Generating common-sense scene graphs using a knowledge base BERT model

Author:
C.J.A. Slewe
6842291

Daily supervisor:
Dr. M. de Boer
Supervisor:
Dr. T. Deoskar
Second Examiner:
Prof. Yoad Winter

Universiteit Utrecht

Graduate School of Natural Sciences
Artificial Intelligence
July 2021

Abstract

Scene graphs can be used to improve upon autonomous robots by describing a variety of environments. This thesis compares performance of transformer models, trained on common knowledge bases such as Wikipedia and ConceptNet, in the creation of common-sense graphs. These graphs are based on images, but the concepts in the graphs are image-independent. For example, a 'living room' can be used as a scene, with a table and a chair present as objects in the common-sense graph. The hypothesis is that the bidirectional encoder representations from transformers (BERT) model can help improve the graph generation by predicting spatial relations between objects. Three different models are compared: a statistical model based on most frequent relations, a ConceptNet-trained KnowBERT model, a Wikipedia-WordNet KnowBERT model. Five scene graphs were generated to evaluate each of the three models. For the spatial relation prediction, the Wikipedia-WordNet model outperformed the ConceptNet model slightly in the 100-relation model (F1 = 0.55, F1 = 0.51± 0.01) but not for the 50-relation model (F1 = 0.54± 0.03, F1 = 0.54± 0.01). This could be due to the fact that the Wikipedia + WordNet model is trained on two knowledge bases. The statistical model proved to be slightly superior over both KnowBERT models, with an accuracy of 0.59 ± 0.01 for the 100-relation model and an accuracy of 0.62 ± 0.04 for the 50-relation model. However, for unseen relations, all KnowBERT models perform far better than the statistical model.

Contents

1 INTRODUCTION 1

1.1 USE CASE: AUTOMATIC LABEL GENERATION 2
1.2 PROBLEM DESCRIPTION 3
1.3 THESIS OUTLINE 4

2 RELATED WORK 6

2.1 SCENE CLASSIFICATION 6
2.2 SCENE GRAPHS 7
2.3 COMMON-SENSE SCENE GRAPHS. 8
2.4 KNOWLEDGE BASES 8
2.4.1 Wikipedia 8
2.4.2 WordNet 9
2.4.3 ConceptNet 10
2.5 BIDIRECTIONAL ENCODER REPRESENTATION FROM TRANSFORMER (BERT) 10
2.5.1 BERT for relation extraction 12
2.5.2 BERT as a knowledge base 12

3 EXPERIMENTAL SETUP 14

3.1 DATASET AND PRE-PROCESSING 14
3.2 METHOD 16
3.2.1 ConceptNet-based label generation 16
3.2.2 Spatial relation predicting 17
3.2.3 Subset selection of spatial relations 20
3.2.4 Constructing a graph 21
3.3 EVALUATION 21
3.3.1 Visual relation prediction 21
3.3.2 Scene graph generation 21

4 RESULTS AND DISCUSSION 22

4.1 VISUAL RELATION PREDICTION 22
4.2 ANALYSIS OF PREDICTIONS 24
4.3 SCENE GRAPH GENERATIONS 27

5 LIMITATIONS 30

6 CONCLUSION 31

6.1 FUTURE RESEARCH 32

7 REFERENCES 33

8 APPENDIX A: ALL GENERATED SCENE GRAPHS 36

1 Introduction

Autonomous robots are able to reach places that are too hostile for humans, such as the Fukushima nuclear plant after the disaster (Nagatani et al., 2013). The radiation causes harm to humans, but a robot wouldn't be affected. This distinction in capabilities between humans and robots is also true for warzones and natural disasters. Autonomous robots not only can explore areas that are too dangerous for humans, but they can search for objects there as well. A specialized version of Boston Dynamics Spot, an agile, mobile robot with automated sensing, was able to detect 16 artifacts such as backpacks, gas leaks, and human survivors in the Defense Advanced Research Projects Agency (DARPA) subterranean challenge, thereby winning first place in the competition (Bouman et al., 2020). However, these artifacts have not yet been integrated to infer information about the robot's environment and provide context for further object detection. Using objects to infer information on the environment is an essential aspect of scene classification.

Scene classification is an important skill for autonomous robots because these robots need to respond to their environment. Scene classification is the task of classifying the overall content of an image, such as distinguishing between different rooms in the house (Yang, Jiang, Hauptmann, & Ngo, 2007). A robot in a living room must act differently from a robot in a desert: there are other obstacles, and there might be other objectives. Scene classification helps the robot collect labels for the objects one would except in a certain scene. If a scene is classified as a living room, one would expect to find chairs, a table, and a television. When classifying a scene, a model uses the information provided by the image classifier (e.g., chairs and a table) to infer a higher hierarchical level (e.g., a living room). The robot needs to have a concept of what different scenes look like in order to be successful. Scene graphs can be generated to differentiate and analyze different scenes structurally. *A scene graph is a symbolic, graphical representation of an image.* The nodes correspond to objects in the image, and the edges represent an interaction (Zareian, A., Karaman, S., & Chang, 2020). Figure 1 shows an example of a scene graph.

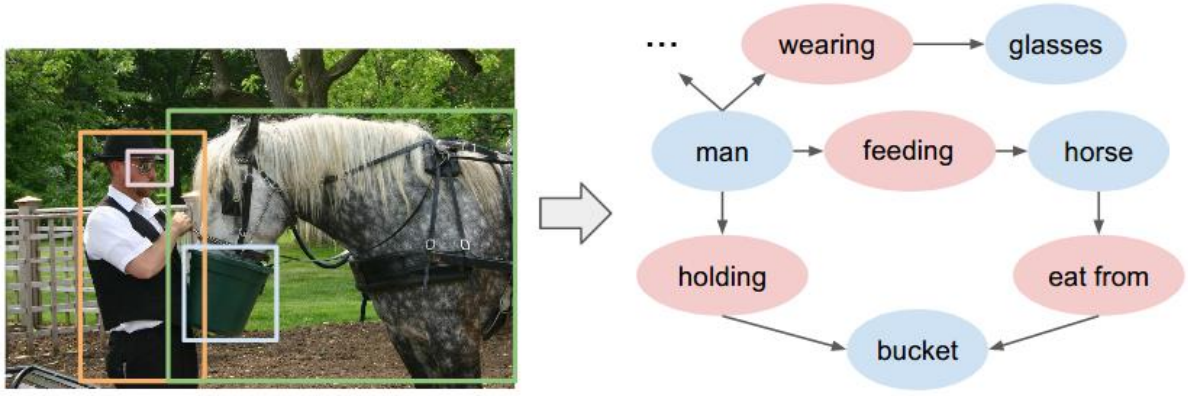

Figure 1. An example of a scene graph for an image a man feeding horse. In this scene graph, blue nodes represent objects in the images, while red nodes represent their relation. from D. Xu, Y. Zhu, C. Choy, B. C., and L. Fei-Fei (2017). *Scene graph generation by iterative message passing. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 5410–5419).*

Research has indicated that knowledge bases can be used to improve scene graph generation. A knowledge base is a system that contains organized information, such as Wikipedia. Connecting the entities in the scene graph to gather more information through knowledge bases improves the performance of scene generation tasks (A. Zareian, S. Karaman, & S.F. Chang, 2020; (Chen, Yu, Chen, & Lin, 2019)(Gu et al., 2019). In the next section, I describe how a knowledge base generates a graph of environmental information.

1.1 Use case: automatic label generation

This research builds upon the previous experiments conducted at Netherlands Organization for Applied Scientific Research (TNO). This project aimed to develop a set of labels before entering a specific scene to help with the classification of objects. There are three phases in the automatic generation of labels for a specific task. First, the relevant concepts for task context are selected (e.g., a house has rooms, and a room contains furniture). Next, the concepts are ordered hierarchically (e.g., a chair and a couch are both furniture). Finally, more meaningful relations between concepts are added (e.g., a lamp hangs above the table). Figure 2 shows the process for automated label generation.

The current method for object selection in a scene classification consists of six steps (Figure 2). First, the relevant objects for the task are collected using the goal description; for example, “office,” “person,” and “bomb” are used if the goal description is to find a person and bombs in an office. In the second step, ConceptNet(Liu & Singh, 2004) is used to create a meaningful knowledge graph as well as objects found in this location. ConceptNet is a knowledge base that contains information on different relations a concept has with other concepts. For an office, objects such as desks and chairs would be retrieved in this phase. In the third phase, irrelevant details are removed, such as plurals or terms that ConceptNet has given a low weight (i.e., unsure whether that object exists in that context). Suggested Upper Merged Ontology (SUMO) furniture (Pease, Niles, & Li, 2002) links the common concepts to their respective taxonomies for the fourth step. Afterward, WordNet (Miller, 1995) generates a proper mapping onto the taxonomy, providing two relevant relations from WordNet to SUMO: equivalence and super/subclass. Finally, a label set is created to train the object detector using the relevant taxonomy. Further explanations of WordNet and ConceptNet can be found in the related work section under the subsection knowledge bases. This pipeline is shown in Figure 2 below. This automated pipeline for relevant knowledge graphs allows the robot to reason about novel objects in a context that it is unfamiliar with.

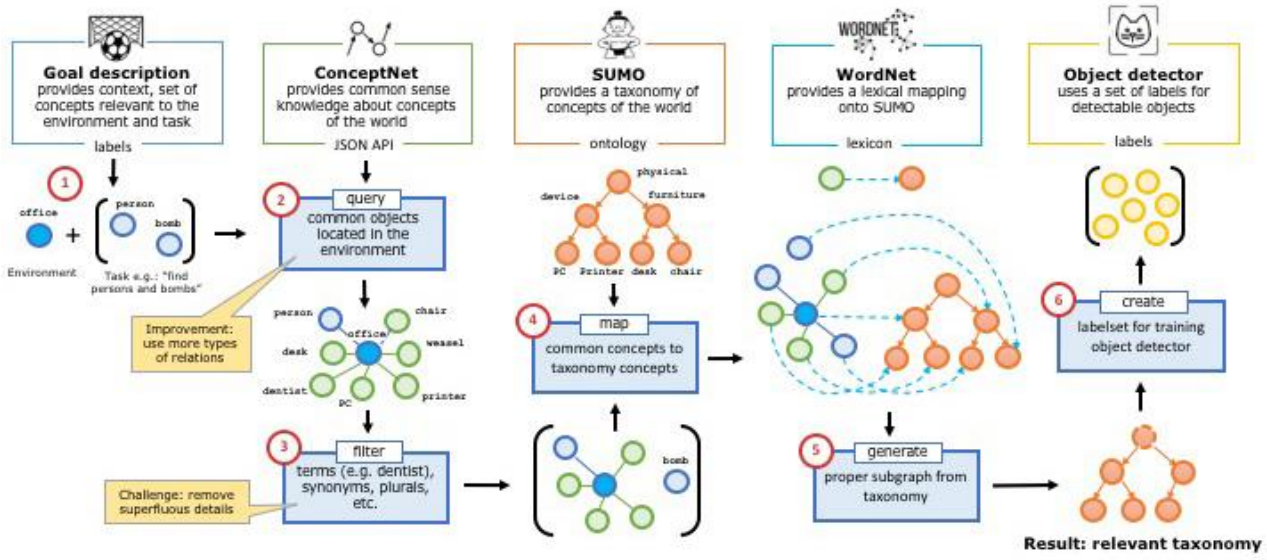

Figure 2. Knowledge graph pipeline for automatic label generation.

1.2 Problem description

This thesis expands upon the TNO method for automatic label generation. The model in this thesis predicts the spatial relation between the objects in the knowledge graph. For this task, the model combines a knowledge base with a language model to generate a complete scene graph using only textual information provided by a particular scene and the knowledge base. With the present method for label generation, there are no relations between these labels. Adding relations to these labels provides a more concrete idea of what a scene looks like. In addition to the labels, the model now has access to a scene graph. The relations provide more information about what can be expected in a scene, facilitating recognition of objects in a later stage.

The task of creating a scene graph is called scene graph generation. An important difference between using knowledge bases for scene graph generation and the model in this thesis is that scene graph generation is a task based on a specific image, while the model created in this thesis generates a hypothetical scene with a textual input. This thesis will discuss a model that uses 'living room' as an input and generates a scene graph based on information from the knowledge base. This model is similar to the model for generating common-sense graphs. *A common-sense graph is image-independent and only contains information that is image-independent.* For example, a common-sense graph can show that the finger is part of the hand. Furthermore, the scene graphs in this thesis are generated, but not evaluated in their ability to help in scene detection. The focus of this thesis is only on the natural language processing part of scene graph generation and common-sense scene graphs, not on computer vision.

Because of its diverse possibilities for finetuning and ability to integrate information knowledge bases, the bidirectional encoder representations from transformers (BERT) model can help improve the current knowledge graph by predicting spatial relations between objects. BERT is a language model that advances state-of-the-art model for a different range of natural language processing (NLP) tasks, such as question answering and language inference (Devlin, Chang, Lee, & Toutanova, 2019). BERT can be trained in combination with a knowledge base (Petroni et al., 2020)(Peters et al., 2019) and for relation extraction (Shi & Lin, 2019).

Several versions of BERT exist, each fine-tuned for a different purpose. While it would be possible to fine-tune the standard version of BERT for predicting spatial relations, research has indicated that an expanded version of BERT's knowledge base might improve performance. Peters et al. (2019) developed KnowBERT, a version that uses an entity linker to provide information from a knowledge base to expand the entity embeddings provided to the language model. Therefore, the research question of this thesis is as follows:

RQ: How can KnowBERT be used to generate common-sense scene graphs?

This model is typically trained on Wikipedia-WordNet. However, the ConceptNet knowledge base is more relationship-focused. Therefore, I hypothesize that a ConceptNet-trained version of KnowBERT will outperform Wikipedia-WordNet-trained KnowBERT.

SQ 1: Is ConceptNet or a Wikipedia-wordnet better suited to predict spatial relations in a KnowBERT model?

It is not certain that KnowBERT will outperform a statistical model. A comparison to a statistical model will be made to determine the impact of the knowledge base.

SQ 2: Which model performs better for spatial relation predictions – a statistical model or KnowBERT?

KnowBERT should also be able to predict relations for unseen pairs of objects. This capability is expected because of the knowledge base entity embeddings it is trained on, allowing for generalization from the training examples. Therefore, it is also useful to separately examine the pairs of objects that are only present in the test set and not in the training set.

SQ 3: Which model performs better for spatial relation prediction on unseen object pairs – a statistical model or KnowBERT?

1.3 Thesis outline

This thesis hypothesizes that a ConceptNet-trained version of KnowBERT will outperform Wikipedia-WordNet-trained KnowBERT. This is expected because ConceptNet contains many spatial relations, making the information in ConceptNet more related to the spatial relation prediction task. The chapter on related work (chapter 2) will discuss previous research on scene classification (section 2.1), scene graphs (section 2.2), common-sense scene graphs (section 2.3), knowledge bases (section 2.4), and the BERT model (section 2.5). Previous research has shown that natural language processing models have improved scene classification in the past. The scene graph section and common-sense scene graph (section 2.2 & 2.3) explain how knowledge bases can improve scene graph generation. In the knowledge base section (section 2.4), I expand upon the different knowledge bases used for this thesis. I will explain their differences, as well as their strengths and weakness. In the BERT section (section 2.5), I explain the architecture of BERT and elaborate on the use of KnowBERT in the context of this thesis.

The experimental setup chapter (chapter 3) discusses the visual genome dataset and its preprocessing. In addition, I discuss three different methods: the previous method at TNO,

which this thesis builds upon; a ConceptNet-based method for retrieving objects based on a scene; and the generation of relations for the relevant object pairs. The results and discussion chapter (chapter 4) compares the performance of the different KnowBERT models (section 4.1 & 4.2) and shows the model's output for different scenes (section 4.3). Furthermore, the reason for the differences and similarities in performance on the spatial relation tasks is discussed. In the limitations chapter (chapter 5) limitations of this study are discussed. In the conclusion (chapter 6), I discuss the answers to the research questions. Furthermore, I will suggest avenues for future work (section 6.1).

2 Related work

In this section, I will discuss the scientific work that this thesis builds upon. I will discuss the current models that are used for scene classification, scene graph generation and common-sense graph generation, as well as the different knowledge bases that the KnowBERT models are trained on. Furthermore, I will discuss the BERT model, explain why BERT is the current state-of-the-art model for a range of NLP tasks and elaborate on the KnowBERT model.

