POLITECNICO DI TORINO

Master's Degree in Mathematical Engineering

Master's Degree Thesis

Integrating Knowledge Graphs With Logic Tensor Networks

Supervisor	Candidate
Professor Lia MORRA	Nicola DI SALVATORE

November 2024

Summary

Knowledge graphs are representations of knowledge structured as a graph. They are made of nodes (entities) that represent objects, edges (relationships) which represent connections or associations between nodes; attributes (properties) that store additional information about nodes or edges; and weights which indicate the strength or confidence of a relationship. This thesis investigates the integration of knowledge graphs such as ConceptNet and WordNet with Logic Tensor Networks (LTNs), a neuro-symbolic framework that combines first-order fuzzy logic with neural networks, to enhance scene graph generation.

The methodology includes aligning WordNet synsets with ConceptNet concepts and employing ConceptNet relationships like "IsA," "CapableOf," "NotCapableOf", and"Synonym" and ConceptNet embeddings (Numberbatch) to automatically generate first-order logic statements that work as input for Logic Tensor Networks, using the Visual Genome dataset as a foundation.

The study utilizes Logic Tensor Networks to inject prior knowledge into neural networks and guide scene graph generation, using the Visual Genome dataset as a foundation. Results highlight a dense number of first-order logic statement generation (1507 for each image), with a focus on range and domain constraints for predicates. The thesis also outlines potential algorithms for expanding automatic axiom generation, though these were not implemented due to time constraints.

This research demonstrates the value of combining symbolic reasoning with prior knowledge to improve the efficiency of AI systems in semantic image interpretation tasks.

Acknowledgements

I would like to express my deepest gratitude to everyone who supported me throughout the journey of completing this master's thesis.

First and foremost, I extend my sincere thanks to my thesis supervisor, Lia Morra, and my thesis advisors, Alessandro Russo and Francesco Manigrasso, for their guidance and insightful feedback. Their expertise has been fundamental in shaping this work.

I am profoundly grateful to the Politecnico di Torino for providing an inspiring academic environment and access to the resources needed to conduct the research.

A special thanks goes to my father, Angelo, my mother Tiziana, my sister Jessica, my grandparents Giuseppina, Luigi, Mirella, and Nicola, and my uncles and cousins Gabriella, Marco, Maria, and Michelangelo, for their love, patience, and encouragement.

Thanks also to my friends Alessandro, Francesca, and Lorenzo, who accompanied me for a lot of time in these mathematical adventures; I cannot forget Alessia, Alessandro, Arianna, Beatrice, Matteo, and Sofia, who managed to make the mathematical lectures fun during these two last years; and Andrea, Claudia, Edoardo, Elena, Emanuela, Marco, Martina, Riccardo, which accompany me from the first year, despite the different courses of studies. Their support and the memories we share have been very important during this challenging but rewarding journey.

I am grateful also to my flatmates Andrea, Duccio, Matteo, Ruggero, and Tiziano, for their everyday conversations and enjoyable moments spent together.

To my colleagues in the Department of Mathematical Engineering, thank you for your support that made this journey enriching and enjoyable.

To all who have contributed in any way, directly or indirectly, to this thesis, I extend all my thanks.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

In recent years, the integration of symbolic reasoning with deep learning has become a focal point in developing more interpretable and efficient AI systems. One promising approach to achieve this integration is through Logic Tensor Networks (LTNs), a neuro-symbolic framework that combines first-order fuzzy logic with neural networks. LTNs offer a way to embed logical reasoning into the learning process, enabling models to reason with uncertainty while leveraging the power of deep learning.

Knowledge graphs, on the other hand, serve as powerful stores of structured knowledge, typically organized as graphs containing nodes (representing concepts), edges (depicting relationships), and weights (signifying the strength or relevance of those relationships). These graphs encapsulate vast amounts of prior knowledge, which can be beneficial when incorporated into machine learning models, especially in domains that require reasoning over complex structures.

The main objective of this thesis is to explore how prior knowledge (in this case ConceptNet and its embeddings Numberbatch) can be employed to inject prior knowledge into neural networks through Logic Tensor Networks, with a particular focus on enhancing the performance of semantic image interpretation or scene graph generation.

Scene graphs are a crucial component of visual understanding, as they represent the relationships between objects and their attributes within an image. By utilizing knowledge from structured knowledge graphs, this work aims to automatically generate first-order logic statements that can guide neural networks in their interpretation of visual data (in this case it comes from Visual Genome).

