation, kend: Kendall’s Tau, knn: K-Nearest Neighbors, svm: Support Vector Machine, fnn: Feed-Forward Network. The gap in Levy is due to its small training set and the corresponding small vocabulary. (A *gray cell* indicates significant values at  $p < 0.01$ )

#### 7.4.4 Scoring Measures and Classifiers

Given a word representation model, we compute the final score for the measures and classifiers to estimate how well they reconstruct the original probability distribution of the source entities relative to the target entities (see Table 7.4, 7.5, and 7.6). This is computed by the distance correlation (Székely et al., 2007) between the target-source probability distributions and the corresponding association distributions of the respective measure or classifier. The advantage of the distance correlation over the Pearson correlation is that it can also measure nonlinear relations. This was calculated both for all targets individually (correlation of all sources to one target) and then *concatenated* for all targets together; we denote this variant by *CONC*. Therefore, *CONC* does not correspond to the average of the individual distance correlations.

## 7.5 Experiments

Using the apparatus of Section 7.4, we now evaluate the classes of word representation models (static, MLMs and CLMs) in conjunction with the similarity measures and classifiers. The results for the static models are shown in Table 7.1, for the MLMs in Table 7.2 and for the CLMs in Table 7.3. Figure 7.2, 7.3 and 7.4 in Appendix show a visualization of the associations computed by means of cosine, masked-target & masked-source increased log similarity measures and the FFN classifier based on BERT-Large using the different datasets. An experiment was also conducted with word frequencies via *Google Ngram*<sup>6</sup> (see Section 7.7 in the appendix).

<sup>6</sup><https://books.google.com/ngrams>

	BERT-Base						BERT-Large						RoBERTa						ElectraGen						Albert						
	cos	m-s	m-t	knn	svm	fnn	cos	m-s	m-t	knn	svm	fnn	cos	m-s	m-t	knn	svm	fnn	cos	m-s	m-t	knn	svm	fnn	cos	m-s	m-t	knn	svm	fnn	
Room	bathroom	0.57	0.13	0.52	0.72	0.87	0.93	0.65	0.30	0.59	0.78	0.93	0.93	0.21	0.24	0.52	0.55	0.83	0.88	0.58	0.32	0.34	0.49	0.72	0.73	0.24	0.18	0.39	0.52	0.75	0.90
	bedroom	0.48	0.33	0.43	0.53	0.66	0.77	0.44	0.41	0.44	0.44	0.87	0.78	0.23	0.18	0.36	0.17	0.53	0.60	0.32	0.31	0.37	0.37	0.37	0.39	0.23	0.22	0.47	0.31	0.44	0.68
	kitchen	0.56	0.25	0.58	0.62	0.81	0.83	0.43	0.24	0.54	0.72	0.77	0.79	0.39	0.27	0.59	0.16	0.62	0.73	0.34	0.24	0.36	0.48	0.34	0.39	0.25	0.17	0.30	0.05	0.56	0.69
	living room	0.30	0.37	0.26	0.51	0.78	0.79	0.23	0.38	0.24	0.57	0.49	0.66	0.13	0.38	0.28	0.49	0.74	0.65	0.26	0.48	0.33	0.15	0.27	0.26	0.15	0.35	0.54	0.20	0.29	0.40
	office	0.46	0.39	0.28	0.40	0.59	0.61	0.40	0.37	0.31	0.25	0.52	0.71	0.14	0.37	0.38	0.18	0.74	0.63	0.17	0.37	0.23	0.42	0.27	0.36	0.23	0.22	0.42	0.45	0.66	0.81
CONC	0.43	0.26	0.33	0.54	0.73	0.78	0.34	0.26	0.36	0.55	0.72	0.78	0.19	0.22	0.31	0.28	0.69	0.71	0.22	0.30	0.27	0.38	0.40	0.43	0.19	0.15	0.23	0.25	0.53	0.69	
Part	bed	0.55	0.41	0.51	0.51	0.69	0.79	0.49	0.41	0.55	0.56	0.69	0.69	0.20	0.42	0.62	0.49	0.52	0.60	0.37	0.31	0.43	0.44	0.44	0.43	0.26	0.40	0.54	0.36	0.66	0.71
	dishwasher	0.22	0.16	0.22	0.27	0.31	0.28	0.30	0.18	0.31	0.29	0.17	0.18	0.16	0.19	0.19	0.13	0.24	0.17	0.26	0.19	0.21	0.01	0.23	0.36	0.17	0.18	0.25	0.26	0.25	0.23
	door	0.19	0.32	0.20	0.34	0.65	0.63	0.13	0.28	0.39	0.47	0.60	0.62	0.15	0.33	0.27	0.52	0.42	0.51	0.14	0.20	0.17	0.41	0.57	0.60	0.13	0.29	0.21	0.36	0.50	0.54
	mortise lock	0.12	0.14	0.09	0.16	0.26	0.28	0.14	0.23	0.11	0.19	0.26	0.35	0.07	0.29	0.12	0.08	0.18	0.28	0.16	0.18	0.15	0.39	0.59	0.39	0.09	0.27	0.22	0.16	0.31	0.39
	refrigerator	0.44	0.21	0.40	0.48	0.47	0.54	0.38	0.21	0.54	0.42	0.51	0.50	0.18	0.38	0.45	0.49	0.43	0.49	0.37	0.33	0.43	0.46	0.45	0.53	0.44	0.27	0.51	0.66	0.51	0.61
toilet	0.18	0.16	0.29	0.16	0.34	0.45	0.25	0.16	0.26	0.36	0.55	0.50	0.22	0.34	0.41	0.45	0.51	0.51	0.34	0.26	0.42	0.26	0.41	0.46	0.24	0.23	0.25	0.22	0.31	0.46	
CONC	0.20	0.20	0.24	0.33	0.45	0.49	0.22	0.21	0.28	0.39	0.46	0.46	0.07	0.29	0.29	0.39	0.39	0.43	0.21	0.19	0.23	0.32	0.45	0.47	0.08	0.23	0.27	0.35	0.42	0.49	
Verb	eat	0.78	0.65	0.67	0.89	0.84	0.90	0.65	0.58	0.72	0.80	0.89	0.90	0.26	0.66	0.81	0.65	0.87	0.86	0.62	0.64	0.76	0.74	0.79	0.79	0.53	0.61	0.74	0.57	0.84	0.85
	listen to	0.46	0.53	0.51	0.42	0.52	0.57	0.50	0.52	0.50	0.43	0.55	0.52	0.30	0.53	0.55	0.23	0.49	0.54	0.57	0.47	0.59	0.00	0.36	0.39	0.23	0.47	0.51	0.07	0.44	0.57
	play	0.63	0.58	0.69	0.54	0.58	0.61	0.55	0.60	0.73	0.54	0.64	0.66	0.37	0.64	0.65	0.38	0.53	0.59	0.64	0.53	0.69	0.64	0.64	0.65	0.37	0.42	0.52	0.45	0.60	0.62
	read	0.42	0.46	0.65	0.34	0.73	0.65	0.30	0.42	0.66	0.42	0.77	0.59	0.26	0.29	0.59	0.21	0.44	0.44	0.41	0.43	0.63	0.51	0.68	0.69	0.31	0.19	0.57	0.35	0.63	0.60
	wash with	0.49	0.46	0.33	0.49	0.66	0.63	0.42	0.53	0.45	0.61	0.62	0.60	0.30	0.56	0.30	0.23	0.60	0.59	0.42	0.50	0.35	0.52	0.40	0.41	0.33	0.42	0.32	0.18	0.46	0.51
wear	0.66	0.64	0.76	0.88	0.90	0.92	0.62	0.57	0.74	0.79	0.90	0.85	0.24	0.64	0.77	0.36	0.72	0.79	0.53	0.62	0.74	0.90	0.84	0.83	0.30	0.61	0.77	0.61	0.77	0.86	
CONC	0.53	0.53	0.38	0.59	0.69	0.71	0.37	0.50	0.44	0.60	0.73	0.68	0.20	0.55	0.37	0.28	0.59	0.64	0.49	0.51	0.37	0.60	0.61	0.62	0.15	0.40	0.26	0.29	0.62	0.67	

Table 7.2: All results of the contextual masked-language models. cos: Cosine Measure, m-s: Masked-Source Log Score, m-t: Masked-Target Log Score, knn: K-Nearest Neighbors, svm: Support Vector Machine, fnn: Feed-Forward Network. (A gray cell indicates significant values at  $p < 0.01$ )

	GPT2									GPT-Neo									GPT-J											
	cos	p-s	p-s-l	p-t	p-t-l	knn	svm	fnn		cos	p-s	p-s-l	p-t	p-t-l	knn	svm	fnn		cos	p-s	p-s-l	p-t	p-t-l	knn	svm	fnn				
Room	bathroom	0.52	0.20	0.38	0.50	0.37	0.31	0.95	0.95	0.30	0.22	0.51	0.36	0.25	0.53	0.89	0.91	0.50	0.26	0.60	0.66	0.48	0.35	0.89	0.92					
	bedroom	0.26	0.31	0.23	0.47	0.38	0.26	0.61	0.54	0.19	0.33	0.21	0.53	0.48	0.55	0.49	0.57	0.24	0.32	0.23	0.62	0.48	0.33	0.70	0.64					
	kitchen	0.34	0.41	0.45	0.69	0.60	0.53	0.82	0.83	0.31	0.49	0.67	0.70	0.57	0.38	0.51	0.81	0.33	0.36	0.52	0.83	0.70	0.70	0.82	0.83					
	living room	0.21	0.43	0.26	0.41	0.33	0.16	0.27	0.46	0.26	0.57	0.39	0.60	0.44	0.28	0.13	0.48	0.21	0.50	0.47	0.67	0.45	0.46	0.40	0.63					
	office	0.13	0.21	0.43	0.37	0.23	0.31	0.44	0.73	0.33	0.30	0.43	0.46	0.34	0.24	0.53	0.72	0.23	0.36	0.53	0.49	0.39	0.37	0.52	0.69					
CONC	0.26	0.23	0.30	0.46	0.35	0.30	0.61	0.72	0.15	0.34	0.44	0.44	0.35	0.40	0.52	0.71	0.23	0.32	0.42	0.56	0.41	0.42	0.66	0.74						
Part	bed	0.36	0.30	0.51	0.67	0.45	0.55	0.59	0.70	0.32	0.38	0.46	0.77	0.70	0.66	0.78	0.88	0.46	0.38	0.36	0.81	0.68	0.71	0.83	0.84					
	dishwasher	0.11	0.34	0.22	0.25	0.23	0.18	0.21	0.28	0.06	0.23	0.30	0.30	0.29	0.15	0.15	0.24	0.09	0.26	0.30	0.44	0.38	0.12	0.15	0.32					
	door	0.23	0.07	0.14	0.20	0.28	0.20	0.65	0.66	0.27	0.10	0.17	0.35	0.42	0.20	0.44	0.66	0.15	0.13	0.12	0.37	0.41	0.25	0.67	0.77					
	mortise lock	0.07	0.34	0.43	0.17	0.18	0.27	0.63	0.65	0.11	0.49	0.42	0.30	0.22	0.27	0.49	0.61	0.15	0.43	0.43	0.47	0.31	0.04	0.63	0.66					
	refrigerator	0.42	0.41	0.24	0.53	0.52	0.47	0.39	0.51	0.29	0.33	0.33	0.47	0.55	0.44	0.47	0.57	0.46	0.51	0.41	0.57	0.63	0.55	0.53	0.63					
toilet	0.29	0.36	0.44	0.20	0.17	0.16	0.49	0.54	0.27	0.42	0.50	0.25	0.26	0.23	0.50	0.58	0.32	0.45	0.48	0.32	0.37	0.26	0.53	0.62						
CONC	0.14	0.24	0.28	0.28	0.25	0.29	0.47	0.54	0.12	0.30	0.34	0.37	0.36	0.32	0.46	0.57	0.16	0.34	0.32	0.42	0.43	0.33	0.54	0.62						
Verb	eat	0.49	0.82	0.65	-	-	0.82	0.87	0.87	0.45	0.86	0.66	-	-	0.76	0.87	0.88	0.48	0.68	0.74	-	-	0.63	0.89	0.89					
	listen to	0.22	0.57	0.51	-	-	0.29	0.50	0.58	0.22	0.47	0.47	-	-	0.20	0.42	0.55	0.21	0.60	0.56	-	-	0.29	0.52	0.60					
	play	0.40	0.64	0.62	-	-	0.62	0.61	0.59	0.32	0.66	0.61	-	-	0.20	0.55	0.62	0.34	0.66	0.70	-	-	0.37	0.67	0.67					
	read	0.32	0.63	0.30	-	-	0.32	0.45	0.45	0.32	0.61	0.34	-	-	0.29	0.52	0.49	0.40	0.63	0.41	-	-	0.20	0.59	0.49					
	Wash with	0.39	0.77	0.51	-	-	0.57	0.61	0.63	0.28	0.66	0.52	-	-	0.39	0.41	0.60	0.23	0.69	0.52	-	-	0.66	0.61	0.64					
wear	0.44	0.38	0.72	-	-	0.76	0.84	0.87	0.16	0.39	0.66	-	-	0.68	0.79	0.85	0.21	0.38	0.62	-	-	0.87	0.84	0.87						
CONC	0.31	0.52	0.53	-	-	0.51	0.64	0.66	0.19	0.49	0.52	-	-	0.40	0.59	0.66	0.23	0.50	0.56	-	-	0.48	0.68	0.69						

Table 7.3: All results of the contextual causal-language models. p-s: Predict Source Score, p-s-l: Predict Source Log Score, p-t: Predict Target Score, p-t-l: Predict Target Log Score. The gap for the verb p-t score is due to the lack of an easily applicable sentence templates in this direction. (A gray cell indicates significant values at  $p < 0.01$ )

### 7.5.1 Model-related Observations

The basic expectation that the cosine measure would generally perform the worst and the FFN classifier the best was met (see Table 7.1–7.3). Interestingly, cosine is also outperformed by distance correlation in almost all cases.

Among the static models, GloVe and fastText performed best in most cases, especially on the room and part dataset (Table 7.1). Although Levy performs by far the worst in the room dataset, it keeps up with all classification results in the verb dataset. One reason for this could be the dependency-based learning strategy, which seems to work very well for verb associations, even though it was trained on a much smaller data set.

Among the masked-language models, BERT-Base surprisingly performed the best (Table 7.2). BERT-Large achieved the better Increased Log Probabilities, but the FFN classifier still worked better with the vector representations of the Base variant. This suggests that although associations are represented in a more fine-grained manner in BERT-Large, they are more difficult to retrieve due to the size of this model.

Among the masked-language models, GPT-J (which was trained with by far the largest training data) performs best (Table 7.3). Context-based models generally seem to determine the target given the source ( $P(\textit{target} \mid \textit{source})$ ) more easily than the reverse ( $P(\textit{source} \mid \textit{target})$ ). With verbs, on the other hand, the reverse effect occurs. The GPT models show that the results for sources are better when weighted, while for targets the results are better without weighting.

In general, the SVM performed surprisingly well, even though only a linear kernel was used. But also the KNN method mostly performed better than the similarity measures. However, FFN performs best in all cases, outperforming cosine (worst case) by increases in the interval [6%, 52%] and outperforming the KNN approach (worst classifier) in each case by increases in the interval [2%, 43%].

### 7.5.2 Dataset-related Observations

In terms of rooms, *bathroom* scores the best, while *living room* or *office* usually score the worst. This may be because many bathroom objects are related to specific bathroom activities (e.g., toothbrush, bathtub), while objects that used to be located in the living room are increasingly found in other rooms (e.g., television in the bedroom). This would also explain why the results for *kitchen* are also better.

On the part dataset, the static models actually performed significantly better than the contextualized models. This relates especially to GloVe and fastText which outperformed almost all contextualized models. Thus, static models are in some cases a good alternative to their contextualized counterparts. However, the more technical the objects become (here *mortise lock* and *dishwasher*), the worse the results become.

On the verb dataset, the contextualized models perform minimally better. As mentioned earlier, these models can associate objects with verbs more easily than the other way around. Here, the largest difference in performance is observed in the case of Levy, where the results are almost equal to those of the other models, probably due to the learning strategy based on dependency trees.

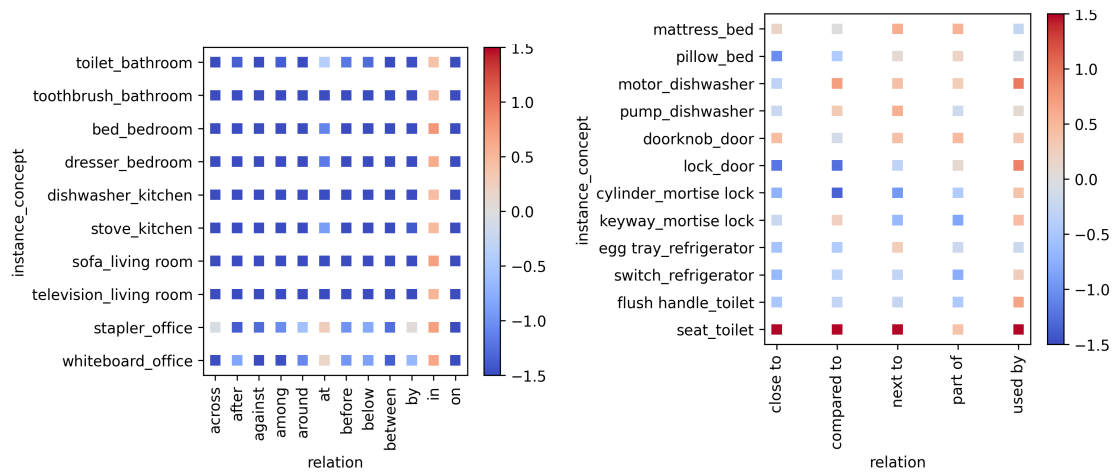


Figure 7.1: Small relation evaluation of BERT-large after the method of Kurita et al. (2019).

In summary, knowledge extraction using language models, whether static or contextualized, is more effective using classifiers than using similarity measures commonly used in the field of bias research: there is potential for this type of knowledge extraction, but at the price of training classifiers – if one uses similarity measures instead, this knowledge is mostly out of reach.

### 7.5.3 Relation Observation

All previous evaluations only examined associations between instances and concepts, but not whether the models represent their true relations. To fill this gap, we repeated the experimental setup of Kurita et al. (2019) for the room and part dataset on BERT-large, but this time masked the relation. The results are shown in Figure 7.1. Our selection of relations does not claim to be exhaustive, but serves as an illustration. It shows that while BERT-large is still very good at assigning objects *in* rooms, the dominant relation predicted for parts is *used by*. This suggests that BERT has problems correctly assigning object parts, an observation that could explain its poorer results while being consistent with findings of (Lin et al., 2020) (e.g., regarding counting parts).

## 7.6 Discussion

As good as the results obtained using classifiers are, they must be viewed with caution. One can attribute their success to the fine-tuning of numerous parameters (and ultimately to overfitting); however, one can also attribute this success to nonlinear structuring of the information encoded in language models. In other words, these models appear to encode object knowledge, but require a sophisticated apparatus to retrieve it. Thus, they should not be considered as an alternative to unsupervised approaches.



Another issue is that our experiments do not yet allow for a comparison of model *architectures*, as the models studied differ significantly in terms of the size of their parameter spaces and training data. Our experiments do suggest that certain smaller models come close to or even outperform the results of larger models. However, a comparison of model architectures would require controlling for these parameters. Nevertheless, the results we have obtained are, in part, promising enough to encourage such research.

Finally, our experiments show that static models can perform better than contextualized models to some extent. This finding is conditioned by our experiments and their context of application. These observations that *older* models perform better on certain tasks are consistent with other work (e.g. LSTMs on small datasets for intent classification (Ezen-Can, 2020) or definiteness prediction (Kabbara & Cheung, 2021)). At this point, a much broader analysis is needed (considering more areas and object relations), which also exploits contextual knowledge represented in contextualized models more than has been done here and in related work. Nevertheless, it is generally difficult to obtain data for such a broader analysis, and our experiments are already broader in scope and consider finer relationships than similar approaches.

## 7.7 Conclusion

We evaluated static and contextualized models as potential resources for object-related knowledge extraction. To this end, we examined three datasets (to identify typical artifacts in rooms, objects of actions, or parts of objects). We also experimented with different similarity measures and classifiers to extract the information contained in the language models. In doing so, we have shown that the models in combination with the measures differ greatly in terms of the amount of knowledge they allow for extracting. There is a weak trend that BERT-Base is the best performer among contextualized models, and GloVe and fastText among static models. Secondly, our results suggest that approaches based on classifiers perform significantly better than similarity measures. Thirdly, we have shown that static models perform almost as well as contextualized models – in some cases even better. This result shows that research on these models needs to be advanced. In future work we will also investigate how grounded language models perform on such datasets. However, as noted above, this requires a significant expansion of bias research, such as that conducted here to enable knowledge extraction.

## Appendix

A tabular breakdown of the datasets used can be seen in Table 7.4, 7.5 and 7.6. The exact models used are listed in Table 7.7. The heatmap visualizations for the other two datasets are in Figure 7.3 and 7.4.

## Word Frequency

We also conducted an experiment to correlate the scores with their frequency. For this purpose, the corresponding objects of each target were selected. And then the distance correlation between the scores and the corresponding word frequency was calculated based on the average of the last 10 years of *Google Ngrams*. The results are shown in Table 7.9. The correlations are not particularly significant (mostly  $p \geq 0.1$ ), but it is noticeable that especially the cosine score depends strongly on the word frequency. The classifiers are generally less sensitive.

bathroom		bedroom		kitchen		living room		office	
object	score	object	score	object	score	object	score	object	score
toilet	1.00	dresser	1.00	drying rack	1.00	coffee table	0.94	whiteboard	1.00
bathhtub	1.00	night stand	1.00	kitchen island	1.00	ottoman	0.93	room divider	0.94
toothbrush holder	1.00	headboard	1.00	pot	1.00	fireplace	0.87	stapler	0.92
toothpaste	1.00	bed	0.97	frying pan	1.00	dvd player	0.69	cork board	0.92
shower curtain	1.00	alarm clock	0.97	spice rack	1.00	sofa	0.68	file	0.88
toothbrush	0.97	laundry basket	0.86	cutting board	1.00	decorative plate	0.61	keyboard	0.85
towel rod	0.96	hat	0.74	blender	1.00	tv stand	0.57	mouse	0.84
toilet paper	0.96	doll	0.70	knife	1.00	blanket	0.55	pen	0.83
squeeze tube	0.95	stuffed animal	0.60	stove	0.98	television	0.53	computer	0.82
faucet handle	0.82	pillow	0.56	dishwasher	0.97	remote control	0.50	column	0.81

Table 7.4: Statistics generated from ScanNet using NYU categories: *score* is the conditional probability  $P(\text{room} \mid \text{object})$  of the room given the object based on the frequencies observable in NYU.

bed		dishwasher		door		mortise lock		refrigerator		toilet	
object	score	object	score	object	score	object	score	object	score	object	score
pillow	1.00	drain hose	1.00	lock	1.00	ring	1.00	switch	1.00	valve seat shaft	1.00
bolster	1.00	overflow protection switch	1.00	cornice	1.00	keyway	1.00	refrigerator compartment	1.00	tank lid	1.00
mattress cover	1.00	tub	1.00	hanging stile	1.00	cotter pin	1.00	egg tray	1.00	conical washer	1.00
leg	1.00	pump	1.00	entablature	1.00	spring	1.00	shelf channel	1.00	lift chain	1.00
box spring	1.00	gasket	1.00	top rail	1.00	rotor	1.00	magnetic gasket	1.00	seat	1.00
headboard	1.00	water hose	1.00	middle panel	1.00	cylinder case	1.00	storage door	1.00	shutoff valve	1.00
mattress	1.00	heating element	1.00	bottom rail	1.00	key	1.00	freezer door	1.00	trip lever	1.00
pillow protector	1.00	rack	1.00	panel	1.00	faceplate	1.00	guard rail	1.00	ball-cock supply valve	1.00
elastic	1.00	cutlery basket	1.00	jamb	1.00	dead bolt	1.00	crisper	1.00	toilet bowl	1.00
footboard	1.00	wash tower	1.00	doorknob	1.00	cylinder	1.00	glass cover	1.00	flush handle	1.00
		motor	1.00	threshold	1.00	stator	1.00	butter compartment	1.00	wax seal	1.00
		detergent dispenser slide	1.00	weatherboard	1.00	strike plate	1.00	thermostat control	1.00	tank ball	1.00
		leveling foot	1.00	lock rail	1.00			freezer compartment	1.00	float ball	1.00
		insulating material	1.00	shutting stile	1.00			ice cube tray	1.00	filler tube	1.00
		spray arm	1.00	header	1.00			meat keeper	1.00	waste pipe	1.00
		rinse-aid dispenser	1.00					door stop	1.00	seat cover	1.00
								shelf	1.00	cold-water supply line	1.00
								dairy compartment	1.00	overflow tube	1.00
								door shelf	1.00	trap	1.00
										refill tube	1.00

Table 7.5: A subset of part-whole relations extracted from *Online-Bildwörterbuch*. All parts have a value of 1.00 in our data set, because they only occur with this object.

eat		listen to		play		read		wash with		wear	
object	score	object	score	object	score	object	score	object	score	object	score
food	0.13	music	0.22	game	0.27	book	0.08	soap	0.29	clothing	0.07
diet	0.08	song	0.03	music	0.06	label	0.06	water	0.29	glove	0.06
meal	0.07	body	0.03	note	0.04	instruction	0.05	vinegar	0.04	shoe	0.05
breakfast	0.04	side	0.02	sport	0.03	review	0.04	solution	0.03	clothes	0.05
balanced diet	0.03	partner	0.02	chord	0.02	body language	0.02	detergent	0.03	shirt	0.02
fruit	0.03	child	0.02	song	0.02	rule	0.02	baking soda	0.03	makeup	0.02
vegetable	0.03	perspective	0.02	video game	0.02	example	0.02	cream	0.02	gear	0.02
plenty	0.03	response	0.02	card	0.02	complaint	0.01	shampoo	0.02	boot	0.02
protein	0.03	parent	0.02	role	0.02	law	0.01	towel	0.02	dress	0.02
snack	0.02	people	0.02	video	0.02	story	0.01	cold water	0.02	sock	0.02

Table 7.6: A subset of verb-object relations extracted from an updated version of HowToKB.

Model	Specification	Dimension	Parameters	Dataset Size (T ; S)	URL
word2vec	GoogleNews-vectors-negative300	300	-	100B ; -	<a href="https://code.google.com/archive/p/word2vec/">https://code.google.com/archive/p/word2vec/</a>
Glove	Common Crawl - glove.840B.300d	300	-	840B ; -	<a href="https://nlp.stanford.edu/projects/glove/">https://nlp.stanford.edu/projects/glove/</a>
Levy	Dependency-Based Words	300	-	English Wiki (~ 2B tokens)	<a href="https://levyomer.wordpress.com/2014/04/25/dependency-based-word-embeddings/">https://levyomer.wordpress.com/2014/04/25/dependency-based-word-embeddings/</a>
fastText	crawl-300d-2M-subword	300	-	600B ; -	<a href="https://fasttext.cc/docs/en/english-vectors.html">https://fasttext.cc/docs/en/english-vectors.html</a>
static-BERT	bert_12layer_sent	768	-	+1.28B ; -	<a href="https://zenodo.org/record/5055755">https://zenodo.org/record/5055755</a>
BERT-Base	bert-base-uncased	768	~ 110M	3.3B ; 16GB	<a href="https://huggingface.co/bert-base-uncased">https://huggingface.co/bert-base-uncased</a>
BERT-Large	bert-large-uncased	1024	~ 336M	3.3B ; 16GB	<a href="https://huggingface.co/bert-large-uncased">https://huggingface.co/bert-large-uncased</a>
RoBERTa	roberta-large	1024	~ 336M	- ; 160GB	<a href="https://huggingface.co/roberta-large">https://huggingface.co/roberta-large</a>
ELECTRA	electra-large-generator	256	~ 51M		<a href="https://huggingface.co/google/electra-large-generator">https://huggingface.co/google/electra-large-generator</a>
ALBERT	albert-xxlarge-v2	4096	~ 223M	3.3B ; 16GB	<a href="https://huggingface.co/albert-xxlarge-v2">https://huggingface.co/albert-xxlarge-v2</a>
GPT2	gpt2-large	1280	~ 774M	- ; 40GB	<a href="https://huggingface.co/gpt2-large">https://huggingface.co/gpt2-large</a>
GPT-Neo	gpt-neo-2.7B	2560	~ 2.7B	420B ; -	<a href="https://huggingface.co/EleutherAI/gpt-neo-2.7B">https://huggingface.co/EleutherAI/gpt-neo-2.7B</a>
GPT-J	gpt-j-6B	4096	~ 6B	- ; 825GB	<a href="https://huggingface.co/EleutherAI/gpt-j-6B">https://huggingface.co/EleutherAI/gpt-j-6B</a>

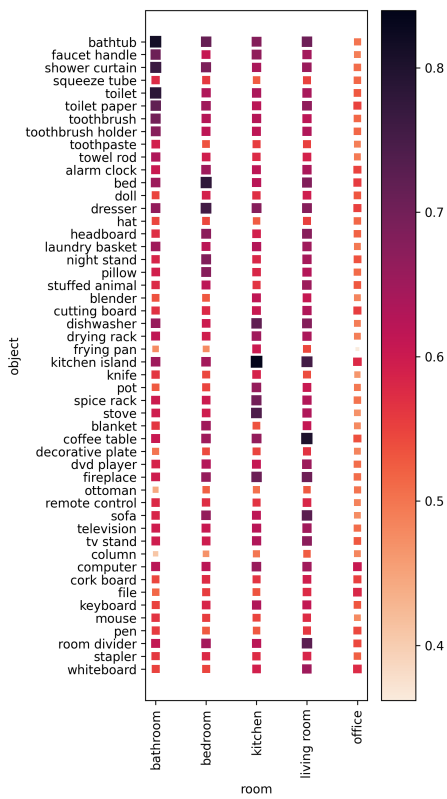
Table 7.7: Model overview. Mostly only the token quantity (T) or the dataset size (S) was given.

Task	Model	Data	Templates
Cosine Score & Classification	MLM & CLM	Room & Objects & Parts	This is a/an {x}. That is a/an {x}. There is a/an {x}. Here is a/an {x}. A/An {x} is here. A/An {x} is there.
		Verbs	I {x} something. I {x} anything. I {x}. You {x} something. You {x} anything. You {x}.
Increased Log Probability	MLM	Room & Object	A/An {obj} is usually in the {room}.
		Object & Part	A/An {part} is usually part of a/an {obj}.
		Verb & Object	I usually {verb} this {obj}.
	CLM	Room & Object	A/An {obj} is usually in the ... In the {room} is usually a/an ...
		Object & Part	A/An {part} is usually part of a ... ...
		Verb & Object	In the {obj} is usually a/an ... I usually {verb} this ...