2.1 Scene classification

In this. Section, I will discuss some of the state-of-the-art models for scene classification. FOSnet combines information from the general scene with the objects in the scene to achieve this performance. In addition, scene coherence loss is used to train FOSnet. Scene coherence loss is a loss function that calculates the consistency of the use over the image's different grids, as seen in Figure 3. FOSnet achieved the state-of-the-art performance on the MIT indoor 67 (accuracy of 90.37 %) and Places 2 (accuracy of 60.14 %) dataset, two common benchmarks for scene classification (Seong, Hyun, & Kim, 2020).

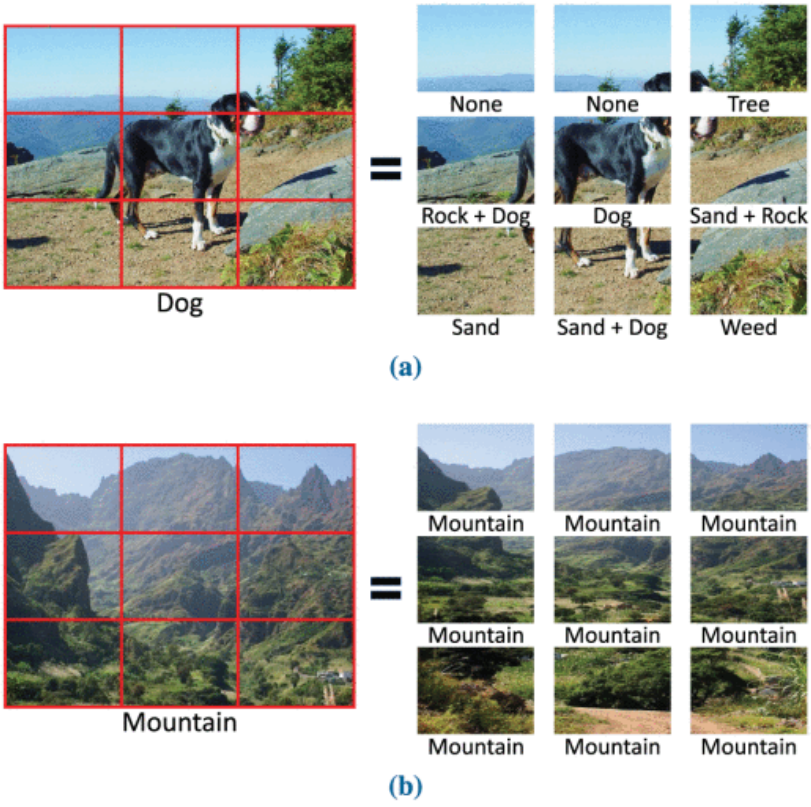

Figure 3. Coherence in object recognition versus scene coherence. The top image (a) shows object recognition, while the bottom image (b) shows scene recognition, which does not vary across the separate tiles. From H. Seong, J. Hyun, and E. Kim (2020). Fosnet: An end-to-end trainable deep neural network for scene recognition. *IEEE Access*, *8*, 82066-82077.

Other natural language processing techniques for improving scene classification include bag-of-words object recognition (Sivic & Zisserman, 2009) and scene classification (Yang et al., 2007). Sivic and Zisserman used clusters of key points, which are image patches containing

rich information. These clusters are treated as visual words and are processed using a bag-of-words model, which represents the words as a list with their respective counts, disregarding the grammar and word order of the sentence.

Other studies have used knowledge bases to improve scene classification. For example, de Boer et al. have shown that the search query can be expanded using ConceptNet or Wikipedia to improve search results when searching for videos. They focused on using the expanded query to detect complex events, such as “birthday party”, “doing homework” and “doing a bike trick”. A complex event is similar to a scene but contains a temporal element. This temporal element allows for more complex events. Their results showed that query expansion improved performance for complex event detection (M. de Boer, Schutte, & Kraaij, 2016).

Other research has shown that scene classification for complex events is possible when there are no labelled images available for these events. Using a vocabulary with high-level concepts and a word2vec-based method for concept selection, the researchers produced a model that outperforms knowledge-based concept selection methods. A word2vec model represents words as vectors based on the words that are present in the same sentences. These vectors then contain semantic information based on the context of the word. (M. H. T. De Boer et al., 2017).

2.2 Scene graphs

A scene graph is a graphical representation of a visual setting with nodes that represent objects, such as “man” or “bike,” and edges that represent relationships, such as “riding”(Zareian, A., Karaman, S., & Chang, 2020). The scene graph representation allows for a semantic, textual analysis of an image. This thesis focuses on knowledge-based scene graphs because of their human-curated and high-quality information (Peters et al., 2019).

Several models implement a combination of scene graphs and knowledge bases. MotifNet uses regularly appearing substructures, called motifs, in scene graphs to improve relation prediction of objects in a scene. MotifNet improves upon prior state-of-the-art models by focusing primarily on these intra-graph interactions (Zellers, Yatskar, Thomson, & Choi, 2018).

One of the challenges of scene graph generation is that datasets are often unbalanced and noisy (Chen et al., 2019; Gu et al., 2019). The relationships in scene graph datasets are often unbalanced because some relationships occur more frequently than others. Using a knowledge-embedded routing network, Chen et al. addressed this issue by using prior knowledge of statistical correlations between objects to regularize the semantic space of relationship prediction (Chen et al., 2019).

Because datasets like visual genome are crowd-sourced, they often miss annotations and contain incorrect or meaningless proposals. Gu et al. (2019) used an external knowledge base, ConceptNet, combined with a reconstructed image to make the scene graph more generalizable.

The work of Xu, Zhu Choy, and Fei-Fei (2017) focused on the visual genome dataset as well in their scene graph generation. Their recurrent neural network (RNN) is trained iteratively via message passing. Similar to the method in this thesis Xu, Zhu Choy, and Fei-Fei split up their model into a relation prediction task and a scene graph generation task. Because their

model is trained on the same dataset for the same task (relation prediction), their model will be used as an comparison.

2.3 Common-sense scene graphs.

A common-sense scene graph links entities using common-sense knowledge. Common-sense knowledge is information on entities that is always true, independent of the situation. For example, between the nodes “hand” and “person,” there is an edge “part of.” While this graph looks similar to a scene graph, it is image-independent.

The Graph Bridging network (GB-Net) links entities in a common-sense knowledge graph to a scene graph. In a common-sense scene graph, each node represents one general concept. The edges represent a general relationship (A. Zareian, S. Karaman, & Chang, 2020). The common-sense knowledge graph of Ilievski, Szekely, and Zhang combines data from seven different sources: the Visual Genome dataset, ConceptNet, Wikidata, WordNet, Roget, ATOMIC, and FrameNet. Their analysis showed that graph and text embeddings of CommonSense Knowledge Graph (CSKG) have complementary notions of similarity since graph embeddings focus on structural patterns while text embeddings focus on lexical features. Their study also showed that using their work for pretraining a language increases recall on common-sense reasoning tasks, such as question answering (Ilievski, Szekely, & Zhang, 2020).

2.4 Knowledge bases

Knowledge bases are able to improve scene graph generation for several reasons. First, the information in knowledge bases is often human-curated (Peters et al., 2019). Knowledge bases are also reviewed, which means the information they contain improves over time. Another advantage of a knowledge base is that they are organized predictably, making information collection easier. Knowledge bases can also contain more general information; the relationships described in a knowledge base are supposed to be true for all instances of an object, while datasets often contain specific examples. Furthermore, knowledge bases can be used in addition to regular training, as discussed in the previous section. Standard BERT is also trained through Wikipedia before being finetuned for a particular task. The finetuning task then requires less training data, another benefit of using knowledge bases (Devlin et al., 2019).

KnowBert has a pre-trained version that integrates entities in Wikipedia-WordNet. This integration is done to explicitly add word sense knowledge and facts about named entities, including those unseen at training time (Peters et al., 2019). For this thesis, KnowBert is also trained on ConceptNet. ConceptNet provides triples with relations between objects; among those are spatial relations. Therefore, ConceptNet contains information that is closer to the task of scene graph generation. In addition, I will expand upon the knowledge bases of Wikipedia, WordNet, and ConceptNet because of their role in the training of KnowBERT.

2.4.1 Wikipedia

Wikipedia is an online multilingual encyclopedia that anyone can edit. Because of its openness and popularity, Wikipedia has over 56 million articles in all languages (“Wikimedia Statistics,” 2021). The strength of Wikipedia as a knowledge base is also its weakness since anyone can edit, and Wikipedia cannot guarantee its validity. The reason Wikipedia is used

for comparison in combination with WordNet against ConceptNet is that the data in Wikipedia is relatively unstructured. Unlike WordNet and ConceptNet, Wikipedia does not have a set structure of relations with other objects. However, Wikipedia does contain a wide range of entries and a lot of textual information on these entries. Wikipedia can be defined as a knowledge base, since it contains structured, human-curated data that can be used by a computer system (Peters et al., 2019). The reason Wikipedia was used over Wikidata is that early experiments with Wikidata embeddings did not improve results on the performance of the KnowBert model for relation extraction tasks (Peters et al., 2019).

2.4.2 WordNet

WordNet is an English lexical database that contains semantical information about words. However, the focus of WordNet is machines. WordNet organizes words into sets of synonyms, called synsets. These synsets are linked through the following semantic relations" synonymy (similar), antonymy (opposite), hyponymy (subordinate), meronymy (part of), toponomy (manner), and entailment. WordNet contains over 118,000 words and more than 90,000 different word senses (Miller, 1995). An example of WordNet is shown in Figure 4 below.

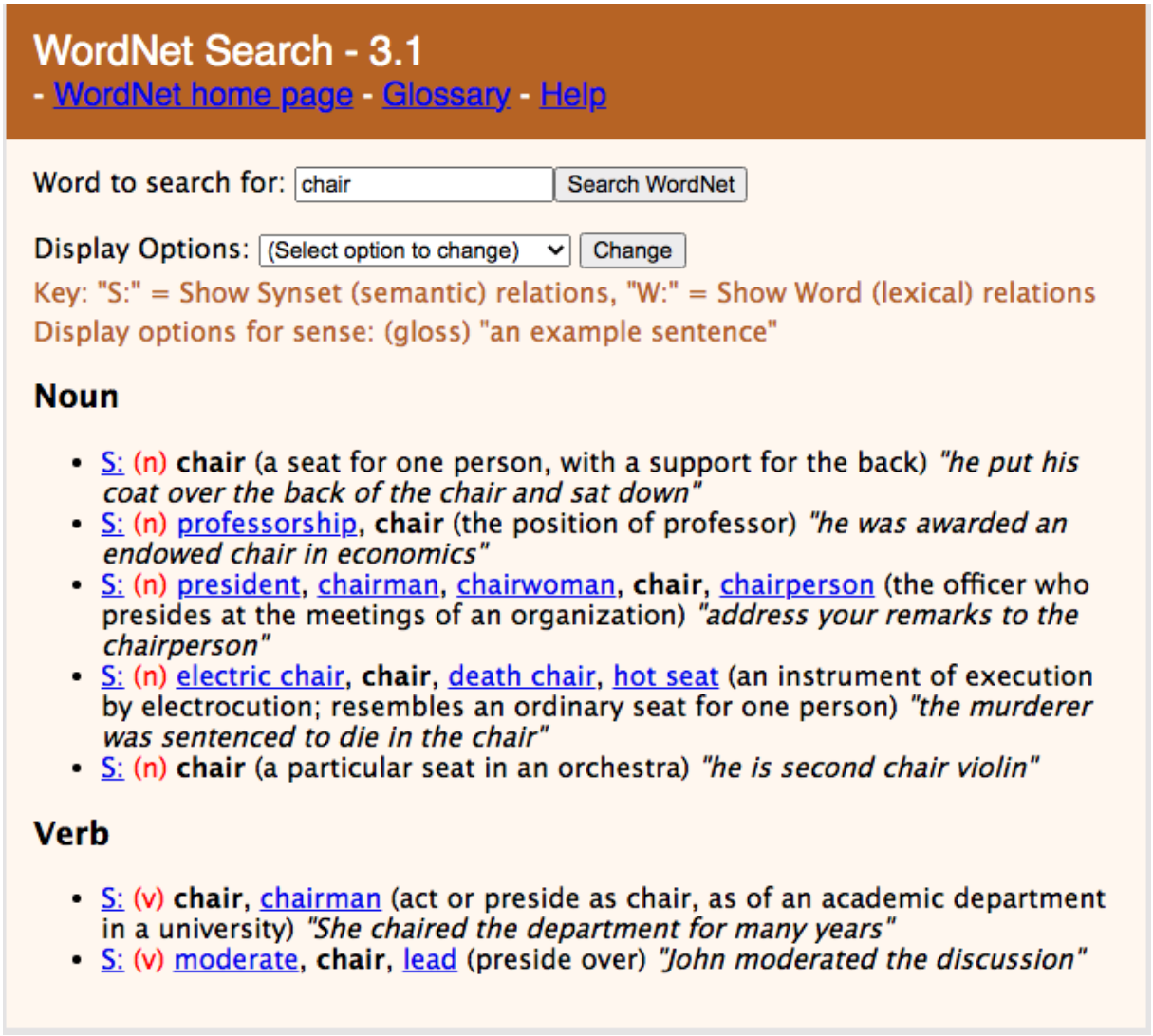

Figure 4. A WordNet page on the chair.

2.4.3 ConceptNet

Similar to WordNet, ConceptNet is a knowledge base created for machine-reading. While WordNet is optimized for lexical categorization, ConceptNet is optimized for context-based inferences. The relationships in WordNet are focused on lexical semantics, but the relationships in ConceptNet are about the real-world relationships between the concepts. Examples of relationships in ConceptNet are "effect of," "part of," and "used for." This attribute makes ConceptNet a useful choice as a knowledge base for inferencing spatial relationships (Liu & Singh, 2004). An example of a ConceptNet page is shown in Figure 5 below.

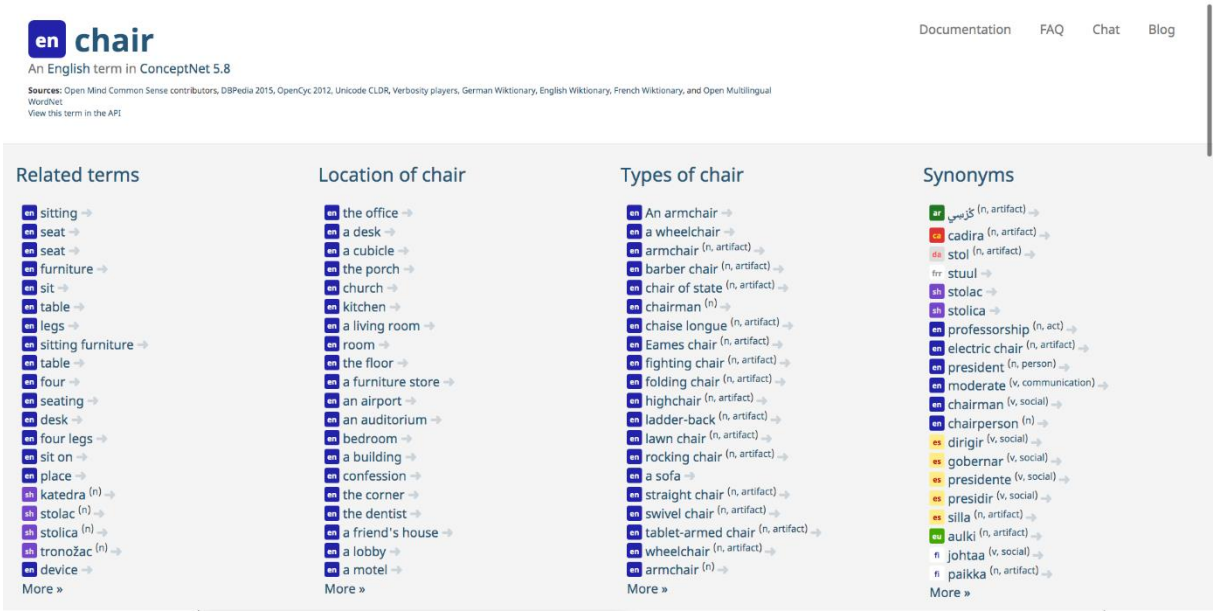

Figure 5. A ConceptNet page on the chair.