In Chapter 3 the external knowledge bases and frameworks have been introduced. In particular, the Visual Genome dataset is the source of images and triples annotation; WordNet and ConceptNet contain the external knowledge injected (in particular, the IsA, $CapableOf$, $Synonym$, and $NotCapableOf$ relationships of ConceptNet); Numberbatch contains the embedding vectors, useful both in the

entity alignment process (among WordNet and ConceptNet) and in the first-order logic statements generation; finally, there is a section on Logic Tensor Networks, the framework that allows to inject prior knowledge into neural networks.

Chapter 4 presents the two main results of the thesis: the entity alignment and the first-order logic statements generation, with both the methodologies and the experimental results.

Lastly, in Chapter 5 there are ideas and algorithms to further expand the automatic first-order logic axiom generation, but they have not been implemented in the thesis for time reasons.

Chapter 2

State of the art

Logic Tensor Networks have been introduced in the paper [1]. It is a neuro-symbolic framework that integrates deep learning with symbolic logic. About the implementation of Logic Tensor Networks for the semantic image interpretation task, the reference works are in [2], [3] and [4]; a more detailed view of the implementation has been described in section 3.5.

Interesting points on the benefits of knowledge graphs in deep learning, together with very useful resources (such as the semantic web and machine learning knowledge graph) and information about the integration of knowledge bases and neural networks one of the first papers studied is [5], which contains also interesting research directions (in particular, the one regarding reasoning with existing Neuro-symbolic frameworks such as LTN).

The semantic web and machine learning system knowledge graph ([6]) has been a great source of papers in the field of Semantic Web and machine learning technologies, where semantic web technologies such as knowledge bases are combined with statistical models. In particular, various papers have been explored to analyze datasets, knowledge bases, the parts of semantic web resources utilized and how they have been integrated into the pipeline.

The paper [7] explores the potentiality of extracting knowledge from knowledge bases for association rule learning in the context of RDF data (that is, uncovering patterns of co-occurrence or dependencies among entities and their attributes). The resources utilized in this work include data from DrugBank and SPARQL queries, which are employed to extract and refine the semantic associations in the RDF data. Moreover, ontology infobox types from the knowledge base DBpedia are filtered on the basis of six core classes—Person, Organization, Place, Work, Event, and Species—, and approximately 300,000 triples are extracted (60,000 triples per class) from the Ontology Infobox Properties. DBpedia plays a crucial role in this process by measuring the quality of association rules through the assessment of rdf:type and rdfs:subClassOf relationships.

In the paper [8] Conceptnet knowledge has been integrated with Visual Genome datasets. In particular, information is extracted from Visual Genome objects in the questions or connected with those in the questions. About these objects, three types of ConceptNet relations are used: properties of entity words (e.g., $HasProperty$, $DefinedAs$, IsA, $HasA$, and $HasContext$), the spatial location of objects (e.g., $AtLocation$, $LocatedNear$, $PartOf$, and $SymbolOf$), and the tendency of objects (e.g., $MadeOf$, $UsedFor$, $ReceivesAction$, $RelatedTo$, $CapableOf$, and $MannerOf$). The external information is used to adaptively determine more relevant information to support the model and locate true solutions to visual questions.

In the paper [9] ImageNet classes have been integrated with WordNet to tag the images with labels that have no training examples available.

A similar knowledge injection is utilized in the paper [10], in which the ConceptNet concepts highly related (using word2vec, GloVe embeddings and cosine similarity) to ImageNet 22K classes are used as auxiliary training data for the CNN.

Instead, in the paper [11] ConceptNet is leveraged for the object detection task. In particular, knowledge from ConceptNet nodes, which are linked to "3d-build" and "Scannet" objects by both AtLocation (indicate in what environment the objects are often located) and UsedFor (describing common use-cases of the objects) relationships, are injected in an embedded form into the relative position vectors, (which incapsulate distance and direction of objects).

In [12] each Visual Genome instance (scene entity and scene predicate node) is connected to its corresponding class (commonsense graph entity and commonsense graph node). Then the knowledge is propagated through scene graphs and common sense graphs in an embedded form.

Lastly, a case in which knowledge bases have been adapted for predicate classification is the paper [13], where ConceptNet knowledge linked with dataset entities is used to generate a scene graph and WikiData knowledge makes the knowledge graph more densely connected.

Chapter 3

Background

In this paragraph, the background elements utilized in the thesis development have been reported. In particular, the first section is focused on the Visual Genome dataset, the source of images and triple annotations; the second, third and fourth sections illustrate respectively two knowledge bases (WordNet and ConceptNet), and the word embeddings Numberbatch, which contain the external knowledge injected in the form of first-order logic statements into neural networks, through the Logic Tensor Networks framework introduced in the fifth section.