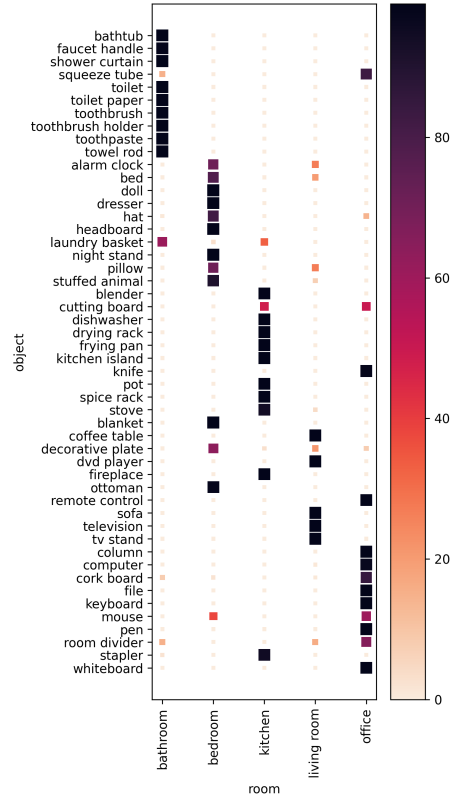
Table 7.8: Templates for calculating scores regarding *Masked Language Models* (MLM) and *Causal Language Models* (CLM). For more details, see Section 7.4.

	Word2Vec					GloVe					Levy					fastText					static-BERT									
	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn	cos	dist	kend	knn	svm	ffn
bathroom	0.73	0.75	0.78	0.31	0.31	0.22	0.53	0.56	0.57	0.23	0.00	0.23	0.65	0.67	0.68	0.25	0.00	0.32	0.65	0.67	0.70	0.00	0.00	0.00	0.74	0.75	0.76	0.56	0.41	0.47
bedroom	0.55	0.53	0.56	0.00	0.36	0.35	0.72	0.72	0.69	0.50	0.35	0.57	0.51	0.51	0.50	0.00	0.00	0.37	0.58	0.54	0.53	0.66	0.55	0.35	0.45	0.44	0.44	0.68	0.35	0.36
kitchen	0.51	0.52	0.51	0.35	0.49	0.42	0.37	0.34	0.34	0.29	0.35	0.33	0.46	0.46	0.46	0.20	0.00	0.20	0.33	0.36	0.40	0.30	0.20	0.37	0.46	0.46	0.46	0.20	0.31	0.42
living room	0.63	0.61	0.61	0.36	0.46	0.50	0.48	0.62	0.63	0.44	0.29	0.30	0.53	0.57	0.57	0.41	0.41	0.54	0.39	0.57	0.59	0.28	0.29	0.27	0.52	0.54	0.53	0.00	0.30	0.52
office	0.65	0.65	0.58	0.38	0.26	0.37	0.52	0.57	0.56	0.38	0.43	0.47	0.66	0.66	0.65	0.27	0.50	0.50	0.45	0.43	0.41	0.58	0.35	0.30	0.37	0.40	0.40	0.57	0.17	0.42
	BERT-Base					BERT-Large					RoBERTa					ElectraGen					Albert									
	cos	m-s	m-t	knn	svm	ffn	cos	m-s	m-t	knn	svm	ffn	cos	m-s	m-t	knn	svm	ffn	cos	m-s	m-t	knn	svm	ffn	cos	m-s	m-t	knn	svm	ffn
bathroom	0.40	0.35	0.30	0.23	0.23	0.23	0.52	0.34	0.50	0.23	0.23	0.24	0.61	0.47	0.42	0.43	0.23	0.35	0.58	0.66	0.39	0.35	0.35	0.40	0.69	0.56	0.36	0.34	0.36	0.39
bedroom	0.61	0.47	0.41	0.45	0.28	0.37	0.63	0.36	0.37	0.42	0.19	0.36	0.67	0.56	0.33	0.42	0.28	0.41	0.41	0.53	0.62	0.68	0.55	0.50	0.54	0.58	0.36	0.18	0.31	0.47
kitchen	0.34	0.60	0.35	0.30	0.23	0.43	0.38	0.73	0.45	0.75	0.75	0.62	0.65	0.38	0.49	0.46	0.34	0.24	0.48	0.37	0.37	0.25	0.44	0.43	0.43	0.75	0.38	0.35	0.22	0.45
living room	0.52	0.57	0.56	0.25	0.20	0.32	0.47	0.47	0.54	0.36	0.36	0.39	0.43	0.51	0.45	0.51	0.27	0.27	0.41	0.36	0.45	0.44	0.49	0.47	0.38	0.54	0.47	0.19	0.30	0.34
office	0.68	0.56	0.64	0.35	0.44	0.56	0.64	0.75	0.67	0.26	0.58	0.45	0.48	0.49	0.53	0.27	0.28	0.47	0.50	0.70	0.55	0.59	0.55	0.25	0.72	0.54	0.61	0.58	0.37	0.36

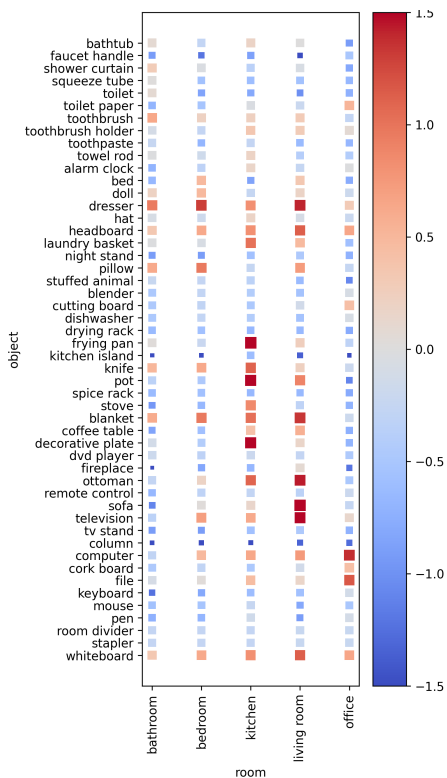
Table 7.9: Distance Correlation calculated on the word frequencies of Google Ngram. (A gray cell indicates significant at  $p < 0.1$ )



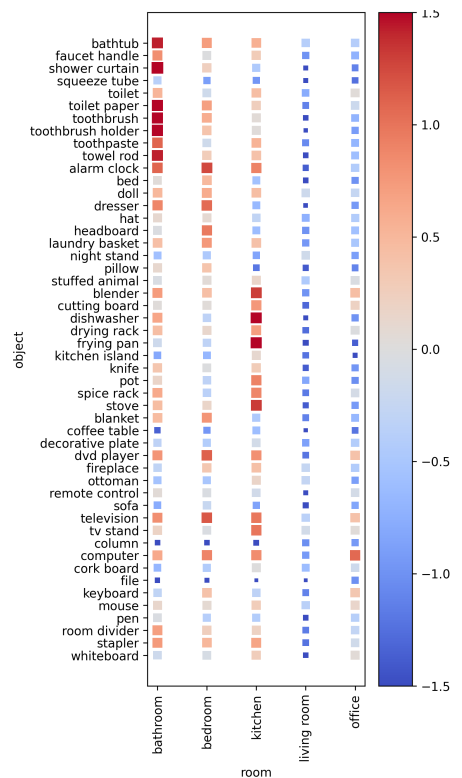
(a) Cosine Score



(b) FFN Classify Score



(c) Mask Object Score



(d) Mask Room Score

Figure 7.2: Heatmap of source-object associations based on BERT-Large and the room dataset. The objects (sources) on the y-axis are grouped by the room in which they are most likely to be located according to the *NYU Depth V2 Dataset*.

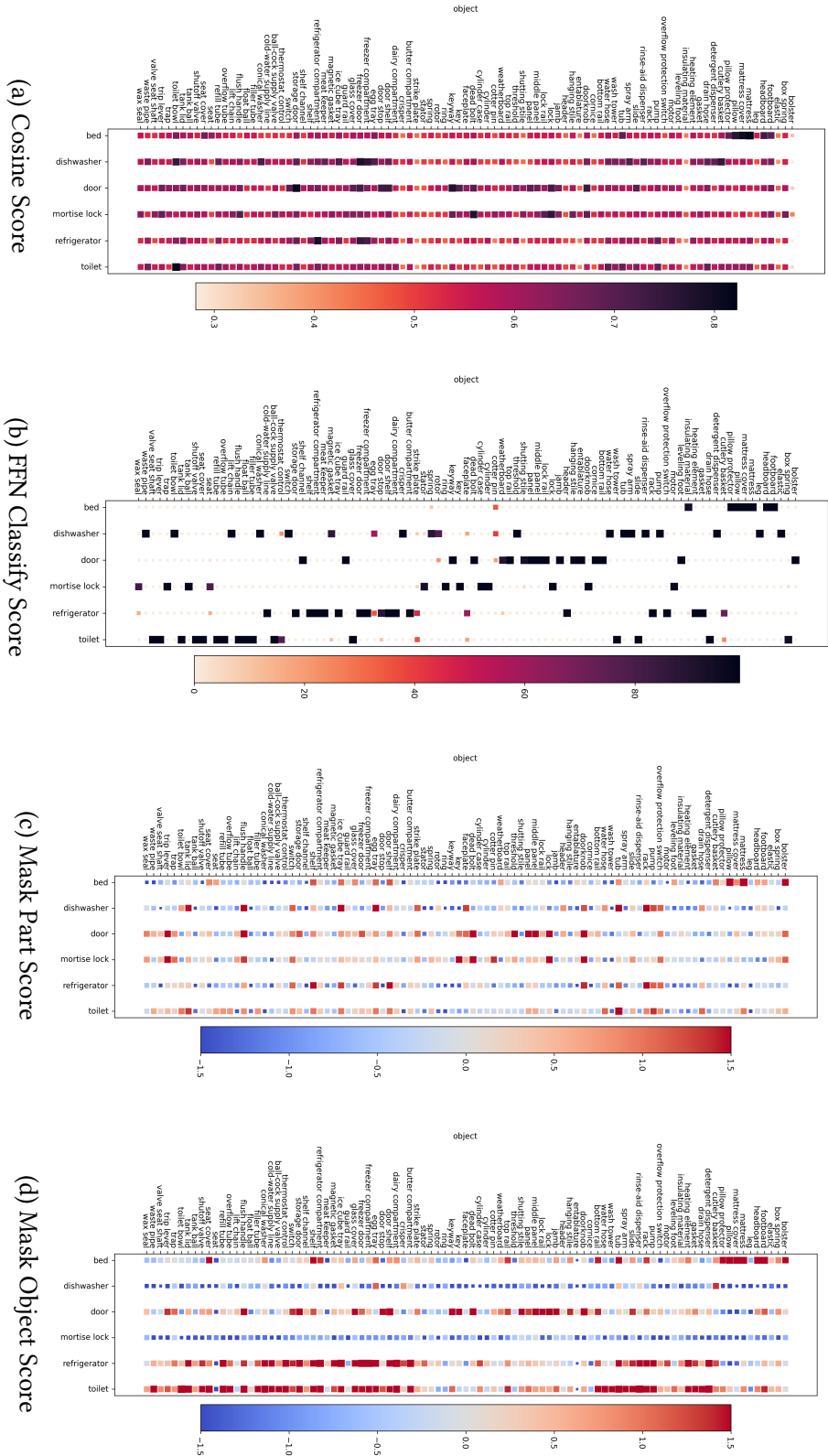


Figure 7.3: Association heatmap of BERT-Large on the part dataset. The parts (sources) on the y-axis are grouped by the room in which they are most likely to be located according to the *Online-Bildwörterbuch Dataset*.

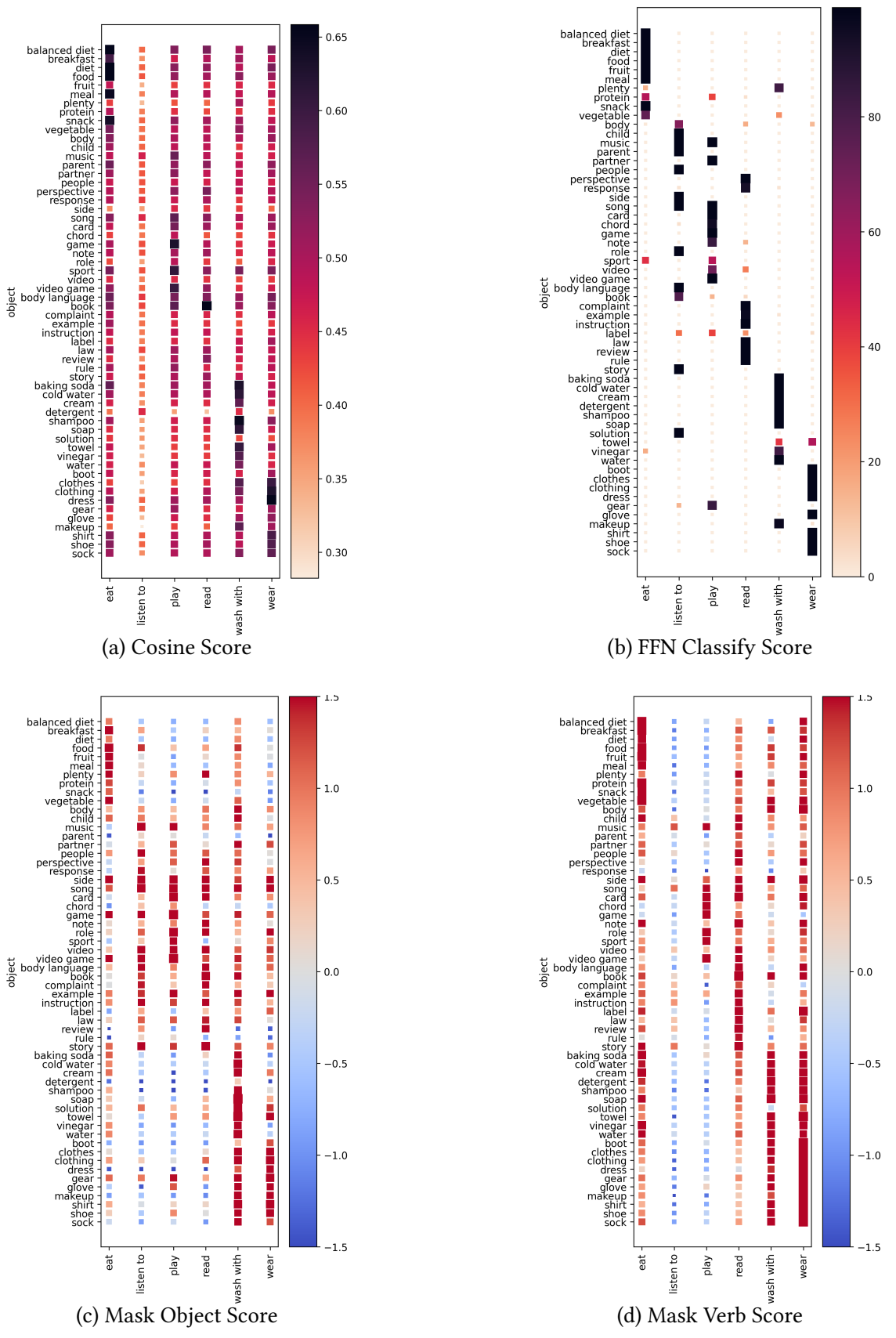


Figure 7.4: Association heatmap of BERT-Large on the verb dataset. The objects (sources) on the y-axis are grouped by the room in which they are most likely to be located according to the *HowToKB Dataset*.





# 8 Grounding Human-Object Interaction to Affordance Behavior in Multimodal Datasets

Henlein, A., Gopinath, A., Krishnaswamy, N., Mehler, A., & Pustejovsky, J. (2023a). Grounding human-object interaction to affordance behavior in multimodal datasets. *Frontiers in Artificial Intelligence*, 6

## Abstract

While affordance detection and Human-Object interaction (HOI) detection tasks are related, the theoretical foundation of affordances makes it clear that the two are distinct. In particular, researchers in affordances make distinctions between J.J. Gibson’s traditional definition of an affordance, “the action possibilities” of the object within the environment, and the definition of a *telic* affordance Pustejovsky (2013), or one defined by conventionalized purpose or use. We augment the HICO-DET dataset with annotations for Gibsonian and telic affordances and a subset of the dataset with annotations for the orientation of the humans and objects involved. We then train an adapted Human-Object Interaction (HOI) model and evaluate a pre-trained viewpoint estimation system on this augmented dataset. Our model, AffordanceUPT, is based on a two-stage adaptation of the Unary-Pairwise Transformer (UPT), which we modularize to make affordance detection independent of object detection. Our approach exhibits generalization to new objects and actions, can effectively make the Gibsonian/telic distinction, and we show that this distinction is correlated with features in the data that are not captured by the HOI annotations of the HICO-DET dataset.

## 8.1 Introduction

Introduced by Gibson in the 1970s, the concept of an “affordance” describes the functional and ecological relationship between organisms and their environments (Gibson, 1977). Gibson formulated the concept as a measure of what the environment “offers the animal” in terms of action possibilities of the object. In modern AI, particularly as it pertains to problems of perception in robotics (Horton et al., 2012) and grounding language to vision (McClelland et al., 2020), to say an object “affords” an action is to say that the object facilitates the action being taken with it. *Gibsonian* affordances are those

behaviors afforded due to the physical object structure, and can be directly perceived by animals. For example, if a cup has a handle, it *affords* grasping and lifting by that handle. Pustejovsky, following from his Generative Lexicon theory (Pustejovsky, 1995) subsequently introduced the notion of a *telic* affordance, or behavior conventionalized due to an object’s typical use or purpose (Pustejovsky, 2013). For example, a cup’s conventional *purpose* is *for drinking from* and a book’s is *for reading*. These conventionalized afforded behaviors are correlated with certain specific configurations between human and object; e.g., a chair must be upright with its seat clear to be sat in. These conditions (or *habitats*) form a precondition to the satisfaction of the intended use of the object; if those conditions are satisfied, the act of sitting on the chair will lead to the expected result of the chair supporting the human (i.e., its Telic qualia role according to Generative Lexicon theory). If not (e.g., the chair is upside down), the human will not be appropriately supported.

On the question of multimodal grounding, the computer vision and natural language processing (NLP) communities have drawn closer together, such that datasets originating in computer vision (e.g., (Goyal et al., 2017; Damen et al., 2018; Boggust et al., 2019)) now have demonstrated utility as benchmarks for NLP grounding tasks (e.g., (Gella & Keller, 2017; Huang et al., 2020; Xu et al., 2020)). One such popular challenge is grounding words to actions in images and video (e.g., (Radford et al., 2021)). As such actions often involve humans interacting with objects, datasets specialized to not just actions (running, jumping, walking, etc.) but to *human-object interaction* (HOI) have also proliferated in recent years (Chao et al., 2018; Krishna et al., 2017; Gupta & Malik, 2015).

Knowledge of how a human interacts with an object, however, is not always revealed through a single modality (language or image), and often even the alignment of multimodal annotations (e.g., bounding box and linguistic caption) does not adequately encode the actual HOI in a situation. For many HOIs, conventional descriptions used to caption them often fail to draw out significant aspects of the interactions that are important for creating visual embeddings. For example, it would be expected that an image with the caption “person driving a car” would share certain visual correlations with images of tools held in the hand, but there is usually no linguistic expression present in the caption to explicitly evidence that the driver is holding a steering wheel, etc.

Humans most often learn about affordances (e.g., “cups contain things”, “spoons are used for stirring”) by using objects or watching them in use (Tomasello, 2004); hence there is a natural alignment between affordance reasoning and various kinds of HOI tasks.

However, it must be noted that affordances and HOIs are not identical. Returning to Gibson’s original formulation of the concept, he expands on it by stating that an affordance “implies the complementarity of the animal and the environment”. That is to say that the Gibsonian affordance, one afforded by an object’s structure, is not just any action which can be taken with an object, but an action that is somewhat specific to that object and that agent in that environment. For example, the hollow geometry of a bottle *affords* containing liquids, while the opening *affords* releasing them. An image of a human drinking from a bottle, with it raised to the mouth, implies both the structure and the purpose of the bottle, even though neither is made explicit from the collocation of

the object *bottle* and the action *drink\_from*. It is this type of intentionality information, or identification of the relation between the object and human that is largely missing from grounded HOI datasets.

In this paper, we address the question of whether HOI models can distinguish the intentionality behind telic affordances from Gibsonian *exploitation* of an object.

Our novel contributions are as follows:

1. We present an augmentation of the HICO-DET (Chao et al., 2018) dataset that is annotated to distinguish Gibsonian from telic affordances at the visual and linguistic levels.
2. We developed AffordanceUPT, an adapted and modularized version of UPT (Zhang et al., 2021a) that is trained over this novel data and can generalize to certain novel objects and actions.
3. We evaluate PoseContrast, a SOTA object orientation model, over the augmented dataset and find that PoseContrast tends to exhibit a strong bias toward the most frequent or default orientation, rather than the appropriate orientation for the action.

AffordanceUPT trained over the augmented HICO-DET dataset is able to accurately distinguish active intentional use from simple Gibsonian exploitation, and we find that the way objects cluster when the model is trained for the Gibsonian/telic distinction exposes additional correlations to the visual features of the specific images themselves.

## 8.2 Related Work

There has been considerable interest in how encoding affordances might be used to improve the accuracy of HOI recognition and scene understanding models (Hassanin et al., 2021), as well as in downstream reasoning tasks in cognitive models of HOI or computational models of HRI. Psychological studies have shown that humans respond faster when objects are observed in canonical configurations (or *habitats* (Pustejovsky, 2013)) for their typical affordances (Yoon et al., 2010; Borghi et al., 2012; Natraj et al., 2015). Roboticists are particularly interested in affordances to model human-like interactions with objects, and work from that community has demonstrated that in order to successfully interact with an object, a robot need not know the object’s name, but only perceive its function (Myers et al., 2015) or object affordances (Kim & Sukhatme, 2014; Saponaro et al., 2017). Affordances have also been recognized as implicating broader decisions for planning and inference (Horton et al., 2012; Antunes et al., 2016; Beßler et al., 2020).

The NLP community has made significant contributions in extracting object-oriented knowledge from language data. Multimodal datasets have been used to associate linguistic descriptions to visual information from action images, e.g., IMAGACT (Russo et al., 2013; Moneglia et al., 2018). Other research has explored integrating different descriptions of affordance information coming from language and visual datasets (Chao

et al., 2015; Saponaro et al., 2017). Several approaches have identified objects' functional roles and factors involved with their creation using standard distributional techniques reflecting PPMI between action verbs and object types (Cimiano & Wenderoth, 2007; Yamada et al., 2007). These correlate with the *telic* (function) and *agentive qualia* (creation) a la Pustejovsky.

Recently it has become clear that not all modes of interacting with an object involve an affordance, while not all relevant object affordances are actually involved in the interaction the human is shown engaging in in an image (Beßler et al., 2020; Hassanin et al., 2021). To address this, Pustejovsky (Pustejovsky, 2013) defines a *habitat* as the precondition for an action to take place. Namely, a habitat is a conditioning environment or context that facilitates the enactment of an afforded behavior, such as how a bottle must be held to be drunk from. A primary component of habitats is object orientation, and therefore a potentially useful multimodal method for habitat detection is *pose detection*.

Pose detection has applications ranging from autonomous driving (Caesar et al., 2020), to robotics (Tremblay et al., 2018), and language grounding (Thomason et al., 2022). Consequently, available datasets are also diverse and specialized (more details in Section 8.3.3). Only recently has object orientation has been introduced into HOI Detection (e.g. D3D-HOI (Xu et al., 2021) or BEHAVE (Bhatnagar et al., 2022)). So far, the focus has been mainly on human pose (e.g. Yao & Fei-Fei, 2010) or object size and positioning (e.g. Li et al., 2020).

## 8.3 An Approach to Detecting Affordances

### 8.3.1 Theory

When we identify and label objects, we not only perform a categorical type assignment (e.g., cup), but more often than not, we understand an entire set of object attributes as well as a network of relations concerning how the object participates in the situation under discussion. Many of these involve human-object interactions (HOIs), and our knowledge of things is predicated on an understanding of how we interact with them. Osieurak et al. (2017) provide a clear operationalization of this mechanical knowledge of affordances in the domain of tool use. In this domain, Norman (Norman, 2002) divided Gibson's formulation into *physical* and *learned* affordances, and Young (Young, 2006) specified the notion of the functional affordance. These specifications divide affordances into *hand-centered* and *tool-centered*, and the divisions map relatively straightforwardly to Gibson's affordances and Pustejovsky's *telic* affordances, but do not per se address the question of object orientation to the human.

For example, there is a conventional presupposition that the orientation of the cup exposes the concavity of the interior to enable the functioning of the cup (Freksa, 1992). Assuming that an object such as a cup, typed as a container, is asymmetric across the plane bisecting it horizontally, but otherwise a symmetrical cylindroid, it would appear that orientation information is critical for enabling the use or function of the object *qua* container. In fact, only when the cup's orientation facilitates containment can the func-

tion be “activated”, as it were. This references two notions that are critical for reasoning about objects and HOI generally: we encode *what* the function associated with an object is (its affordance) (Gibson, 1977), but just as critically, we also identify *when* it is active (its habitat) (Pustejovsky, 2013). Therefore, as given by Pustejovsky’s original definition of the telic affordance, in this study we consider telic as a proper subset of the Gibsonian affordance, that overrides it; a telic affordance necessarily exploits the structural properties of the object, but does so in a way that *selects for a conventionalized configuration to activate a conventionalized function*.

To capture object type and human-object interaction potential, we adopt conventions used in the modeling language VoxML (Pustejovsky & Krishnaswamy, 2016), where habitats, including orientation, are modeled as preconditions on affordances, that is, the situational information about when/how an object is used. This allows modeling contextual and common-sense information about objects and events that is otherwise hard to capture in unimodal corpora, e.g., *balls roll because they are round*.

Hence the task of extracting dependencies between object habitats and affordances is consequential for tasks like automatic annotation of VoxML or Text-to-3D Scene applications (Chang et al., 2015a). The current study focuses on adapting HOI models for affordance type classification using the Gibsonian/telic distinction and object orientation.

### 8.3.2 Annotation

#### Image Context Annotations

Our dataset consists of images taken from HICO-DET, a benchmark for HOI detection (Chao et al., 2018). Every image contains annotations for each HOI instance—bounding boxes for the humans and the objects with labels for the interactions. We annotated 120 images taken from 10 object categories for a total of 1,200 images. The 10 object categories are *apple, bicycle, bottle, car, chair, cup, dog, horse, knife* and *umbrella*, chosen for being representative of the full set of HICO-DET object categories, which includes animals, vehicles, and household objects. Using a modification of the VIA tool (Dutta et al., 2016; Dutta & Zisserman, 2019) as shown in Figure 8.1, each image was annotated for the *action, affordance* class (Gibsonian/telic), and direction of *front* and *up* orientation of the objects therein. *Action* and *affordance* were annotated for all the relevant humans in an image, and orientation fields *up* and *front* were annotated for both the objects and the humans. Additionally, fields *is\_part\_of* and *changes?* were used to track whether an item being annotated was part of another annotated item and whether any changes were made in the annotations (new object or action) from those specified in the HICO-DET dataset respectively.

The possible options for the field *affordance* are *None, Gibsonian (G)* and *telic (T)*. The affordance is marked as G when the action performed is by virtue of the object’s structure and T if by virtue of the object’s conventionalized use or purpose (see Section 8.3.1). The fields *action* and *obj name* are chosen from the list of actions and object names respectively provided in the HICO-DET dataset. Front and upward orientations

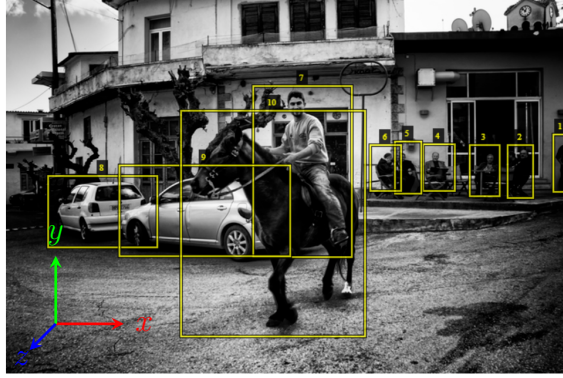


Figure 8.1: Example image context annotation. The image shows a telic affordance between horse (10) and person (7) and both with orientation:  $front(-1, 0, 1)$   $up(0, 1, 0)$ .

Object	Action	Affordance
bicycle	ride	T
bicycle	hold	G
bottle	hold	G
bottle	drink_with	T
cow	milk	T
cat	feed	T
banana	carry	G
skis	pick_up	G
knife	cut_with	T

Table 8.1: A small subset of text annotations. G stands for Gibsonian and T for telic.

are selected from the world orthogonal axes  $[x, y, z]$ . When viewing an image face-on,  $+x$  is to the right of the screen,  $-x$  is to the left,  $+y$  is upward and  $-y$  is downward, while  $+z$  extends out of the screen toward the annotator and  $-z$  is pointing away from them into the screen. This assumed a standard right-hand coordinate system as shows in Figure 8.1. Axes can be combined. If the front of the human or object faces both leftward and forward (out of the image), then the *front* orientation is  $-x + z$ , and  $+x + z$  if turned halfway towards the right. If no clear front or top was apparent (e.g., for a ball), it was annotated as  $[0, 0, 0]$ . In this paper we denote orientation using the notation *front\_up* with each vector represented as  $(x, y, z)$ . The horse in Figure 8.1 would be denoted  $[-1, 0, 1]$   $[0, 1, 0]$ , because its forward vector is facing toward the left ( $-x$ ) and out of the image ( $z$ ) while its intrinsic up vector is pointing up ( $y$ ).

These annotations were later used to evaluate Object Pose Detection (see Section 8.3.3) and to evaluate the overall Habitat Extraction approach (Section 8.4.4).

## Text Annotations

Each of the 600 object-verb pairs in the HICO-DET dataset were also annotated with the affordance (G for Gibsonian or T for telic). Table 8.1 shows a few examples. In HICO-DET, people and objects are often associated with multiple verbs (e.g., a person sits, rides, and races a motorcycle). If only one action of such a set has been defined as telic, we define this as a telic action as telic affordances are considered to supercede Gibsonian affordances—see Section 8.3.1). These text-only annotations have the advantage of rapidly generating data for training HOI models, while lacking some additional contextual information that may be provided by an image, as in Section 8.3.2. These annotations were later used to train and evaluate the AffordanceUPT Model (see Section 8.3.3).

Image and text annotation were each performed by a different person. The calculated IAA is listed in appendix (Table 8.5).

### 8.3.3 Models

#### Human-Object-Interaction

We adapted the UPT (*Unary-Pairwise Transformer*; Zhang et al., 2022) model as the basis for Gibsonian/telic affordance classification. UPT is a two-step transformer-based (Vaswani et al., 2017) HOI classifier and its authors demonstrate that it is comparatively performant and memory efficient compared to other state-of-the-art HOI models (e.g., Tamura et al. (2021); Zhang et al. (2021b)). In the first step, it determines all relevant entities and in the second step their relations (in contrast to single-task models, where entities and relations are considered together in multi-task learning (Zhang et al., 2021a)). UPT is therefore composed of two parts: a *cooperative transformer*, which operates on *unary tokens* to generate a representation of entities, and a *competitive transformer*, which subsequently operates on *pairwise tokens* to represent their relations.

Moreover, the two-step approach enables the analysis of both representations of objects (*unary tokens*) and of their interactions (*pairwise tokens*) (see Section 8.4).