2.5 Bidirectional Encoder Representation from Transformer (BERT)

BERT is a pretrained language model that can be finetuned for various tasks, such as part-of-speech tagging, named entity recognition, sentiment analysis, and question answering tasks (Devlin et al., 2019). Bidirectional refers to the way BERT is trained. While directional models read a sentence from left to right or from right to left, the BERT encoder reads the entire sequence simultaneously. BERT achieves this by using *masked language modeling* (Masked LM) and *next sentence prediction*, as shown in Figure 6. Masked language modeling prevents a word vector with multiple layers from seeing itself through previous layers, in turn, predicting itself. Figure 6 shows a bidirectional model without masking.

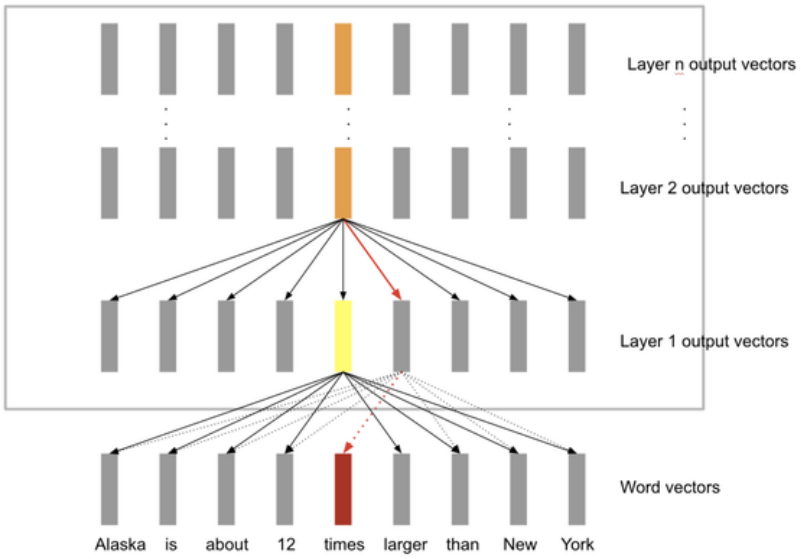

Figure 6. The problem with bidirectional language models without masking. The red lines display the word "times" seeing itself. From Ajit Rajasekharan (2019). What is a masked language model, and how is it related to BERT? – Quora. Retrieved June 21, 2021, from https://www.quora.com/What-is-a-masked-language-model-and-how-is-it-related-to-BERT

This bidirectional context makes BERT the current state-of-the-art method for natural language processing, improving RNN, long-short term memory (LSTM), and regular transformer models. However, since BERT contains numerous neural layers, it is a computationally intensive model. The issue of computational power is partially solved by having a pre-trained BERT model available that is trained for next sentence prediction and masked word prediction. Afterward, it can be finetuned for a wide range of tasks, decreasing the required training time (Devlin et al., 2019). The process of pre-training and fine-tuning is shown in figure 7 below.

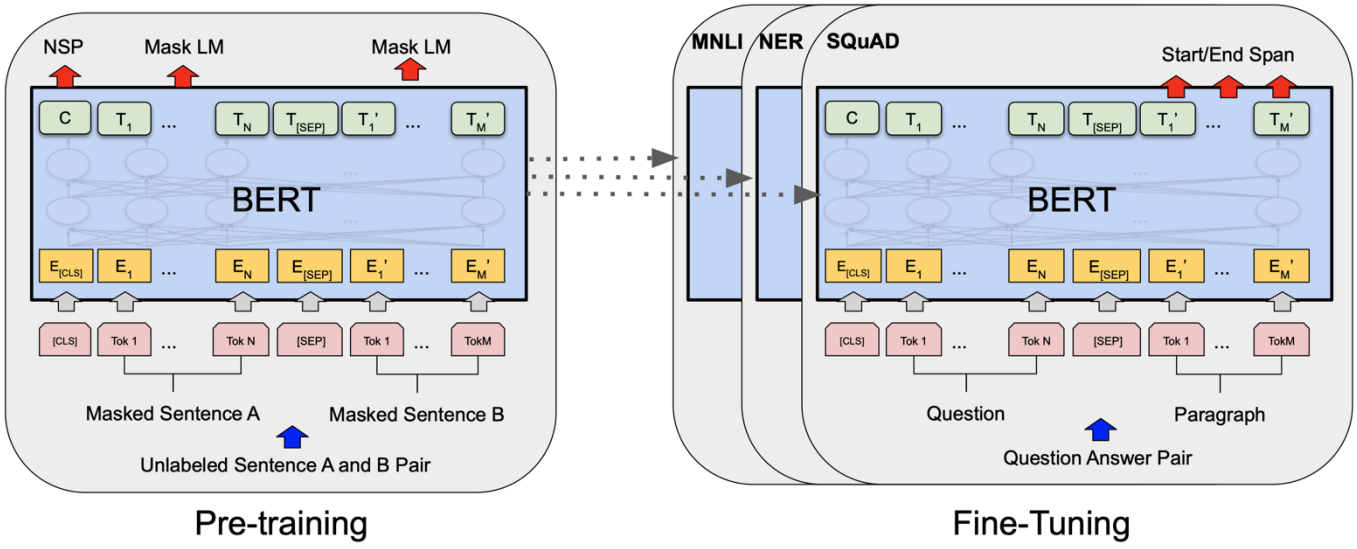

Figure 7. The architecture of BERT for pre-training and finetuning. From J. Devlin, M.W. Chang, K. Lee, & K. Toutanova (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT 2019 – 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies – Proceedings of the Conference*.

Masked LM is used because it is a language model trained in a bidirectional manner without masking. The task of next word prediction would be trivial for the model because the words can "see themselves" when predicting in the reverse order. With masking, 15% of tokens are randomly masked and predicted by the model. The downside of masking is that it creates a mismatch between the pre-training and the finetuning of the model. To mitigate this effect, when a token is 'masked,' it is replaced with a mask token 80% of the time, replaced with a random token 10% of the time, and preserved unchanged 10% of the time (Devlin et al., 2019). When the token is either masked or replaced with a random token, the model never learns the correct token. When the token is either masked or kept the same, the model copies the non-contextual embedding (Horev, 2018).

The other task BERT uses for pre-training is next sentence prediction. Next sentence prediction is a task where the model predicts whether sentence B follows sentence A. Sentence B is a random sentence from another place in the corpus 50% of the time. The other 50% is the sentence that follows sentence A (Devlin et al., 2019).

The pre-training data for BERT consists of the BookCorpus (800M words) and English Wikipedia (2,500M words) (Devlin et al., 2019). This elaborate pre-training leads to a language model that can fine-tune a specific task with relatively small datasets. The following sections explain two of BERT's relevant tasks that can be finetuned for relation extraction and performance as a knowledge base.

2.5.1 BERT for relation extraction

Relation extraction is the task of predicting the relation between two entities, given a sentence. BERT works well for relation extraction: a simple BERT-based model reached state-of-the art-performance out of all individual models (Shi & Lin, 2019). The model Shi and Lin used had no human-designed constraints or syntactic features, showing the power of BERT as a simple neural architecture for specific tasks.

Other research has expanded upon the BERT model, for example, by building a structured prediction layer to enable BERT to predict multiple relations with one-pass encoding, making it more scalable to larger datasets (Wang et al., 2020). Soares et al. (2020) have shown that the use of architecture for finetuning relation representations outperforms human accuracy in relational matching, and it is also effective in a low-resource environment (Soares, FitzGerald, Ling, & Kwiatkowski, 2020).

2.5.2 BERT as a knowledge base

Without fine-tuning, BERT already contains relation knowledge, similar to the older language model connected to a knowledge base, and is able to answer open-domain questions (Petroni et al., 2020). However, BERT is limited in size compared to a knowledge base such as Wikipedia (around 100 GB, compressed) and ConceptNet (which recommends around 300

GB of free disk space to download). Peters et al. (2019) used a knowledge attention and recontextualization layer (KAR) to benefit from the vast amount of information contained in these knowledge bases. The architecture of the KAR is shown in figure 8 below.

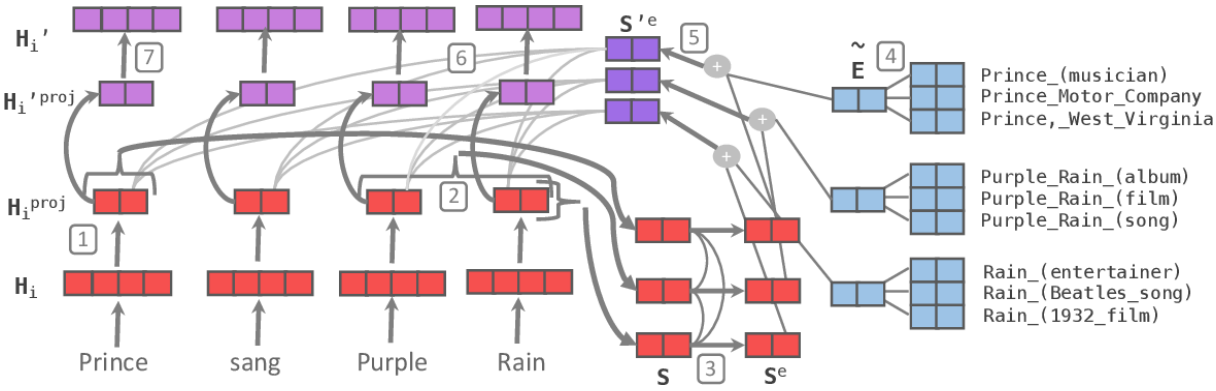

Figure 8. The KAR component. The KAR component. BERT word piece representations (Hi) are first projected to H proj i to fit the entity dimensions (1), then pooled over candidate mentions spans (2) to compute S, and contextualized into S e using mention-span self-attention, similar to a regular transformer layer (3). An integrated entity linker computes the weighted average entity embeddings E˜ (4), which in turn combines the span representations with knowledge from the knowledge base (5), computing S˙e. Finally, the BERT word piece representations are recontextualized with word-to-entity-span attention (6) and projected back to the BERT dimension (7), resulting in H'i. The red objects represent embeddings that are only based on the input, while the blue objects represent embeddings from the knowledge base. The purple objects represent embeddings that contain information from both the knowledge base and the sentence input. From "Knowledge enhanced contextual word representations." From M.E. Peters, M. Neumann, R.L. Logan, R. Schwartz, V. Joshi, S. Singh, & N.A. Smith. In *EMNLP-IJCNLP 2019 – 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*. https://doi.org/10.18653/v1/d19-1005

This KAR is the central component of KnowBERT. KAR consists of five main components: mention-span representation, an entity linker, knowledge-enhanced entity span representations, recontextualization, and alignment of BERT and the entity vectors.
The KAR is an extra layer added between two layers in the middle of a pre-trained BERT. Devlin et al. embedded WordNet and a subset of Wikipedia into BERT by using entity linkers. Entity linkers assign a unique identity to concepts in a text. The entity linker retrieves relevant entity embeddings from a knowledge base to form "knowledge enhanced entity-span representations."(Devlin et al., 2019). As a final step, the model recontextualizes these knowledge-enhanced entity-span representations using word-to-entity attention, allowing long-range interactions between the contextual word representing and all the entity spans used in that context (Peters et al., 2019).

3 Experimental setup

This thesis entailed two different experiments: one that created models capable of predicting spatial relations between tuples of objects and another that integrated these models into a pipeline for the scene graph generation. This chapter will discuss the dataset and its pre-processing for the spatial relation prediction task.

3.1 Dataset and pre-processing

Similar to many other scene graph generation experiments, the Visual Genome dataset was used for training because it contains many different relationships compared to similar datasets. Moreover, previous research has used the dataset as well, allowing for performance comparison. Visual Genome is a dataset consisting of 108K images, an average of 35 objects, 26 attributes, and 21 pairwise relationships. Figure 9 shows all data available in this dataset.

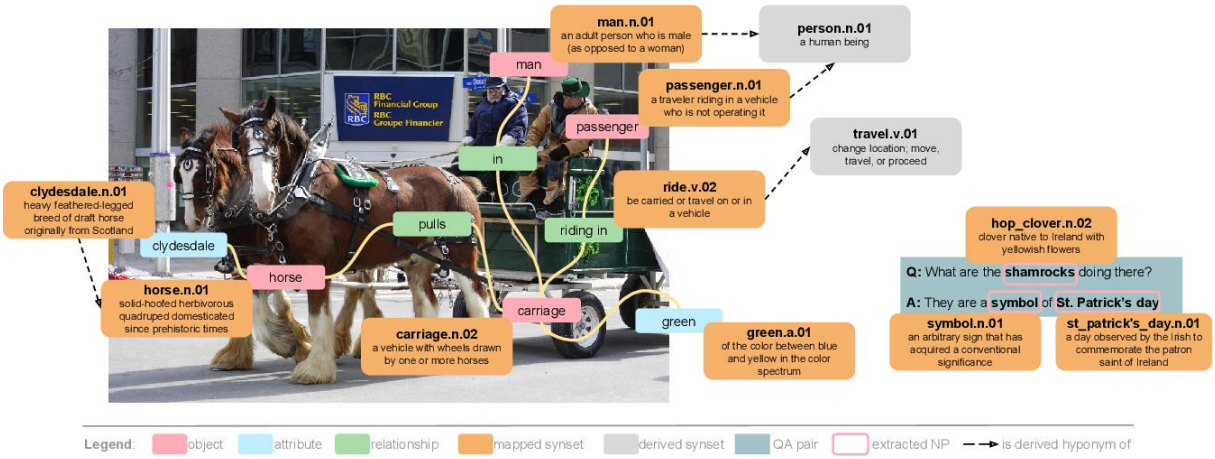

Figure 9. The information in the Visual Genome dataset. For this thesis, the triplets consisting of two objects (shown in red) and one relationship (shown in green) were used. From R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, … Saenko Ranjay K.B. Krishna (2017). Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *Int J Comput Vis*, *123*, 32–73. https://doi.org/10.1007/s11263-016-0981-7

In this thesis, only the triplets with two objects and a relationship were used in order to create a scene graph from a scene as input. The extra information in this image would be difficult to retrieve and use in the context of this task. The relationships of the triplets can be verbs (e.g., sits on, wears) or prepositions (e.g., on, with) describing relations between objects in an image. The graphs representing an image from Visual Genome can be described as scene graphs (Krishna et al., 2017).

Based on current research developments (Chen et al., 2019; Yu, Li, Morariu, & Davis, 2017; H. Zhang, Kyaw, Chang, & Chua, 2017; Zellers et al., 2018), two different datasets were derived from the Visual Genome dataset: 1) one training and testing dataset with the 100 most common relationships and 2) one training and testing dataset with the 50 most common relationships. The occurrence of the 10 most occurring relations is shown in Figures 10 and 11 below.

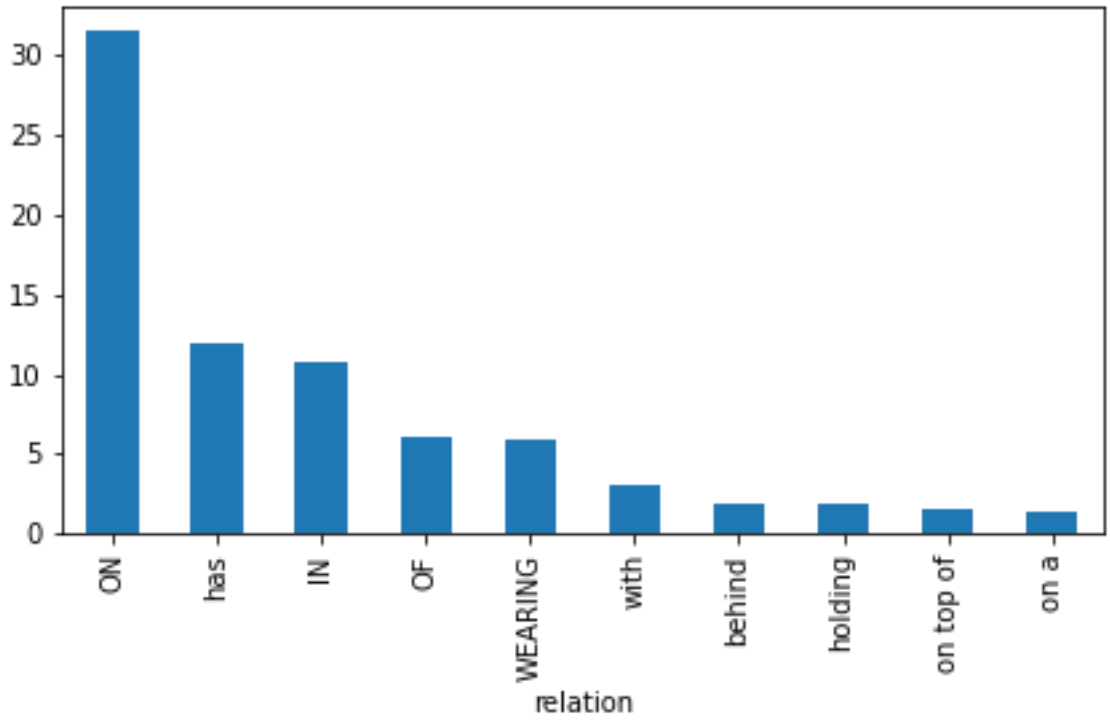

Figure 10. Distribution of the top ten relationships in the Visual Genome test dataset with the 100 most common relationships.