3.1 Visual Genome

Visual Genome is a rich dataset and knowledge base developed by Krishna et al. ([14]) to enable computer vision tasks such as scene graph generation, object detection, and visual question answering. The main components of Visual Genome are:

- **Images:** there are 108,077 images obtained from the intersection of MS-COCO's 328000 images and YFCC100M's 100 million images; they range from 72 pixels wide to 1280 pixels wide, with an average width of 500 pixels;
- **Synsets:** they are the WordNet synsets that represent categories of objects, attributes and relationships. In Visual Genome there are more than 18K WordNet synsets;
- **Objects:** main entities present in an image. There are 3843636 objects, approximately 35 per image. They come from various categories: if are considered only the top 200 categories, there are 2239 objects per category;
- **Attributes:** entities that describe properties or qualities of objects. There are 28 million total attributes with 68111 unique attributes. On average, each

image in Visual Genome contains 26 of them, and each region contains on average 1 attribute;

- **Relationships:** the core components in our scene graphs that connect two objects, one acting as the subject and one as the object, with verbs or prepositions. There are 42374 unique relationships, with over 2347187 million total relationships;
- **Region Description:** regions in an image delimited characterized by a bounding box and a descriptive phrase. In Visual Genome, every image includes an average of 50 regions;
- : **Region Graph:** the union of objects, attributes, and relationships for a region description;
- **Scene Graph:** the union of all region graphs for an image, that is all the objects, attributes, and relationships from each region description for a specific image;
- **Question-Answers:** each pair consists of a question and its correct answer regarding the content of an image. There are 1773258 question-answers (QA) in the entire dataset; on average, every image has 17 QA pairs. They can be distinguished between pair freeform QAs, based on the entire image, and region-based QAs, based on selected regions of the image. We collect 6 different types of questions per image: what, where, how, when, who, and why.

In this thesis, it has been preferred to work with objects, attributes and relationships synsets, since they are more general and precise, making it easier to work with hierarchies in combination with other knowledge bases. For the preceding reasons, they represent a better starting point for the construction of FOL statements for LTN.

3.2 WordNet

WordNet is a knowledge base developed by Princeton University [15] characterized by a large lexical database structure. Its core elements are the synsets, that represent a certain concept and contain various words linked by the relation of synonymy. Moreover, these synsets are also characterized by their part of speech, which can be nouns, verbs, adjectives and adverbs. In addition, each synset also contains a set of lemmas, individual words with the same meaning as the synset, and a definition, that describes its meaning in a few words. But the most important thing about WordNet is that the synsets are linked to others through various types of relations.

The main relationships present in WordNet are the following:

- **Hypernym:** links a synset to a broader one (for example, dog is linked with animal by the hypernym relation);
- **Hyponym:** connects a synset to a more specific one (for example, animal is linked with dog by an hyponym relation);
- **Holonym:** connects a synset with one of its part (for example, tree is linked to leaf by the holonym relation);
- **Meronym:** links a synset with one of its whole synset (for example, leaf is linked with tree by a meronym relation);
- **Antonym:** connects synsets (in particular adjectives and adverbs) with opposite meanings;
- **Similar To:** links synsets with highly related meanings;

3.3 ConceptNet

ConceptNet is a multilingual knowledge graph developed to represent common-sense human knowledge, created by contributors to Commonsense Computing projects, Wikimedia projects, Games with a Purpose, Princeton University's, WordNet, DBPedia, OpenCyc, and Umbel ([16]).

Differently from WordNet, the main elements are not lexical entities but common-use general terms (that can be words or phrases) with a more flexible meaning, that do not need to have a precise definition. Moreover, ConceptNet has the structure of a knowledge graph, where concepts are nodes and relationships are expressed as labelled, weighted edges that connect them. The weights quantify the strength or confidence of the relationship based on the frequency and reliability of data sources, which include other knowledge bases such as Open Multilingual WordNet, Wiktionary, Open-Cyc, and DBPedia.