To utilize UPT for affordance detection, we changed the classification from a variable number of verbs to a two-label Gibsonian/telic classification. We also modularized UPT to make the affordance detection independent of object detection based in DETR (*Detection Transformer*; Carion et al., 2020), which uses ResNet (He et al., 2016) as a backbone. That is, we replaced the pre-trained, inflexibly implemented DETR variant (supporting 80 object types) with a modular variant from Huggingface<sup>1</sup> (supporting 90 object types) and froze all DETR/ResNet weights. This makes our UPT variant independent of the object detection module, so that it can be replaced by models that support other object types. We will refer to the model as **AffordanceUPT** in the remainder of this paper. The performance of AffordanceUPT on unknown objects and actions is also part of our evaluation (see Section 8.4.1). Our approach to affordance detection shows how methods such as UPT can be applied to this and related tasks in multimodal semantics.

<sup>1</sup><https://huggingface.co/facebook/detr-resnet-50>

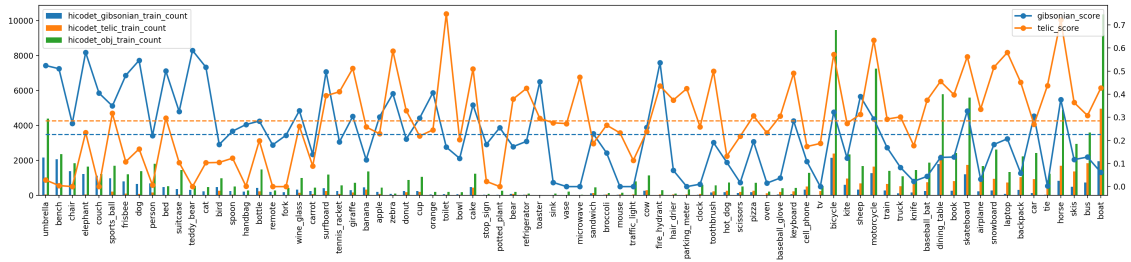


Figure 8.2: AffordanceUPT evaluation regarding object types and training data size. The bottom axis lists the object labels. The left axis and associated bar graphs show the number of *Gibsonian* (blue), *telic* (orange), and general object occurrences (green) in the HICO-DET training subset. The right axis and corresponding line graph show the mAP for each object. Dashed lines denote overall mean values for the two affordance types. The objects are sorted by the ratio between G and T training samples.

### Object Pose Estimation

To estimate object orientation, we use PoseContrast (Xiao et al., 2021). This model has the advantage of not requiring additional information such as CAD references or class information, while still providing strong results (cf. Xiao et al., 2019; Dani et al., 2021; Nguyen et al., 2022). We retrained the model on the ObjectNet3D dataset (Xiang et al., 2016), which is still one of the largest datasets for this task with 100 object categories and over 90 000 images. Other common datasets are still very limited in their domain or object categories (see Table 8.4 in appendix).

### Training

AffordanceUPT was trained for 20 epochs on 2 GeForce RTX 8000 devices with a batch size of 8 per GPU—an effective batch size of 16. Hyperparameter optimization was performed using W&B (Biewald, 2020). The resulting parameters are listed in Table 8.6 (appendix). The respective HICO-DET dataset, annotated with Gibsonian/telic labels as described in Section 8.3.2, served as training and test data. Images without Gibsonian/telic text annotations were removed, resulting in a dataset size of 33 593 training images and 8 527 testing images. In addition to training with the regular HICO-DET split, we also trained variants to evaluate generalization to unknown objects and actions (see Section 8.4.1).

PoseContrast was trained on one GeForce RTX 8000 with default parameters. Different hyperparameters and additional methods of augmenting the training data were tested, but did not result in significant improvements.



	Model	Subset	mAP x 100
	AffordanceUPT	test all	27.58
object	w/o bicycle	all bicycle	35.74
	AffordanceUPT	test bicycle	46.69
	w/o car	all car	20.44
	AffordanceUPT	test car	33.54
verb	w/o wield	all wield	32.99
	AffordanceUPT	test wield	37.23
	w/o drive	all drive	21.40
	AffordanceUPT	test drive	26.05
obj+verb	w/o book+read	all book+read	24.11
	AffordanceUPT	test book+read	31.46
	w/o car+drive	all car+drive	15.63
	AffordanceUPT	test car+drive	22.63

Table 8.2: UPT Results on the Gibsonian/telic text annotated HicoDet Test dataset. The first column denotes the model, where *AffordanceUPT* stands our default model trained on the regular Gibsonian/telic HicoDet dataset. *w/o* denotes models that have been trained without the respective object/verb (e.g. bicycle).

## 8.4 Evaluation & Analyses

### 8.4.1 Evaluation of AffordanceUPT

For the evaluation of AffordanceUPT see Table 8.2 and Figure 8.2. The results show that HOI models can also be used for affordance detection with a few adjustments, as shown in the example of UPT. The mAP values are within  $\sim 1-5$  mAP) of HOI detection on the regular HICO-DET dataset (cf. Zhang et al. (2022); Tamura et al. (2021); Hou et al. (2021b)). The differences are for a few reasons:

- i) The distributions of our target classes are much more complex in nature, subsuming multiple diverse actions;
- ii) HICO-DET has separate bounding boxes for each action, and these can vary widely, resulting in multiple boxes for the same object or person;
- iii) Not every affordance in HICO-DET is always annotated but AffordanceUPT detects them anyway;
- iv) Our object detection model is not trained on HICO-DET, so there can be major deviations for the boundary boxes that cannot be merged.

A few examples can be found in the appendix (see Figure 8.15). These do not significantly affect training and inference, but are reflected in the evaluation score since the problem

primarily concerns the boundary boxes and not the affordance label itself. We deliberately decided against alternative datasets like V-COCO (Lin et al., 2014; Gupta & Malik, 2015) or VisualGenome (Krishna et al., 2017), as V-COCO has a very limited set of verbs (26) and VisualGenome is too unstructured for now.

To evaluate AffordanceUPT on novel objects, we examine a few specific examples, specifically: the nouns *bicycle* and *car*, the verbs *wield* and *drive*, and the HOIs *book+read* and *car+drive*. We re-split HICO-DET such that for each example, the test set comprised all images containing the example the training data comprised all remaining images (i.e., for *car+drive*, images of boats being driven or cars being washed were omitted from both training and evaluation). These results were then compared against the results of the normal AffordanceUPT model on the objects/verbs in the regular HICO-DET test dataset.

Table 8.2 shows that AffordanceUPT can detect affordances on novel objects, albeit with an appreciable drop in mAP (e.g., ~10–13%). The effect is less strong for unknown actions such as *driving* (only a drop of around 5%). AffordanceUPT can even generalize to some extent to novel objects and actions (e.g. detecting that driving a car is a telic affordance, despite never seeing a car or a driving action). Meanwhile, regular HOI models generalize only on unknown HOI combinations (e.g. Hou et al., 2021b; Shen et al., 2018) or on unknown objects (e.g. Hou et al., 2021a; Wang et al., 2020), not both.

Because each re-split requires retraining, the evaluation could not be carried out for all combinations due to runtime reasons. However, the tendencies are clearly apparent.

The generalization on display here is only made possible by our abstraction to the two affordance types that point to specific kinds of action classes that can be contained under the same label. This means affordance detection supports a higher level of generalization due to greater abstraction, and makes affordance detection interesting for applications where the exact action does not need to be detected, but a distinction of intentional use is sufficient.

Such situations could be, for example,

- i) monitoring an object’s active usage time (e.g., is a knife likely to be getting dull from continued use?).
- ii) for autonomous driving (e.g. whether a pedestrian is distracted by the active use of an object and therefore more caution is required (Papini et al., 2021)).
- iii) language grounding applications, such as grounding for robotics (Ahn et al., 2022), visual question answering (Antol et al., 2015) or image captioning (Nguyen et al., 2021)—specifically in cases where the verb implies one kind of affordance but the image indicates the other (e.g., an image of someone driving a car captioned as “riding”).

## 8.4.2 Evaluation of PoseContrast

We used the 1 200 image annotations of HICO-DET from Section 8.3.2 to evaluate PoseContrast. Since PoseContrast outputs object rotation as Euler angles, but the annotations

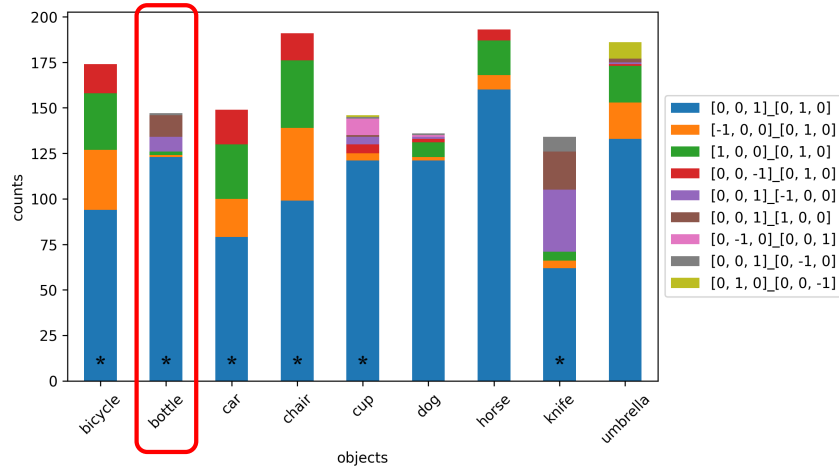


Figure 8.3: PoseContrast orientation predictions on the 1 200 annotated HICO-DET images for 9 object classes. Predicted orientations with a frequency of  $< 5$  were filtered out. \* marks objects that are also in ObjectNet3D.

Model	apple	bicycle	<b>bottle</b>	car	chair	cup	dog	horse	knife	person	umbrella
$[0, 0, 1]_{[0, 1, 0]}$	0.18	0.13	0.57	0.19	0.27	0.72	0.20	0.21	0.01	0.40	0.73
Most Frequent	0.65	0.41	0.57	0.38	0.31	0.72	0.21	0.41	0.18	0.40	0.73
PoseContrast	0.83	0.44	0.67	0.51	0.58	0.75	0.31	0.25	0.08	0.44	0.67

Table 8.3: PoseContrast results on the image annotated HicoDet dataset. The object names in bold are also represented in ObjectNet3D.

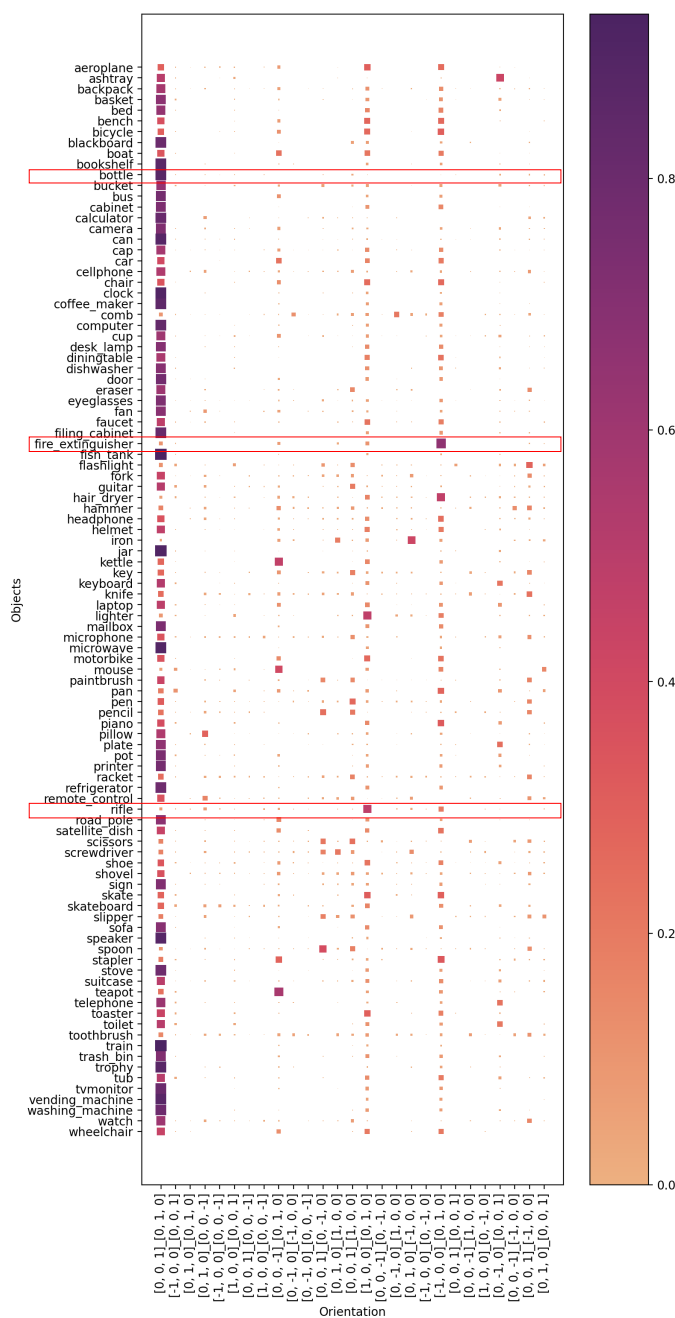


Figure 8.4: ObjectNet3D dataset mapped to main orientations. Scores are weighted for every object. An interesting example is “bottle” (red box), which occurs almost exclusively in upright position in the dataset. Other interesting examples include “fire extinguisher” and “rifle”, which also exist in the dataset in stereotypical pose (cf. Barbu et al., 2019), but which for these objects means that the front of the object points to the side of the image.

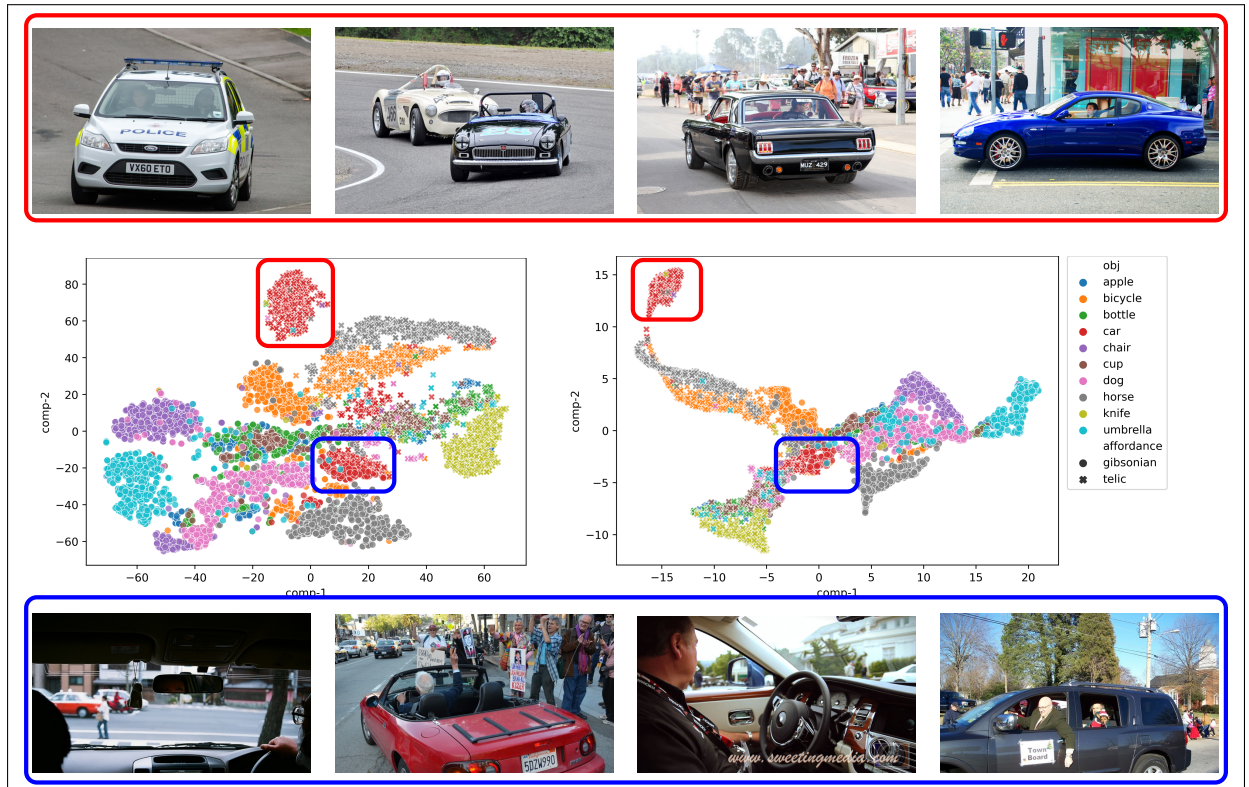


Figure 8.5: AffordanceUPT token-pair visualization using t-SNE (left) and PaCMAP (right). The vehicle images above and below are “ride” images from the HicoDet dataset and classified as telic by the model. The images in the top row are in the red cluster and the images in the bottom row are in the blue cluster.

indicate the major axis orientation, the PoseContrast output was mapped to these axes. The evaluation scores thus describe the accuracy with which the objects were aligned with the correct major axes. We compare PoseContrast with two baselines: one, in which the object is always predicted to be facing forward and upright ( $[0, 0, 1]_{-}[0, 1, 0]$ ), and a second, which always predicts to the most frequent orientation in the HICO-DET annotations (*Most Frequent*). The results are listed in Table 8.3. PoseContrast appears to generalize very poorly on the HICO-DET dataset. Notably, the default orientation  $[0, 0, 1]_{-}[0, 1, 0]$  is predicted for almost all objects (see Figure 8.3), including for object classes in the training set. Examining the ObjectNet3D dataset, we find that it almost exclusively contains objects in this orientation (e.g., upright bottles, forward-facing TVs), rather than in orientations where they are manipulated by humans (i.e., Gibsonian or telic affordances) (see Figure 8.4). Rotating the image is used as an augmentation method during training but is of limited use, e.g., if only side views of weapons are available, it is not possible to generate views from the front or back. We also tried additional augmentation methods such as blur filters and dpi variations, but they did not produce significantly better results. Further analyses can be found in the appendix (see Figure 8.10

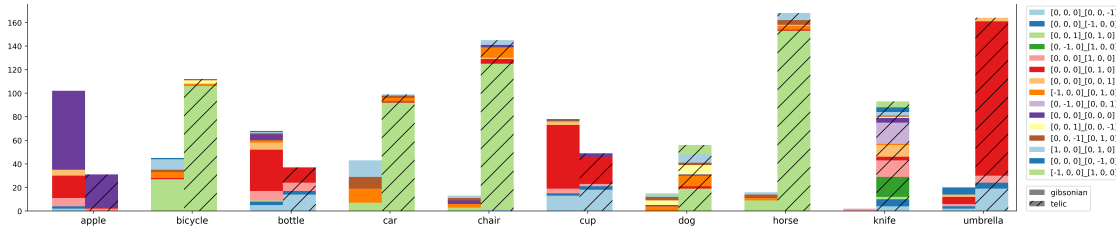


Figure 8.6: Habitats based on the 1 200 image annotations. The colors here represent the relative alignments in relation to the person.

and Figure 8.9).

### 8.4.3 Analysis of AffordanceUPT Tokens

To show how AffordanceUPT distinguishes between Gibsonian and telic affordances, in Figure 8.5 we visualize the token-pair representations for the 10 test categories using t-SNE and PaCMAP (Wang et al., 2021b). We see that objects that are interacted with in a similar way and have similar affordances appear closer together. For example, the occurrences of *bottle* and *cup* (i.e., containers to drink liquids from) are strongly overlapping. Also, *bicycles* and *horses*, both rideable, are placed close to each other when considering telic affordances. Gibsonian interactions with *horses*, on the other hand, are closer to those with *dogs* (and do not occur in the large Gibsonian *bicycle* cluster). In addition, all objects (e.g., *apple*, *bottle*, *cup*, *knife*) that imply interaction primarily with the hand are in the same region, which includes some images of cars (blue marked cluster), an initially rather unintuitive observation. But a look at the different images for “ride” in the two car clusters, explains this. In the blue cluster (closer to the hand-held objects), the interactions of the hand with the car (e.g. steering wheel) are more clearly visible, while in the red cluster the people (and therefore hands) are less visible, and the images focus more on the entire car and the actual “driving” aspect. The same apparent HOI action class (in this case, “ride”), as given by the original labels in HICO-DET, in fact divides into distinct clusters based simply on how the model is trained to represent the two-way affordance type distinction (Gibsonian and telic). Such information is essential for accurately grounding visual human-object interactions to language, and thus leads us back to the motivation from the introduction. This work paves the way for systematically extracting such visual information and linking it to language. Complete visualizations of the *unary tokens* are in the appendix (Figure 8.12 and 8.13).

### 8.4.4 Automated Habitat Annotation

As automatic determination of object orientation is still limited, we analyze habitats based on our HICO-DET image annotations. We converted object orientations in world space to be relative to the interacting person (e.g., the person’s front is now  $+z$ ). In Figure 8.1, the horse would have the orientation  $[0, 0, 1]_{-}[0, 1, 0]$ , since it is oriented in the same direction as the person. Figure 8.6 depicts the resulting statistics, and shows

the relationship between affordance and object orientation as a habitat condition. The orientation of objects like *bicycles*, *cars*, *chairs*, *horses*, and *dogs* is relatively independent of their affordances, but these objects are often aligned in the same way as the person in the case of a telic scenario. Bottles and cups, on the other hand, show a strong relative increase in orientation to  $[0, 0, 0]$ – $[0, 0, -1]$ , indicating that the object’s upward is oriented opposite to the person’s front (typical orientation when drinking). Knives, on the other hand, can be held in any orientation, however the majority of orientations (green segment plus orange segment) indicate that knives are often held with the blade facing down, away from the person.

Figure 8.6 shows the interdependence of affordance and orientation (as a subcondition of habitat): affordances presuppose certain orientations, and conversely, certain object orientations make certain affordances possible in the first place. Therefore, both variables should be considered in relation to each other (in relation to HOI as a whole) and not as independent phenomena.

## 8.5 Discussion and Conclusions

We presented AffordanceUPT, an adaptation of UPT to distinguish between Gibsonian and telic affordances. With some augmentations to HICO-DET and modularization of UPT, we can alter a powerful HOI detection model to detect distinctions in affordances specifically. This greater level of abstraction lends itself to generalization that was not possible before from a forced-choice HOI detection model, and in the process we uncovered properties of the data that have important implications for grounding images to language.

We found that how AffordanceUPT clusters objects indicates what can be detected by automatic entity and intention detection. Such distinctions are useful for (semi) automatically populating a multimodal representation like VoxML Pustejovsky & Krishnaswamy (2016) by inferring possible affordances for an object and their preconditions. AffordanceUPT also shows promise in generalization for novel objects and actions, meaning it could also infer partial information about novel objects or events for such a representation.

### 8.5.1 Future Work

In future work, we plan a comprehensive analysis of AffordanceUPT’s performance on novel entities with respect to which training conditions must be fulfilled for the model to classify which attributes.

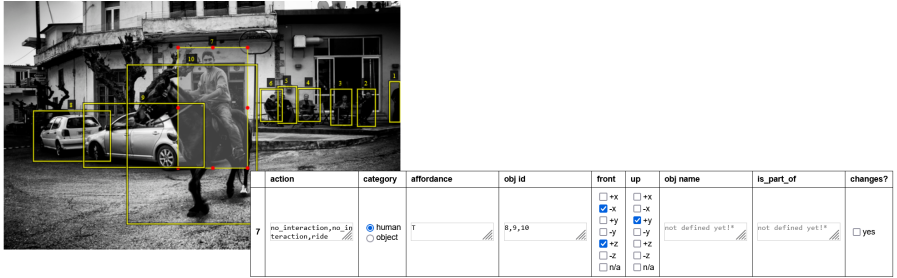
Results and interpretations like Figure 8.5 apply to a manageable subset of data. Further analysis could determine how our method scales when dealing with big data, using automated analysis techniques.

The division into Gibsonian and telic affordances can also be further refined. For example, the act of “repairing a car” is not a telic affordance, but an act of *maintaining* telic functionality.

Successful habitat detection depends on improving performance on the remaining challenge of object orientation detection. In the future, we plan to test our approach on a larger scale and expand the dataset for this purpose. This may involve combining AffordanceUPT with grounded language models e.g., CLIP (Radford et al., 2021).

## Appendix

### Annotation



(a) Example of human annotation.



(b) Example of object annotation.

Figure 8.8 shows few sample object orientations. The *front* orientations are (a) +z (b) -x+z (turned halfway towards the left) (c) +x+z (turned halfway towards the right). Since the object has two pairs of identical edges (parallel edges), we can ignore the *up* orientations or mark it *n/a* in this case. In Figure 8.7a, the human (inside the red dots - with id 7) exhibits actions *no\_interaction*, *no\_interaction*, *ride* with respect to objects with ids 8, 9 and 10 (Figure 8.1). The affordances for the actions are None, None and telic respectively. The front side of the human is pointed towards the -x+z direction and the top in the +y direction. In Figure 8.7b, the object (the horse - object id 10) is oriented in the -x+z direction (front) and in the +y direction (top).

## PoseContrast

Since PoseContrast uses ResNet for feature generation, we visualized a random subset of each 2 000 random images from HICO-DET and ObjectNet3D using t-SNE (Van der



Maaten & Hinton, 2008; Van Der Maaten, 2014) (see Figure 8.10). We additionally analyzed the correct prediction as a function of object size (in pixels) and blur factor, but could not find any particular correlation (see Figure 8.9). Newer posture prediction models seem to handle unknown objects even better, but are currently not open source (Goodwin et al., 2022; Liu et al., 2022). Based on the underlying training data (e.g. CO3D (Reizenstein et al., 2021)), where the objects are again mainly in rest positions, it is unlikely to provide sufficient improvement for our application. Table 8.4 contains a small selection of pose datasets to illustrate the diversity of the data sets. And Figure 8.11 shows a selection of example images for different objects to illustrate the difference between HICO-DET and ObjectNet3D again.

Dataset	Domain	Obj Classes	Images / Videos
300W (Sagonas et al., 2013)	Faces	1	600 Images
Animal-Pose (Cao et al., 2019)	Animals	5	4 000 Images
BEHAVE (Bhatnagar et al., 2022)	HOI	20	321 Videos
CO3D (Reizenstein et al., 2021)	Objects	50	1.5M Images
COCO (Lin et al., 2014)	Human Pose	1*	66 808* Images
D3D-HOI (Xu et al., 2021)	HOI	8	256 Videos
IKEA (Lim et al., 2013)	Furniture	8	800 Images
KITTI-360 (Liao et al., 2022)	Traffic	37	320 000 Images
Linemod (Hinterstoisser et al., 2012)	Houshold Objects	15	1 100 Images
MPII (Andriluka et al., 2014)	Human Pose	1	25 000 Images
NOCS (Wang et al., 2019)	Tabletop Scenes	6	14 Real, 300K AR
ObjectNet3D (Xiang et al., 2016)	Objects	100	90 000 Images
Objectron (Ahmadyan et al., 2021)	Objects	9	15K Videos, 4M Images
Pascal3D+ (Xiang et al., 2014)	Objects	12	36 000 Images
Pix3D (Sun et al., 2018)	Indoor	9	10 000 Images

Table 8.4: Selection of object orientation datasets with information about their size and domain coverage. COCO is a subset with pose information for the persons.

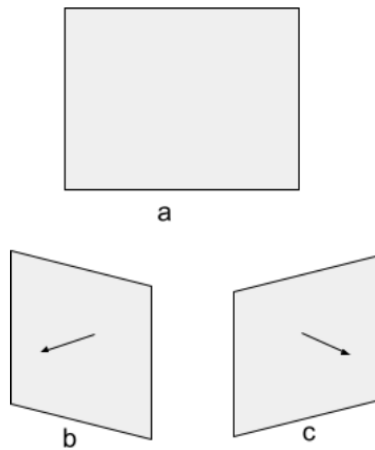


Figure 8.8: Example Object Orientation.

Obj Category	Cohen's Kappa
apple	0.6028
bicycle	0.6105
bottle	0.6411
car	0.4234
chair	0.0
cup	0.6321
dog	0.0451
horse	0.4226
knife	0.4785
umbrella	-0.0080

Table 8.5: Calculated IAA between the image and text annotations. The low kappa values for some classes can be explained by the fact that the language in the caption may not capture the unique telic affordance that is present in the image (e.g., “standing under an umbrella”), so the caption annotation alone would mark this as G, which still allows T.

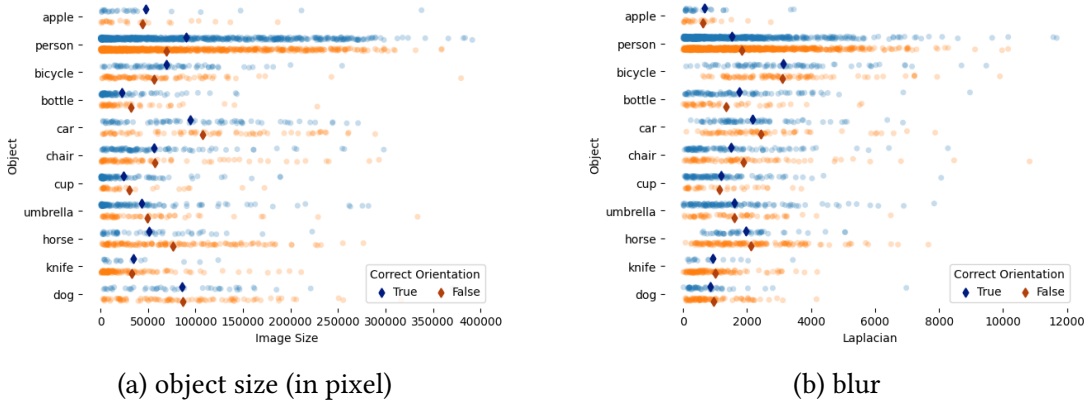


Figure 8.9: PoseContrast Object orientation determination considering image size (a) and blur (b). Blur was calculated with the variance of Laplacian (Bansal et al., 2016). A higher value means a sharper image.

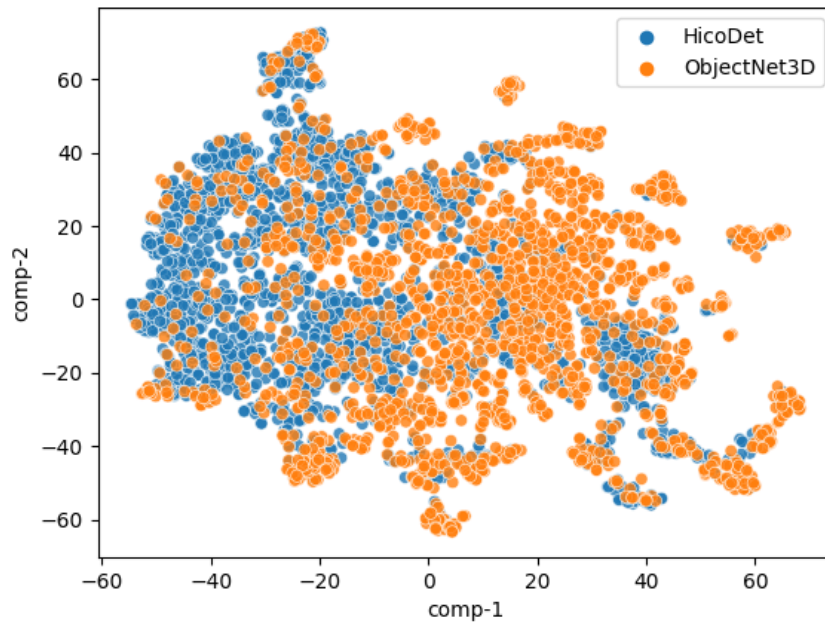


Figure 8.10: t-SNE visualization of ResNet features for 2 000 images each from ObjectNet3D and HICO-DET. While the densities of the two datasets differ significantly, ObjectNet3D finds complete overlap with HICO-DET (in two dimensions) as the number of samples increases.