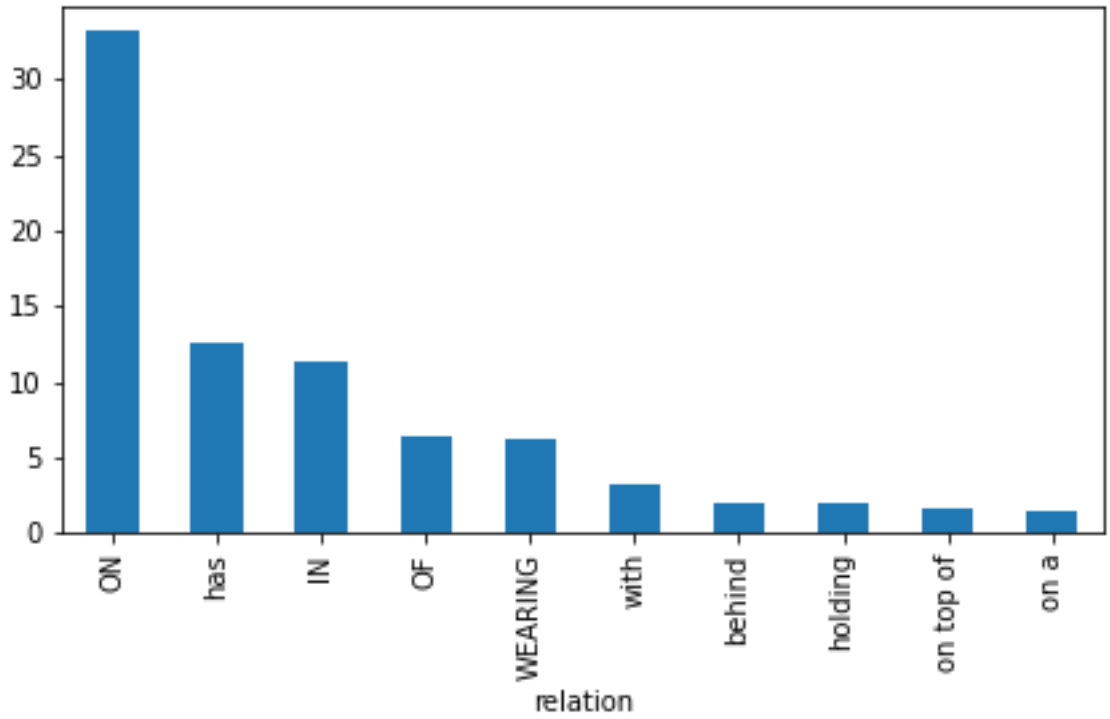

Figure 11. Distribution of the top ten relationships in the Visual Genome test dataset with the 50 most common relationships.

3.2 Method

The generation of the scene graph consisted of four phases. In the first phase, objects located at the scene were collected using the ConceptNet API. Next, all possible pairs between objects were stored as tuples. In the second phase, the spatial relations between these objects were predicted using one of the models. Based on these predictions, the most certain predictions were kept in the dataset. In the third phase, a scene graph was generated from this dataset using the RDFLIB Python library ("rdflib 5.0.0 – rdflib 5.0.0 documentation," n.d.). This pipeline is shown in Figure 12 below. These steps are elaborated in the following sections.

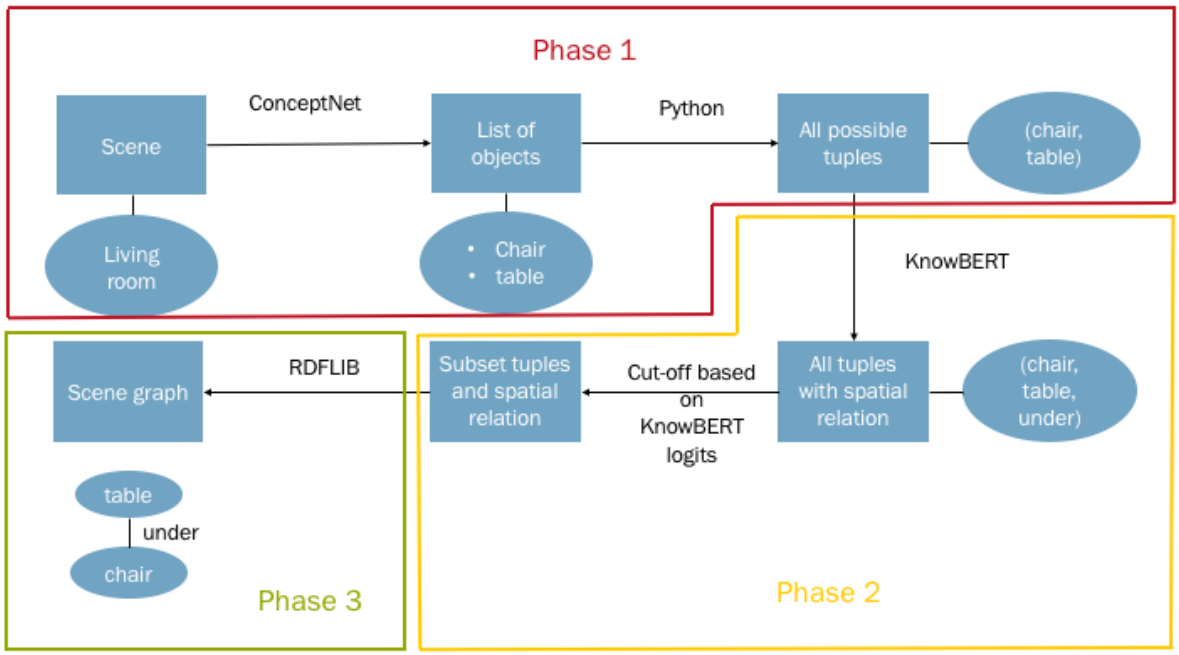

Figure 12. The pipeline of the model for scene generation. The data is shown in the squares, with examples of the data in the circles. The arrows show the transformation of the data. In phase one a list of tuples with two objects is created for the scene. In phase two, a spatial relation is generated for a subset of these tuples. In phase three, these triples of objects and their relation are used to generate a scene graph.

3.2.1 ConceptNet-based label generation

For this phase, the ConceptNet web-API was used in combination with the Python Requests library. All objects with the ConceptNet relation *AtLocation* for that particular scene were collected. An example for the scene *living room* is shown in Figure 13 below.

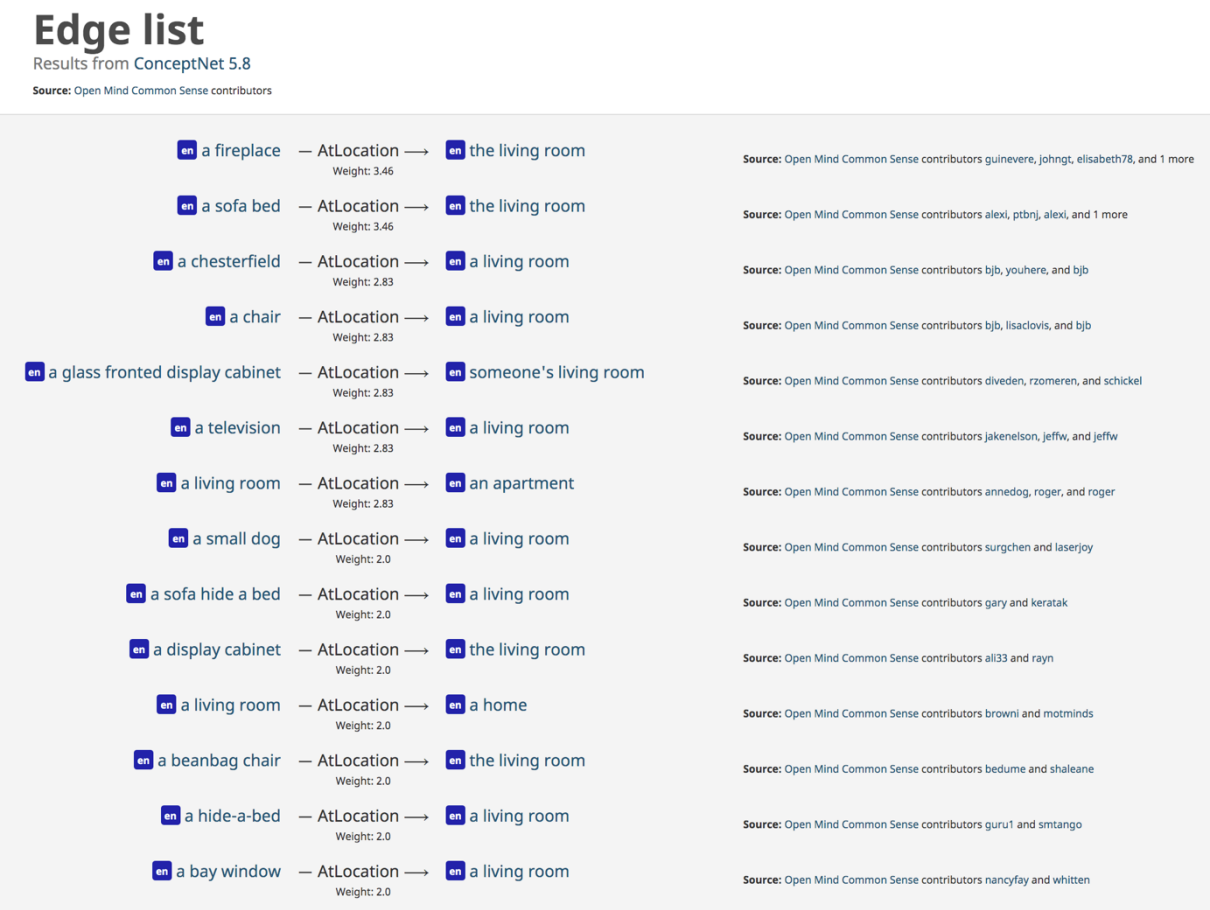

Figure 13. A subset of the items with the relation *AtLocation* for the scene *living room*.

3.2.2 Spatial relation predicting

In this section, I will discuss the different models that have been used in this thesis for the spatial relation prediction task. The spatial relation task consists of predicting the relation between two given objects within a scene. As a baseline, a statistical model is used. Furthermore, two different KnowBERT models have been trained for this task: a ConceptNet-trained KnowBERT model and a Wikipedia-WordNet KnowBERT model. The ConceptNet KnowBERT model is pre-trained for the purpose of this thesis. Both KnowBERT models are finetuned for spatial relation prediction for the purpose of this thesis.

The training pipeline of the two KnowBERT models is shown in Figure 14 on the next page.

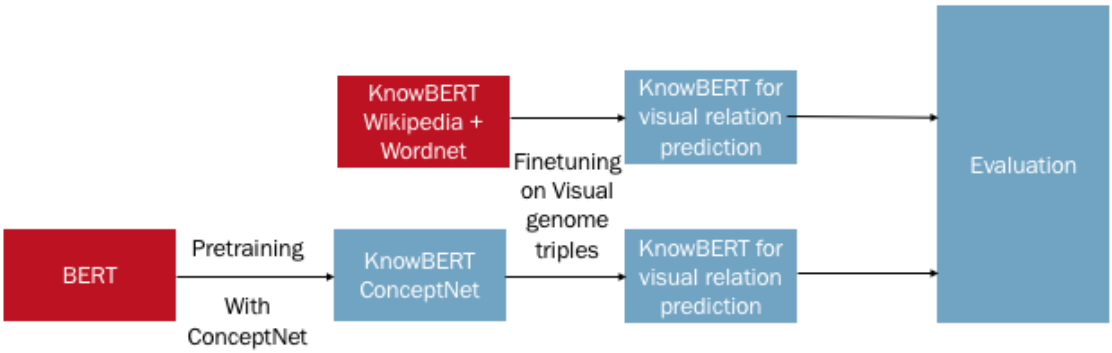

Figure 14. Creation of the Wikipedia-WordNet KnowBERT model and the ConceptNet KnowBERT model. As shown, the ConceptNet model was trained for this thesis, while the KnowBERT Wikipedia-WordNet model already existed. Both models were finetuned on the same dataset. The red models are downloaded, while the models in blue are trained for the purpose of this thesis.

3.2.2.1 Baseline

A statistical model was created to serve as a baseline. The model is not trained with a loss function. However, the model does not only predict the most-occurring relation for the whole dataset. The model first predicts the most-occurring relation for that specific tuple of objects if it exists in the training dataset. For example, if the model has to predict the relation between laptop and table, and "on top of" occurs 25 times in the training dataset and "next to" 22 times, it will predict "on top of." If the tuple of objects does not exist in the training dataset, the most common relationship is predicted, which is "ON".

3.2.2.2 ConceptNet

In order to create a ConceptNet-based version of KnowBERT, a guide for training a new KnowBERT model was followed as described on https://github.com/allenai/kb in the section "*How to pretrain KnowBERT*" (Peters et al., 2019). However, some adjustments were made to train BERT on the ConceptNet data. For example, synsets are not available in ConceptNet, so information could not be used for the embeddings. However, ConceptNet and WordNet also contain similar information categories. For example, the relations found in ConceptNet can be used for training using the same method. These relations were extracted using the ConceptNet API. Furthermore, ConceptNet and WordNet both contain examples. The generation of these ConceptNet datasets for training using entities and training using relations can be found on the GitHub of this thesis in the notebooks located at *clean_kb/kb/notebooks/*conceptnet_entity_creator.*ipynb* and *clean_kb/kb/notebooks/conceptnet_relation_creator.ipynb*. Due to size and training constrains, not all of ConceptNet was used. All entities in the visual genome dataset that are available in the English ConceptNet were used to create an entity file and a relation file. The entity file contained the entity and the 'is defined as' relation as the definition of the word. The relation file contained all relations each entity has. For example, the chair concept, shown in figure 5, would contain the relation triplets 'chair' 'is related to' 'sitting', 'chair' 'is related to' 'seat', chair' 'is located at' 'the office', etcetera. This relation file is then split up into a training and test file, with a 0.99/0.01 training/test split. This is the standard setting for pre-training a KnowBERT model. The entity file and the relation file are then combined into a

graph where each node is a concept. Then, TuckER (Tucker Entities Relations) embeddings are trained on extracted graphs. TuckER is a linear model for link prediction in knowledge graphs based on Tucker decomposition (Balažević, Allen, & Hospedales, 2019).The Tucker decomposition models a three-way data using a least-square approximation (Tucker, 1966). Then, a vocabulary file is generated based on the entities, including a separation ([SEP]) and classification ([CLS]) token. After that, GenSen embeddings are calculated on the definition of entities. The GenSen Framework uses a single recurrent sentence encored across multiple tasks (Subramanian, Trischler, Bengio, & Pal, 2018). The Tucker and GenSen are then combined into one file. To train the model supervised, the Semantic Concordance (SemCor) word sense disambiguation dataset is used. The SemCor dataset is a benchmark dataset for automatic sense identification (Miller, Chodorow, Landes, Leacock, & Thomas, 1994). The loss function is shown in equation 1 below. The function uses a position-wise multilayer perceptron (MLP). This MLP function is used to calculate ψ_{mk} (for the knowledge base embeddings) and ψ_{mg} (for the gold entities gathered from the SemCor dataset), as shown in equation 2 and 3.

$$\mathcal{L}_{\text{EL}} = -\sum_{m} \log \left(\frac{\exp(\psi_{mg})}{\sum_{k} \exp(\psi_{mk})} \right)$$

Equation 1. The loss function for pretraining the KnowBERT models. From M.E. Peters, M. Neumann, R.L. Logan, R. Schwartz, V. Joshi, S. Singh, & N.A. Smith. In *EMNLP-IJCNLP 2019 – 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*. https://doi.org/10.18653/v1/d19-1005

$$\psi_{mk} = \text{MLP}(p_{mk}, \mathbf{s}_m^e \cdot \mathbf{e}_{mk})$$

Equation 2. The MLP function for calculating ψ_{mk}.. p_{mk} represents the prior probability of the knowledge base, s^e_{mk} represents the associated mention span vector for the knowledge base, e_{mk} represents the candidate entity with its embeddings for the knowledge base. From M.E. Peters, M. Neumann, R.L. Logan, R. Schwartz, V. Joshi, S. Singh, & N.A. Smith. In *EMNLP-IJCNLP 2019 – 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*. https://doi.org/10.18653/v1/d19-1005

$$\psi_{mg} = \text{MLP}(p_{mg}, \mathbf{s}_m^e \cdot \mathbf{e}_{mg})$$

Equation 3. The MLP function for calculating ψ_{mg}.. p_{mg} represents the prior probability of the gold standard from the SemCor dataset, s^e_{mk} represents the associated mention span vector of the gold standard from the SemCor dataset, e_{mk} represents the candidate entity with its embeddings of the gold standard from the SemCor dataset. From M.E. Peters, M. Neumann, R.L. Logan, R. Schwartz, V. Joshi, S. Singh, & N.A. Smith. In *EMNLP-IJCNLP 2019 – 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*. https://doi.org/10.18653/v1/d19-1005

The training was performed in ten epochs, with 434 steps per epoch and a learning rate of 0.001. Further details on the parameters can be found in the configuration file in the GitHub repository at *clean_kb/kb/pretrain_conceptnet_kb/config.json*.