ConceptNet's relationships can be categorized into two main types: symmetric relations (bidirectional ones, that apply in both directions) and asymmetric relations (directed, not invertible relations). The following symmetric relationships are the ones present in Visual Genome:

- **Antonym:** connects opposites concepts;
- **DistinctFrom:** links two different concepts;
- **EtymologicallyRelatedTo:** connects concepts with the same etymological roots.
- **LocatedNear:** indicates that two concepts have spatial proximity;
- **RelatedTo:** captures a general association between concepts.
- **SimilarTo:** connects concepts with similar meanings.
- **Synonym:** groups concepts with nearly identical meanings;

Moreover, the asymmetric relationships found in Visual Genome are the following:

- **AtLocation:** describes that a concept is where the linked concept is commonly found;
- **CapableOf:** indicates that a concept is able to do the linked concept;
- **Causes:** a concept causes the event represented by the linked concept

- **CausesDesire:** the concept makes someone want the linked concept relationships;
- **CreatedBy:** connects a connect that represents a creation to the concept of its creator;
- **DefinedAs:** links a concept to its definition;
- **Desires:** describes indicates that a concept typically desires the linked concept;
- **Entails:** indicates actions that imply another;
- **EtymologicallyDerivedFrom**: indicates that a concept is etymologically derived from another one
- **ExternalURL:** points to relations outside ConceptNet;
- **FormOf**: indicates that a concept is an inflected form of the linked concept;
- **HasA:** represents that a concept possesses an other concept;
- **HasContext:** specifies situational or contextual relevance;
- **HasFirstSubevent:** a concept is an event that begins with the subevent given by the linked concept;
- **HasLastSubevent:** a concept is an event that ends with the subevent given by the linked concept;
- **HasPrerequisite:** indicates that a concept happens only if the linked concept happens;
- **HasProperty:** a concept has the property represented by the linked concept;
- **HasSubevent:** the linked concept is the subevent of the starting concept;
- **InstanceOf** and **IsA:** connect a concept to its category, similar to the hypernymy-hyponomy WordNet relationships;
- **MadeOf:** the concept is made of the linked concept;
- **MannerOf:** similar to the IsA relationship, but for verbs;
- **MotivatedByGoal:** says that a concept has as a point the linked concept;
- **NotHasProperty:** a concept hasn't the property represented by the linked concept;

- **NotCapableOf:** indicates that a concept is not able to do the linked concept;
- **NotDesires:** indicates that a concept typically not desire the linked concept;
- **PartOf**: part-whole relationships, similar to meronymy-holonymy relationships in WordNet;
- **ReceivesAction**: indicates that the linked concept can be done to the starting concept;
- **SymbolOf**: indicates that a concept symbolically represents another;
- **UsedFor:** indicates that the starting concept is typically used for the ending concept.

Moreover, restricting the relationships present between the Visual Genome entities, the ConceptNet relationships has been divided in 6 different categories:

- **Semantic Relationships:** $Antonym$, $DefinedAs$, $DistinctFrom$, $EtymologicallyDerivedFrom$, $EtymologicallyRelatedTo$, $FormOf$, $RelatedTo$, $SimilarTo$, $SymbolOf$, $Synonym$;
- **Spatial Relationships:** $AtLocation$, $LocatedNear$;
- **Ontological Relationships:** $InstanceOf$, IsA, $HasA$, $PartOf$, $MannerOf$;
- **Functional Relationships:** $CapableOf$, $Desires$, $HasProperty$, $NotHasProperty$, $Causes$, $CausesDesire$, $HasFirstSubevent$, $HasLastSubevent$, $HasPrerequisite$, $HasProperty$, $HasSubevent$, $MotivatedByGoal$, $NotCapableOf$, $NotDesires$, $ReceivesAction$, $UsedFor$;
- **Creation Relationships:** $CreatedBy$, $MadeOf$;
- **External Information Links:** $ExternalURL$.

Moreover, since Visual Genome is a dataset built for computer vision, the only types of relationships that have been taken into account for our task of building FOL statements for LTN are the semantic, spatial and ontological relationships. The structure of ConceptNet, which is a knowledge graph that connects general terms, makes it particularly useful to combine it with knowledge embedding frameworks such as ConceptNet NumberBatch, word2vec, GloVe: in fact, this allows for extracting information from it in numerical form.

3.4 ConceptNet Numberbatch

Knowledge embeddings are representations of entities and relationships from structured knowledge bases, mapped to vectors in a continuous and high-dimensional space. These embeddings allow the transformation in numerical form of the complex relationships and semantics in a knowledge graph, which makes them suitable for performing machine learning tasks.

Among the available knowledge embeddings, one of them is ConceptNet Numberbatch ([16]), a set of pre-trained word embeddings that associate each ConceptNet concept with a 300-dimensional array. The difference between ConceptNet Numberbatch and other embeddings is that Numberbatch is built by combining linguistic and commonsense knowledge from ConceptNet, while others include only the context in which words appear and others represent only relational knowledge. This characteristic makes it especially effective for tasks that need to capture real-world relationships, such as in the case of this thesis, where it is used in combination with the Visual Genome dataset.