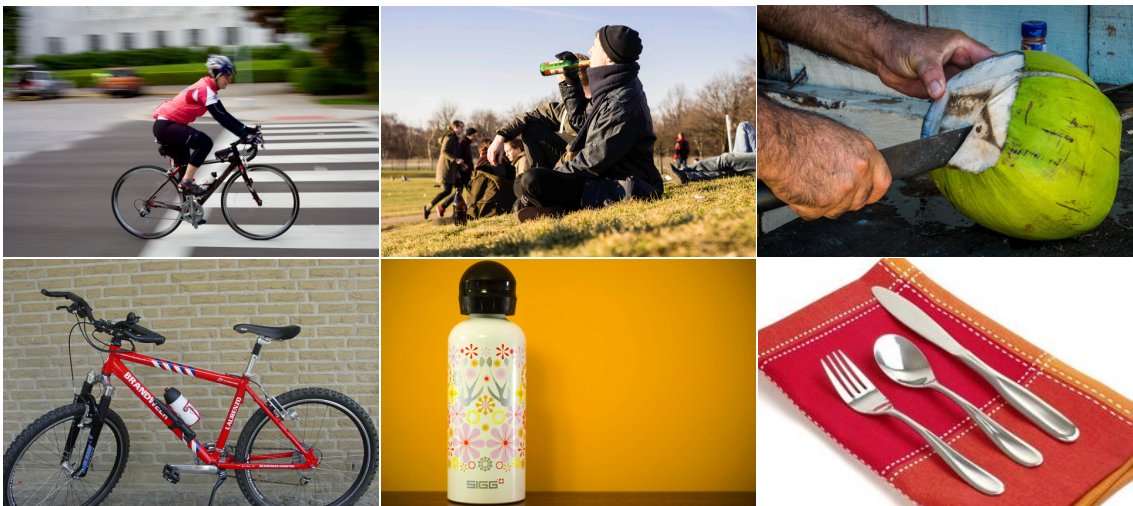


Figure 8.11: Examples from HICO-DET (upper row) and ObjectNet3D (lower row). The selection is biased because there are also images of bikes being ridden in ObjectNet3D, but these are in the minority.

## AffordanceUPT

The optimized hyperparameter are listed in Table 8.6. We used HDBSCAN (McInnes et al., 2017) to determine the main clusters for the t-SNE variant from Figure 8.5 with HDBSCAN. The result is shown in Figure 8.14.

Figure 8.12 and Figure 8.13 show the visualizations of the unary tokens. It can be seen that these probably represent mostly class information. And in Figure 8.13 it can be seen that only from the representation of the person no affordance can be derived either.

Figure 8.15 shows some more examples of boundary box distinguished between AffordanceUPT and HICO-DET.

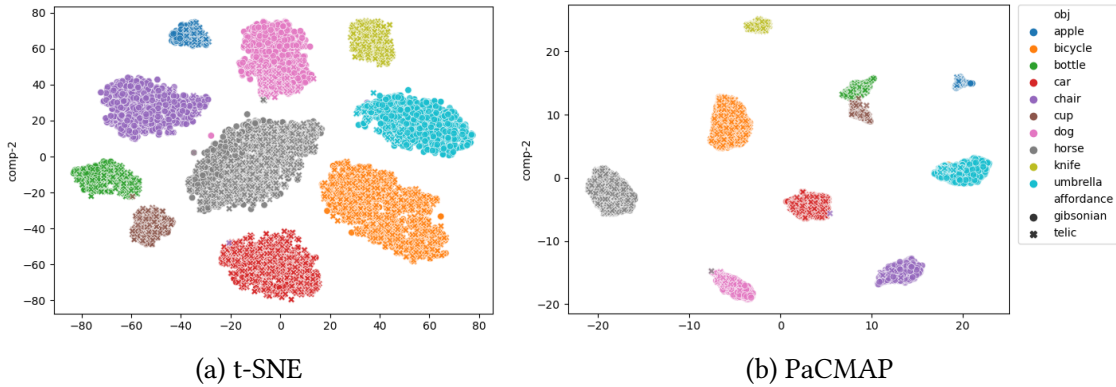


Figure 8.12: UPT Object Unary Token Visualization. Labeled according to their classes and in which affordance was determined for this object.

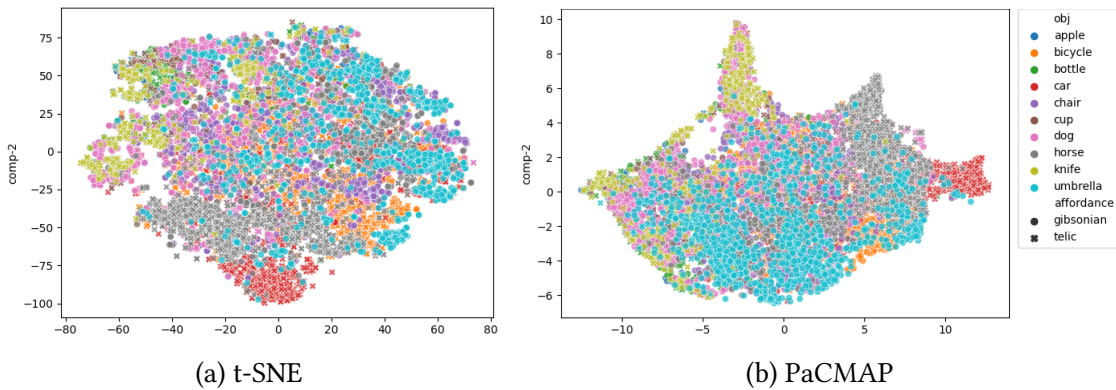


Figure 8.13: UPT Person Unary Token Visualization. Labeled according to their affordance and interacted object.

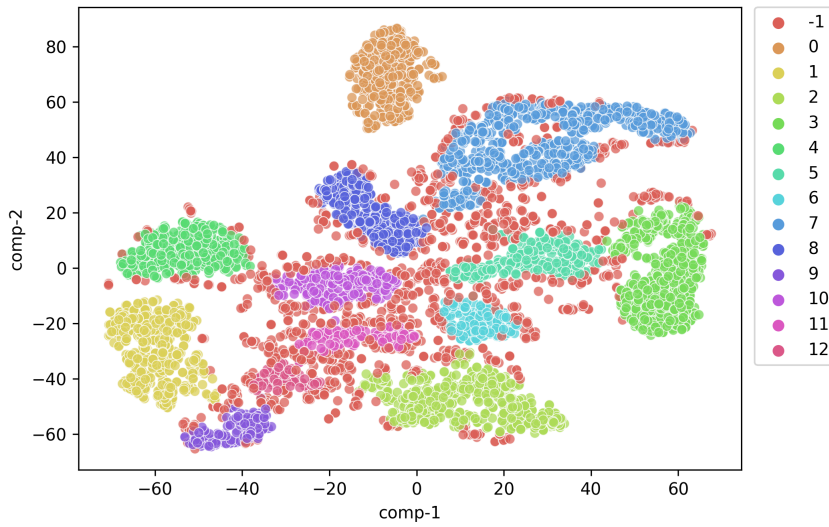


Figure 8.14: HDBSCAN results with *minimal cluster size*: 50 and *maximal cluster size*: 1 000 on Figure 8.5. Homogeneity: 0.603; Completeness: 0.675; V-measure: 0.637; Adjusted Rand Index: 0.436; Adjusted Mutual Information: 0.634. Points with label -1 were classified as noise.

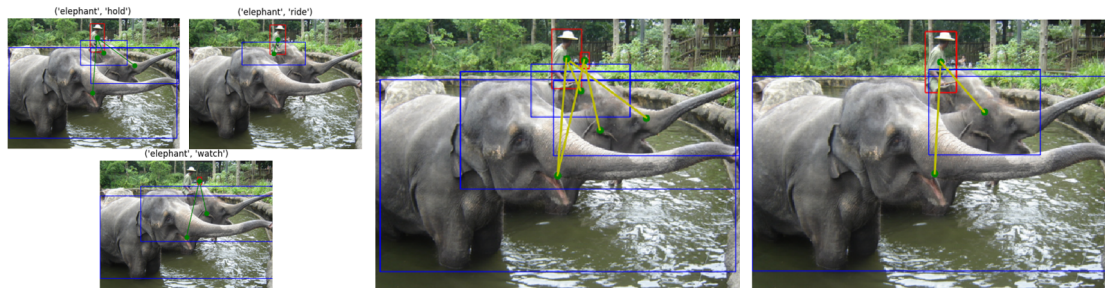
Hyperparameter	Value
learning rate	1.3e-4
weight decay	4.7e-4
learning rate drop	82
gradient clip	0.18
loss alpha	0.25
loss gamma	0.85

Table 8.6: Hyperparameter for AffordanceUPT.

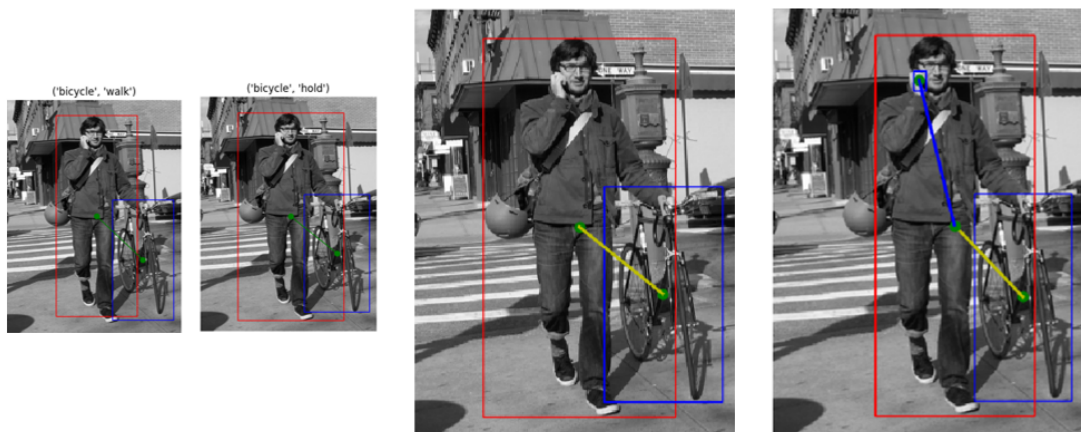




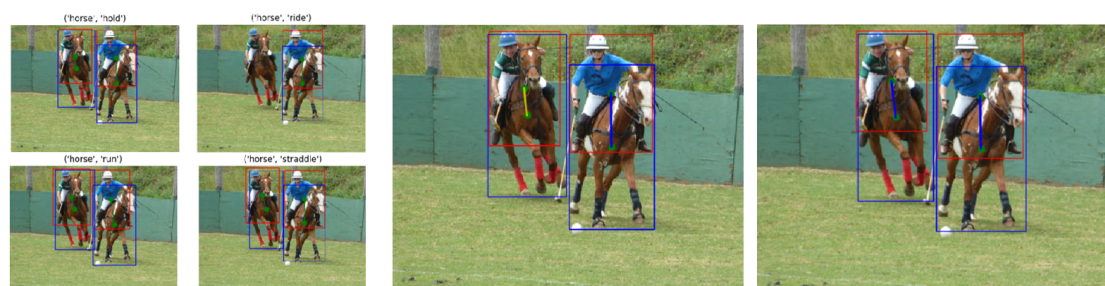
(a) The right person is not part of the HICO-DET annotation.



(b) Different bounding boxes for the same objects, which are unmergable.



(c) Cell phone is not part of the HICO-DET annotation



(d) The rear left horse was not annotated as a “ride”, which is why it shows up in the dataset as Gibsonian.

Figure 8.15: Various error cases of AffordanceUPT based on HICO-DET annotations. The left images are the original HICO-DET annotations, the middle images are the merged variants with Gibsonian (yellow) and telic (blue) connections and the right images are the predictions by AffordanceUPT.

# 9 Semantic Scene Builder: Towards a context sensitive Text-to-3D Scene Framework

Henlein, A., Kett, A., Baumartz, D., Abrami, G., Mehler, A., Bastian, J., Blecher, Y., Budgenhagen, D., Christof, R., Ewald, T.-O., Fauerbach, T., Masny, P., Mende, J., Schnüre, P., & Viel, M. (2023b). Semantic scene builder: Towards a context sensitive text-to-3d scene framework. In *Semantic, artificial and computational interaction studies: Towards a behavioromics of multimodal communication, Held as Part of the 25rd HCI International Conference, HCII 2023, Copenhagen, Denmark, July 23– 28, 2023, Proceedings*: Springer. accepted

## Abstract

We introduce Semantic Scene Builder (S<sub>ESB</sub>), a VR-based text-to-3D scene framework using SemAF (Semantic Annotation Framework) as a scheme for annotating discourse structures. S<sub>ESB</sub> integrates a variety of tools and resources by using SemAF and UIMA as a unified data structure to generate 3D scenes from textual descriptions. Based on VR, S<sub>ESB</sub> allows its users to change annotations through body movements instead of symbolic manipulations: from annotations in texts to corrections in editing steps to adjustments in generated scenes, all this is done by grabbing and moving objects. We evaluate S<sub>ESB</sub> in comparison with a state-of-the-art open source text-to-scene method (the only one which is publicly available) and find that our approach not only performs better, but also allows for modeling a greater variety of scenes.

## 9.1 Introduction

Humans are able to describe visual scenes linguistically and, conversely, to generate visual representations, e.g., in their mind's eye or on a sheet of paper, on the basis of linguistic descriptions (Sadoski et al., 1990; Sadoski & Paivio, 2013). These modality changes require mental capabilities in the area of multimodal fusion and fission (Dumas et al., 2009). From a computational point of view, the second of these capabilities is modeled in terms of text-to-scene systems (e.g. Tan et al., 2019), namely when it comes to generating 3D scenes from text descriptions (e.g. Coyne & Sproat, 2001).

While language-assisted image generation has received a lot of attention recently (e.g.

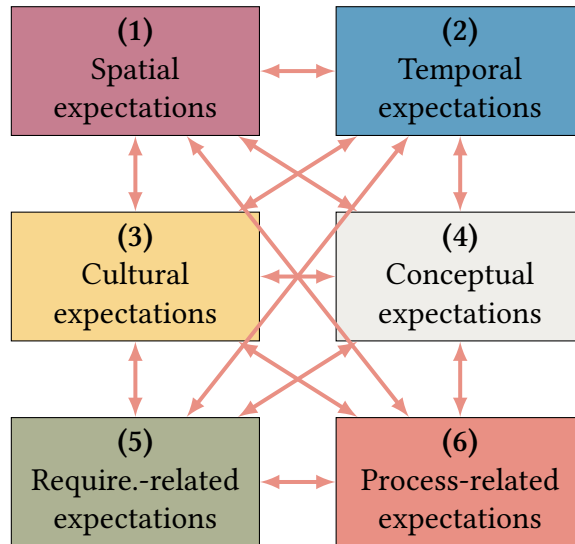


Figure 9.1: Areas of contextual expectations that are relevant for scene generation.

Ramesh et al., 2021, 2022; Saharia et al., 2022; Ding et al., 2022; Alayrac et al., 2022), 3D scene generation from language has not been explored that much (Hassani & Lee, 2016; Ma et al., 2018; Wang et al., 2021a). Image-generating models benefit immensely from the advances in grounded language modeling (like CLIP; Radford et al., 2021) and the sheer amount of data available (cf. LAION-5b<sup>1</sup> which provides 5,85 billion image-text pairs). This amount of data is currently unthinkable for text-to-3D scene applications. There are large-scale 3D object datasets like ShapeNet (Chang et al., 2015b) or 3D-FUTURE (Fu et al., 2021b), and large-scale 3D scene datasets, like Matterport3D (Chang et al., 2017a), 3D-Front (Fu et al., 2021a) or SUNCG (Song et al., 2017, which is currently not available due to license problems). But non of these datasets is annotated with textual descriptions. As a result, recent work increasingly emphasizes generating ever more realistic scenes (c.f., Scene Synthesis; Zhang et al., 2019), where the use of language is increasingly reduced to imposing constraints on the generated scenes so that the alignment of natural language and scenes takes a back seat. This is exemplified by Ma et al. (2018) where language processing does not go beyond pattern matching of dependency trees and keywords. That is, the quality of the generated scenes is primarily achieved via co-occurrence patterns of objects in already modeled scenes.

Yet human language is so versatile and flexible in describing events, often based on elliptical or underspecified constructions, which, however, are perfectly understandable by exploring knowledge shared by speakers and listeners, i.e., their common ground (Clark, 1996; Garrod & Pickering, 2004): Hearing, for example, a sentence like

“After eating my croissant, I read the newspaper.”

the listener is likely to assume that the speaker is describing an event in which he ate a croissant in the kitchen or dining room, that the croissant was eaten with a coffee,

<sup>1</sup><https://laion.ai/>



that the event took place in the morning, etc. But none of these common ground-related expectations are expressed in the sentence – although they are relevant to the imagination of a sufficiently complete scene. The breadth and depth of such expectation-driven understanding of scene descriptions is contrasted with a lack of data that make them explicit and link them to image representations. Figure 9.1 lists ranges of such information implied by scene descriptions concerning

- (1) expectations about **spatial** relations regarding, e.g., the placement of objects (e.g. piece of cake *on* a plate) (Chang et al., 2014b),
- (2) **temporal** relations and epoch-related expectations (e.g. a medieval kitchen compared to today) (Baden-Powell, 2006),
- (3) **cultural** expectations (e.g. a classic German vs. a French breakfast) (Gibney et al., 2018),
- (4) expectations about **conceptual** relations and object affordances (e.g., chairs are for sitting) (Pustejovsky & Krishnaswamy, 2016),
- (5) **requirements-related** expectations (e.g. eggs are needed to make omelets) (Sap et al., 2019),
- (6) and **process-related** expectations (in terms of what has been processed so far or will likely happen next) (Pustejovsky et al., 2005a).

These domains are interrelated and give rise to complex expectations about, e.g., courses of events (Anderson, 1983). Thus, there is a large body of work dealing with descriptive models of contexts (e.g. Mainwaring et al., 2003; Neumann & Möller, 2008; Marszalek et al., 2009; Oliva & Torralba, 2007; Dennerlein, 2009; Tosi et al., 2020).

Much has been done to generate realistic scenes, but the range of linguistic descriptions of such scenes is far from exhausted. We argue that this is mainly due to the lack of available data and a computational framework for its generation, processing and maintenance. We present *Semantic Scene Builder* (SESB), a VR-based<sup>2</sup> text-to-3D scene framework to fill this gap. Its interactive approach, based on VR and a unified data model, allows the system to be used for every step of text-to-3D scene generation, from annotation of data to integration of individual specialized tools to complete end-to-end models. For each of its processing steps we implemented 1-3 modules based on state-of-the-art tools, including a self-trained BERT (Devlin et al., 2019) model for extracting spatial entities and relations, and a dataset for processing associations between actions and objects. For evaluation, we generated scenes with SESB and with the system of Ma et al. (2018) and compared them regarding two criteria: naturalness and plausibility.

The paper is structured as follows: Section 9.2 describes current text-to-3D scene systems and their limitations. We review the range of linguistic variants of scene descriptions and outline IsoSpace and SemAF. Section 9.3 presents the functionality of SESB

---

<sup>2</sup>VR here stands for *fully-immersive virtual reality*, supported by hand tracking and head-mounted displays (Riva, 2006).

and Section 9.4 its implementation. Section 9.5 evaluates S<sub>ESB</sub> in comparison to a state-of-the-art text-to-3D scene system. Section 9.6 describes future work and Section 9.7 gives a conclusion.

## 9.2 Related Work

### 9.2.1 Text-to-3D Scene

One of the first successful text-to-3D scene systems is WordsEye (Coyne & Sproat, 2001). To date, thanks to various additions, it is one of the linguistically most comprehensive systems (Hassani & Lee, 2016). The basic version already supported representations of actions, avatars, negations and of proverbs. This functionality was later extended by means of frame semantics (VigNet, Coyne et al., 2011) and SpatialNet (Ulinski et al., 2019), a hand-annotated resource for spatial relations. These manually created and non-open-source resources allow to disambiguate and resolve ambiguous prepositions and verbs.

Another well known text-to-3D scene system is that of Chang et al. (2014a,b, 2015a), meanwhile referred to as SceneSeer (Chang et al., 2017b). It creates a scene  $s$  given an utterance  $u$  using a conditional probability:

$$\begin{aligned} P(s|u) &= P(t|u)P(t'|t, u)P(s|t', t, u) \\ &= P(t|u)P(t'|t)P(s|t') \end{aligned} \tag{9.1}$$

That is,  $P(s|u)$  is decomposed into the product of the parsing probability  $P(t|u)$ , the inference probability  $P(t'|t)$  and the generation probability  $P(s|t')$ .  $t$  stands for the scene template given utterance  $u$ , while  $t'$  is the completed scene template. This model assumes that  $s$  is independent of  $t$  and  $u$ , and  $t'$  is assumed to be independent of  $u$  – a weakness that Chang et al. (2017b) already noted, but most systems retain to this day. There is also a transformer-based (Vaswani et al., 2017) end-to-end approach (SceneFormer; Wang et al., 2021a), but the text-conditioned model has not been published yet.

In the *parsing* step,  $u$  is preprocessed and the objects and relations mentioned in  $u$  are mapped to elements of the scene template  $t$ . In the *inference* step, objects and constraints implied by  $t$  (and optimally from  $u$ ) are inferred to generate the expanded template  $t'$ . This is done using coincidence probabilities learned *a priori* from spatial datasets. Finally, in the *generation* step, the output scene  $s$  is produced starting from  $t'$ .  $s$  can be adjusted by the user to allow the system to continue learning.

Few systems actively use external language resources in addition to pre-configured rooms to enable more diverse language inputs (Hassani & Lee, 2016). At first glance, these systems generate very realistic scenes from scene descriptions, but are rather application scenario specific. Thus, while they learn, e.g., from large 3D corpora (like, SUNCG (Song et al., 2017) and 3D-FRONT (Fu et al., 2021a)) how a kitchen is typically set up and that a pan is usually placed on the stove, the same is not true for expressions that express ambiguous linguistic relations (Herskovits, 1986; Feist & Gentner, 1998). Take the following examples

- (1) “I am *on* the wall.”
- (2) “The mirror is *on* the wall.”
- (3) “I am *on* the airplane”.

which illustrate three different meanings of *on*<sup>3</sup>. WordsEye, for example, tries to resolve such ambiguities by means of SpatialNet. However, the underlying system is not open source and thus not extensible. Such problems can be solved with the help of IsoSpace (see below).

There are many other systems that address the creation of 3D scenes, which are based, e.g., on images (Kermani et al., 2016), relation patterns (Zhao et al., 2016) and scene categories (Li et al., 2019). Since we focus on text-based scene generation, we do not consider this approach. The same applies to text-based generations of avatar movements (Petrovich et al., 2022) or 3D objects (Chen et al., 2018).

Value	Description	Example
DC	disconnected	[Europe] - [America]
EC	externally connected	the [book] on the [table]
PO	partial overlap	the [light switch] on the [wall]
EQ	equal	[The White House] - [1600 Pennsylvania Avenue]
TPP	tangential proper part	the [windows] of the [house]
NTTP	non-tangential proper part	the [heart] of the [city]
IN	disjunction of TTP and NTTP	the [table] in the [room]

Table 9.1: The list of RCC8+ (*Region Connection Calculus*) relations (ISO, 2020).

### 9.2.2 SemAF & IsoSpace

IsoSpace (Pustejovsky et al., 2011a; ISO, 2020) is part of the *Semantic Annotation Framework* (SemAF; Ide & Pustejovsky, 2017, p. 128), “an annotation scheme for the markup of spatial relations, both static and dynamic, as expressed in text and other media” (Ide & Pustejovsky, 2017, p. 989). SemAF is a further development of *Spatial Role Labeling* (Kordjamshidi et al., 2010, 2011) and consists of two main components: *entities* marked directly in the text and *links* that relate these entities to each other. According to IsoSpace, entities are divided into

- (i) spatial entities, such as objects, persons or places,
- (ii) signal words, mostly prepositions,
- (iii) events, mostly verbs, and
- (iv) measures.

Entities can be provided with attributes (e.g., *type*, *dimensionality* and *cardinality*) and linked by

<sup>3</sup><https://dictionary.cambridge.org/dictionary/english/on>

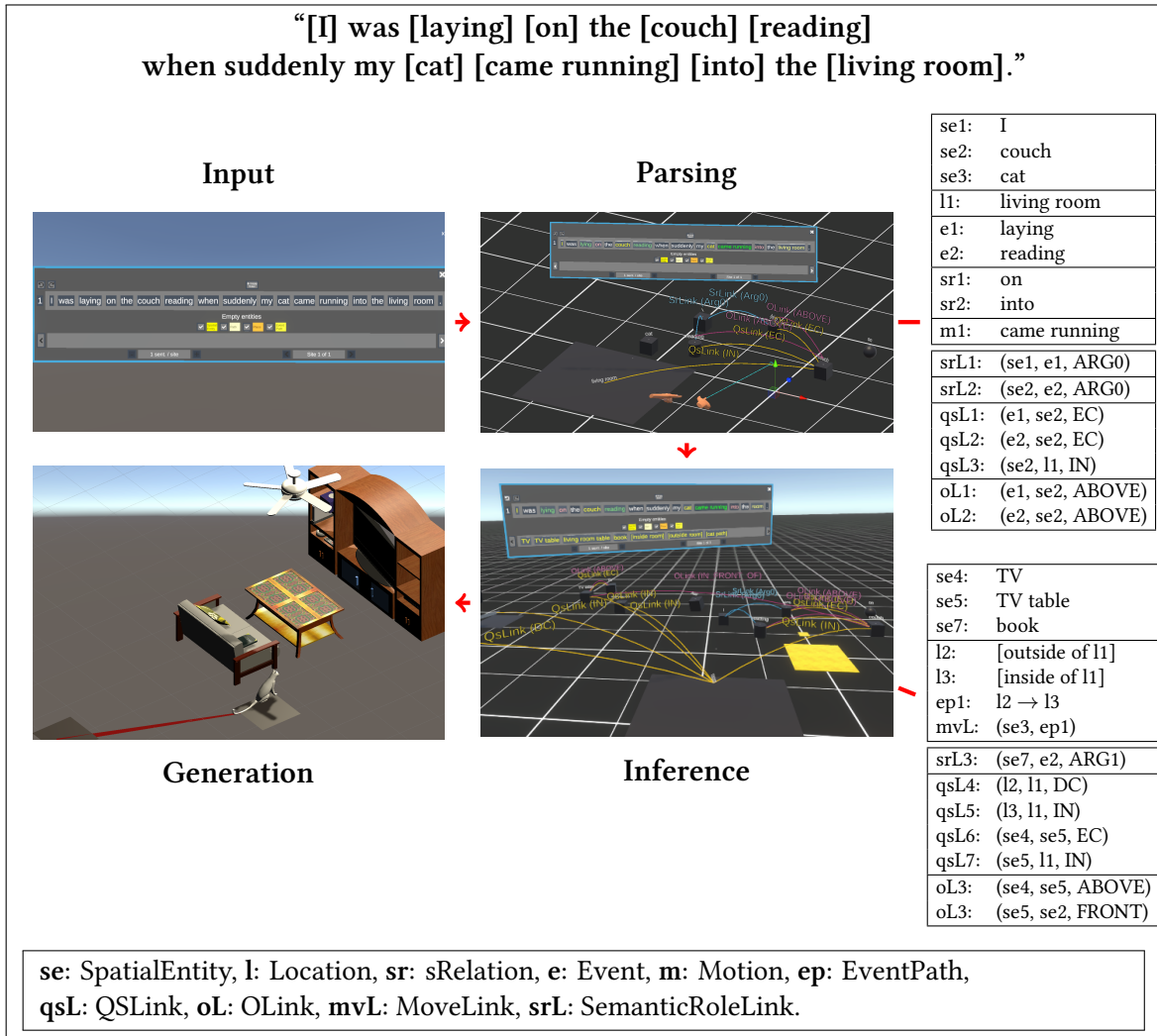


Figure 9.2: Example of an IsoSpace-based, user-supported text-to-3D scene result generated with SESB (not all attributes are displayed as, e.g., trigger references to the sRelations for the IsoSpace links; the list of IsoSpace links is only displayed in part). The images show the complete text-to-3D scene process, from the initial scene description to the final scene. Start and end points of the *cat* were added manually by the user during the processing steps. In the last image (Generation), the links have been omitted for clarity. Main points: (1) Entities not explicitly mentioned in the text are added [e.g. se4 and se5]. (2) Implicit Semantic Role Labeling [srL3] is annotated. (3) Entity movements are also annotated. For this purpose, an EventPath is created that describes the path [ep1] and is linked to the moving entity via a MoveLink [mvL1]. (4) Beyond entities, events are linked using IsoSpace links. For dynamic scenes, this allows the acting entity to not be directly linked to objects and therefore perform actions in different areas [qsL1, oL1]. (5) Though not being shown in this example, SESB allows for annotating (sub-)coreferences and partonymy relations using MetaLinks.

- (i) *Qualitative Spatial Links* (QSLinks, representing topological RCC8+ relations (Randell et al., 1992); see Table 9.1),
- (ii) *Orientation Links* (OLinks, denoting all other spatial relations; e.g. *in front of*, *north*, *across*),
- (iii) *Movement Links* (MoveLinks regarding movements of entities in space), and
- (iv) *Measure Links* (MLinks, used to represent sizes and distances, e.g., *2m*, *4l*).

We also use the SemAF specifications for semantic roles (SrLinks, ISO, 2014b) and coreference annotation (MetaLinks, ISO, 2014a). In the case of MetaLinks, we use the type value *Part* to indicate meronymy and holonymy relations, a relation not covered by IsoSpace by default. The links distinguished so far allow, for example, for resolving ambiguous prepositions:

- (1) “I am *on* the wall.”  
→ QSLink(EC), OLink(ABOVE)
- (2) “The mirror is *on* the wall.”  
→ QSLink(EC), OLink(FRONT)
- (3) “I am *on* the airplane”.  
→ QSLink(IN)

For a second example of using SemAF for disambiguation see Figure 9.2.

IsoSpace does not claim to cover all aspects of spatial language; however, it is still the most comprehensive annotation model of this sort. IsoSpace is not limited to spatial descriptions but can also be used for representing tasks such as (ISO, 2020):

- (a) creating routes based on route descriptions;
- (b) tracking moving objects based on motion descriptions;
- (c) conversion of viewer-centered descriptions into other-oriented descriptions or descriptions based on absolute coordinates.