3.2.2.3 Wikipedia+WordNet

The pretrained Wikipedia+WordNet model was downloaded from the KnowBert GitHub. This WordNet part of the model is trained by Peters et al. (2019) using mostly the same method that was used for pretraining the ConceptNet model. For the Wikipedia apart of the model, Peterse et al. (2019) create candidate selectors and priors by combing CrossWikis, a precomputed dictionary that combines statistics from Wikipedia with a web corpus (Spitkovsky & Chang, 2012), and the YAGO2 dictionary, a knowledge base with focus on temporal and spatial knowledge (Hoffart et al., 2011) . The entity embeddings make use of a skip-gram method to learn embeddings of Wikipedia page titles. Unlike the WordNet and ConceptNet pretraining, the pretraining for the Wikipedia model does not use an explicit graph structure. Both the Wikipedia+WordNet and the ConceptNet model needed to be finetuned for spatial relation prediction.

3.2.2.4 Finetuning

The finetuning of the KnowBERT models was based on the finetuning method Peters et al. used for the text analysis conference relation extraction dataset (TACRED), another relation extraction task. The advantage of adjusting the TACRED task instead of building one from scratch is that the TACRED task was similar to our relation prediction task, saving time and effort. Furthermore, training the models was conducted using the AllenNLP terminal commands, as recommended by Peters et al. (2019). The labels in the dataset reader for the TACRED task were adjusted to read the Visual Genome dataset. This dataset reader can be found on the GitHub page of this thesis on *clean_kb/kb/kb/tacred_dataset_reader.py*. Furthermore, the Visual Genome was adjusted to the same structure as the TACRED dataset to facilitate the reading process. The notebook where this data is adjusted can be found at *../notebooks/conceptnet_relationcreator.ipynb*. Both models were trained using the standard setting of the KnowBERT configuration file with the batch size adjusted to fit the new training dataset. The batch size was 32 for three epochs, with 19,199 steps per epoch. The learning rate was 3e-05. Further information on the parameters can be found in the configuration files in the GitHub repository at *clean_kb/kb/pretrain_conceptnet_kb/config.json*, *clean_kb/kb/conceptnet_visgen_model_r100/config.json*, and *clean_kb/kb/wordnet_wiki_visgen_model_r100/config.json*.

3.2.3 Subset selection of spatial relations

Because the models will always create a relation between objects, a select subset from all the relations must be created to avoid a graph that is too cluttered and contains non-existing relations. The KnowBert model predicts the relation by choosing the highest logit in the last layer. The cut-off point in the logits was created for each scene. If the highest logit was over two standard deviations from the mean for that particular relation, the relation was kept. Otherwise, the relation was removed from the set.

For the statistical model, all relations that were seen in the Visual Genome dataset were kept.

3.2.4 Constructing a graph

For the subset of the spatial relations, a graph was constructed. In this graph, each object is shown once as a node, with an edge for each spatial relation with other objects. The RDFlib Python library was used to construct these graphs. The code for this process can be found in *clean_kb/kb/notebooks/scene_graph_generation*. RDFlib takes triples as an input to generate a graph. This step of the model does not transform the data, but allows the data to be visualized.

3.3 Evaluation

3.3.1 Visual relation prediction

The Visual Genome dataset was split into a 70/30 training/test dataset to compare the models. The training dataset contained 1,433,479 items, while the test dataset contained 614,348 items.

3.3.2 Scene graph generation

Five scene graphs were generated for each of the three models: a garden, a bathroom, a living room, a bedroom, and a kitchen. Using the model to generate scene graphs exemplifies the usefulness of the model, because it showcases the capabilities of the model in terms of common-sense knowledge. This knowledge can be used in tasks in tasks that would require an autonomous robot to gather information on expected objects and their respective spatial relations in a place it has not seen before.

4 Results and Discussion

The metric section compares the two KnowBERT models and the statistical model for the visual relation prediction task using the Visual Genome training and testing dataset. This comparison was done separately for the Visual Genome dataset with 50 different relations and one with 100 different relations. All models were trained and tested on a randomized train-test split three times. The tables show the mean performance with the variance of all three runs. The next section analyzes the predicted relations of the different models. Afterward, the performance of the different models for each relation is examined. Furthermore, only the tuples of objects that were not present in the training dataset are compared. The final section examines the different scene graphs that were generated.

4.1 Visual relation prediction

Table 1 shows the performance of the Wikipedia-WordNet and the ConceptNet model on the Visual Genome dataset. The statistical model, which had an accuracy score of 0.62, outperformed the Wikipedia-WordNet and the ConceptNet mode significantly, which had an F1 score of 0.55 and 0.52, respectively. Predicting only the most occurring relation would yield an accuracy of 0.36 for the 50 relation-subset and an accuracy of 0.29 for the 100 relation-subset, since this is the percentage of relations that correspond to 'ON'.

Models R50- relation dataset

	Wikipedia-WordNet	ConceptNet	Statistical Model
Training micro precision	0.65 ± 0.01	0.60 ± 0.00	-
Training micro recall	0.55 ± 0.01	0.48 ± 0.00	-
Training micro F1	0.69± 0.01	0.54 ± 0.01	0.73 ± 0.00*
Test micro precision	0.61 ± 0.02	0.59 ± 0.00	-
Test micro recall	0.50 ± 0.02	0.50 ± 0.01	-
Test micro F1	0.54 ± 0.03	0.54 ± 0.01	0.62 ± 0.04*

Table 1. Average performance and standard deviation for spatial prediction for the Visual Genome 50 relation subset.
*The statistical model's performance is measured through accuracy only.

Models R100- relation dataset

	Wikipedia-WordNet	ConceptNet	Statistical Model
Training micro precision	0.60 ± 0.01	0.57 ± 0.00	-
Training micro recall	0.50 ± 0.00	0.45 ± 0.01	-
Training micro F1	0.55 ± 0.00	0.50 ± 0.00	0.71 ± 0.00*
8Test micro precision	0.60 ± 0.00	0.57 ± 0.01	-
Test micro recall	0.50 ± 0.00	0.48 ± 0.01	-
Test micro F1	0.55 ± 0.00	0.51 ± 0.01	0.59 ± 0.01*

Table 2. Average performance and standard deviation for spatial prediction for Visual Genome 50 relation subset

*The statistical model's performance is measured through accuracy only.

All three models had a similar performance for the spatial relation prediction task. The ConceptNet and Wikipedia-WordNet had an F1-score of 0.52 and 0.55, respectively, while the statistical model scored 0.59. The relatively high performance of the statistical model is not surprising; other research has shown that triplet frequency is a strong predictor of the relationship (Zareian, Wang, You, & Chang, 2020).

4.2 Analysis of predictions

Models R50-relation dataset			
	Wikipedia-WordNet	ConceptNet	Statistical Model
Test Accuracy	0.53	0.56	0.36

Table 3. Performance of spatial relation prediction for tuples of objects that were not present in the training dataset.

Models R100-relation dataset			
	Wikipedia-WordNet	ConceptNet	Statistical Model
Test Accuracy	0.53	0.48	0.29

Table 4. Performance of spatial relation prediction for tuples of objects that were not present in the training dataset.

Models R50-relation dataset			
	Wikipedia-WordNet	ConceptNet	Statistical Model
Test Accuracy	0.17	0.18	0.19

Table 5. Performance of spatial relation prediction for tuples of objects that were present in the training dataset only once.

Models R100-relation dataset			
	Wikipedia-WordNet	ConceptNet	Statistical Model
Test Accuracy	0.56	0.50	0.44

Table 6. Performance of spatial relation prediction for tuples of objects that were present in the training dataset only once.

The baseline model performed significantly worse for unseen relations. The Wikipedia-WordNet and the ConceptNet KnowBERT models have an accuracy of 0.53 and 0.48, respectively, while the statistical model had an accuracy of 0.29. This result shows that the

KnowBERT model is far more generalizable. The result could be due to the fact that the knowledge base entities are used as embeddings in the model, while the statistical model only learns about the information in the training set.

As previously discussed, both KnowBERT models performed similarly for the unseen and seen data. The similarity in performance could be due to several factors. The slight performance edge of the Wikipedia-WordNet model may be due to the fact that it is trained on two knowledge bases; in addition, Wikipedia contains a significant amount of textual information on entities. In contrast, the ConceptNet model is trained on a subset of ConceptNet. Furthermore, WordNet synsets also provide information on similar entities, which, combined with Wikipedia, enables a large possibility for knowledge.

The performance of the models differs per relation. As shown in figure 15 and 16 on the next page, the ConceptNet model performs relatively good on relations that are more uncommon. ConceptNet performs better at predicting all relations that contain ‘on’ or ‘off’.

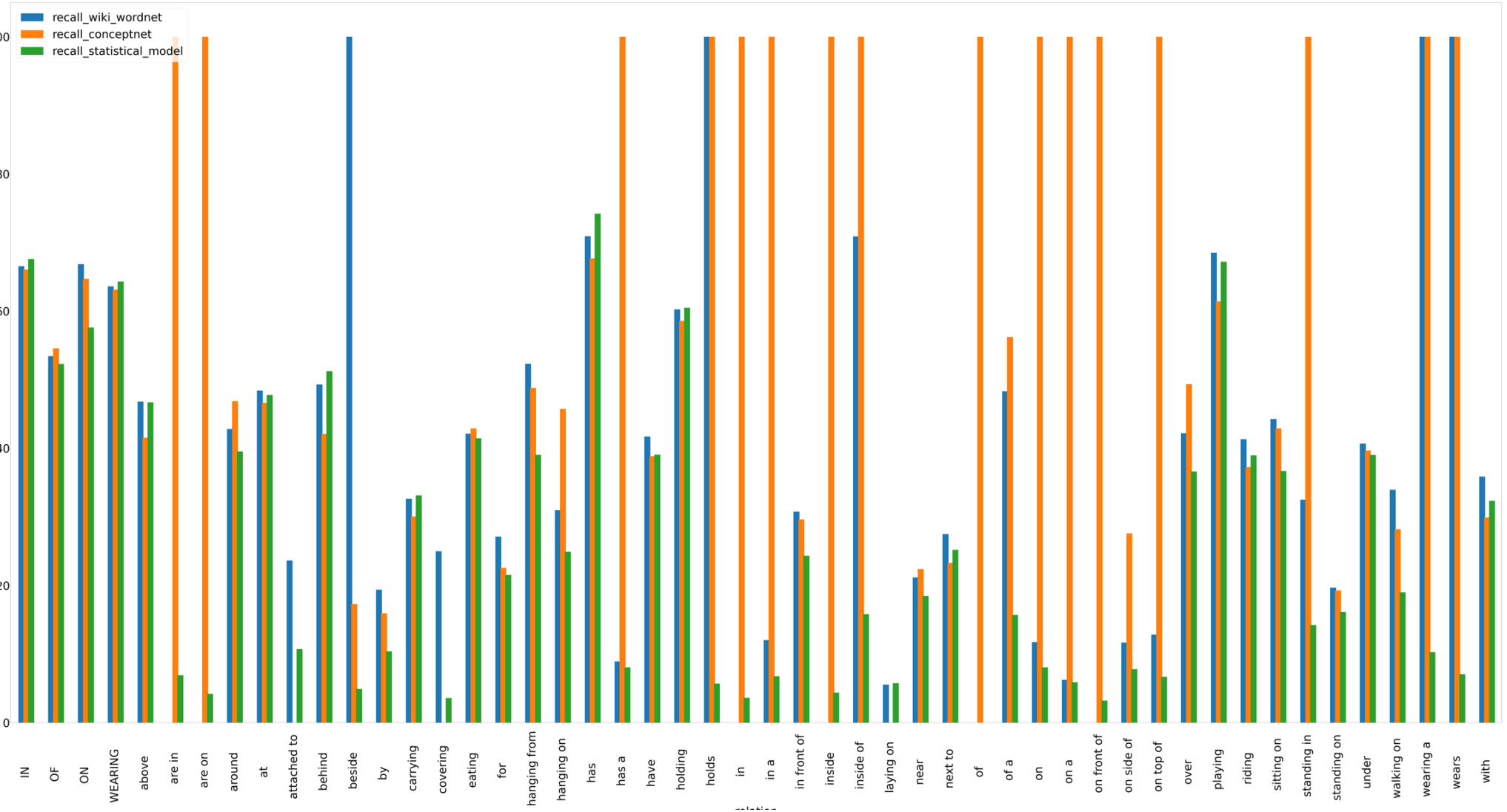

Figure 15. Recall per relation of the 50-relation Visgen dataset. The relation from left to right are ordered in order of number of occurrences, with the relation on the most left occurring the most. Each color represents one of the models.

4.3 Scene graph generations

In this section, I will show examples of scene graphs generated for five scenes: bathroom, bedroom, living room, garden and kitchen. These scene graphs are generated with three different models: the ConceptNet KnowBERT model, the Wikipedia-WordNet KnowBERT model and the statistical model. All three models are finetuned on the Visual Genome dataset, which contains 100 different relations. The scene graphs for the scene ‘bathroom’ are shown on the next page in figure 16 up until 18. The scene graphs for the other scenes are available in the appendix. The graphs for Wikipedia-WordNet KnowBERT model contain less nodes than the ConceptNet KnowBERT model. This means that the final layer of the ConceptNet model is focused more towards one relation, while the final layer of the Wikipedia-WordNet model has a lower standard deviation

In general, the scene graph generation task was conducted successfully. The capacity for scene graph generation shows one of the possibilities of a hybrid model. This hybrid model combines ConceptNet for gathering things located at a location in combination with a spatial relation predictor in the form of KnowBERT. Both the objects and relations in the scene graphs are sound in their respective context. Notably, most graphs are fully connected. This indicates that some objects have more certainty in all their relations. However, it should be noted that unconnected nodes are not possible for these models since the cut-off is based on relations.

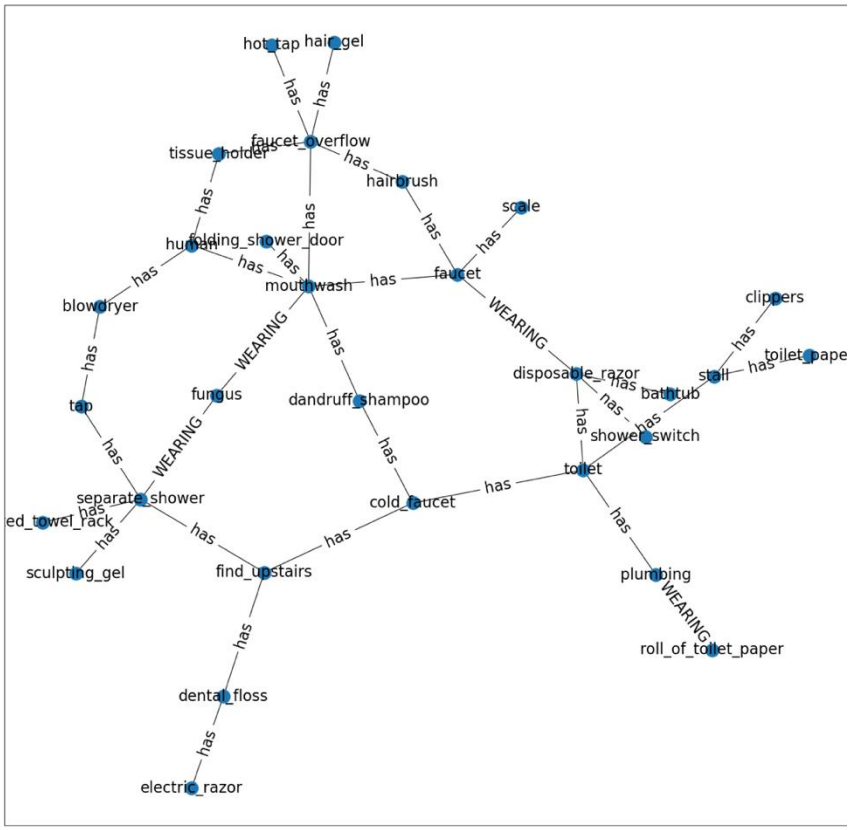

Figure 16. ConceptNet KnowBERT model generated scene graph for “bathroom.”