Related work concerning the transfer of IsoSpace into VR environments is presented by Henlein et al. (2020). However, this work focuses primarily on annotations, while we focus on their application for text-to-3D scene systems.

## 9.3 Semantic Scene Builder

To date, there is no extensible open source system that integrates comprehensive SemAF-related annotations to generate data for training text-to-3D scene systems. We propose SESB, to fill this gap. SESB implements the following features:

1. Its SemAF-based data model allows for modeling a wide range of semantic relations. In this way, not only spatial objects but also events and movements can be represented.
2. S<sub>ESB</sub> is not distributed across a number of heterogeneous, proprietary, barely interoperable systems. Rather, S<sub>ESB</sub> integrates its functionality into a single system that is freely accessible and extensible.
3. Users maintain control at all times and can adjust scene generation according to their needs. Changes to data representations can be made in the course of scene generation, so that S<sub>ESB</sub> can be used as an active learning environment (Settles, 2009).

S<sub>ESB</sub> stores all data (e.g., input texts, entity labels, links, object placements) in a UIMA-based (Ferrucci & Lally, 2004) XMI format based on SemAF. We now describe the modules of S<sub>ESB</sub> by distinguishing four steps of generating text-to-3D scenes (for the first three steps cf. Section 9.2.1), that is, *parsing*, *inference*, *generation* and *annotation*. In this way, we stepwise generate a scene  $s$  starting from a given scene description  $u$ .

### 9.3.1 Parsing

For preprocessing of scene description  $u$  (including, e.g., tokenization, sentence splitting, POS tagging and lemmatization) we use the Stanza (Qi et al., 2020) interface to CoreNLP (Manning et al., 2014). For semantic role labeling, we use the AllenNLP (Gardner et al., 2018) implementation of Shi & Lin (2019).

We use a two-step approach for detecting objects and their relations in  $u$ : first we extract entities and links with IsoSpaceSpERT (see below), and then augment and correct them with a rule-based model.

#### Rule-based Model

As a baseline, we reimplemented the rule-based approach of Ma et al. (2018). That is, anything recognized as a noun by POS tagging is labeled as a spatial entity. Attributes and relations are assigned using hand-generated dependency rules, which we transferred to the QS-/OLink schema: e.g. “The cat is in front of the table.” is mapped onto (cat, in\_front\_of, table) and OLink(cat, table, front).

#### IsoSpaceSpERT

Since no open source models are yet available for IsoSpace tagging (Shin et al., 2020; Nichols & Botros, 2015; D’Souza & Ng, 2015; Salaberry et al., 2015), we trained IsoSpaceSpERT based on SpERT (Eberts & Ulges, 2019), which in turn is based on BERT (Devlin et al., 2019) using the data from SpaceEval (Pustejovsky et al., 2015). We also conducted experiments with REBEL (Huguet Cabot & Navigli, 2021) and PL-Marker (Ye et al., 2022). However, both models performed significantly worse – probably due to the limited amount of training data.

	SpRL-CWW	S-BERT	Prec	Rec	F1
Place	74.7	86.8	81.4	82.6	82.0
Path	61.7	94.9	81.0	76.4	78.7
Spatial entity	80.8	89.9	79.7	87.6	83.5
Motion	76.9	94.3	79.2	87.4	83.1
Motion signal	78.6	90.7	84.4	88.8	86.6
Spatial signal	70.9	85.9	76.1	83.6	79.6
Measure	79.1	98.3	88.6	91.2	89.9
Non-motion	56.4	89.4	59.4	63.3	61.3
average	74.6	90.0	78.7	82.6	80.6
QSLink	-	-	58.2	40.8	48.0
OLink	-	-	35.5	32.4	33.9
average	3.0*	-	46.9	36.6	41.0

Table 9.2: Evaluation of IsoSpaceSpERT. \*: Evaluated on (figure, ground, trigger, rel\_type) and MoveLinks.

Since no link annotations were published for the SpaceEval test data, we added QSLink and OLink annotations, arriving at 92% of the data reported in the official statistics for each of these types (all these data are made publicly available via this publication). For annotation we used the Multi-purpose Annotation Environment (Stubbs, 2011) that was originally used for generating the SpaceEval data<sup>4</sup>. We deleted files in the test data that also appeared in the training data and did not consider empty entities and their links.

We trained IsoSpaceSpERT to detect the type of spatial relation that connects entities: e.g., for QSLinks RCC8+ relations are tagged. Note that S-BERT (Shin et al., 2020) and SpRL-CWW (Nichols & Botros, 2015) focus on finding relation triples (figure, ground, trigger) and therefore do not predict relation types. In addition, S-BERT only considers relations manifested by prepositions. These differences make it difficult to directly compare our results with previous work on QSLink and OLink detection.

Table 9.2 shows the results of evaluating IsoSpaceSpERT. The results of hyperparameter optimization using wandb (Biewald, 2020) are shown in Table 9.3 (appendix). We trained separate models for QSLinks and OLinks, as this resulted in significantly better link detection results, even though entity detection benefits from joint training. The results for entity detection are generated by the QSLink model. We achieve an F1-Score of 41% for QSLinks (48%) and OLinks (33.9%). There is one dominant error due to our model architecture: since the prediction of relations between two entities only takes into account the context that lies between them, the model has problems with statements like “On  $x$  is  $y$ ”, where the preposition is to the left of  $x$  and  $y$ .

<sup>4</sup><http://jamespusto.com/wp-content/uploads/2014/07/SpaceEval-guidelines.pdf>

### 9.3.2 Inference

SESB contains two mechanisms for inferring contextual information for expectation-driven understanding (see Figure 9.1). This relates to aspects of spatial and process-related expectations.

#### Spatial or Room-related Inference

To exploit knowledge about rooms, we estimate the conditional probability  $P(r \mid o)$  of a room  $r$  given an object  $o$ .  $P(r \mid o)$  estimates, e.g., the probability of objects like bathtubs being typically located in bathrooms. To this end, we use the NYU Depth V2 dataset (Silberman et al., 2012). It contains 464 labeled real-world scenes and nearly 900 different object labels, and therefore significantly more than, e.g., COCO (Lin et al., 2014) with only 80 object categories. Using NYU Depth V2 to estimate  $P(r \mid o)$ , we add the five most strongly associated objects to the scene. Note that using  $P(r \mid o)$  to determine these associations generated the better results, because of filtering out uninteresting objects (like ceiling or curtain), while selecting objects that are interesting for a room type (e.g. bed  $\rightarrow$  bedroom). This is done for each room label occurring in the input scene description  $u$  detected by the parsing module.

#### Process- or Task-related Inference

To insert objects inferred from described actions into the scene  $s$ , we use a version of HowToKB (Chu et al., 2017). HowToKB represents task-related knowledge along with attributes for the parent task, the preceding and the succeeding subtask. This knowledge is extracted from WikiHow<sup>5</sup> articles by means of OpenIE (Etzioni et al., 2011). HowToKB also contains information about tools and objects required to perform a task, if they are explicitly listed in a separate section of the original article.

We created a new crawl of WikiHow and updated the whole pipeline based on HowToKB. We also performed WordNet-related (Miller, 1995) *Word Sense Disambiguation* (WSD) using LMMS (Loureiro & Jorge, 2019; Loureiro & Camacho-Collados, 2020) and updated task clustering to include disambiguation as well. Using WSD, we expanded the number of objects extracted from WikiHow articles and added to task-specific lists of required objects (increasing the amount of labeled *objects involved* from 1.4M to 2.2M). To process queries, we imported HowToKB into Neo4J.<sup>6</sup>

For each event  $e$  described in the input description  $u$ , we then search our extended knowledge base for all object-event combinations that contain  $e$ . For each such entry, the corresponding object is finally inserted into the scene  $s$  if it is missing there.

In future work, we will experiment with systems such as COMET-ATOMIC 2020 (Hwang et al., 2021), ConceptNet (Speer et al., 2017) or TransOMCS (Zhang et al., 2020) to provide SESB as an application for evaluating commonsense systems.

<sup>5</sup><https://www.wikihow.com/>

<sup>6</sup><https://neo4j.com/>



### 9.3.3 Generation

The next step is to select 3D objects for all the objects detected so far and place them in the scene  $s$  in a meaningful way (taking into account constraints mentioned in  $u$ ). For this purpose we use the scene generation tool of Ma et al. (2018). This tool creates a scene from all extracted objects and extends it with objects and relations that are still missing (e.g., a given plate implies a table on which it is placed). The models that Ma et al. trained for this purpose were created using various annotated scene resources, that is, SUNCG (Song et al., 2017), SceneSynth (Fisher et al., 2012) and SceneNN (Hua et al., 2016). This includes

- (1) the **support model** for adding matching supports of objects (e.g. a plate as a support under a piece of cake).
- (2) the **co-occurrence model** for adding relevant objects based on co-occurrence probabilities (e.g. a mouse next to keyboard).
- (3) the **pairwise model** to predict the relative positioning between two objects.
- (4) the **group model** for handling with group relations (e.g. “messy table”).
- (5) and the **relative model** to handle conflicts between explicit relations specified in the input record and implicit relations specified by existing objects.

This approach is originally based on Fisher et al. (2012). It shows that modules do not have to strictly adhere to the succession of parsing, inference, and generation, since objects and relations can also be inferred and added during generation.

### 9.3.4 Annotation

Based on VR, SESB allows users to make changes at each processing step. These can be changes to the final scene by grabbing objects and repositioning, rotating, or scaling them. It may also concern the placement of inferred objects into the scene or deleting them. Furthermore, it is possible to interact with the input text  $u$  by means of a text window to annotate entities, set their attributes or insert links between them. In this way, the user has full access to SESB’s data structure. In this way, SESB provides a 3D, VR-based annotation environment for SemAF and 3D scenes.

## 9.4 Implementation

SESB is based on VANNOTATOR, which builds on TEXTANNOTATOR<sup>7</sup>. SESB is implemented in Unity3D<sup>8</sup> and can be used by means of 3D glasses. VANNOTATOR creates a

<sup>7</sup>VANNOTATOR, TEXTANNOTATOR are TEXTIMAGER are synonyms to comply with the guidelines for author anonymity.

<sup>8</sup><https://unity.com/>

virtual 3D environment in which scenes can be visualized and modified, with both operations provided by TEXTANNOTATOR. TEXTANNOTATOR is a platform-independent, WebSocket-based multi-user annotation framework which enables collaborative, simultaneous annotations based on UIMA (Ferrucci & Lally, 2004). Thus, different users can annotate the same scene  $s$  at the same time. To this end, scenes are modeled as UIMA documents, which are annotated with TEXTANNOTATOR. Any change to a scene (e.g., by creating, moving, scaling, texturing, or relating objects) is interpreted as an annotation instruction that is communicated to each annotator of the same scene to update her or his view. To this end, all representations of object, their attributes and relations are modeled as annotation objects.

Since the representation of 3D objects is an essential part of scene generation, we use ShapeNetSem (Savva et al., 2015), a sub-project of ShapeNet (Chang et al., 2015b), to visualize 3D objects. Through ShapeNetSem, it is possible to access 12 000 semantically annotated objects, which allows SESB to create and annotate a wide range of concrete, visualizable objects in addition to abstract objects such as cubes, planes and spheres.

All tools from Section 9.3 are included into VANNOTATOR via a Python implementation of TEXTIMAGER and work directly on UIMA documents; this is enabled by means of dkpro-cassis (Klie & de Castilho, 2020).

## 9.5 Evaluation

We compare SESB with the system of Ma et al. (2018), the only related system that is freely available. We used both systems to generate scenes from 21 different scene descriptions. These descriptions each contain 1-3 sentences from the following three categories:

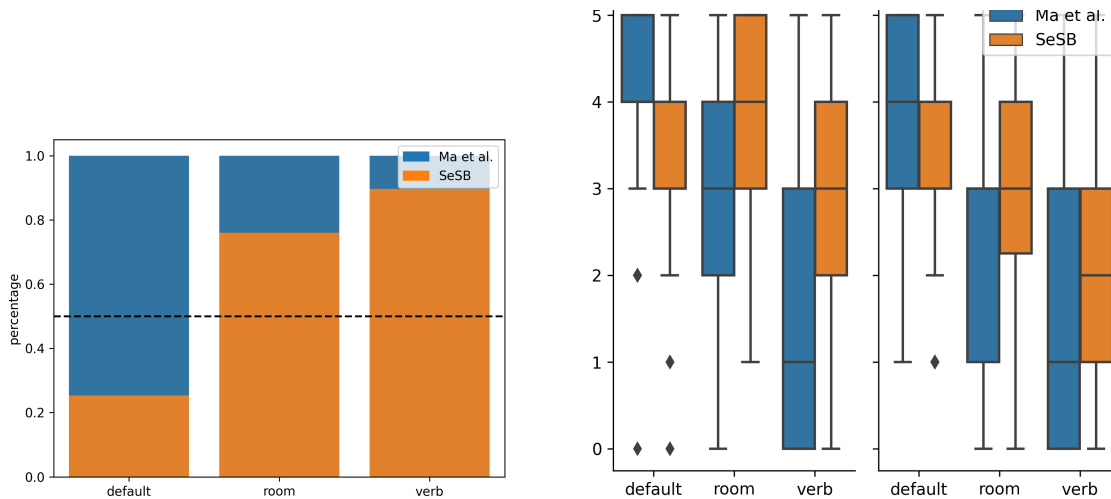
- (a) original descriptions from the appendix of Ma et al. (2018),
- (b) room name-based descriptions (e.g. “I ate an apple in the kitchen.”), and
- (c) action-based descriptions (e.g. “I like to make music.”).

Examples of the generated scenes can be found in the appendix (Figure 9.5).

The annotators employed for our comparative evaluation were assigned a three-part task: each annotator was asked to determine for each pair of images the better scene representation produced either by Ma’s approach or by SESB. Furthermore, each of the images was to be assigned a value between 0 and 5 for *naturalness* and *plausibility*. This approach follows the evaluation method of Ma et al. (2018). Which image of which system was displayed on which side of the screen was randomly selected; however, the images shown always referred to the same input description. The results of our evaluation are shown in Figure 9.3a and 9.3b; they are based on a total of 22 participants.

Figure 9.3a shows that SESB performs slightly worse for concrete spatial descriptions. But when the spatial description is tagged with a room label or an action is described, scenes generated by SESB are clearly preferred by annotators. We hypothesize that the initially poorer results are due to the integration of additional systems (such as SpERT

in particular) that interfere with room generation, as these systems produce increased noise. However, this integration allows SeSB to ultimately process more complex input texts (as Figure 9.5 (appendix) shows). Our findings are also reflected in the naturalness and plausibility ratings (Figure 9.3b), where we perform slightly worse regarding concrete descriptions, but better in the other two scenarios. We hypothesize that the plausibility of SeSB was rated somewhat lower than the naturalness of its action representations because we do not yet have the data to place the objects involved appropriately. That is, although objects are generally placed meaningfully in a room according to the annotators’ ratings, they are not necessarily always relevant to the action being described.



(a) Comparative evaluation of Ma et al.’s (2018) approach and SeSB. (b) Rating of generated scenes regarding their naturalness (left) and plausibility (right).

## 9.6 Discussion & Future Work

While it is possible to create 3D scenes from less constrained or more natural descriptions using SeSB, the possibilities offered by natural languages for scene descriptions are far from exhausted. This becomes clear when looking at Figure 9.1. While the system of Ma et al. (2018) covers aspects (1) and (4), we extended aspect (4) by including additional spatial room concepts and partially consider aspect (5) as well. However, with the restriction that the positioning of objects is not conditioned by the described activity. Aspects (2) and (3) of Figure 9.1 are still not considered. While considering these two aspects could generate a manageable amount of work, the true complexity comes from combinations of the aspects (1-6):

“A person listens to music in the 50s.”

Starting from the token “listens” (aspect 5) and the time expression “50s” (aspect 2), a tube radio or a record player (aspect 4) seems more likely as the instrument involved.

This instrument is then more likely to be found in a living room (aspect 1), while it can vary greatly in design (aspect 3) depending on the assumed region of the speaker (aspect 1). Obviously, text-image systems also have problems with such examples (Marcus et al., 2022), regardless of the ever-increasing training datasets available to them. This is exemplified in the appendix (Figure 9.4a and 9.4b) by means of DALL-E Mini (Dayma et al., 2021)<sup>9</sup>. Approaching this complexity will be part of future work and is unlikely to be realistically accomplished without active learning and far more sophisticated approaches to human computation (McClelland et al., 2019; Bisk et al., 2020; Kumar, 2021).

## 9.7 Conclusion

We presented *Semantic Scene Builder* (SESB), a VR-based text-to-3D scene framework that generates 3D scenes based on scene descriptions. It uses SemAF and UIMA as underlying data structures and integrates a wide range of resources such as HowToKB and IsoSpaceSpERT to cover more complex scene descriptions. By enabling annotations in VR and the expressive power of SemAF that SESB covers, SESB is usable to generate training corpora for text-to-3D scene systems. This is important because this area of language understanding is still in its early stages and relevant data sets are therefore rare. We evaluated SESB against a state-of-the-art open-source text-to-scene method (the only one publicly available yet) and found that our approach not only performed better, but also allowed us to model a wider variety of scenes.

---

<sup>9</sup><https://huggingface.co/spaces/dalle-mini/dalle-mini>

## Appendix

Parameter	QSLink-SpERT	OLink-SpERT
BERT	bert-base-cased	
Epochs	30	
Batch size	10	
Negative entity count	300	320
Negative relation count	80	15
Learning rate	6.0e-5	6.3e-5
Weight decay	0.0082	0.0085
Relation filter threshold	0.44	0.23
Size embedding	60	45

Table 9.3: IsoSpaceSpERT Hyperparameter



(a) Images generated by DALL-E Mini for the sentence: “After eating my croissant, I read the newspaper”.



(b) Images generated by DALL-E Mini for the sentence: “A person listens to music in the 50s”.

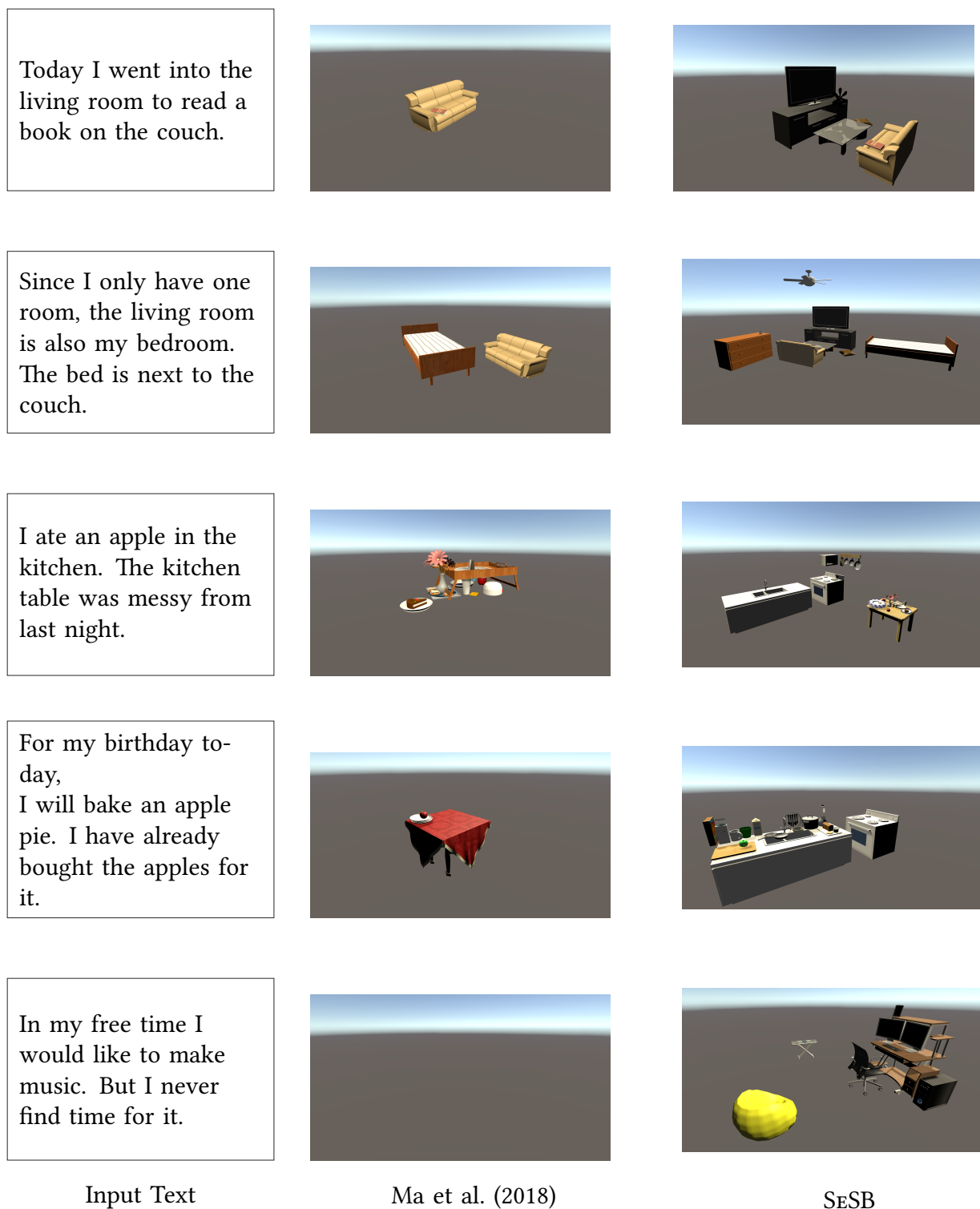


Figure 9.5: Generated examples scenes from the evaluation.

# 10 Conclusion

## 10.1 Summary

The contributions of this dissertation can be divided into four main points.

**Analysis & Evaluation of Language Models** We were able to show that static word embeddings do not significantly benefit from pronoun substitution as a reprocessing step and explained the results that we ended up with exactly what we were trying to prevent with the approach: the loss of contextual information.

On the other hand, we were able to show that purely text-based models contain knowledge about object relations. The various models (whether static or contextualized) differ significantly in how much knowledge they allow to extract, with the static models actually performing sometimes better than the transformer-based contextual models.

**Grounding of Human-Object Interactions** We introduced a self-annotated extension of HICO-DET to include Gibsonian and telic affordances. On this data, we trained a variant of UPT adapted by us, named AffordanceUPT, and could show for this model that it can effectively distinguish between Gibsonian and telic affordances in images. AffordanceUPT also learns other correlations in the data to make such distinctions, which are important for grounding these objects.

**SESB** We presented SeSB, a VR and SemAF-based text-to-3D scene system, and its corresponding preliminary work. The system supports both the annotation of semantically expressive spatial data via IsoSpace and the automatic generation of 3D scenes. For the latter, several modules have been improved and implemented that allow solving both spatial and requirement-related contexts and evaluated against another state-of-the-art tool. We were able to show that our approach not only performs better but also allows the modeling of a wider variety of scenes.

**VR as a Tool for Digital Learning** We reviewed existing practices and tools for digital and virtual teaching and learning based on derived rules and were able to show that the possibilities of VR are far from exhausted, as most applications only attempt to emulate reality rather than build and improve on it. Complementing the existing tools, we show the possibilities in the area of virtual and three-dimensional teaching and learning environments using the example of VANNOTATOR.

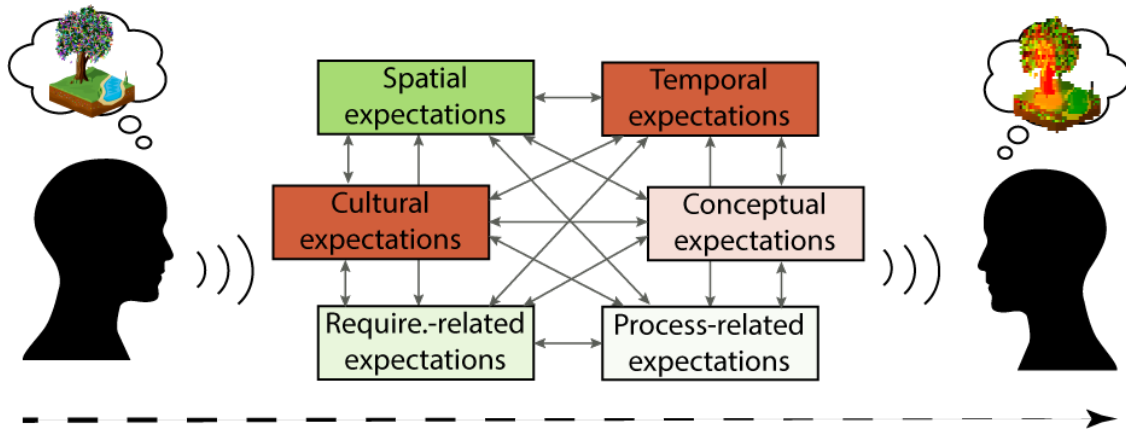


Figure 10.1: Adaptation of Figure 1.1 for future work. Green stands for contexts that have already been solved well, and the redder the box, the more problematic it is.

## 10.2 Future Work

With models such as SESB and AffordanceUPT, we have created systems that will enable even more realistic text-to-3D scene generation in the future and are already showing initial success with the resources that have been integrated. But Figure 10.1 shows that we are still a long way from the quality that humans are capable of.

### 10.2.1 Spatial expectations

Standing alone, this is probably the best-solved point of all. Scene Synthesis models are already very good at generating realistic scenes, based only on the spatial relationships between objects (cf. Li et al., 2019; Wang et al., 2021a). However, these are still very domain restricted. SceneFormer (Wang et al., 2021a) only supports bedrooms and living rooms and this also applies to other work that supports outdoor scenes (e.g. Karacan et al., 2016; Yang et al., 2021). And there is still the problem of actively incorporating these models into text-to-3D scene pipelines, as it is not simply enough to create a realistic kitchen. The kitchen must also correspond to the description from the text, which is why we still resort to such basic systems as that of Ma et al. (2018).

### 10.2.2 Temporal expectations

The time factor is not yet considered in this work. By implementing SemAF as a data structure it should not be a problem in the future to extend the function by IsoTimeML (see Section 2.1). Systems such as HeidelTime (Strötgen & Gertz, 2013; Kuzey et al., 2016), for example, can then be used to mark the temporal relationships within a text. This would then also make it possible to hide objects that are no longer part of the scene, or are yet to become part of the scene, or to display scene changes over time.



### 10.2.3 Cultural expectations

The cultural aspect has also not yet been addressed. To our knowledge, there is simply not enough data to expand on this aspect. The first step would probably be to build an interior scene database that maps kitchens, living rooms, bedrooms, etc. in different cultural spaces and label them as well. Accordingly, the data sets with 3D objects, such as ShapeNetSem, would also have to be adapted so that appropriately labeled 3D objects also appear there (so that, for example, not only beds but also sofa beds are included and annotated).

### 10.2.4 Conceptual expectations

For this point, the ground work was done with Henlein et al. (2023a) and AffordanceUPT. In the future, these analyses need to be extended to a larger scale and tested for wetting subdivisions other than Gibsonian and Telic. In addition, some work needs to be done on object orientation detection in images with human-object interactions. This should later make it possible to generate large-scale affordances and habitats and thus VoxML descriptions for 3D objects. Grounded language models such as CLIP could also help with this (Radford et al., 2021).

### 10.2.5 Requirements-related expectations

To this end, we have shown preliminary work with the update, improvement, and expansion of HowToKB (Chu et al., 2017). However, it needs to be evaluated in more detail and provided with further filtering mechanisms to filter out unwanted objects. There are also many more “common sense reasoning datasets”, such as: COMET-ATOMIC 2020 (Hwang et al., 2021), ConceptNet (Speer et al., 2017) or TransOMCS (Zhang et al., 2020). These should also be evaluated and included for text-to-3D scene applications to create a richer dataset.

### 10.2.6 Process-related expectations

This point consists on the one hand of the improvement of the models implemented so far and extension by new ones. Of the models implemented so far, this applies in particular to IsoSpaceSpERT. The model can benefit significantly from more training data and an architecture designed for SemAF links in the future. In the future, this will lead to a SemAF model that indexes not only IsoSpace but all SemAF-relevant entities and links, thus benefiting from the combination of the different tasks (multi-task learning; Chen et al., 2021; Bingel & Søgaard, 2017).

The second main point is processing contexts over a long time. It is much easier to process 2-3 sentences for a model than complete documents or scripts. This was a bigger problem with RNN- and LSTM-based methods with phenomena like gradient vanishing or explosion (Hanin, 2018; Pascanu et al., 2013). Regular transformer models, on the other hand, cannot process long sequences because of their self-attention mechanism.

There are adapted variants, like Big Bird (Zaheer et al., 2020) or Longformer (Beltagy et al., 2020) which in turn only benefit if such long sequences occur in the training data. Accordingly, the task will be to generate just such training data, e.g., using book scenes or theater scene descriptions.

### 10.2.7 Annotation

As described in Chapter 9, SE<sub>SB</sub> can also be used for annotation purposes via the VR controller. The possibilities offered by VR have not yet been exhausted. Future work could use the user's movements and hand gestures as additional annotation input, which would further speed and simplify it. Thus, the characters' movements could be generated via the direct movements of the person commenting, or the direction of gaze could be used as another interaction medium.

### 10.2.8 Combination

The last goal is to bring all these contexts and expectations together. Starting from single pairs (e.g. *Spatial + Temporal* → *Kitchen in the Middle Ages*) until finally there are enough resources and corresponding models that can connect everything. In the future, it will be possible to describe e.g. birthday scenes with only a few words, because the system can process the contexts independently (age of the person, cultural setting, appropriate gifts, guest clothes, ...).

# Bibliography

- Abrami, G., Henlein, A., Kett, A., & Mehler, A. (2020a). Text2SceneVR: Generating hyper-texts with vannotator as a pre-processing step for text2scene systems. In *Proceedings of the 31st ACM Conference on Hypertext and Social Media, HT '20* (pp. 177–186). New York, NY, USA: Association for Computing Machinery.
- Abrami, G., Henlein, A., Lücking, A., Kett, A., Adeberg, P., & Mehler, A. (2021). Unleashing annotations with TextAnnotator: Multimedia, multi-perspective document views for ubiquitous annotation. In *Proceedings of the Seventeenth Joint ACL - ISO Workshop on Interoperable Semantic Annotation (ISA-17)*, ISA-17. accepted.
- Abrami, G. & Mehler, A. (2018). A UIMA database interface for managing NLP-related text annotations. In *Proc. of LREC 2018*, LREC 2018 Miyazaki, Japan.
- Abrami, G., Mehler, A., Lücking, A., Rieb, E., & Helfrich, P. (2019a). TextAnnotator: A flexible framework for semantic annotations. In *Proc. of ISA-15*.
- Abrami, G., Mehler, A., & Spiekermann, C. (2019b). Graph-based format for modeling multimodal annotations in virtual reality by means of VAnnotatoR. In *Proc. of HCI International 2019*.
- Abrami, G., Mehler, A., Spiekermann, C., Kett, A., Lööck, S., & Schwarz, L. (2020b). Educational technologies in the area of ubiquitous historical computing in virtual reality: Finding new ways to teach in a transformed learning environment. In L. Daniela (Ed.), *New Perspectives on Virtual and Augmented Reality*. Taylor & Francis. in press.
- Abrami, G., Spiekermann, C., & Mehler, A. (2019c). VAnnotatoR: Ein Werkzeug zur Annotation multimodaler Netzwerke in dreidimensionalen virtuellen Umgebungen. In *Proceedings of the 6th Digital Humanities Conference in the German-speaking Countries, DHd 2019*, DHd 2019.
- Abrami, G., Stoeckel, M., & Mehler, A. (2020c). TextAnnotator: A uima based tool for simultaneous and collaborative annotation of texts. In *Proc. of LREC 2020*, LREC 2020.
- Achlioptas, P., Diamanti, O., Mitliagkas, I., & Guibas, L. (2018). Learning representations and generative models for 3d point clouds. In *International conference on machine learning* (pp. 40–49): PMLR.
- Adeberg, P. (2020). MobileAnnotator: an App for TextAnnotator. Original title: MobileAnnotator: eine App für den TextAnnotator.

## Bibliography

- Afelt, A., Frutos, R., & Devaux, C. (2018). Bats, coronaviruses, and deforestation: Toward the emergence of novel infectious diseases? *Frontiers in microbiology*, 9, 702.
- Ahmadyan, A., Zhang, L., Ablavatski, A., Wei, J., & Grundmann, M. (2021). Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.
- Ahmed, S., Stoeckel, M., Driller, C., Pachzelt, A., & Mehler, A. (2019). BIOfid Dataset: Publishing a german gold standard for named entity recognition in historical biodiversity literature. In *Proc. of CoNLL 2019*.
- Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Ruano, R. J., Jeffrey, K., Jesmonth, S., Joshi, N., Julian, R., Kalashnikov, D., Kuang, Y., Lee, K.-H., Levine, S., Lu, Y., Luu, L., Parada, C., Pastor, P., Quiambao, J., Rao, K., Rettinghouse, J., Reyes, D., Sermanet, P., Sievers, N., Tan, C., Toshev, A., Vanhoucke, V., Xia, F., Xiao, T., Xu, P., Xu, S., & Yan, M. (2022). Do as i can and not as i say: Grounding language in robotic affordances. In *arXiv preprint arXiv:2204.01691*.
- Akbik, A., Blythe, D., & Vollgraf, R. (2018). Contextual string embeddings for sequence labeling. In *Proc. of COLING 2018* (pp. 1638–1649).
- Akizuki, H., Uno, A., Arai, K., Morioka, S., Ohyama, S., Nishiike, S., Tamura, K., & Takeda, N. (2005). Effects of immersion in virtual reality on postural control. *Neuroscience letters*, 379(1), 23–26.
- Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. (2022). Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*.
- Allcoat, D. & von Mühlennen, A. (2018). Learning in virtual reality: Effects on performance, emotion and engagement. *Research in Learning Technology*, 26.
- Anderson, C. A. (1983). Imagination and expectation: The effect of imagining. *Journal of Personality and Social Psychology*, 4(2), 293–30.
- Andriluka, M., Pishchulin, L., Gehler, P., & Schiele, B. (2014). 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C. L., & Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision* (pp. 2425–2433).
- Antunes, A., Jamone, L., Saponaro, G., Bernardino, A., & Ventura, R. (2016). From human instructions to robot actions: Formulation of goals, affordances and probabilistic planning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 5449–5454).: IEEE.

- Azarpanah, H. & Farhadloo, M. (2021). Measuring biases of word embeddings: What similarity measures and descriptive statistics to use? In *Proceedings of the First Workshop on Trustworthy Natural Language Processing* (pp. 8–14).
- Baden-Powell, C. (2006). *Architect's pocket book of kitchen design*. Routledge.
- Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.
- Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). The berkeley framenet project. In *Proc. of COLING 98* (pp. 86–90).: ACL.
- Bansal, R., Raj, G., & Choudhury, T. (2016). Blur image detection using laplacian operator and open-cv. In *2016 International Conference System Modeling & Advancement in Research Trends (SMART)* (pp. 63–67).: IEEE.
- Barbu, A., Mayo, D., Alverio, J., Luo, W., Wang, C., Gutfreund, D., Tenenbaum, J., & Katz, B. (2019). Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32.
- Barricelli, B. R., De Bonis, A., Di Gaetano, S., & Valtolina, S. (2018). Semiotic framework for virtual reality usability and UX evaluation. In *Proc. of GHIItaly18*.
- Barricelli, B. R., Gadia, D., Rizzi, A., & Marini, D. L. R. (2016). Semiotics of virtual reality as a communication process. *Behav Inform Technol*, 35(11), 879–896.
- Bateman, J. A. (2010). Language and space: a two-level semantic approach based on principles of ontological engineering. *Int. J. Speech Technol.*, 13(1), 29–48.
- Bateman, J. A., Hois, J., Ross, R., & Tenbrink, T. (2010). A linguistic ontology of space for natural language processing. *Artificial Intelligence*, 174(14), 1027–1071.
- Belinkov, Y. & Glass, J. (2019). Analysis Methods in Neural Language Processing: A Survey. *Transactions of the Association for Computational Linguistics*, 7, 49–72.
- Beltagy, I., Peters, M. E., & Cohan, A. (2020). Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.
- Benbouriche, M., Nolet, K., Trottier, D., & Renaud, P. (2014). Virtual reality applications in forensic psychiatry. In *Proc. of VRIC '14* (pp. 7:1–7:4). New York: ACM.
- Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency, FAccT '21* (pp. 610–623). New York, NY, USA: Association for Computing Machinery.
- Benesty, J., Chen, J., Huang, Y., & Cohen, I. (2009). Pearson correlation coefficient. In *Noise reduction in speech processing* (pp. 1–4). Springer.

## Bibliography

- Benford, S., Greenhalgh, C., & Lloyd, D. (1997). Crowded collaborative virtual environments. In *Proc. of CHI 1997* (pp. 59–66). New York: ACM.
- Bernstein, M. (2011). Can we talk about spatial hypertext. In *Proc. of HT 11, HT '11* (pp. 103–112). New York, NY, USA: ACM.
- Bertino, E. & Ferrari, E. (2018). Big data security and privacy. In *A Comprehensive Guide Through the Italian Database Research Over the Last 25 Years* (pp. 425–439). Springer.
- Beßler, D., Porzel, R., Pomarlan, M., Beetz, M., Malaka, R., & Bateman, J. (2020). A formal model of affordances for flexible robotic task execution. In *ECAI 2020* (pp. 2425–2432). IOS Press.
- Bhatnagar, B. L., Xie, X., Petrov, I., Sminchisescu, C., Theobalt, C., & Pons-Moll, G. (2022). Behave: Dataset and method for tracking human object interactions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*: IEEE.
- Biederman, I., Mezzanotte, R. J., & Rabinowitz, J. C. (1982). Scene perception: Detecting and judging objects undergoing relational violations. *Cognitive psychology*, 14(2), 143–177.
- Biewald, L. (2020). Experiment tracking with weights and biases. Software available from wandb.com.
- Biggs, J. B. (2011). *Teaching for quality learning at university: What the student does*. McGraw-hill education (UK).
- Bingel, J. & Søgaard, A. (2017). Identifying beneficial task relations for multi-task learning in deep neural networks. *arXiv preprint arXiv:1702.08303*.
- Bisk, Y., Holtzman, A., Thomason, J., Andreas, J., Bengio, Y., Chai, J., Lapata, M., Lazaridou, A., May, J., Nisnevich, A., Pinto, N., & Turian, J. (2020). Experience grounds language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 8718–8735). Online: Association for Computational Linguistics.
- Black, S., Leo, G., Wang, P., Leahy, C., & Biderman, S. (2021). GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow. If you use this software, please cite it using these metadata.
- Boggust, A. W., Audhkhasi, K., Joshi, D., Harwath, D., Thomas, S., Feris, R. S., Gutfreund, D., Zhang, Y., Torralba, A., Picheny, M., et al. (2019). Grounding spoken words in unlabeled video. In *CVPR Workshops*, volume 2.
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 135–146.

- Boleda, G., Gupta, A., & Padó, S. (2017). Instances and concepts in distributional space. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers* (pp. 79–85).
- Bommasani, R., Davis, K., & Cardie, C. (2020). Interpreting Pretrained Contextualized Representations via Reductions to Static Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 4758–4781). Online: Association for Computational Linguistics.
- Borghini, A. M., Flumini, A., Natraj, N., & Wheaton, L. A. (2012). One hand, two objects: Emergence of affordance in contexts. *Brain and cognition*, 80(1), 64–73.
- Bouraoui, Z., Camacho-Collados, J., & Schockaert, S. (2020). Inducing relational knowledge from bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34 (pp. 7456–7463).
- Boyce, S. J. & Pollatsek, A. (1992). Identification of objects in scenes: the role of scene background in object naming. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 18(3), 531.
- Bozgeyikli, E., Raj, A., Katkooi, S., & Dubey, R. (2016). Point & teleport locomotion technique for virtual reality. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play, CHI PLAY '16* (pp. 205–216). New York, NY, USA: Association for Computing Machinery.
- Brooks, S. K., Webster, R. K., Smith, L. E., Woodland, L., Wessely, S., Greenberg, N., & Rubin, G. J. (2020). The psychological impact of quarantine and how to reduce it: rapid review of the evidence. *The Lancet*.
- Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., & Amodei, D. (2020). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems*, volume 33 (pp. 1877–1901).: Curran Associates, Inc.
- Bruni, E., Tran, N.-K., & Baroni, M. (2014). Multimodal distributional semantics. *Journal of Artificial Intelligence Research*, 49, 1–47.
- Buciluă, C., Caruana, R., & Niculescu-Mizil, A. (2006). Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 535–541).
- Bunt, H., Pustejovsky, J., & Lee, K. (2018). Towards an ISO standard for the annotation of quantification. In *Proc. of LREC 2018*.

## Bibliography

- Butchart, B. (2011). *Augmented reality for smartphones*. UKOLN, University of Bath.
- Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., & Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 11621–11631).
- Caliskan, A., Bryson, J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334), 183–186.
- Cao, J., Tang, H., Fang, H.-S., Shen, X., Lu, C., & Tai, Y.-W. (2019). Cross-domain adaptation for animal pose estimation. In *The IEEE International Conference on Computer Vision (ICCV)*.
- Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The information visualizer, an information workspace. In *Proc. of CHI 1991* (pp. 181–186). New York: ACM.
- Card, S. K., Robertson, G. G., & York, W. (1996). The WebBook and the Web Forager: An information workspace for the World-Wide Web. In *Proc. of CHI 1996*.
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020). End-to-end object detection with transformers. *CoRR*, abs/2005.12872.
- Challenger, J. R., Cwiklik, J., Degenaro, L. R., Epstein, E. A., & Lewis, B. L. (2016). Distributed uima cluster computing (ducc) facility. US Patent 9,396,031.
- Chang, A. X., Dai, A., Funkhouser, T., Halber, M., Niessner, M., Savva, M., Song, S., Zeng, A., & Zhang, Y. (2017a). Matterport3d: Learning from rgb-d data in indoor environments. *International Conference on 3D Vision (3DV)*.
- Chang, A. X., Eric, M., Savva, M., & Manning, C. D. (2017b). SceneSeer: 3D scene design with natural language. *arXiv preprint arXiv:1703.00050*.
- Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., & Yu, F. (2015a). *ShapeNet: An Information-Rich 3D Model Repository*. Technical Report arXiv:1512.03012 [cs.GR], Stanford University – Princeton University – Toyota Technological Institute at Chicago.
- Chang, A. X., Monroe, W., Savva, M., Potts, C., & Manning, C. D. (2015b). Text to 3D scene generation with rich lexical grounding. In *Proc. of IJCNLP 15* (pp. 53–62). Beijing, China: ACL.
- Chang, A. X., Savva, M., & Manning, C. D. (2014a). Interactive learning of spatial knowledge for text to 3D scene generation. In *Association for Computational Linguistics (ACL) Workshop on Interactive Language Learning, Visualization, and Interfaces (ILLVI)*.



- Chang, A. X., Savva, M., & Manning, C. D. (2014b). Learning spatial knowledge for text to 3D scene generation. In *Proc. of EMNLP 14*.
- Chao, Y.-W., Liu, Y., Liu, X., Zeng, H., & Deng, J. (2018). Learning to detect human-object interactions. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 381–389).: IEEE.
- Chao, Y.-W., Wang, Z., Mihalcea, R., & Deng, J. (2015). Mining semantic affordances of visual object categories. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4259–4267).
- Chen, K., Choy, C. B., Savva, M., Chang, A. X., Funkhouser, T., & Savarese, S. (2018). Text2shape: Generating shapes from natural language by learning joint embeddings. In *Asian conference on computer vision* (pp. 100–116).: Springer.
- Chen, S., Zhang, Y., & Yang, Q. (2021). Multi-task learning in natural language processing: An overview. *arXiv preprint arXiv:2109.09138*.
- Chen, Z. & Zhang, H. (2019). Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 5939–5948).
- Chlup, D. T. & Collins, T. E. (2010). Breaking the ice: Using ice-breakers and re-energizers with adult learners. *Adult Learning*, 21(3-4), 34–39.
- Chu, C. X., Tandon, N., & Weikum, G. (2017). Distilling task knowledge from how-to communities. In *Proceedings of the 26th International Conference on World Wide Web* (pp. 805–814).
- Cimiano, P. & Wenderoth, J. (2007). Automatic acquisition of ranked qualia structures from the web. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (pp. 888–895).
- Clark, H. H. (1996). *Using Language*. Cambridge: Cambridge University Press.
- Clark, K., Khandelwal, U., Levy, O., & Manning, C. D. (2019). What does BERT look at? an analysis of BERT’s attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP* (pp. 276–286). Florence, Italy: Association for Computational Linguistics.
- Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.
- Clark, K. & Manning, C. D. (2015). Entity-centric coreference resolution with model stacking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1 (pp. 1405–1415).

## Bibliography

- Clark, K. & Manning, C. D. (2016). Deep reinforcement learning for mention-ranking coreference models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 2256–2262). Austin, Texas: Association for Computational Linguistics.
- Coyne, B., Bauer, D., & Rambow, O. (2011). Vignet: Grounding language in graphics using frame semantics. In *RELMS@ACL*.
- Coyne, B., Rambow, O., Hirschberg, J., & Sproat, R. (2010). Frame semantics in text-to-scene generation. In *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems* (pp. 375–384).: Springer.
- Coyne, B. & Sproat, R. (2001). WordsEye: An automatic text-to-scene conversion system. In *Proc. of SIGGRAPH 2001* (pp. 487–496).
- Da, J. & Kasai, J. (2019). Cracking the contextual commonsense code: Understanding commonsense reasoning aptitude of deep contextual representations. In *Proceedings of the First Workshop on Commonsense Inference in Natural Language Processing* (pp. 1–12). Hong Kong, China: Association for Computational Linguistics.
- Dalgarno, B. & Lee, M. J. (2010). What are the learning affordances of 3-d virtual environments? *British Journal of Educational Technology*, 41(1), 10–32.
- Damen, D., Doughty, H., Farinella, G. M., Fidler, S., Furnari, A., Kazakos, E., Moltisanti, D., Munro, J., Perrett, T., Price, W., et al. (2018). Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)* (pp. 720–736).
- Dani, M., Narain, K., & Hebbalaguppe, R. (2021). 3dposelite: A compact 3d pose estimation using node embeddings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)* (pp. 1878–1887).
- Daniela, L., Visvizi, A., & Lytras, M. D. (2018). How to predict the unpredictable: technology-enhanced learning and learning innovations in higher education. In *The future of innovation and technology in education: Policies and practices for teaching and learning excellence* (pp. 11–26). Emerald Publishing Limited.
- Dayma, B., Patil, S., Cuenca, P., Saifullah, K., Abraham, T., Lê Khác, P., Melas, L., & Ghosh, R. (2021). Dall-e mini.
- de Reichenfeld, C. H. (2022). Textannotator-basierte szenenerstellung mit objekten aus shapenet.
- Dennerlein, K. (2009). *Narratologie des Raumes*, volume 22. Walter de Gruyter.
- Dessing, D., Pierik, F. H., Sterkenburg, R. P., van Dommelen, P., Maas, J., & de Vries, S. I. (2013). Schoolyard physical activity of 6–11 year old children assessed by gps and accelerometry. *International Journal of Behavioral Nutrition and Physical Activity*, 10(1), 1–9.

- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics.
- Dhamo, H., Manhardt, F., Navab, N., & Tombari, F. (2021). Graph-to-3d: End-to-end generation and manipulation of 3d scenes using scene graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 16352–16361).
- Ding, M., Zheng, W., Hong, W., & Tang, J. (2022). Cogview2: Faster and better text-to-image generation via hierarchical transformers. *arXiv preprint arXiv:2204.14217*.
- Ding, Q., Wu, S., Sun, H., Guo, J., & Guo, J. (2020). Hierarchical multi-scale gaussian transformer for stock movement prediction. In *IJCAI* (pp. 4640–4646).
- Dozat, T. & Manning, C. D. (2017). Deep biaffine attention for neural dependency parsing. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*: OpenReview.net.
- Dumas, B., Lalanne, D., & Oviatt, S. (2009). Multimodal interfaces: A survey of principles, models and frameworks. In *Human machine interaction* (pp. 3–26). Springer.
- Dutta, A., Gupta, A., & Zissermann, A. (2016). VGG image annotator (VIA). <http://www.robots.ox.ac.uk/vgg/software/via/>. Version: 2.0.11, Accessed: 2022-04-24.
- Dutta, A. & Zisserman, A. (2019). The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia, MM '19* New York, NY, USA: ACM.
- D'Souza, J. & Ng, V. (2015). Utd: Ensemble-based spatial relation extraction. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (pp. 862–869).
- Eberts, M. & Ulges, A. (2019). Span-based joint entity and relation extraction with transformer pre-training. *CoRR*, abs/1909.07755.
- Ethayarajh, K. (2019). How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 55–65). Hong Kong, China: Association for Computational Linguistics.
- Etzioni, O., Fader, A., Christensen, J., Soderland, S., et al. (2011). Open information extraction: The second generation. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

## Bibliography

- Ezen-Can, A. (2020). A comparison of LSTM and BERT for small corpus. *CoRR*, abs/2009.05451.
- Fabola, A., Miller, A., & Fawcett, R. (2015). Exploring the past with google cardboard. In *2015 Digital Heritage*, volume 1 (pp. 277–284).: IEEE.
- Feist, M. I. & Gentner, D. (1998). On plates, bowls, and dishes: Factors in the use of english in and on. In *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society* (pp. 345–349).: Routledge.
- Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., & Lehmann, S. (2017). Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 1615–1625). Copenhagen, Denmark: Association for Computational Linguistics.
- Ferrucci, D. & Lally, A. (2004). UIMA: An architectural approach to unstructured information processing in the corporate research environment. *Natural Language Engineering*, 10(3-4), 327–348.
- Ferrucci, D., Lally, A., Verspoor, K., & Nyberg, E. (2009). Unstructured Information Management Architecture (UIMA) Version 1.0. OASIS Standard.
- Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., & Ruppin, E. (2001). Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web* (pp. 406–414).: ACM.
- Finstad, K. (2010). The usability metric for user experience. *Interacting with Computers*, 22(5), 323–327.
- Fisher, M., Ritchie, D., Savva, M., Funkhouser, T., & Hanrahan, P. (2012). Example-based synthesis of 3d object arrangements. In *ACM SIGGRAPH Asia 2012 papers*, SIGGRAPH Asia '12.
- Foote, B. (2020). New ford race car designed entirely through virtual reality [online]. Available at: <https://fordauthority.com/2020/05/new-ford-race-car-designed-entirely-through-virtual-reality/>. Accessed 26 October 2020.
- Forbes, M., Holtzman, A., & Choi, Y. (2019). Do neural language representations learn physical commonsense? In *CogSci*.
- Fowler, C. (2015). Virtual reality and learning: Where is the pedagogy? *British journal of educational technology*, 46(2), 412–422.
- Francisco-Revilla, L. & Shipman, F. (2005). Parsing and interpreting ambiguous structures in spatial hypermedia. In *Proc. of HT 05*, HT '05 (pp. 107–116). New York, NY, USA: ACM.

- Fraser, B. J. & Goh, S. C. (2003). Classroom learning environments. In *International handbook of educational research in the Asia-Pacific region* (pp. 463–475). Springer.
- Freksa, C. (1992). *Using orientation information for qualitative spatial reasoning*. Springer.
- Frutos, R., Lopez Roig, M., Serra-Cobo, J., & Devaux, C. A. (2020). Covid-19: The conjunction of events leading to the coronavirus pandemic and lessons to learn for future threats. *Frontiers in Medicine*, 7, 223.
- Fu, H., Cai, B., Gao, L., Zhang, L.-X., Wang, J., Li, C., Zeng, Q., Sun, C., Jia, R., Zhao, B., et al. (2021a). 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10933–10942).
- Fu, H., Jia, R., Gao, L., Gong, M., Zhao, B., Maybank, S., & Tao, D. (2021b). 3d-future: 3d furniture shape with texture. *International Journal of Computer Vision*, 129(12), 3313–3337.
- Fussell, S. R. (2002). *The verbal communication of emotions: interdisciplinary perspectives*. Psychology Press.
- Gaizauskas, R. & Alrashid, T. (2019). SceneML: A proposal for annotating scenes in narrative text. In *Workshop on ISA-15* (pp.13).
- Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., & Leahy, C. (2021). The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027.
- Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N. F., Peters, M., Schmitz, M., & Zettlemoyer, L. (2018). AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)* (pp. 1–6). Melbourne, Australia: Association for Computational Linguistics.
- Garrod, S. & Pickering, M. J. (2004). Why is conversation so easy? *Trends in cognitive sciences*, 8(1), 8–11.
- Gella, S. & Keller, F. (2017). An analysis of action recognition datasets for language and vision tasks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)* (pp. 64–71). Vancouver, Canada: Association for Computational Linguistics.
- Gibney, M. J., Barr, S. I., Bellisle, F., Drewnowski, A., Fagt, S., Livingstone, B., Masset, G., Varela Moreiras, G., Moreno, L. A., Smith, J., et al. (2018). Breakfast in human nutrition: The international breakfast research initiative. *Nutrients*, 10(5), 559.
- Gibson, J. J. (1977). The theory of affordances. *Hilldale, USA*, 1(2), 67–82.
- Gigante, M. A. (1993). Virtual reality: definitions, history and applications. In *Virtual reality systems* (pp. 3–14). Elsevier.

## Bibliography

- Girard, C., Ecalle, J., & Magnan, A. (2013). Serious games as new educational tools: how effective are they? a meta-analysis of recent studies. *Journal of computer assisted learning*, 29(3), 207–219.
- Gladkova, A., Drozd, A., & Matsuoka, S. (2016). Analogy-based detection of morphological and semantic relations with word embeddings: What works and what doesn't. In *Proceedings of the NAACL-HLT SRW* (pp. 47–54). San Diego, California, June 12-17, 2016: ACL.
- Glasmachers, T. (2017). Limits of end-to-end learning. In *Asian conference on machine learning* (pp. 17–32): PMLR.
- Gleim, R., Mehler, A., & Ernst, A. (2012). SOA implementation of the eHumanities Desktop. In *Proc. of the Workshop on SOAs for the Humanities: Solutions and Impacts, Digital Humanities 2012*.
- Goodwin, W., Vaze, S., Havoutis, I., & Posner, I. (2022). Zero-shot category-level object pose estimation. *arXiv preprint arXiv:2204.03635*.
- Götz, T. & Suhre, O. (2004). Design and implementation of the UIMA Common Analysis System. *IBM Systems Journal*, 43(3), 476–489.
- Goyal, R., Ebrahimi Kahou, S., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fruend, I., Yianilos, P., Mueller-Freitag, M., et al. (2017). The” something something” video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision* (pp. 5842–5850).
- Grave, E., Bojanowski, P., Gupta, P., Joulin, A., & Mikolov, T. (2018). Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.
- Greene, M. R. (2013). Statistics of high-level scene context. *Frontiers in psychology*, 4, 777.
- Greenwald, A. G., McGhee, D. E., & Schwartz, J. L. (1998). Measuring individual differences in implicit cognition: the implicit association test. *Journal of personality and social psychology*, 74(6), 1464.
- Gupta, A., Boleda, G., & Padó, S. (2017). Distributed prediction of relations for entities: The easy, the difficult, and the impossible. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (\* SEM 2017)* (pp. 104–109).
- Gupta, P. & Jaggi, M. (2021). Obtaining better static word embeddings using contextual embedding models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 5241–5253). Online: Association for Computational Linguistics.

- Gupta, S. & Malik, J. (2015). Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*.
- Gurusamy, K. S., Aggarwal, R., Palanivelu, L., & Davidson, B. R. (2009). Virtual reality training for surgical trainees in laparoscopic surgery. *Cochrane database of systematic reviews*, CD006575(1).
- Guttentag, D. A. (2010). Virtual reality: Applications and implications for tourism. *Tourism management*, 31(5), 637–651.
- Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., et al. (2022). A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*.
- Hancock, J. T., Landrigan, C., & Silver, C. (2007). Expressing emotion in text-based communication. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 929–932).
- Hanin, B. (2018). Which neural net architectures give rise to exploding and vanishing gradients? *Advances in neural information processing systems*, 31.
- Hannun, A., Case, C., Casper, J., Catanzaro, B., Diamos, G., Elsen, E., Prenger, R., Satheesh, S., Sengupta, S., Coates, A., et al. (2014). Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*.
- Hanser, E., Mc Kevitt, P., Lunney, T., & Condell, J. (2009a). Scenemaker: automatic visualisation of screenplays. In *Proc. of AAAI 09* (pp. 265–272): Springer.
- Hanser, E., Mc Kevitt, P., Lunney, T., & Condell, J. (2009b). Text-to-animation: Affective, intelligent and multimodal visualisation of natural language scripts. *School of Computing and Intelligent Systems, Univ. Ulster*.
- Hanser, E., Mc Kevitt, P., Lunney, T., Condell, J., & Ma, M. (2010). Scenemaker: multimodal visualisation of natural language film scripts. In *Proc. of KES 2010* (pp. 430–439): Springer.
- Harfouche, A. L. & Nakhle, F. (2020). Creating bioethics distance learning through virtual reality. *Trends in Biotechnology*.
- Hassani, K. & Lee, W.-S. (2016). Visualizing natural language descriptions: A survey. *ACM Comput. Surv.*, 49(1).
- Hassanin, M., Khan, S., & Tahtali, M. (2021). Visual affordance and function understanding: A survey. *ACM Computing Surveys (CSUR)*, 54(3), 1–35.
- Haun, D. B., Rapold, C. J., Janzen, G., & Levinson, S. C. (2011). Plasticity of human spatial cognition: Spatial language and cognition covary across cultures. *Cognition*, 119(1), 70–80.

## Bibliography

- Hayashi, M., Inoue, S., Douke, M., Hamaguchi, N., Kaneko, H., Bachelder, S., & Nakajima, M. (2014). T2v: New technology of converting text to cg animation. *ITE Transactions on Media Technology and Applications*, 2(1), 74–81.
- Hayward, W. G. & Tarr, M. J. (1995). Spatial language and spatial representation. *Cognition*, 55(1), 39–84.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770–778).
- Hedberg, J. & Alexander, S. (1994). Virtual reality in education: Defining researchable issues. *Educational Media International*, 31(4), 214–220.
- Heinzerling, B. & Inui, K. (2021). Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume* (pp. 1772–1791). Online: Association for Computational Linguistics.
- Helfrich, P., Rieb, E., Abrami, G., Lücking, A., & Mehler, A. (2018). TreeAnnotator: Versatile visual annotation of hierarchical text relations. In *Proc. of LREC 2018*.
- Hellewell, J., Abbott, S., Gimma, A., Bosse, N. I., Jarvis, C. I., Russell, T. W., Munday, J. D., Kucharski, A. J., Edmunds, W. J., Sun, F., et al. (2020). Feasibility of controlling covid-19 outbreaks by isolation of cases and contacts. *The Lancet Global Health*.
- Hemati, W., Uslu, T., & Mehler, A. (2016). Textimager: a distributed uima-based system for nlp. In *Proc. of COLING 2016 System Demonstrations: Federated Conference on Computer Science and Information Systems*.
- Henlein, A., Abrami, G., Kett, A., & Mehler, A. (2020). Transfer of isospace into a 3d environment for annotations and applications. In *16th Joint ACL - ISO Workshop on Interoperable Semantic Annotation PROCEEDINGS* (pp. 32–35). Marseille: European Language Resources Association.
- Henlein, A., Abrami, G., Kett, A., Spiekermann, C., & Mehler, A. (2021). Digital learning, teaching and collaboration in an era of ubiquitous quarantine. In L. Daniela & A. Visvizin (Eds.), *Remote Learning in Times of Pandemic - Issues, Implications and Best Practice* chapter 3. Thames, Oxfordshire, England, UK: Routledge.
- Henlein, A., Gopinath, A., Krishnaswamy, N., Mehler, A., & Pustejovsky, J. (2023a). Grounding human-object interaction to affordance behavior in multimodal datasets. *Frontiers in Artificial Intelligence*, 6.
- Henlein, A., Kett, A., Baumartz, D., Abrami, G., Mehler, A., Bastian, J., Blecher, Y., Budgenhagen, D., Christof, R., Ewald, T.-O., Fauerbach, T., Masny, P., Mende, J., Schnüre, P., & Viel, M. (2023b). Semantic scene builder: Towards a context sensitive text-to-3d



- scene framework. In *Semantic, artificial and computational interaction studies: Towards a behavioromics of multimodal communication, Held as Part of the 25rd HCI International Conference, HCII 2023, Copenhagen, Denmark, July 23– 28, 2023, Proceedings*: Springer. accepted.
- Henlein, A. & Mehler, A. (2020). On the Influence of Coreference Resolution on Word Embeddings in Lexical-semantic Evaluation Tasks. In *Proceedings of The 12th Language Resources and Evaluation Conference* (pp. 27–33). Marseille, France: European Language Resources Association.
- Henlein, A. & Mehler, A. (2022). What do toothbrushes do in the kitchen? how transformers think our world is structured. In *Proceedings of the 2022 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2022)*. accepted.
- Herskovits, A. (1986). *Language and spatial cognition*, volume 12. Cambridge university press Cambridge.
- Hewitt, J. & Manning, C. D. (2019). A structural probe for finding syntax in word representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4129–4138). Minneapolis, Minnesota: Association for Computational Linguistics.
- Hill, F., Reichart, R., & Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4), 665–695.
- Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., & Navab, N. (2012). Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Asian conference on computer vision* (pp. 548–562): Springer.
- Hinton, G., Vinyals, O., Dean, J., et al. (2015). Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7).
- Hochreiter, S. & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735–1780.
- Honnibal, M. & Montani, I. (2017). spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7.
- Horton, T. E., Chakraborty, A., & Amant, R. S. (2012). Affordances for robots: a brief survey. *AVANT. Pismo Awangardy Filozoficzno-Naukowej*, 2, 70–84.
- Hou, Z., Yu, B., Qiao, Y., Peng, X., & Tao, D. (2021a). Affordance transfer learning for human-object interaction detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 495–504).