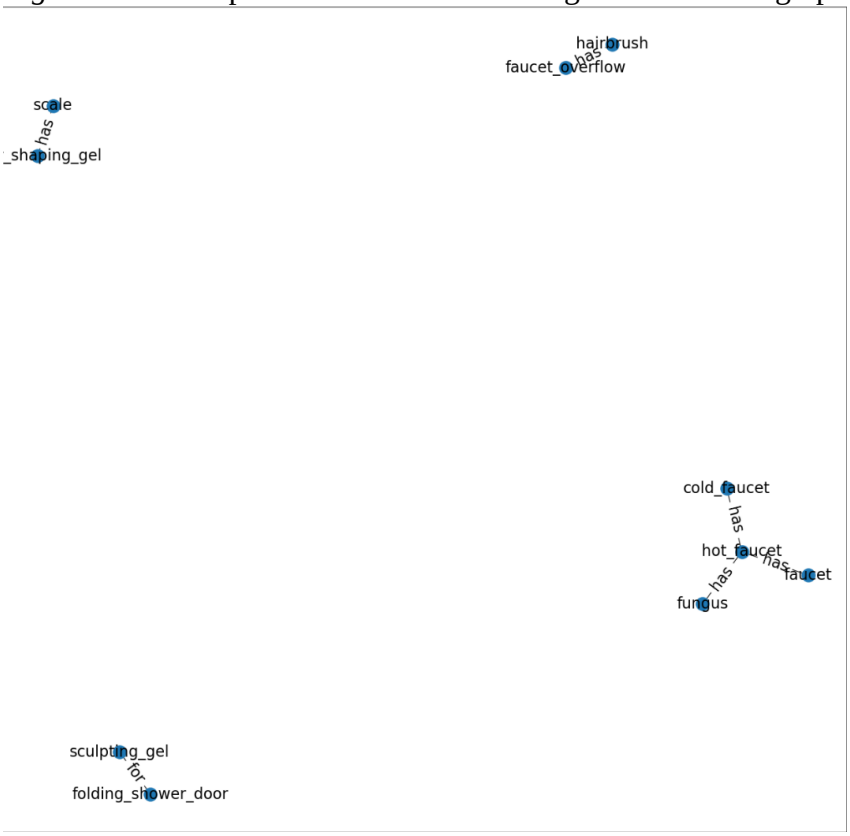

Figure 17. Wikipedia-WordNet KnowBERT model generated scene graph for “bathroom.”

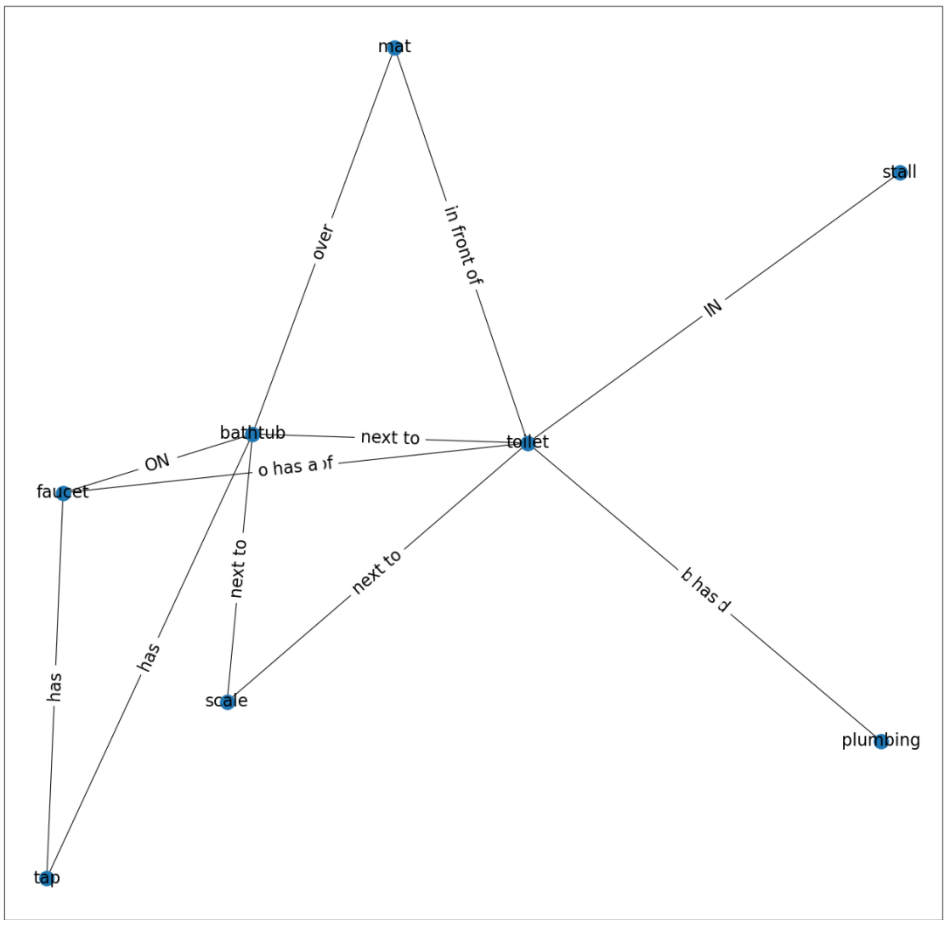

Figure 18. Statistical model generated scene graph for “bathroom.”

5 Limitations

When reviewing the limitations of this study, certain aspects of the spatial relation task must be considered. First, the model itself will never predict “no relation” between objects. The inability to predict ‘no relation’ was fixed in a later stage by implementing a threshold on the logits in the final layers. However, both models could also be trained to predict “no relation,” which could improve the consistency of the results. This problem can be seen in the scene graphs, such as the relation like ‘has’ for ‘small dog’ and ‘chair’.

Another limitation was the size of ConceptNet. While all entities in the training and testing dataset were used for the pretraining of the KnowBERT model, these entities still consist of only a subset of ConceptNet. This was done due to constraints in training time and graphics processing unit (GPU) availability. The current set-up took about eight hours of training; therefore, training on the whole of ConceptNet might not have been feasible. However, a model trained on the whole of ConceptNet would be more generalizable for finetuning other tasks.

Furthermore, Visual Genome dataset is not just focused on spatial relations. One could argue that the relations ‘with’ and ‘has,’ which are often present in the dataset, are not spatial relations. They describe a relationship but not a direction in space. The usefulness of these relations depends on the context. While the spatial relations do not provide information about the location of the objects relative to each other, they do indicate proximity.

Finally, the scene graphs were not evaluated on their ability to help in scene detection. Further research could focus on extending the pipeline of the model and trying to actually detect scenes in a computer vision context using the textual knowledge in these scene graphs.

6 Conclusion

In this work, a model has been developed to predict spatial relations between objects in a hypothetical scene. The main research question of this paper was:

"How can KnowBERT be used to generate common-sense scene graphs?"

This was done by finetuning the original Wikipedia + WordNet model as well as a ConceptNet model on tuples extracted from the visual Genome task. Both of these models achieved performance similar to the state-of-the art model of Xu, Zhu Choy & Fei-Fei (2017, with the Wikipedia-Wordnet model performing slightly better. Using these models, a pipeline was created that integrated the ConceptNet API with the spatial relation prediction models. With the ConceptNet API, objects that are located at the scene are collected using the AtLocation relation in ConceptNet. Then, all possible pairs between objects are stored as tuples. In the second phase, the spatial relations between these objects are predicted using one of the models. Based on these predictions, the most certain predictions are kept in the dataset using a threshold of a max logit-score of two standard deviations from the mean for that relation. From this dataset, a scene graph is generated using the RDFLIB Python library. This pipeline is successful at creating common-sense graphs. Furthermore, three sub questions were examined in this thesis:

SQ 1: Is ConceptNet or a Wikipedia-wordnet is better suited to predict spatial relations in a KnowBERT model?

The Wikipedia-WordNet model outperformed the ConceptNet model slightly in the 100-relation model (F1 = 0.55, F1 = 0.52) and the 50-relation model (F1 = 0.54, F1 = 0.53). This could be because the Wikipedia + WordNet model is trained on two knowledge bases. Wikipedia contains a lot of textual information on an entity, while WordNet synsets give information on what entities are related.

SQ 2: Which model performs better for spatial relation prediction, a statistical model or KnowBERT?

The statistical model predicts the most-occurring relation for that specific object pair. This model proved to be slightly superior over both KnowBERT models, with an accuracy of 0.58 for the 100-relation model and an accuracy of 0.59 for the 50-relation model. The dataset did contain many object pairs that were seen in the training dataset, which gives an advantage to the statistical model.

SQ 3: Which model performs better for spatial relation prediction on unseen object pairs, a statistical model or KnowBERT?

For unseen relations, all KnowBERT models perform far better than the statistical model. The model for 100 relations has an accuracy of 0.53 and 0.48 for the Wikipedia-WordNet and the ConceptNet model respectively against an accuracy of 0.29 for the statistical model. The model for 50 relations has an accuracy of 0.53 and 0.56 for the Wikipedia-WordNet and the ConceptNet model respectively against an accuracy of 0.36 for the statistical model.

6.1 Future research

One interesting avenue for future research is the use of Graph Neural Networks (GNN). Different state-of-the-art methods use GNNs for scene graph generation (Chen et al., 2019; A. Zareian; S. Karaman, & Chang, 2020). However, graph neural networks often have problems, such as over-smoothing. Graph-BERT is a graph neural network that uses an attention mechanism to address these problems (J. Zhang, Zhang, Xia, & Sun, 2020). Graph-BERT could be used in future studies, potentially combined with a knowledge base, to create scene graphs from the fully connected graphs created by the spatial relation prediction model and ConceptNet.

Another interesting model for future work would be a hybrid between a statistical model and a machine-learning model. While the statistical model outperformed the KnowBERT models, it performed far worse on unseen object pairs. A model that uses the statistical model for seen object pairs and the KnowBERT machine learning model for unseen object pairs would deliver superior performance.

7 References

Balažević, I., Allen, C., & Hospedales, T. (2019). TuckER: Tensor Factorization for Knowledge Graph Completion. *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, 5185–5194. https://doi.org/10.18653/V1/D19-1522

Bouman, A., Ginting, M. F., Alatur, N., Palieri, M., Fan, D. D., Touma, T., ... & Agha-Mohammadi, A. A. (2020, October). Autonomous spot: Long-range autonomous exploration of extreme environments with legged locomotion. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp. 2518-2525). IEEE.

Chen, T., Yu, W., Chen, R., & Lin, L. (2019). Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 6163-6171).

Boer, Maaike HT De, Yi-Jie Lu, Hao Zhang, Klamer Schutte, Chong-Wah Ngo, and Wessel Kraaij. "Semantic reasoning in zero example video event retrieval." *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 13, no. 4 (2017): 1-17.

de Boer, M., Schutte, K., & Kraaij, W. (2016). Knowledge based query expansion in complex multimedia event detection. *Multimedia Tools and Applications*, *75*(15), 9025-9043.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*.

Gu, J., Zhao, H., Lin, Z., Li, S., Cai, J., & Ling, M. (2019). Scene graph generation with external knowledge and image reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 1969-1978).

Hoffart, J., Suchanek, F. M., Berberich, K., Lewis-Kelham, E., De Melo, G., & Weikum, G. (2011). YAGO2: Exploring and querying world knowledge in time, space, context, and many languages. *Proceedings of the 20th International Conference Companion on World Wide Web, WWW 2011*, 229–232. https://doi.org/10.1145/1963192.1963296

Horev, R. (2018). BERT – State of the Art Language Model for NLP | Lyrn.AI. Retrieved April 13, 2021, from https://www.lyrn.ai/2018/11/07/explained-bert-state-of-the-art-language-model-for-nlp/#appendix-A

Ilievski, F., Szekely, P., & Zhang, B. (2021, June). Cskg: The commonsense knowledge graph. In *European Semantic Web Conference* (pp. 680-696). Springer, Cham.

Krishna, Ranjay, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen et al. "Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations." *International Journal of Computer Vision* 123, no. 1 (2017).

Liu, H., & Singh, P. (2004). ConceptNet—a practical commonsense reasoning tool-kit. *BT technology journal*, *22*(4), 211-226.

Miller, G. A. (1995). WordNet: a lexical database for English. *Communications of the ACM*, *38*(11), 39-41.

Miller, G. A., Chodorow, M., Landes, S., Leacock, C., & Thomas, R. G. (1994). Using a

semantic concordance for sense identification. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.
Nagatani, K., Kiribayashi, S., Okada, Y., Otake, K., Yoshida, K., Tadokoro, S., ... & Kawatsuma, S. (2013). Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots. *Journal of Field Robotics*, *30*(1), 44-63.
Pease, A., Niles, I., & Li, J. (2002, July). The suggested upper merged ontology: A large ontology for the semantic web and its applications. In *Working notes of the AAAI-2002 workshop on ontologies and the semantic web* (Vol. 28, pp. 7-10).
Peters, M. E., Neumann, M., Logan, R. L., Schwartz, R., Joshi, V., Singh, S., & Smith, N. A. (n.d.). allenai/kb: KnowBert -- Knowledge Enhanced Contextual Word Representations. Retrieved June 26, 2021, from https://github.com/allenai/kb
Peters, M. E., Neumann, M., Logan, R. L., Schwartz, R., Joshi, V., Singh, S., & Smith, N. A. (2019). Knowledge enhanced contextual word representations. In *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*. https://doi.org/10.18653/v1/d19-1005
Petroni, F., Rocktäschel, T., Riedel, S., Lewis, P., Bakhtin, A., Wu, Y., & Miller, A. (2019, November). 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).
rdflib 5.0.0 — rdflib 5.0.0 documentation. (n.d.). Retrieved June 28, 2021, from https://rdflib.readthedocs.io/en/stable/
Seong, H., Hyun, J., & Kim, E. (2020). FOSNet: An end-to-end trainable deep neural network for scene recognition. *IEEE Access*, *8*, 82066–82077. https://doi.org/10.1109/ACCESS.2020.2989863
Shi, P., & Lin, J. (2019). Simple BERT Models for Relation Extraction and Semantic Role Labeling. *arXiv preprint arXiv:1904.05255*.
Sivic, J., & Zisserman, A. (2008). Efficient visual search of videos cast as text retrieval. *IEEE transactions on pattern analysis and machine intelligence*, *31*(4), 591-606.
Soares, L. B., FitzGerald, N., Ling, J., & Kwiatkowski, T. (2019, July). Matching the Blanks: Distributional Similarity for Relation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 2895-2905).
Spitkovsky, V. I., & Chang, A. (2012, May). A Cross-Lingual Dictionary for English Wikipedia Concepts. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)* (pp. 3168-3175).
Subramanian, S., Trischler, A., Bengio, Y., & Pal, C. J. (2018). Learning General Purpose Distributed Sentence Representations via Large Scale Multi-task Learning. *International Conference on Learning Representations*.
Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrika 1966 31:3*, *31*(3), 279–311. https://doi.org/10.1007/BF02289464
Wang, H., Tan, M., Yu, M., Chang, S., Wang, D., Xu, K., ... & Potdar, S. (2019, July). Extracting Multiple-Relations in One-Pass with Pre-Trained Transformers. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics* (pp. 1371-1377).
Wikimedia statistics. (n.d.). Retrieved from https://stats.wikimedia.org/#/all-wikipedia-projects
Yang, J., Jiang, Y. G., Hauptmann, A. G., & Ngo, C. W. (2007, September). Evaluating bag-

of-visual-words representations in scene classification. In *Proceedings of the international workshop on Workshop on multimedia information retrieval* (pp. 197-206).