## Bibliography

- Hou, Z., Yu, B., Qiao, Y., Peng, X., & Tao, D. (2021b). Detecting human-object interaction via fabricated compositional learning. In *CVPR*.
- Hua, B.-S., Pham, Q.-H., Nguyen, D. T., Tran, M.-K., Yu, L.-F., & Yeung, S.-K. (2016). Scenenn: A scene meshes dataset with annotations. In *International Conference on 3D Vision (3DV)*.
- Huan, L., Zheng, X., & Gong, J. (2022). Georec: Geometry-enhanced semantic 3d reconstruction of rgb-d indoor scenes. *ISPRS Journal of Photogrammetry and Remote Sensing*, 186, 301–314.
- Huang, G., Pang, B., Zhu, Z., Rivera, C., & Soricut, R. (2020). Multimodal pretraining for dense video captioning. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing* (pp. 470–490).
- Huguet Cabot, P.-L. & Navigli, R. (2021). REBEL: Relation extraction by end-to-end language generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021* (pp. 2370–2381). Punta Cana, Dominican Republic: Association for Computational Linguistics.
- Hupkes, D., Dankers, V., Mul, M., & Bruni, E. (2020). Compositionality decomposed: How do neural networks generalise? (extended abstract). In C. Bessiere (Ed.), *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20* (pp. 5065–5069).: International Joint Conferences on Artificial Intelligence Organization. Journal track.
- Hürlimann, M. & Bos, J. (2016). Combining lexical and spatial knowledge to predict spatial relations between objects in images. In *Proc. of CVPR 2016* (pp. 10–18).
- Hwang, J. D., Bhagavatula, C., Le Bras, R., Da, J., Sakaguchi, K., Bosselut, A., & Choi, Y. (2021). Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. In *AAAI*.
- Ide, N. & Pustejovsky, J. (2017). *Handbook of linguistic annotation*. Springer.
- Ide, N. & Suderman, K. (2009). Bridging the gaps: Interoperability for GrAF, GATE, and UIMA. In *Proc. of LAW III* (pp. 27–34). Suntec, Singapore: ACL.
- Indraprastha, A. & Shinozaki, M. (2009). The investigation on using unity3d game engine in urban design study. *Journal of ICT Research and Applications*, 3(1), 1–18.
- İrsoy, O., Benton, A., & Stratos, K. (2021). Corrected CBOW performs as well as skip-gram. In *Proceedings of the Second Workshop on Insights from Negative Results in NLP* (pp. 1–8). Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

- ISO (2012a). *Language resource management – Semantic annotation framework (SemAF) – Part 1: Time and events (SemAF-Time, ISO-TimeML)*. Standard ISO/IEC TR 24617-1:2012, International Organization for Standardization.
- ISO (2012b). *Language resource management – Semantic annotation framework – Part 2: Dialogue acts*. Standard ISO/IEC TR 24617-2:2012, International Organization for Standardization.
- ISO (2014a). *Language resource management – Semantic annotation framework (SemAF) – Part 7: Spatial information (ISO-Space)*. Standard ISO/IEC TR 24617-7:2014, International Organization for Standardization.
- ISO (2014b). *Language resource management – Semantic annotation framework – Part 4: Semantic roles (SemAF-SR)*. Standard ISO/IEC TR 24617-4:2014, International Organization for Standardization.
- ISO (2014c). *Language resource management – Semantic annotation framework – Part 5: Discourse structure*. Standard ISO/IEC TR 24617-5:2014, International Organization for Standardization.
- ISO (2019). *Language resource management – Semantic annotation framework – Part 11: Reference annotation framework (RAF)*. Standard ISO/IEC TR 24617-9:2019, International Organization for Standardization.
- ISO (2020). *Language resource management – Semantic annotation framework (SemAF) – Part 7: Spatial information (ISO-Space)*. Standard ISO/IEC TR 24617-7:2020, International Organization for Standardization.
- ISO (2021). *Language resource management – Semantic annotation framework – Part 11: Measurable quantitative information (MQI)*. Standard ISO/IEC TR 24617-11:2021, International Organization for Standardization.
- ISO (2022a). *Language resource management – Semantic annotation framework – Part 10: Visual information (VoxML)*. Standard ISO/IEC TR 24617-10, International Organization for Standardization.
- ISO (2022b). *Language resource management – Semantic annotation framework – Part 14: Spatial semantics*. Standard ISO/IEC TR 24617-14, International Organization for Standardization.
- Jastrzebski, S., Leśniak, D., & Czarnecki, W. M. (2017). How to evaluate word embeddings? on importance of data efficiency and simple supervised tasks. *arXiv preprint arXiv:1702.02170*.
- Jiang, Z., Xu, F. F., Araki, J., & Neubig, G. (2020). How Can We Know What Language Models Know? *Transactions of the Association for Computational Linguistics*, 8, 423–438.

## Bibliography

- Johansson, R., Berglund, A., Danielsson, M., & Nugues, P. (2005). Automatic text-to-scene conversion in the traffic accident domain. In *IJCAI*, volume 5 (pp. 1073–1078).
- Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., & Levy, O. (2020). SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8, 64–77.
- Joshi, M., Levy, O., Weld, D. S., & Zettlemoyer, L. (2019). BERT for coreference resolution: Baselines and analysis. In *Empirical Methods in Natural Language Processing (EMNLP)*.
- Jurgens, D. A., Turney, P. D., Mohammad, S. M., & Holyoak, K. J. (2012). Semeval-2012 task 2: Measuring degrees of relational similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation* (pp. 356–364).: Association for Computational Linguistics.
- Kabbara, J. & Cheung, J. C. K. (2021). Post-editing extractive summaries by definiteness prediction. In *Findings of the Association for Computational Linguistics: EMNLP 2021* (pp. 3682–3692). Punta Cana, Dominican Republic: Association for Computational Linguistics.
- Kamp, H. (1975). Two theories about adjectives. In E. L. Keenan (Ed.), *Formal Semantics of Natural Language* (pp. 123–155). Cambridge University Press.
- Karacan, L., Akata, Z., Erdem, A., & Erdem, E. (2016). Learning to generate images of outdoor scenes from attributes and semantic layouts. *arXiv preprint arXiv:1612.00215*.
- Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika*, 30(1/2), 81–93.
- Kermani, Z. S., Liao, Z., Tan, P., & Zhang, H. (2016). Learning 3d scene synthesis from annotated rgb-d images. In *Proceedings of the Symposium on Geometry Processing, SGP '16* (pp. 197–206). Goslar, DEU: Eurographics Association.
- Kett, A. (2020). text2city: Räumliche visualisierung textueller strukturen.
- Kett, A., Abrami, G., Mehler, A., & Spiekermann, C. (2018). Resources2City Explorer: A system for generating interactive walkable virtual cities out of file systems. In *Proc. of the 31st ACM UIST*.
- Khan, S., Naseer, M., Hayat, M., Zamir, S. W., Khan, F. S., & Shah, M. (2021). Transformers in vision: A survey. *ACM Computing Surveys (CSUR)*.
- Kim, D. I. & Sukhatme, G. S. (2014). Semantic labeling of 3d point clouds with object affordance for robot manipulation. In *2014 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 5578–5584).: IEEE.

- Kingma, D. P. & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Klie, J.-C. & de Castilho, R. E. (2020). Dkpro cassis - reading and writing uima cas files in python.
- Kokoska, S. & Zwillinger, D. (2000). *CRC standard probability and statistics tables and formulae*. Crc Press.
- Komninos, A. & Manandhar, S. (2016). Dependency based embeddings for sentence classification tasks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1490–1500).
- Kordjamshidi, P., Moens, M.-F., & van Otterlo, M. (2010). Spatial Role Labeling: Task definition and annotation scheme. In *Proc. of LREC 2010* (pp. 413–420).
- Kordjamshidi, P., Van Otterlo, M., & Moens, M.-F. (2011). Spatial role labeling: Towards extraction of spatial relations from natural language. *ACM Transactions on Speech and Language Processing (TSLP)*, 8(3), 1–36.
- Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.-J., Shamma, D. A., et al. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1), 32–73.
- Kuehn, B. M. (2018). Virtual and augmented reality put a twist on medical education. *JAMA*, 319(8), 756–758.
- Kühn, V., Abrami, G., & Mehler, A. (2020). WikiNectVR: A gesture-based approach for interacting in virtual reality based on wkinect and gestural writing. In *Proc. of HCI 2020*.
- Kumar, A. A. (2021). Semantic memory: A review of methods, models, and current challenges. *Psychonomic Bulletin & Review*, 28(1), 40–80.
- Kurita, K., Vyas, N., Pareek, A., Black, A. W., & Tsvetkov, Y. (2019). Measuring bias in contextualized word representations. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing* (pp. 166–172). Florence, Italy: Association for Computational Linguistics.
- Kuzey, E., Strötgen, J., Setty, V., & Weikum, G. (2016). Temponym Tagging: Temporal Scopes for Textual Phrases. In *Proceedings of the 6th Temporal Web Analytics Workshop (TempWeb '16)* (pp. 841–842).: ACM.
- Lage, M. J., Platt, G. J., & Treglia, M. (2000). Inverting the classroom: A gateway to creating an inclusive learning environment. *The Journal of Economic Education*, 31(1), 30–43.

## Bibliography

- Lakoff, G. (1987). *Women, Fire, and Dangerous Things: What Categories Reveal about the Mind*. Chicago: University of Chicago Press.
- Lam, M. H.-B., Wing, Y.-K., Yu, M. W.-M., Leung, C.-M., Ma, R. C. W., Kong, A. P. S., So, W., Fong, S. Y.-Y., & Lam, S.-P. (2009). Mental Morbidities and Chronic Fatigue in Severe Acute Respiratory Syndrome Survivors: Long-term Follow-up. *Archives of Internal Medicine*, 169(22), 2142–2147.
- Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942.
- Land, S. & Jonassen, D. (2012). *Theoretical foundations of learning environments*. Routledge.
- Landau, B. & Jackendoff, R. (1993). “what” and “where” in spatial language and spatial cognition. *Behavioral and Brain Sciences*, 16(2), 217–238.
- Lang, B. (2020). 34 vr apps for remote work, education, training, design review, and more [online]. Available at: <https://www.roadtovr.com/vr-apps-work-from-home-remote-office-design-review-training-education-cad-telepresence-wf> Accessed 26 October 2020.
- Lauer, T., Willenbockel, V., Maffongelli, L., & Vö, M. L.-H. (2020). The influence of scene and object orientation on the scene consistency effect. *Behavioural Brain Research*, 394, 112812.
- Lee, K., He, L., Lewis, M., & Zettlemoyer, L. (2017). End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 188–197). Copenhagen, Denmark: Association for Computational Linguistics.
- Lee, K., He, L., & Zettlemoyer, L. (2018). Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)* (pp. 687–692). New Orleans, Louisiana: Association for Computational Linguistics.
- Levy, O. & Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2 (pp. 302–308).
- Li, L., Yu, F., Shi, D., Shi, J., Tian, Z., Yang, J., Wang, X., & Jiang, Q. (2017). Application of virtual reality technology in clinical medicine. *American journal of translational research*, 9(9), 3867.

- Li, L. H., Yatskar, M., Yin, D., Hsieh, C.-J., & Chang, K.-W. (2020). What does BERT with vision look at? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics* (pp. 5265–5275). Online: Association for Computational Linguistics.
- Li, M., Patil, A. G., Xu, K., Chaudhuri, S., Khan, O., Shamir, A., Tu, C., Chen, B., Cohen-Or, D., & Zhang, H. (2019). Grains: Generative recursive autoencoders for indoor scenes. *ACM Transactions on Graphics (TOG)*, 38(2), 1–16.
- Liao, Y., Xie, J., & Geiger, A. (2022). Kitti-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Lim, J. J., Pirsiavash, H., & Torralba, A. (2013). Parsing IKEA Objects: Fine Pose Estimation. In *ICCV*.
- Lin, B. Y., Lee, S., Khanna, R., & Ren, X. (2020). Birds have four legs?! NumerSense: Probing Numerical Commonsense Knowledge of Pre-Trained Language Models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6862–6868). Online: Association for Computational Linguistics.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740–755).: Springer.
- Ling, W., Dyer, C., Black, A., & Trancoso, I. (2015). Two/Too Simple Adaptations of word2vec for Syntax Problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*: Association for Computational Linguistics.
- Liu, X., He, P., Chen, W., & Gao, J. (2019a). Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 4487–4496). Florence, Italy: Association for Computational Linguistics.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019b). Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.
- Liu, Y., Wen, Y., Peng, S., Lin, C., Long, X., Komura, T., & Wang, W. (2022). Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images. *arXiv preprint arXiv:2204.10776*.
- Loucaides, C. A., Jago, R., & Charalambous, I. (2009). Promoting physical activity during school break times: piloting a simple, low cost intervention. *Preventive Medicine*, 48(4), 332–334.

## Bibliography

- Loureiro, D. & Camacho-Collados, J. (2020). Don't neglect the obvious: On the role of unambiguous words in word sense disambiguation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 3514–3520). Online: Association for Computational Linguistics.
- Loureiro, D. & Jorge, A. (2019). Language modelling makes sense: Propagating representations through WordNet for full-coverage word sense disambiguation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 5682–5691). Florence, Italy: Association for Computational Linguistics.
- Luong, T., Socher, R., & Manning, C. (2013). Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning* (pp. 104–113).
- Ma, R., Patil, A. G., Fisher, M., Li, M., Pirk, S., Hua, B.-S., Yeung, S.-K., Tong, X., Guibas, L., & Zhang, H. (2018). Language-driven synthesis of 3D scenes from scene databases. In *SIGGRAPH Asia 2018 Technical Papers* (pp. 212): ACM.
- Mahalanobis, P. C. (1936). On the generalized distance in statistics. *Proceedings of the National Institute of Sciences (Calcutta)*, 2, 49–55.
- Mainwaring, S. D., Tversky, B., Ohgishi, M., & Schiano, D. J. (2003). Descriptions of simple spatial scenes in english and japanese. *Spatial cognition and computation*, 3(1), 3–42.
- Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proc. of ACL System Demonstrations* (pp. 55–60).
- Marcus, G., Davis, E., & Aaronson, S. (2022). A very preliminary analysis of dall-e 2. *arXiv preprint arXiv:2204.13807*.
- Marini, D., Folgieri, R., Gadia, D., & Rizzi, A. (2012). Virtual reality as a communication process. *Virtual Reality*, 16(3), 233–241.
- Marshall, C. C., Shipman, F. M., & Coombs, J. H. (1994). Viki: Spatial hypertext supporting emergent structure. In *Proc. of ECHT 94, ECHT '94* (pp. 13–23). New York, NY, USA: ACM.
- Marshall, C. C. & Shipman III., F. M. (1995). Spatial hypertext: designing for change. *Communications of the ACM*, 38(8), 88–97.
- Marshall, C. C. & Shipman III., F. M. (1997). Spatial hypertext and the practice of information triage. In *Proc. of HT 97* (pp. 124–133).
- Marszalek, M., Laptev, I., & Schmid, C. (2009). Actions in context. In *2009 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 2929–2936): IEEE.



- Martín-Gutiérrez, J., Mora, C. E., Añorbe-Díaz, B., & González-Marrero, A. (2017). Virtual technologies trends in education. *EURASIA Journal of Mathematics, Science and Technology Education*, 13(2), 469–486.
- May, C., Wang, A., Bordia, S., Bowman, S. R., & Rudinger, R. (2019). On measuring social biases in sentence encoders. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 622–628). Minneapolis, Minnesota: Association for Computational Linguistics.
- Mayes, J. T. & Fowler, C. J. (1999). Learning technology and usability: a framework for understanding courseware. *Interacting with computers*, 11(5), 485–497.
- McClelland, J. L., Hill, F., Rudolph, M., Baldridge, J., & Schütze, H. (2019). Extending machine language models toward human-level language understanding. *CoRR*, abs/1912.05877.
- McClelland, J. L., Hill, F., Rudolph, M., Baldridge, J., & Schütze, H. (2020). Placing language in an integrated understanding system: Next steps toward human-level performance in neural language models. *Proceedings of the National Academy of Sciences*, 117(42), 25966–25974.
- McInnes, L., Healy, J., & Astels, S. (2017). hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11).
- McNeill, D. (2016). *Why we gesture: The surprising role of hand movements in communication*. Cambridge University Press.
- Mehler, A. (2009). Generalized shortest paths trees: A novel graph class applied to semi-otic networks. In M. Dehmer & F. Emmert-Streib (Eds.), *Analysis of Complex Networks: From Biology to Linguistics* (pp. 175–220). Weinheim: Wiley-VCH.
- Mehler, A., Abrami, G., Bruendel, S., Felder, L., Ostertag, T., & Spiekermann, C. (2017). Stolperwege: an app for a digital public history of the Holocaust. In *Proc. of HT 17, HT '17* (pp. 319–320). New York, NY, USA: ACM.
- Mehler, A., Abrami, G., Spiekermann, C., & Jostock, M. (2018). VAnnotatoR: A framework for generating multimodal hypertexts. In *Proc. HT 2018* New York, NY, USA: ACM.
- Mehler, A., Uslu, T., & Hemati, W. (2016a). Text2voronoi: An image-driven approach to differential diagnosis. In *Proceedings of the 5th Workshop on Vision and Language (VL'16) hosted by the 54th Annual Meeting of the Association for Computational Linguistics (ACL), Berlin*.
- Mehler, A., Wagner, B., & Gleim, R. (2016b). Wikidition: Towards a multi-layer network model of intertextuality. In *Proc. of DH 2016, DH 2016*.
- Mehrabian, A. (1972). *Nonverbal communication*. Transaction Publishers.

## Bibliography

- Merrill, W., Goldberg, Y., Schwartz, R., & Smith, N. A. (2021). Provable limitations of acquiring meaning from ungrounded form: What will future language models understand? *Transactions of the Association for Computational Linguistics*, 9, 1047–1060.
- Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., & Geiger, A. (2019). Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 4460–4470).
- Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Mikolov, T., Grave, E., Bojanowski, P., Puhersch, C., & Joulin, A. (2018). Advances in pre-training distributed word representations. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).
- Mikropoulos, T. A. & Natsis, A. (2011). Educational virtual environments: A ten-year review of empirical research (1999–2009). *Computers & Education*, 56(3), 769–780.
- Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11), 39–41.
- Mo, K., Zhu, S., Chang, A. X., Yi, L., Tripathi, S., Guibas, L. J., & Su, H. (2019). Partnet: A large-scale benchmark for fine-grained and hierarchical part-level 3d object understanding. In *Proc. of CVPR 2019* (pp. 909–918).
- Moneglia, M., Panunzi, A., & Gregori, L. (2018). Action identification and local equivalence of action verbs: the annotation framework of the imagact ontology. In *Proceedings of the LREC 2018 Workshop AREA. Annotation, Recognition and Evaluation of Actions* (pp. 23–30).
- Moore, J. L., Dickson-Deane, C., & Galyen, K. (2011). e-learning, online learning, and distance learning environments: Are they the same? *The Internet and Higher Education*, 14(2), 129–135.
- Moreno, R. & Mayer, R. (2007). Interactive multimodal learning environments. *Educational psychology review*, 19(3), 309–326.
- Myers, A., Teo, C. L., Fermüller, C., & Aloimonos, Y. (2015). Affordance detection of tool parts from geometric features. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 1374–1381).: IEEE.
- Naranjo, C. A., Ortiz, J. S., Álvarez, V. M., Sánchez, J. S., Tamayo, V. M., Acosta, F. A., Proaño, L. E., & Andaluz, V. H. (2017). Teaching process for children with autism in virtual reality environments. In *Proc. of ICETC 17* (pp. 41–45). New York: ACM.

- Nash, C., Ganin, Y., Eslami, S. A., & Battaglia, P. (2020). Polygen: An autoregressive generative model of 3d meshes. In *International conference on machine learning* (pp. 7220–7229).: PMLR.
- Natraj, N., Pella, Y. M., Borghi, A. M., & Wheaton, L. (2015). The visual encoding of tool–object affordances. *Neuroscience*, 310, 512–527.
- Neumann, B. & Möller, R. (2008). On scene interpretation with description logics. *Image and Vision Computing*, 26(1), 82–101.
- Nguyen, C., DiVerdi, S., Hertzmann, A., & Liu, F. (2017). Vremiere: In-headset virtual reality video editing. In *Proc. of CHI 17* (pp. 5428–5438). New York: ACM.
- Nguyen, K., Tripathi, S., Du, B., Guha, T., & Nguyen, T. Q. (2021). In defense of scene graphs for image captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 1407–1416).
- Nguyen, V. N., Hu, Y., Xiao, Y., Salzmann, M., & Lepetit, V. (2022). Templates for 3d object pose estimation revisited: Generalization to new objects and robustness to occlusions. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*.
- Nichols, E. & Botros, F. (2015). Sprl-cww: Spatial relation classification with independent multi-class models. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (pp. 895–901).
- Norman, J. (2002). Two visual systems and two theories of perception: An attempt to reconcile the constructivist and ecological approaches. *Behavioral and brain sciences*, 25(1), 73–96.
- Oberhauser, R. & Lecon, C. (2017). Virtual reality flythrough of program code structures. In *Proc. of VRIC 17* (pp. 10:1–10:4). New York: ACM.
- Oliva, A. & Torralba, A. (2007). The role of context in object recognition. *Trends in cognitive sciences*, 11(12), 520–527.
- Osgood, C. E. (1966). Dimensionality of the semantic space for communication via facial expressions. *Scandinavian journal of psychology*, 7(1), 1–30.
- Osiurak, F., Rossetti, Y., & Badets, A. (2017). What is an affordance? 40 years later. *Neuroscience & Biobehavioral Reviews*, 77, 403–417.
- Ouerhani, N., Maalel, A., Ghézala, H. B., & Chouri, S. (2020). Smart ubiquitous chatbot for covid-19 assistance with deep learning sentiment analysis model during and after quarantine.
- Palmer, M., Gildea, D., & Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational linguistics*, 31(1), 71–106.

## Bibliography

- Papini, G. P. R., Plebe, A., Da Lio, M., & Donà, R. (2021). A reinforcement learning approach for enacting cautious behaviours in autonomous driving system: Safe speed choice in the interaction with distracted pedestrians. *IEEE Transactions on Intelligent Transportation Systems*.
- Pascanu, R., Mikolov, T., & Bengio, Y. (2013). On the difficulty of training recurrent neural networks. In *International conference on machine learning* (pp. 1310–1318): PMLR.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543).
- Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of NAACL-HLT 2018* (pp. 2227–2237).
- Petroni, F., Lewis, P., Piktus, A., Rocktäschel, T., Wu, Y., Miller, A. H., & Riedel, S. (2020). How context affects language models' factual predictions. In *Automated Knowledge Base Construction*.
- Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019). Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 2463–2473). Hong Kong, China: Association for Computational Linguistics.
- Petrovich, M., Black, M. J., & Varol, G. (2022). Temos: Generating diverse human motions from textual descriptions. *arXiv preprint arXiv:2204.14109*.
- Pickering, M. J. & Garrod, S. (2004). Toward a mechanistic psychology of dialogue. *Behavioral and brain sciences*, 27(2), 169–190.
- Poesio, M., Stuckardt, R., & Versley, Y. (2016). *Anaphora resolution*. Springer.
- Ponzetto, S. P. & Poesio, M. (2009). State-of-the-art nlp approaches to coreference resolution: Theory and practical recipes. In *Tutorial Abstracts of ACL-IJCNLP 2009* (pp. 6–6): Association for Computational Linguistics.
- Potka, J. (1995). Immersive training systems: Virtual reality and education and training. *Instructional science*, 23(5-6), 405–431.
- Pustejovsky, J. (1995). *The generative lexicon*. MIT press.

- Pustejovsky, J. (2013). Dynamic event structure and habitat theory. In *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)* (pp. 1–10).
- Pustejovsky, J., Ingria, B., Sauri, R., Castano, J., Littman, J., Gaizauskas, R., Setzer, A., Katz, G., & Mani, I. (2005a). The specification language TimeML. *The language of time: A reader*, (pp. 545–557).
- Pustejovsky, J., Knippen, R., Littman, J., & Sauri, R. (2005b). Temporal and event information in natural language text. *Language resources and evaluation*, 39(2), 123–164.
- Pustejovsky, J., Kordjamshidi, P., Moens, M.-F., Levine, A., Dworman, S., & Yocum, Z. (2015). SemEval-2015 Task 8: SpaceEval. In *Proc. of SemEval 2015* (pp. 884–894).
- Pustejovsky, J. & Krishnaswamy, N. (2016). VoxML: A visualization modeling language. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)* (pp. 4606–4613). Portorož, Slovenia: European Language Resources Association (ELRA).
- Pustejovsky, J., Lee, K., Bunt, H., & Romary, L. (2010). ISO-TimeML: An international standard for semantic annotation. In *Proc. of LREC 2010*.
- Pustejovsky, J., Moszkowicz, J. L., & Verhagen, M. (2011a). ISO-Space: The annotation of spatial information in language. In *Proc. of the Sixth Joint ISO-ACL SIGSEM Workshop on ISA* (pp. 1–9).
- Pustejovsky, J., Moszkowicz, J. L., & Verhagen, M. (2011b). Using iso-space for annotating spatial information. In *Proc. of the International Conference on Spatial Information Theory*.
- Pustejovsky, J. & Yocum, Z. (2014). Image annotation with ISO-Space: Distinguishing content from structure. In *Proc. of LREC 2014* (pp. 426–431): ELRA.
- Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages.
- Radford, A. (2004). *English syntax: An introduction*. Cambridge University Press.
- Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Aspell, A., Mishkin, P., Clark, J., et al. (2021). Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning* (pp. 8748–8763): PMLR.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8), 9.
- Radinsky, K., Agichtein, E., Gabrilovich, E., & Markovitch, S. (2011). A word at a time: computing word relatedness using temporal semantic analysis. In *Proceedings of the 20th international conference on World wide web* (pp. 337–346): ACM.

## Bibliography

- Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., & Chen, M. (2022). Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.
- Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., & Sutskever, I. (2021). Zero-shot text-to-image generation. In M. Meila & T. Zhang (Eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research* (pp. 8821–8831).: PMLR.
- Randell, D. A., Cui, Z., & Cohn, A. G. (1992). A spatial logic based on regions and connection. *KR*, (pp. 165–176).
- Reif, E., Yuan, A., Wattenberg, M., Viégas, F. B., Coenen, A., Pearce, A., & Kim, B. (2019). Visualizing and measuring the geometry of BERT. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada* (pp. 8592–8600).
- Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., & Novotny, D. (2021). Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 10901–10911).
- Renz, J. (2002). A canonical model of the region connection calculus. *Journal of Applied Non-Classical Logics*, 12(3-4), 469–494.
- Rezaeinia, S. M., Ghodsi, A., & Rahmani, R. (2017). Improving the accuracy of pre-trained word embeddings for sentiment analysis. *arXiv preprint arXiv:1711.08609*.
- Ritchie, D., Wang, K., & Lin, Y.-a. (2019). Fast and flexible indoor scene synthesis via deep convolutional generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 6182–6190).
- Riva, G. (2006). Virtual reality. *Wiley encyclopedia of biomedical engineering*.
- Rodriguez, N. (2016). Teaching virtual reality with affordable technologies. In *International Conference on Human-Computer Interaction* (pp. 89–97).: Springer.
- Rogers, A., Kovaleva, O., & Rumshisky, A. (2020). A primer in BERTology: What we know about how BERT works. *Transactions of the Association for Computational Linguistics*, 8, 842–866.
- Roßner, D., Atzenbeck, C., & Gross, T. (2019). Visualization of the relevance: Using physics simulations for encoding context. In *Proc. of HT 19, HT '19* (pp. 67–76). New York, NY, USA: ACM.
- Ruan, Y., Lee, H.-H., Zhang, K., & Chang, A. X. (2022). Tricolo: Trimodal contrastive loss for fine-grained text to shape retrieval. *arXiv preprint arXiv:2201.07366*.

- Rubart, J. (2019). On managing spatial hypermedia with document stores. In *Proc. of HUMAN 19, HUMAN '19* (pp. 13–18). New York, NY, USA: ACM.
- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1985). *Learning internal representations by error propagation*. Technical report, California Univ San Diego La Jolla Inst for Cognitive Science.
- Russo, I., Frontini, F., De Felice, I., Khan, F., & Monachini, M. (2013). Disambiguation of basic action types through nouns' telic qualia. In *Proceedings of the 6th International Conference on Generative Approaches to the Lexicon (GL2013)* (pp. 70–75).
- Ryan, M.-L. (2012). Space. *Hühn, Peter et al. (eds.): the living handbook of narratology*. view date:12 Feb 2019.
- Sadoski, M., Goetz, E. T., Olivarez Jr, A., Lee, S., & Roberts, N. M. (1990). Imagination in story reading: The role of imagery, verbal recall, story analysis, and processing levels. *Journal of Reading Behavior*, 22(1), 55–70.
- Sadoski, M. & Paivio, A. (2013). *Imagery and text: A dual coding theory of reading and writing*. Routledge.
- Sagonas, C., Tzimiropoulos, G., Zafeiriou, S., & Pantic, M. (2013). A semi-automatic methodology for facial landmark annotation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 896–903).
- Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E., Ghasemipour, S. K. S., Ayan, B. K., Mahdavi, S. S., Lopes, R. G., Salimans, T., Ho, J., Fleet, D. J., & Norouzi, M. (2022). Photorealistic text-to-image diffusion models with deep language understanding.
- Salaberri, H., Arregi, O., & Zapirain, B. (2015). Ixagroupehuspaceeval:(x-space) a wordnet-based approach towards the automatic recognition of spatial information following the iso-space annotation scheme. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)* (pp. 856–861).
- Sampaio, A. Z., Rosario, D., Gomes, A., & Santos, J. (2013). Virtual reality applied on civil engineering education: Construction activity supported on interactive models. *Int. Journal of Engineering Education*, 29(6), 1331–1347.
- Sanghi, A., Chu, H., Lambourne, J. G., Wang, Y., Cheng, C.-Y., Fumero, M., & Malekshan, K. R. (2022). Clip-forge: Towards zero-shot text-to-shape generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 18603–18613).
- Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS EMC2 Workshop*.