Yu, R., Li, A., Morariu, V. I., & Davis, L. S. (2017). Visual Relationship Detection with Internal and External Linguistic Knowledge Distillation. *Proceedings of the IEEE International Conference on Computer Vision*, *2017-October*, 1068–1076. Retrieved from http://arxiv.org/abs/1707.09423

Zareian, A., Karaman, S., & Chang, S. F. (2020). Bridging knowledge graphs to generate scene graphs. *European Conference on Computer Vision . Springer, Cham.*, (pp. 606-623).

Zareian, A., Wang, Z., You, H., & Chang, S.-F. (2020). Learning Visual Commonsense for Robust Scene Graph Generation. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12368 LNCS*, 642–657. Retrieved from http://arxiv.org/abs/2006.09623

Zellers, R., Yatskar, M., Thomson, S., & Choi, Y. (2018). Neural Motifs: Scene Graph Parsing with Global Context. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 5831–5840. https://doi.org/10.1109/CVPR.2018.00611

Zhang, H., Kyaw, Z., Chang, S. F., & Chua, T. S. (2017). Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5532-5540).

Zhang, J., Zhang, H., Xia, C., & Sun, L. (2020). Graph-Bert: Only Attention is Needed for Learning Graph Representations. Retrieved from http://arxiv.org/abs/2001.05140

Appendix A: all generated scene graphs

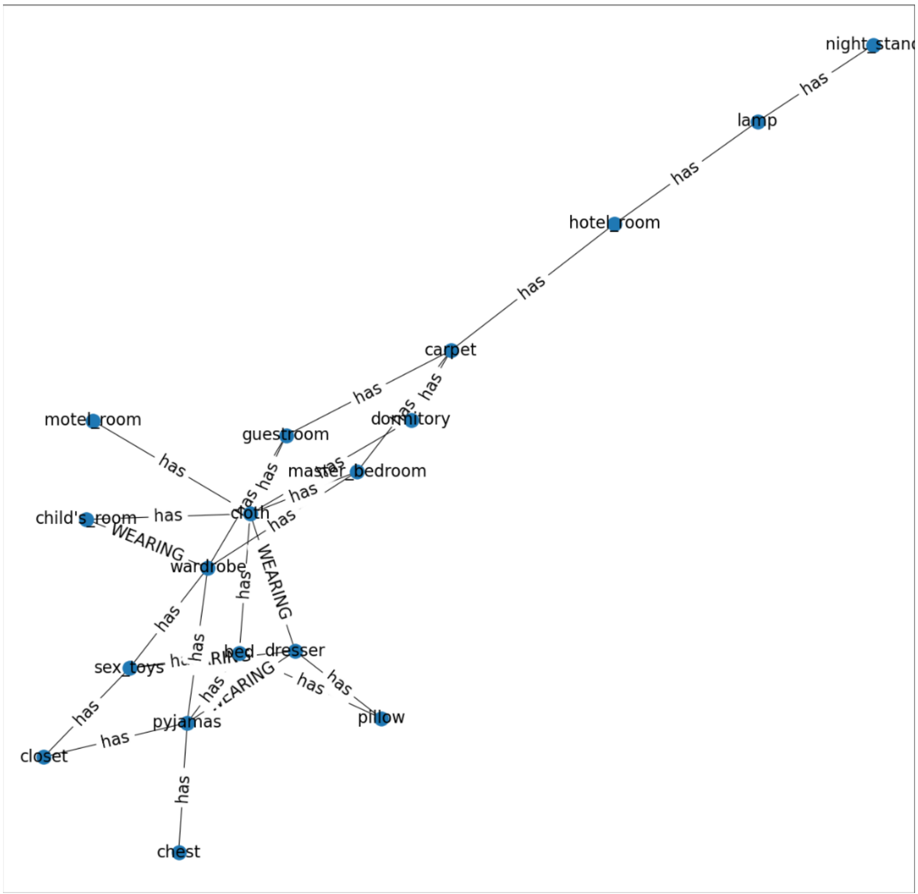

Figure 20. ConceptNet KnowBERT model generated scene graph for “bedroom.”

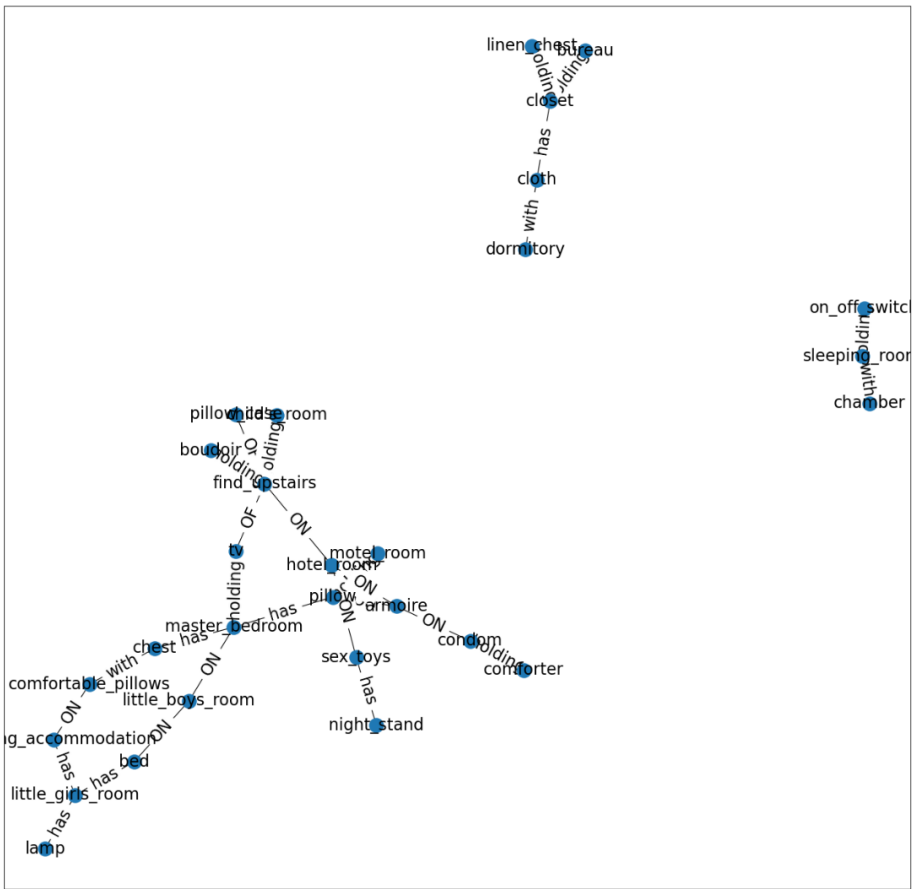

Figure 21. Wikipedia-WordNet KnowBERT model generated scene graph for bedroom.”

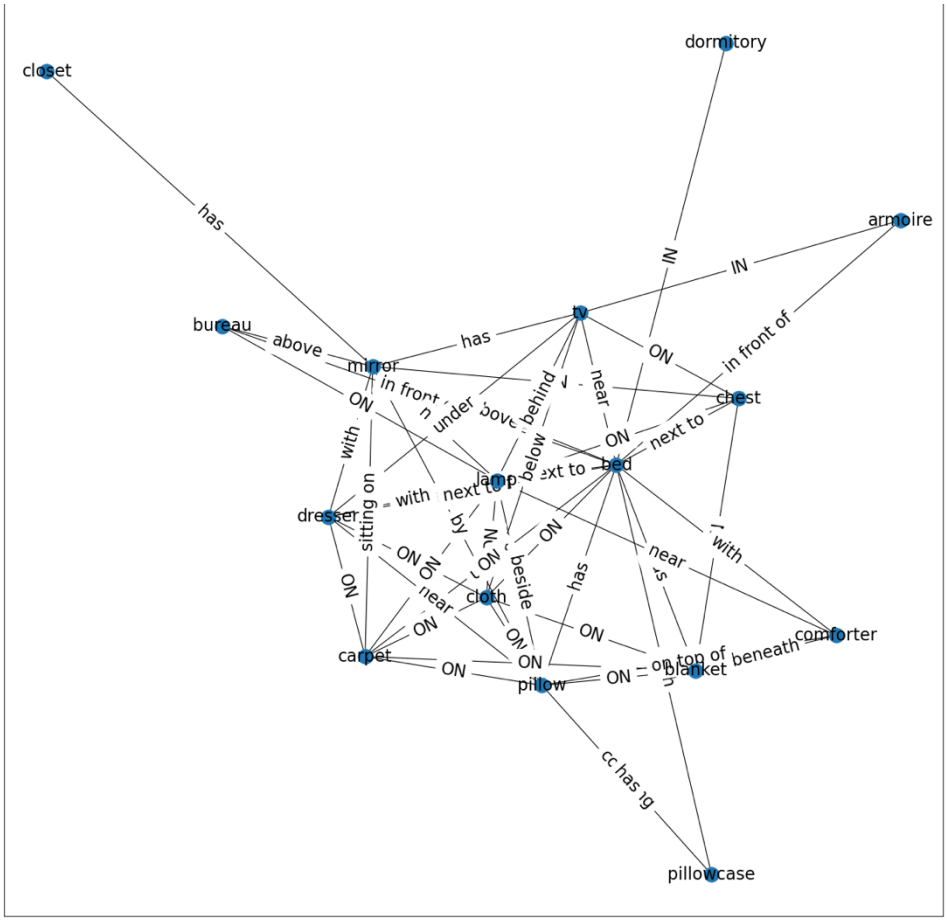

Figure 22. Statistical model generated scene graph for “bedroom.”

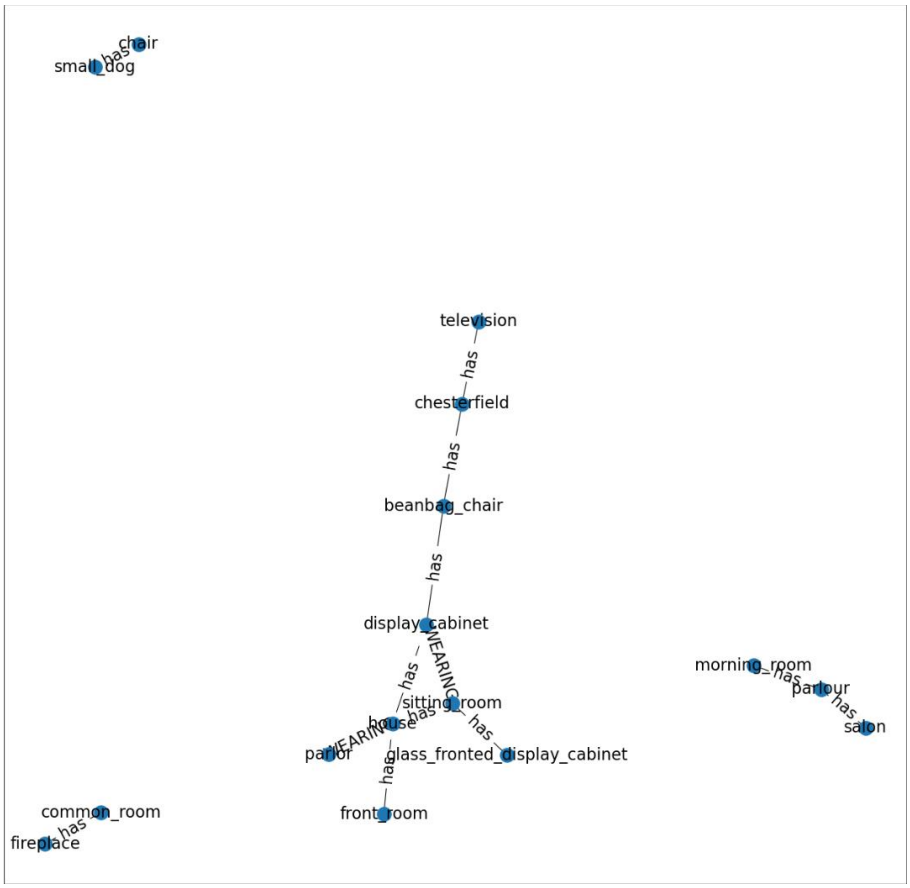

Figure 23. ConceptNet KnowBERT model generated scene graph for ‘living room’

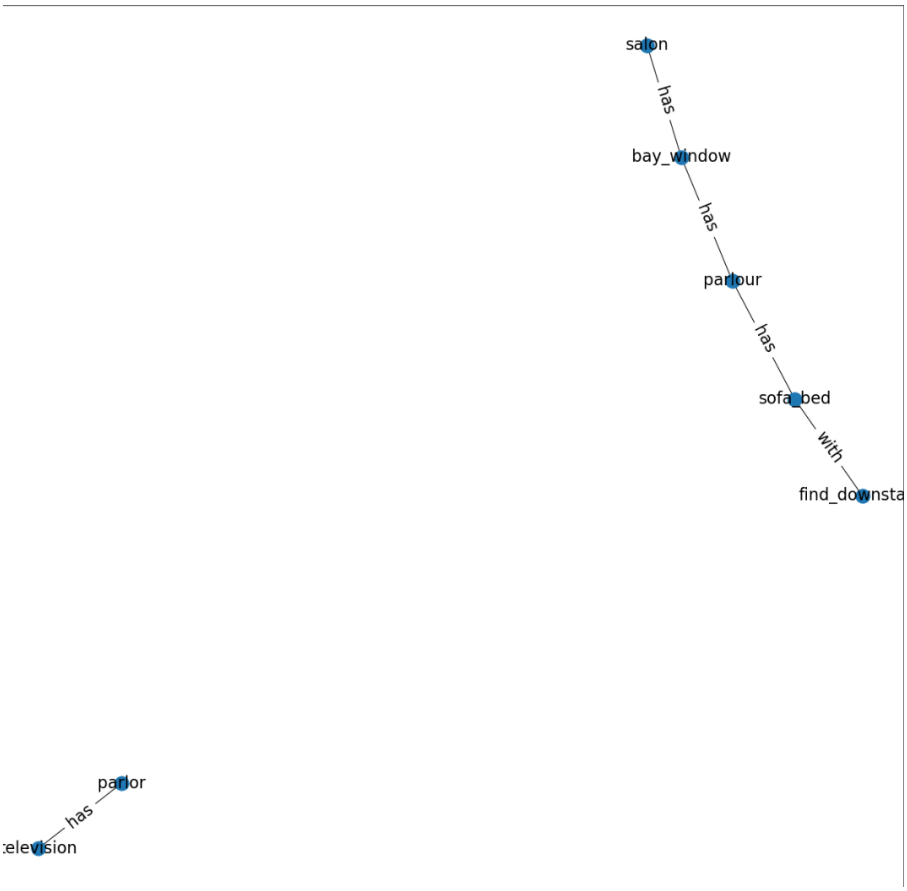

Figure 24. Wikipedia-WordNet KnowBERT model generated scene graph for living room.”

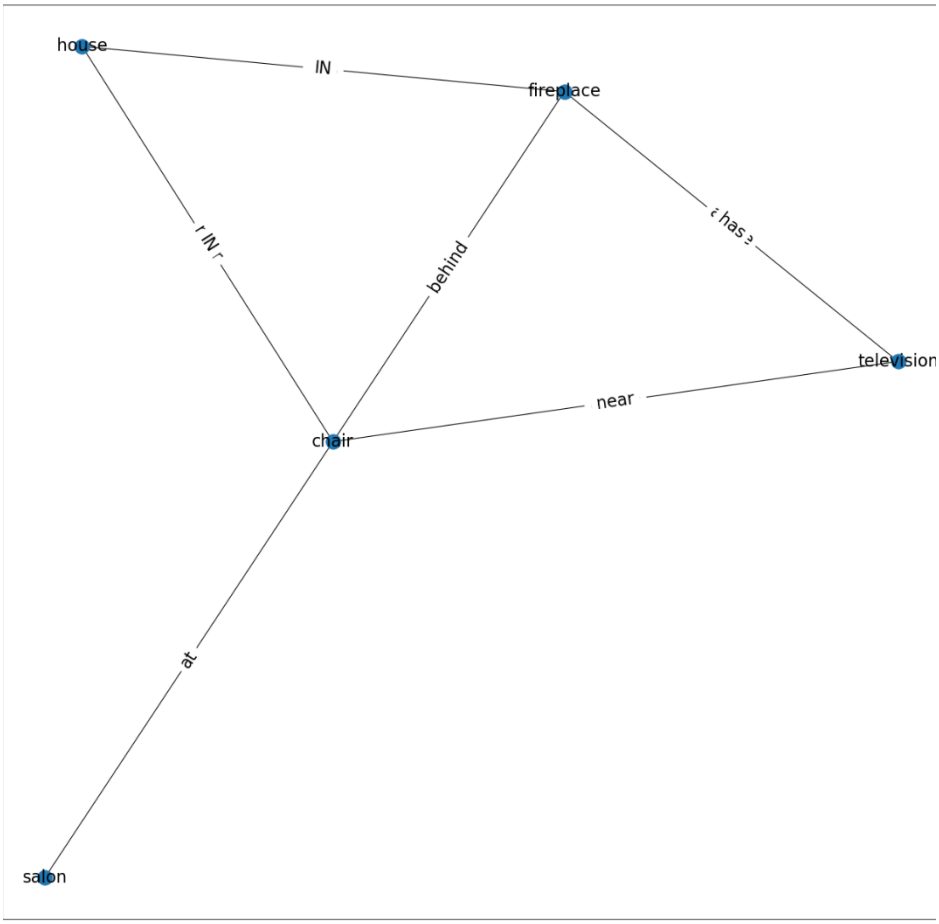

Figure 25. Statistical model generated scene graph for “living room.”

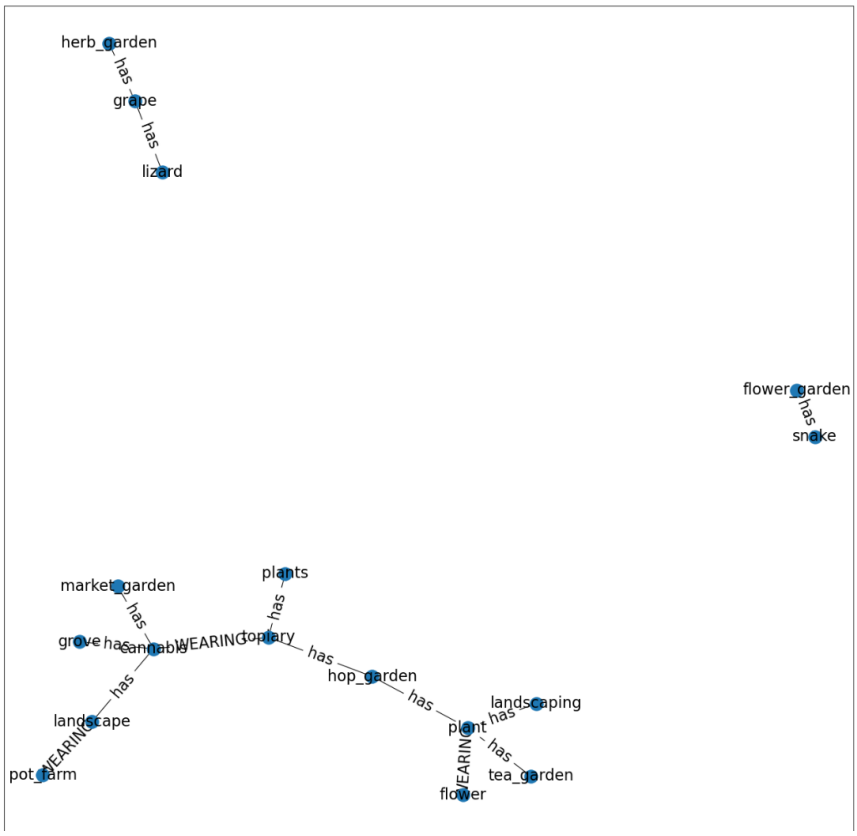

Figure 26. ConceptNet KnowBERT model generated scene graph for “garden.”

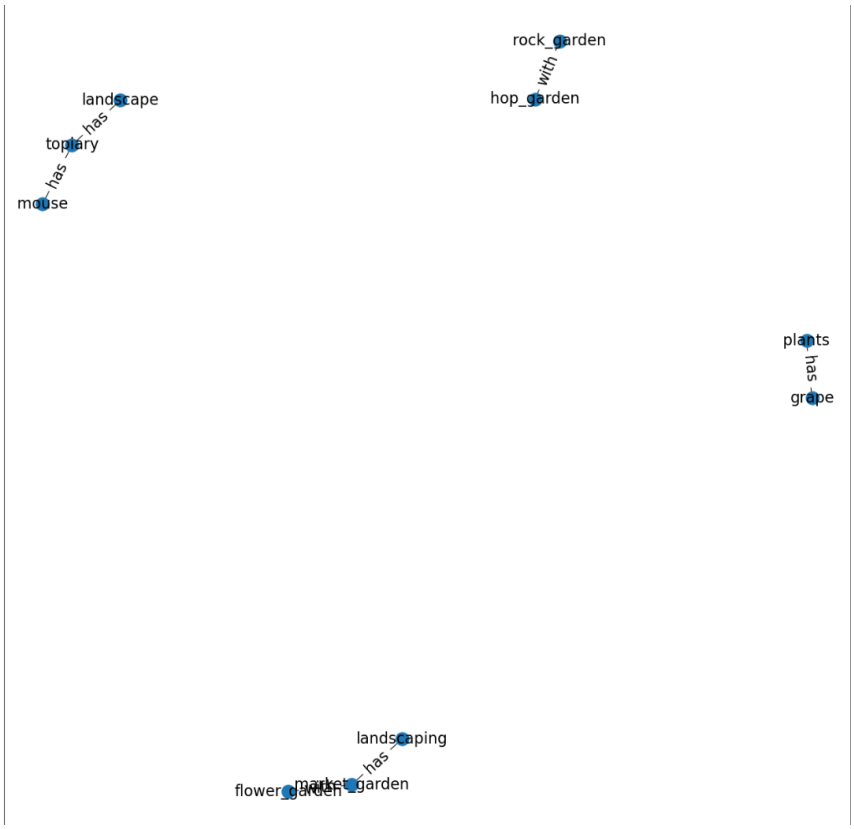

Figure 27. Wikipedia and wordnet KnowBERT model generated scene graph for “garden.”

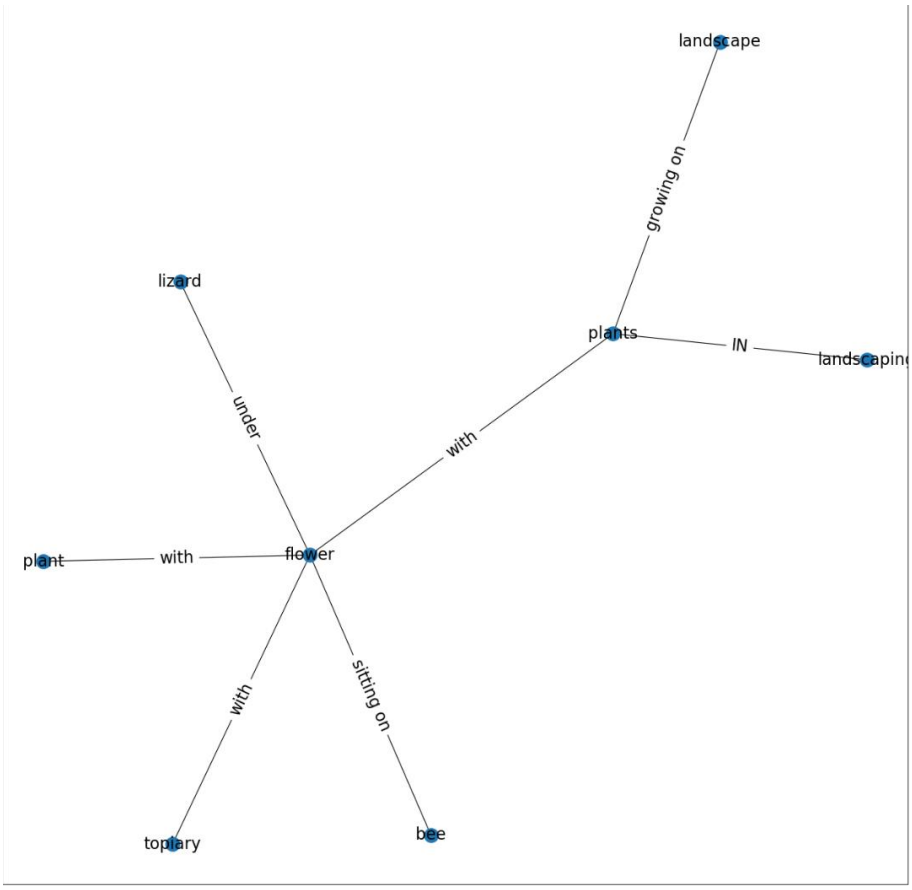

Figure 28. Statistical model generated scene graph for “garden.”

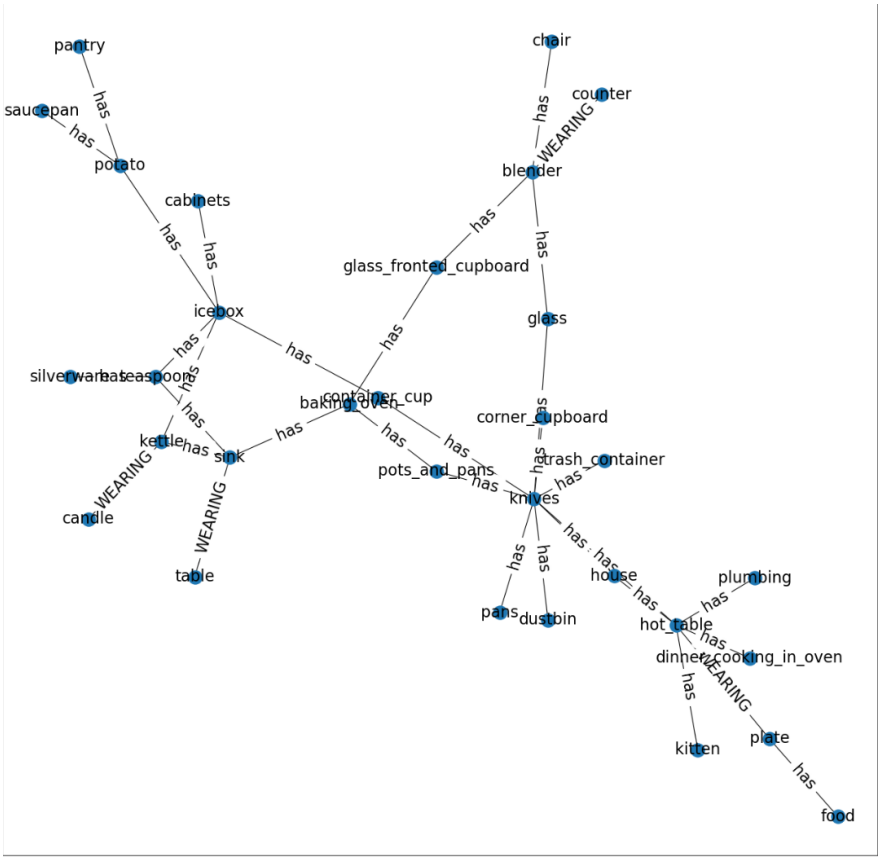

Figure 29. ConceptNet KnowBERT model generated scene graph for “kitchen.”

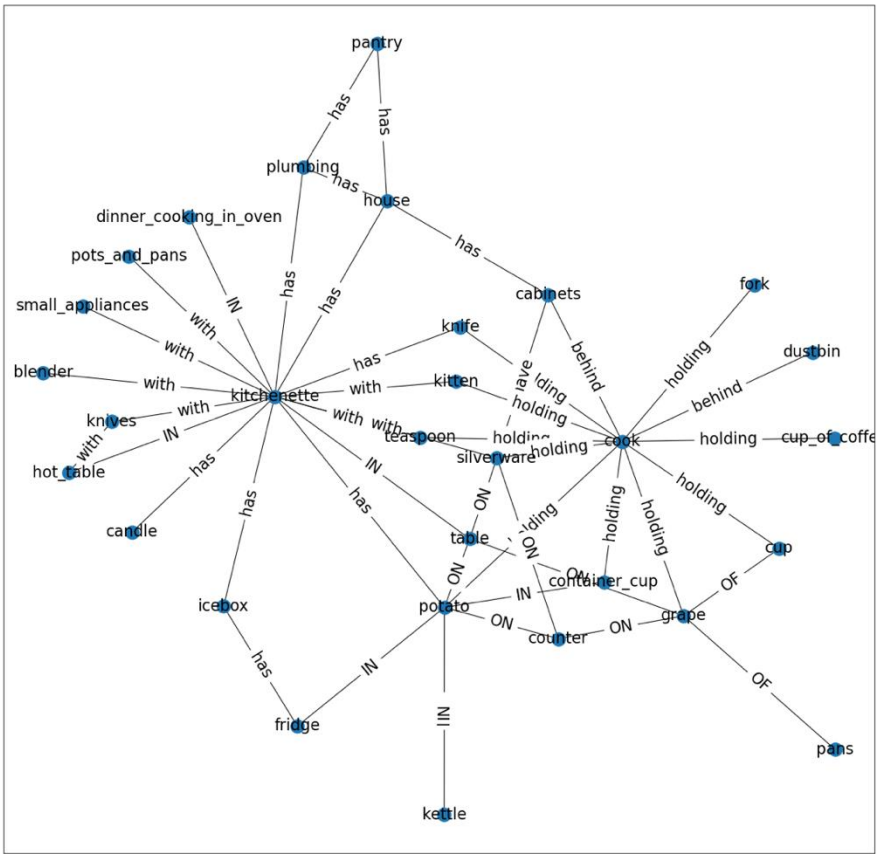

Figure 30. Wikipedia-WordNet KnowBERT model generated scene graph for “kitchen.”

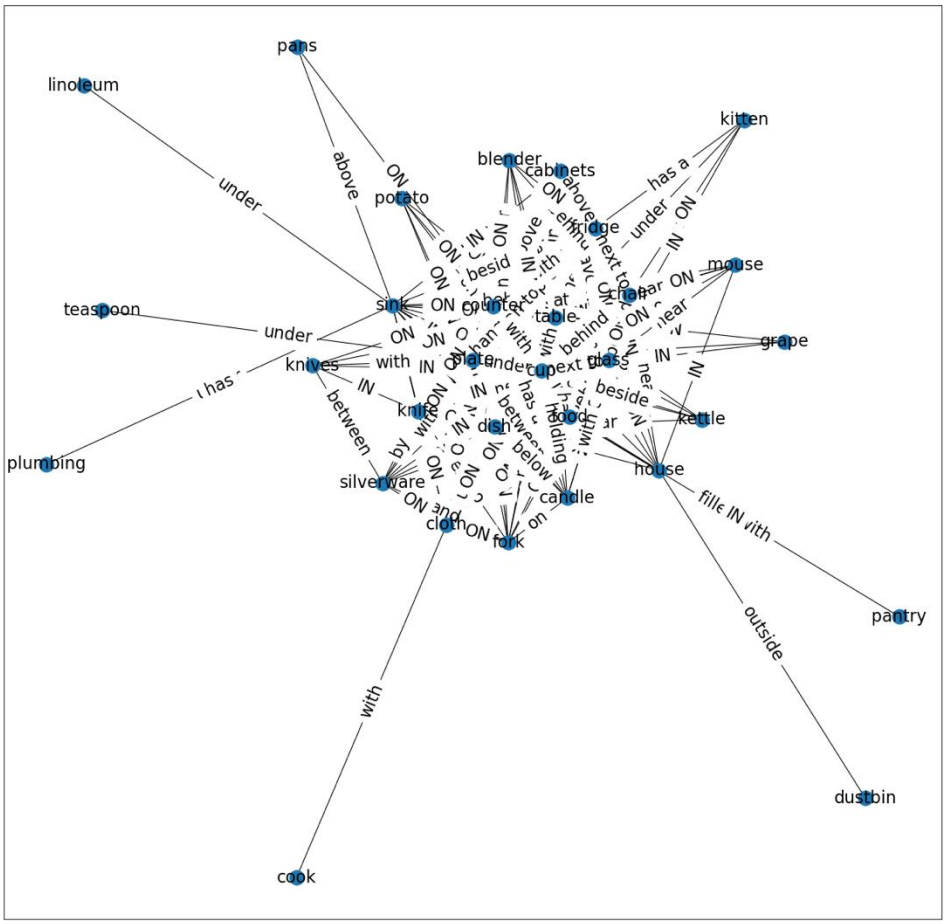

Figure 31. Statistical model generated scene graph for “kitchen.”