## Bibliography

- Sap, M., LeBras, R., Allaway, E., Bhagavatula, C., Lourie, N., Rashkin, H., Roof, B., Smith, N. A., & Choi, Y. (2019). Atomic: An atlas of machine commonsense for if-then reasoning. In *AAAI*.
- Saponaro, G., Jamone, L., Bernardino, A., & Salvi, G. (2017). Interactive robot learning of gestures, language and affordances. *arXiv preprint arXiv:1711.09055*.
- Savva, M., Chang, A. X., & Hanrahan, P. (2015). Semantically-enriched 3D models for common-sense knowledge. *CVPR 2015 Workshop on Functionality, Physics, Intentionality and Causality*.
- Scao, T. L., Wang, T., Hesslow, D., Saulnier, L., Bekman, S., Bari, M. S., Biderman, S., Elshahar, H., Phang, J., Press, O., Raffel, C., Sanh, V., Shen, S., Sutawika, L., Tao, J., Yong, Z. X., Launay, J., & Beltagy, I. (2022). What language model to train if you have one million GPU hours? In *Challenges & Perspectives in Creating Large Language Models*.
- Sedrakyan, G., Malmberg, J., Verbert, K., Järvelä, S., & Kirschner, P. A. (2020). Linking learning behavior analytics and learning science concepts: Designing a learning analytics dashboard for feedback to support learning regulation. *Computers in Human Behavior*, 107, 105512.
- Settles, B. (2009). *Active learning literature survey*. Technical report, University of Wisconsin-Madison Department of Computer Sciences.
- Setzer, A., Gaizauskas, R., & Hepple, M. (2005). The role of inference in the temporal annotation and analysis of text. *Language Resources and Evaluation*, 39(2-3), 243–265.
- Sharma, P., Ding, N., Goodman, S., & Soricut, R. (2018). Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*.
- Shen, L., Yeung, S., Hoffman, J., Mori, G., & Fei-Fei, L. (2018). Scaling human-object interaction recognition through zero-shot learning. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 1568–1576).: IEEE.
- Shi, P. & Lin, J. (2019). Simple BERT models for relation extraction and semantic role labeling. *CoRR*, abs/1904.05255.
- Shin, T., Razeghi, Y., IV, R. L. L., Wallace, E., & Singh, S. (2020). AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In *Empirical Methods in Natural Language Processing (EMNLP)*.
- Shipman III., F. M., Hsieh, H., Maloor, P., & Moore, J. M. (2001). The visual knowledge builder: a second generation spatial hypertext. In *Proc. of HT 01* (pp. 113–122).



- Silberman, N., Hoiem, D., Kohli, P., & Fergus, R. (2012). Indoor segmentation and support inference from rgbd images. In *European conference on computer vision* (pp. 746–760).: Springer.
- Sileo, D. (2021). Visual grounding strategies for text-only natural language processing. In *Proceedings of the Third Workshop on Beyond Vision and LANguage: inTEgrating Real-world kNowledge (LANTERN)* (pp. 19–29). Kyiv, Ukraine: Association for Computational Linguistics.
- Solís, C. & Ali, N. (2008). Shywiki-a spatial hypertext wiki. In *Proc. WikiSym 08, WikiSym '08* New York, NY, USA: ACM.
- Song, S., Yu, F., Zeng, A., Chang, A. X., Savva, M., & Funkhouser, T. (2017). Semantic scene completion from a single depth image. *Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition*.
- Speer, R., Chin, J., & Havasi, C. (2017). Conceptnet 5.5: An open multilingual graph of general knowledge. In *AAAI Conference on Artificial Intelligence* (pp. 4444–4451).
- Spiekermann, C., Abrami, G., & Mehler, A. (2018). VAnnotatoR: a gesture-driven annotation framework for linguistic and multimodal annotation. In *Proc. AREA 2018, AREA*.
- Stirling, L. (2019). Lindsey stirring virtual concert [online]. Available at: <https://youtu.be/mK5Jb1vgrgw>. Accessed 26 October 2020.
- Strötgen, J. & Gertz, M. (2013). Multilingual and cross-domain temporal tagging. *Language Resources and Evaluation*, 47(2), 269–298.
- Stubbs, A. (2011). Mae and mai: lightweight annotation and adjudication tools. In *Proceedings of the 5th Linguistic Annotation Workshop* (pp. 129–133).
- Sun, X., Wu, J., Zhang, X., Zhang, Z., Zhang, C., Xue, T., Tenenbaum, J. B., & Freeman, W. T. (2018). Pix3d: Dataset and methods for single-image 3d shape modeling. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Sutherland, I. E. (1968). A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I* (pp. 757–764).
- Sutton, R. S. & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
- Swan, M. (2015). *Blockchain: Blueprint for a new economy*. ” O’Reilly Media, Inc.”.
- Systap LLC (2015). Blazegraph. <https://blazegraph.com/>. Accessed: 2020-02-15.
- Székely, G. J., Rizzo, M. L., & Bakirov, N. K. (2007). Measuring and testing dependence by correlation of distances. *The annals of statistics*, 35(6), 2769–2794.

## Bibliography

- Tamura, M., Ohashi, H., & Yoshinaga, T. (2021). QPIC: Query-based pairwise human-object interaction detection with image-wide contextual information. In *CVPR*.
- Tan, F., Feng, S., & Ordonez, V. (2019). Text2scene: Generating compositional scenes from textual descriptions. In *Proc. of CVPR 2019*.
- Tenney, I., Xia, P., Chen, B., Wang, A., Poliak, A., McCoy, R. T., Kim, N., Durme, B. V., Bowman, S. R., Das, D., & Pavlick, E. (2019). What do you learn from context? probing for sentence structure in contextualized word representations. *CoRR*, abs/1905.06316.
- Thomason, J., Shridhar, M., Bisk, Y., Paxton, C., & Zettlemoyer, L. (2022). Language grounding with 3d objects. In *Conference on Robot Learning* (pp. 1691–1701): PMLR.
- Thüring, M., Haake, J. M., & Hannemann, J. (1991). What's eliza doing in the chinese room? incoherent hyperdocuments—and how to avoid them. In *Proc. HT 91* (pp. 161–177).
- Tomasello, M. (2004). Learning through others. *Daedalus*, 133(1), 51–58.
- Torregrossa, F., Claveau, V., Kooli, N., Gravier, G., & Allesiardo, R. (2020). On the correlation of word embedding evaluation metrics. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)* (pp. 4789–4797).
- Tosi, A., Pickering, M. J., & Branigan, H. P. (2020). Speakers' use of agency and visual context in spatial descriptions. *Cognition*, 194, 104070.
- Tremblay, J., To, T., Sundaralingam, B., Xiang, Y., Fox, D., & Birchfield, S. (2018). Deep object pose estimation for semantic robotic grasping of household objects. *arXiv preprint arXiv:1809.10790*.
- Tversky, A. & Gati, I. (2004). Studies of similarity. In E. Shafir (Ed.), *Preference, Belief, and Similarity. Selected Writing of Amos Tversky* (pp. 75–95). MIT Press.
- Ulinski, M., Coyne, B., & Hirschberg, J. (2019). Spatialnet: A declarative resource for spatial relations. In *Proc. of SpLU and RoboNLP 2019* (pp. 61–70).
- Uslu, T., Mehler, A., & Baumartz, D. (2019). Computing Classifier-based Embeddings with the Help of text2ddc. In *Proceedings of the 20th International Conference on Computational Linguistics and Intelligent Text Processing, CICLing 2019*.
- Uslu, T., Mehler, A., Baumartz, D., Henlein, A., & Hemati, W. (2018a). fastsense: An efficient word sense disambiguation classifier. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference, May 7 - 12, LREC 2018 Miyazaki, Japan*.
- Uslu, T., Mehler, A., & Meyer, D. (2018b). LitViz: Visualizing Literary Data by Means of text2voronoi. In *Proceedings of the Digital Humanities 2018, DH2018*.

- Uslu, T., Mehler, A., Niekler, A., & Baumartz, D. (2018c). Towards a DDC-based topic network model of wikipedia. In *Proceedings of 2nd International Workshop on Modeling, Analysis, and Management of Social Networks and their Applications (SOCNET 2018), February 28, 2018*.
- Van Der Maaten, L. (2014). Accelerating t-sne using tree-based algorithms. *The journal of machine learning research*, 15(1), 3221–3245.
- Van der Maaten, L. & Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research*, 9(11).
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., & Polosukhin, I. (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, volume 30: Curran Associates, Inc.
- Verhagen, M. (2007). Drawing TimeML relations with TBox. In *Annotating, Extracting and Reasoning about Time and Events* (pp. 7–28). Springer.
- Verhagen, M., Knippen, R., Mani, I., & Pustejovsky, J. (2006). Annotation of temporal relations with Tango. In *LREC* (pp. 2249–2252).
- Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3156–3164).
- Visvizi, A., Daniela, L., & Chen, C.-W. (2020). Beyond the ict- and sustainability hypes: A case for quality education. *Computers in Human Behavior*, 107, 106304.
- Visvizi, A., Lytras, M. D., & Daniela, L. (2019). *The future of innovation and technology in education: policies and practices for teaching and learning excellence*. Emerald Publishing.
- Võ, M. L.-H., Boettcher, S. E., & Draschkow, D. (2019). Reading scenes: How scene grammar guides attention and aids perception in real-world environments. *Current opinion in psychology*.
- Vylomova, E., Rimell, L., Cohn, T., & Baldwin, T. (2016). Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1671–1682). Berlin, Germany: Association for Computational Linguistics.
- Wang, B. & Komatsuzaki, A. (2021). GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. <https://github.com/kingoflolz/mesh-transformer-jax>.

## Bibliography

- Wang, H., Sridhar, S., Huang, J., Valentin, J., Song, S., & Guibas, L. J. (2019). Normalized object coordinate space for category-level 6d object pose and size estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Wang, S., Yap, K.-H., Yuan, J., & Tan, Y.-P. (2020). Discovering human interactions with novel objects via zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 11652–11661).
- Wang, T., Roberts, A., Hesslow, D., Scao, T. L., Chung, H. W., Beltagy, I., Launay, J., & Raffel, C. (2022). What language model architecture and pretraining objective work best for zero-shot generalization? *arXiv preprint arXiv:2204.05832*.
- Wang, X., Yeshwanth, C., & Nießner, M. (2021a). Sceneformer: Indoor scene generation with transformers. In *2021 International Conference on 3D Vision (3DV)* (pp. 106–115): IEEE.
- Wang, Y., Huang, H., Rudin, C., & Shaposhnik, Y. (2021b). Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization. *Journal of Machine Learning Research*, 22(201), 1–73.
- Wang, Y., Liu, S., Afzal, N., Rastegar-Mojarad, M., Wang, L., Shen, F., Kingsbury, P., & Liu, H. (2018). A comparison of word embeddings for the biomedical natural language processing. *Journal of biomedical informatics*, 87, 12–20.
- Warstadt, A., Parrish, A., Liu, H., Mohananey, A., Peng, W., Wang, S.-F., & Bowman, S. R. (2020). BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8, 377–392.
- Wendlandt, L., Kummerfeld, J. K., & Mihalcea, R. (2018). Factors influencing the surprising instability of word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)* (pp. 2092–2102). New Orleans, Louisiana: Association for Computational Linguistics.
- Wiseman, S., Rush, A. M., & Shieber, S. M. (2016). Learning global features for coreference resolution. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 994–1004). San Diego, California: Association for Computational Linguistics.
- Wolf, K., Funk, M., Khalil, R., & Knierim, P. (2017). Using virtual reality for prototyping interactive architecture. In *Proc. of MUM 17* (pp. 457–464). New York: ACM.
- Xiang, Y., Kim, W., Chen, W., Ji, J., Choy, C., Su, H., Mottaghi, R., Guibas, L., & Savarese, S. (2016). Objectnet3d: A large scale database for 3d object recognition. In *European Conference Computer Vision (ECCV)*.

- Xiang, Y., Mottaghi, R., & Savarese, S. (2014). Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*.
- Xiao, Y., Du, Y., & Marlet, R. (2021). Posecontrast: Class-agnostic object viewpoint estimation in the wild with pose-aware contrastive learning. In *International Conference on 3D Vision (3DV)*.
- Xiao, Y., Qiu, X., Langlois, P., Aubry, M., & Marlet, R. (2019). Pose from shape: Deep pose estimation for arbitrary 3D objects. In *British Machine Vision Conference (BMVC)*.
- Xu, F. F., Ji, L., Shi, B., Du, J., Neubig, G., Bisk, Y., & Duan, N. (2020). A benchmark for structured procedural knowledge extraction from cooking videos. In *Proceedings of the First International Workshop on Natural Language Processing Beyond Text* (pp. 30–40).
- Xu, X., Joo, H., Mori, G., & Savva, M. (2021). D3d-hoi: Dynamic 3d human-object interactions from videos. *arXiv preprint arXiv:2108.08420*.
- Yamada, I., Baldwin, T., Sumiyoshi, H., Shibata, M., & Yagi, N. (2007). Automatic acquisition of qualia structure from corpus data. *IEICE transactions on information and systems*, 90(10), 1534–1541.
- Yang, C., Shen, Y., & Zhou, B. (2021). Semantic hierarchy emerges in deep generative representations for scene synthesis. *International Journal of Computer Vision*, 129(5), 1451–1466.
- Yang, G., Huang, X., Hao, Z., Liu, M.-Y., Belongie, S., & Hariharan, B. (2019). Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 4541–4550).
- Yao, B. & Fei-Fei, L. (2010). Modeling mutual context of object and human pose in human-object interaction activities. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 17–24).: IEEE.
- Yao, L., Mao, C., & Luo, Y. (2019). Kg-bert: Bert for knowledge graph completion. *arXiv preprint arXiv:1909.03193*.
- Ye, D., Lin, Y., Li, P., & Sun, M. (2022). Packed levitated marker for entity and relation extraction. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022* (pp. 4904–4917).: Association for Computational Linguistics.
- Yoon, E. Y., Humphreys, G. W., & Riddoch, M. J. (2010). The paired-object affordance effect. *Journal of Experimental Psychology: Human Perception and Performance*, 36(4), 812.

## Bibliography

- Young, G. (2006). Are different affordances subserved by different neural pathways? *Brain and cognition*, 62(2), 134–142.
- Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. (2020). Big bird: Transformers for longer sequences. *Advances in Neural Information Processing Systems*, 33, 17283–17297.
- Zantua, L. S. O. (2017). Utilization of virtual reality content in grade 6 social studies using affordable virtual reality technology. *Asia Pacific Journal of Multidisciplinary Research*, 5(2), 1–10.
- Zhang, A., Liao, Y., Liu, S., Lu, M., Wang, Y., Gao, C., & Li, X. (2021a). Mining the benefits of two-stage and one-stage hoi detection. *Advances in Neural Information Processing Systems*, 34.
- Zhang, D. (2018). Big data security and privacy protection. In *8th International Conference on Management and Computer Science (ICMCS 2018)*: Atlantis Press.
- Zhang, F. Z., Campbell, D., & Gould, S. (2021b). Spatially conditioned graphs for detecting human-object interactions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 13319–13327).
- Zhang, F. Z., Campbell, D., & Gould, S. (2022). Efficient two-stage detection of human-object interactions with a novel unary-pairwise transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 20104–20112).
- Zhang, H., Khashabi, D., Song, Y., & Roth, D. (2020). Transomcs: From linguistic graphs to commonsense knowledge. In *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI) 2020*.
- Zhang, J.-Q., Xu, X., Shen, Z.-M., Huang, Z.-H., Zhao, Y., Cao, Y.-P., Wan, P., & Wang, M. (2021c). Write-an-animation: High-level text-based animation editing with character-scene interaction. In *Computer Graphics Forum*, volume 40 (pp. 217–228): Wiley Online Library.
- Zhang, L., Li, J., & Wang, C. (2017). Automatic synonym extraction using word2vec and spectral clustering. In *2017 36th Chinese Control Conference (CCC)* (pp. 5629–5632): IEEE.
- Zhang, S.-H., Zhang, S.-K., Liang, Y., & Hall, P. (2019). A survey of 3d indoor scene synthesis. *Computer Science and Technology*, 34(3), 594–608.
- Zhang, S.-K., Li, Y.-X., He, Y., Yang, Y.-L., & Zhang, S.-H. (2021d). Mageadd: Real-time interaction simulation for scene synthesis. In *Proceedings of the 29th ACM International Conference on Multimedia* (pp. 965–973).
- Zhao, X., Hu, R., Guerrero, P., Mitra, N., & Komura, T. (2016). Relationship templates for creating scene variations. *ACM Transactions on Graphics (TOG)*, 35(6), 1–13.

- Zhou, W., Du, J., & Ren, X. (2019). Improving bert fine-tuning with embedding normalization. *arXiv preprint arXiv:1911.03918*.
- Zitnick, C. L., Parikh, D., & Vanderwende, L. (2013). Learning the visual interpretation of sentences. In *Proc. of IVVC 2013* (pp. 1681–1688).





# Appendix: Zusammenfassung

Menschen können räumliche Szenen mit Hilfe von Sprache beschreiben und umgekehrt Szenen auf der Grundlage von sprachlichen Beschreibungen rekonstruieren. Die derzeitigen State-of-the-Art Systeme kommen jedoch nicht einmal annähernd an die Komplexität des Menschen heran, wenn es darum geht, eine Szene aus einem gegebenen Text zu erzeugen. Auch die immer weiter fortschreitende Entwicklung immer besserer transformerbasierter Sprachmodelle konnte dabei bisher nicht helfen. Diese Aufgabe, die automatische Generierung einer 3D-Szene auf der Grundlage eines Eingabetextes, wird als Text-to-3D Scene Generierung bezeichnet. In den letzten Jahren haben sich die Arbeiten im Bereich der Text-to-3D Scene Generierung zunehmend darauf konzentriert, immer realistischere Szenen auf der Grundlage vorhandener Szenendatensätze zu erzeugen. Die eigentliche Sprachverarbeitung ist zunehmend in den Hintergrund getreten. Sie dient nur noch dazu, konkrete Relationen (parsbar durch vordefinierte Dependency Regeln) für die Szenengenerierung bereitzustellen (e.g Ma et al., 2018; Chang et al., 2017b). Dementsprechend bleiben aus linguistischer Sicht viele Aufgaben ungelöst (Hassani & Lee, 2016). So müssen alle relevanten Entitäten (welche sind relevant und welche nicht) in einem Text erkannt, Anaphern aufgelöst und schließlich räumliche Beziehungen und semantische Rollen identifiziert werden. Dies erfordert eine umfangreiche Tool-Pipeline von nicht immer homogenen Werkzeugen, die hier zusammenarbeiten müssen. Die meisten Modelle basieren auf vortrainierten Wortvektoren oder Sprachmodellen, die nur auf Text trainiert wurden, und es ist daher nicht klar, wie gut diese räumliche Beziehungen erfassen können. Erschwerend zu der Tatsache, dass die benötigten Daten nicht in ausreichender Menge zur Verfügung stehen, kommt hinzu, dass es keine geeignete Annotationsumgebung gibt, die die Generierung dieser Daten unterstützt.

Weiterhin reicht es nicht aus, nur auf in denen im Text vorhandenen expliziten Informationen zuzugreifen, sondern es müssen verschiedenen zusammenhängenden Beziehungen und Kontexte aufgelöst werden. Für die meisten impliziten Informationen, die in Szenenbeschreibungen enthalten sein können, fehlt es aber an notwendigen Ressourcen. Und schließlich müssen die sprachlichen Einheiten mit 3D-Objekten verknüpft und diese in der Szene sinnvoll angeordnet werden. Auch hier kann es notwendig sein, der Szene zusätzliche Objekte hinzuzufügen, um diese realistischer aussehen zu lassen.

Die meisten der bisherigen Arbeiten sind zudem nicht open-source, so dass es schwierig ist, auf die einzelnen Punkte einzugehen und die Ergebnisse zu vergleichen, ohne direkt ein komplett neues Text-to-3D Scene System zu entwickeln und zu implementieren.

Schließlich stellt sich die Frage, ob solche Systeme in der VR auch für andere Anwendungen als die Erstellung einfacher Szenen geeignet sind, etwa für die digitale Bildung.

Im folgenden Abschnitt wird nun erörtert, wie diese Dissertation zu den verschiedenen Herausforderungen beiträgt.

- (a) Analysen, wie gut aktuelle Sprachmodelle räumliche Informationen verstehen, wie statische Einbettungen im Vergleich dazu abschneiden und ob sie durch Anaphora-Auflösung verbessert werden können.
- (b) Automatisierte Ressourcengenerierung für Kontexterweiterung und *Grounding*, die bei der Erstellung realistischer Szenen helfen können.
- (c) Schaffung eines VR-basierten Text-to-3D Scene Systems, das als Annotations- und aktive Lernumgebung verwendet werden kann, aber auch leicht mit zusätzlichen Funktionen erweitert werden kann, um in Zukunft weitere Kontexte auflösen zu können.
- (d) Analysieren von bestehende Praktiken und Werkzeuge für digitales und virtuelles Lehren, Lernen und Kollaboration sowie Bedingungen und Strategien im Kontext von VR.

Die Zuordnung der verschiedenen Arbeiten zu den Schwerpunkten befinden sich hinter den Namen der Arbeiten.

## **On the Influence of Coreference Resolution on Word Embeddings in Lexical-semantic Evaluation Tasks (a)**

Statische Wordvektoren wie Word2Vec oder transformerbasierte Sprachmodelle wie BERT sind ein wesentlicher Bestandteil jeder modernen NLP-Anwendung. Mit der wachsenden Beliebtheit dieser Modelle wächst auch der Wunsch, die Qualität dieser Grundlagen zu verbessern. Obwohl statische Wordvektoren zunehmend durch kontextbasierte Varianten ersetzt werden, haben sie immer noch ihre Berechtigung, da ihre Anwendung viel schneller, ressourceneffizienter und einfacher zu interpretieren ist (Gupta & Jaggi, 2021).

Ziel der Koreferenzauflösung (CR) ist es, alle Teile eines Textes zu finden, die sich auf dieselbe Entität beziehen. Die F1-Scores bei diesen Aufgaben wurden durch neuentwickelte End2End-Ansätze (Lee et al., 2017) und Transformer-Netzwerke (Joshi et al., 2019) stark verbessert. In dieser Arbeit wurde dieser Effekt in Bezug auf Wortvektoren untersucht, d.h. die Hypothese, dass die Einbeziehung von CR als Vorverarbeitungsschritt zu Verbesserungen bei nachgelagerten Aufgaben führt. Konkret wurden die Auswirkungen von CR auf sechs verschiedene Einbettungsmethoden analysiert und im Kontext von sieben lexikalisch-semantischen Evaluationsaufgaben und der Instanzen- / Hypernymerkennung bewertet. Insbesondere bei der letzten Aufgabe erhofften wir uns eine signifikante Leistungssteigerung. Wir zeigten, dass alle Ansätze nicht signifikant von der Pronomen-Ersetzung profitierten. Die messbaren Verbesserungen waren nur marginal (etwa 0,5 % in den meisten Testfällen). Die Ergebnisse lassen sich dadurch erklären, dass wahrscheinlich genau das erreicht wurde, was der Ansatz zu verhindern versuchte: der Verlust von Kontextinformationen.

## Transfer of ISOSpace into a 3D Environment for Annotations and Applications (c)

Ausgehend aus der Motivation aus der Einleitung beschreibt diese Arbeit ein Projekt zur Unterstützung der räumlichen Annotation. In den letzten Jahren wurden viele Anstrengungen unternommen, ein sprachliches Schema für räumliche und raum-zeitliche Beziehungen zu entwickeln. Allerdings haben sich die Systeme bisher nicht wirklich durchgesetzt, was wahrscheinlich an den komplexen Modellen liegt, auf denen sie beruhen, und am Mangel an verfügbaren Trainingsdaten und automatischen Taggern. Die Erleichterung der Annotation sollte durch eine VR-Umgebung erreicht werden, mit der räumliche Beziehungen besser visualisiert und mit realen Objekten verbunden werden können. Als Objektdatenbank diente dazu ShapeNetSem (Savva et al., 2015). Alleine durch das Platzieren der Objekte sollten dabei z.B. entsprechende IsoSpace Links gesetzt werden, die die räumliche Relation zwischen den Objekten beschreiben. In dieser Arbeit wurden damit die ersten Ansätze und Funktionalitäten des später so genannten Semantic Scene Builders (SeSB) beschrieben.

## Text2SceneVR: Generating Hypertexts with VAnnotatoR as a Pre-processing Step for Text2Scene System (c)

In dieser Arbeit wurde TEXT2SCENEVR vorgestellt, ein VANNOTATOR-basiertes Werkzeug zur Erzeugung von räumlichen Hypertexten. Das Hauptziel dieses Projekts war die Entwicklung eines Tools, das den Engpass der fehlenden Daten für die Generierung von Text-zu-Szene-Daten beseitigt. TEXT2SCENEVR erlaubt seinen Nutzern dafür räumlichen Hypertext in VR zu erstellen. Als Grundlage dient dazu der VANNOTATOR. TEXT2SCENEVR verfügt dabei über folgende Funktionen:

1. **Erstellung von Räumen:** Der erste Schritt zur Erstellung eines räumlichen Hypertextes in VR ist die Erstellung eines Raumes. Dabei kann der Benutzer die Eckpunkte des Raums frei einzeichnen, wodurch die Wände erzeugt werden, die später mit weiteren Details versehen werden können.
2. **Anlegen von Fenstern, Türen und Verwendung von Texturen:** Die Räume können mit Türen und Fenstern ausgestattet und auch texturiert werden. Die Räume können beliebig in der virtuellen Umgebung platziert und angeordnet werden. Es ist möglich, sie nebeneinander anzuordnen, sie zu verbinden und Raumensembles zu bilden.
3. **Objektplatzierung:** Weitere Funktionen umfassen die Auswahl und Konfiguration von Rauminhalten und deren räumliche Anordnung. Objekte werden wie bei der vorherigen vorgestellten Arbeit aus ShapeNetSem bereit gestellt und können überall in der virtuellen Umgebung platziert werden. Neben der Positionierung können Objekte auch skaliert, gedreht und in organisatorischen Gruppen zusammengefasst werden.

Das Datenmodell selbst ist auf Einfachheit und Flexibilität ausgelegt. Evaluiert wurde das Ganze mit Hilfe einer Benutzerstudie.

## **Digital learning, teaching, and collaboration in a time of ubiquitous quarantine (d)**

In dieser Arbeit geht es um die Anwendung von VR-basierten Systemen für digitale Lern- und Lehrzwecke. Ein Punkt, der vor allem in den letzten Jahren durch die Quarantäne- und Corona-Regelungen an Bedeutung gewonnen hat, da z.B. die umfangreichen Quarantänemaßnahmen und die damit verbundene Schließung von Bildungseinrichtungen erhebliche Defizite im Bildungsbereich sichtbar wurde. Basierend auf der Arbeit von Fowler (2015); Mikropoulos & Natsis (2011); Mayes & Fowler (1999), haben wir mehrere Anforderungen an pädagogische VR-Anwendungen abgeleitet und analysieren eine Vielzahl aktueller Programme, um zu prüfen, in wie weit diese erfüllt werden. Die Analysen zeigten, dass die meisten Systeme darauf abzielen, vertraute Lernumgebungen wie Hörsäle, Büros, Seminare oder Klassenzimmer zu simulieren, ohne neue pädagogische Ansätze zu entwickeln, die in VR umgesetzt werden könnten. Das betrifft besonders Funktionen wie das Platzieren, Verknüpfen und Gruppieren von multimodalen Inhalten sowie benutzerspezifische Bearbeitungen und Ansichten von Informationen. Funktionen, die teilweise schon im VANNOTATOR implementiert sind.

## **What do Toothbrushes do in the Kitchen? How Transformers Think our World is Structured (a)**

Transformator-basierte Modelle sind heute im NLP vorherrschend. Sie übertreffen Ansätze, die auf statischen Modellen basieren, in vielerlei Hinsicht. Leider leidet die Interpretierbarkeit dieser Modelle dabei stark. Im Hinblick auf die Generationen von Szenen stellt sich daher die Frage, in wie weit bei transformerbasierte Sprachmodelle die Extraktion von Wissen über Objektbeziehungen möglich ist. Um dies zu analysieren nutzen wir diverse Ansätze aus der Bias-Forschung, um zu untersuchen, in wie weit transformatorbasierte Sprachmodelle es ermöglichen, Wissen über Objektbeziehungen zu extrahieren (*X kommt in Y vor; X besteht aus Z; Aktion A beinhaltet die Verwendung von X*). Dabei kamen folgende Ergebnisse heraus: Erstens zeigen wir, dass sich die Modelle, die mit den verschiedenen Ähnlichkeitsmaßen kombiniert werden, in Bezug auf die Menge des Wissens, das sie zu extrahieren erlauben, stark unterscheiden. Zweitens deuten unsere Ergebnisse darauf hin, dass Ähnlichkeitsmaße viel schlechter abschneiden als klassifikatorbasierte Ansätze. Und zu guter Letzt, dass statische Modelle überraschenderweise fast genauso gut abschneiden wie kontextualisierte Modelle - in einigen Fällen sogar besser.

## **Grounding Human-Object Interaction to Affordance Behavior in Multimodal Datasets (b)**

Diese Arbeit stellte einen Ansatz zur automatischen Erkennung von Objekt-Affordanzen vor. Im Gegensatz zu Aktionen, die mit Objekten durchgeführt werden, beschreiben Affordanzen die Natur dieser Aktion. Dabei wird zwischen Gibsonian (Gibson, 1977) und Telische Pustejovsky (2013) Affordanzen unterschieden. Gibsonian Affordanzen sind jene Verhaltensweisen, die aufgrund der physischen Objektstruktur ermöglicht werden und von Tieren direkt wahrgenommen werden können. Telische Affordanz bezeichnet im Gegenzug ein Verhalten, das durch den typischen Gebrauch oder Zweck eines Objekts konventionalisiert wird.

Wir ergänzten den HICO-DET-Datensatz (Chao et al., 2018) mit Annotationen für Gibsonsche und Telische Affordanzen und eine Teilmenge des Datensatzes mit Annotationen für die Orientierung der beteiligten Menschen und Objekte. Mit diesen Daten wurde dann ein angepasstes Mensch-Objekt-Interaktionsmodell (Human-Object-Model; HOI) trainiert und ein vortrainiertes Object-Orientierungs-Erkennungs Modell evaluiert. Unser Modell, AffordanceUPT, basiert auf einem Unary-Pairwise Transformer (UPT; (Zhang et al., 2021a)), das wir modularisierten, um die Erkennung von Affordanzen unabhängig von der Objekterkennung zu gestalten. Das Modell ist in der Lage, die Gibsonsche/Telische Unterscheidung effektiv zu treffen und dass unser Modell andere Korrelationen in den Daten lernt, um solche Unterscheidungen zu treffen (z.B. das Vorhandensein von Händen im Bild). Die Erkennung der Objektorientierung gestaltet sich aber weiterhin als schwierig.

## **Semantic Scene Builder: Towards a context sensitive Text-to-3D Scene Framework (c)**

In dieser Arbeit stellten wir nun schließlich Semantic Scene Builder (SESB) vor, ein VR-basiertes Text-to-3D Scene Framework, das SemAF (Semantic Annotation Framework) als Schema für die Annotation von Diskursstrukturen verwendet. In SESB sind eine Vielzahl von Werkzeugen und Ressourcen integriert. Als Grundlage dient dazu SemAF und UIMA als einheitliche Datenstruktur, um 3D-Szenen aus textuellen Beschreibungen zu generieren. Dazu gehören ein selbst trainiertes BERT-Modell auf IsoSpace Daten, diverse Ressourcen zum Auflösen von Raumnamen (z.B. Wohnzimmer → Sofa) und objektbezogene Handlungen (z.B. Musik machen → Keyboards) und der kompletten Integrationen eines schon existierenden Text-to-3D Scene Systems (Ma et al., 2018). Die VR-Umgebung ermöglicht SESB seinen Nutzern eine intuitive Erstellung und Annotation der Szene: von Anmerkungen in Texten über Korrekturen in Bearbeitungsschritten bis hin zu Anpassungen in generierten Szenen, all dies geschieht durch das Greifen und Bewegen von Objekten. Wir evaluierten SESB gegen ein anderes State-of-the-Art Text-zu-Szene System und konnten zeigen, dass unser Ansatz nicht nur besser abschneidet, sondern auch die Modellierung einer größeren Vielfalt von Szenen ermöglicht.