In addition, Numberbatch was built by merging data from ConceptNet, word2vec, GloVe, and OpenSubtitles 2016, and is characterized by its out-of-vocabulary strategy, which makes it perform better in the presence of unfamiliar words. The strategy can be summarized as follows:

- For an unknown word in a language different from English, it tries to find an English word with the same spelling;
- If step 1 fails, it continues removing letters from the end of the unknown word until some known ones are found. If so, it averages the embeddings of those known words. Otherwise, if a single character has remained, the process is stopped.

In [16], ConceptNet Numberbatch has been evaluated on various tasks, performance measures and datasets, and it has been compared with various knowledge embeddings, representing only distributional semantics (word2vec, GloVe, and LexVec) and others represent only relational knowledge (ConceptNet PPMI). It is shown that it performs generally better than the other embeddings, making it an optimal choice for this thesis.

3.5 Logic Tensor Networks

Logic Tensor Networks is a neurosymbolic framework introduced by Serafini and Garcez in [1] that allows querying, learning and reasoning with data through a differentiable first-order logic language $\mathcal{L}$ called Real Logic.

In the case of this thesis, Logic Tensor Networks has been applied to the task of semantic-image interpretation (following the works in [2] and [3]) to improve scene graph generation, in combination with Visual Genome dataset and ConceptNet.

Real-Logic, the first-order logic language, has a signature (that is, the list of non-logical symbols) composed of the following elements:

- $\mathcal{C}$: set of constants symbols, where each constant is an object;
- $\mathcal{F}$: set of functions symbols, which transform one or more elements into another;
- $\mathcal{P}$: set of predicates symbols, representing relationships or properties;
- $\mathcal{X}$: set of variables symbols, that can assume different values.

In Real Logic, every constant and variable is interpreted as a real value tensor, each function as a tensor operation or real value function, and each predicate as a real value tensor or real value function with a codomain of [0,1].

More specifically, in the field of Logic Tensor Networks, this interpretation is made through a function called grounding, designed on the signature of $\mathcal{L}$, and made as follows:

- $\mathcal{G}(c) \in \mathbb{R}^n$; represents the grounding of variables $c \in \mathcal{C}$ by a real-valued vector in $\mathbb{R}^n$;
- $\mathcal{G}(f) : \mathbb{R}^{n \cdot \alpha(f)} \to \mathbb{R}^n$: represents the grounding of a function $f \in \mathcal{F}$, which takes in input a real-valued vector of dimension $n \cdot \alpha(f)$ (where $\alpha(f)$ indicates the arity of f, that is the number of arguments of f) and map it to a another n-dimensional real-valued vector;
- $\mathcal{G}(p) : \mathbb{R}^{n \cdot \alpha(p)} \to [0,1]$: represents the grounding of a predicate $p \in \mathcal{P}$, which takes in input a real-valued vector of dimension $n \cdot \alpha(p)$ (where $\alpha(p)$ indicates the arity of p, that is the number of arguments of p) and map it to a value between 0 and 1.

The output of $\mathcal{G}(p)$ (restrained in the continuous interval [0,1]) indicates the truth level of the predicate p (0 indicates it is completely false, 1 it is completely true). This means that the truth level is interpreted as an interval between 0 and 1 rather than a boolean True, or False. For this reason, Real Logic is defined as a fuzzy logic.

This grounding is utilized to build first-order logical statements, that are a combination of predicates, connectors (like AND, OR), and quantifiers (like $\forall, \exists$), and are the components of the knowledge base $\mathcal{K}$. In this thesis, the referring grounding for predicates is the one present in [4].

Let Θ be the set of parameters, and $\mathcal{G}(\cdot, \Theta)$ the grounding obtained by setting the parameters of the grounding functions to Θ, the Logic Tensor Networks objective can be stated as an optimization problem which aims to find the set of parameters $Theta^*$ that maximize the truth values of all statements in the knowledge base $\mathcal{K}$:

$$\Theta^* = \arg\max_{\Theta} \mathcal{G}\left(\bigwedge_{\phi \in K} \phi \middle| \Theta\right) - \lambda \|\Theta\|_2^2,$$

where the first term represents the grounding of the conjunction of all the elements in the knowledge base $\mathcal{K}$, and the second term is a regularization parameter.

Chapter 4

Methodologies and Experimental Results

This chapter shows the two core elements of the thesis: the entity alignment process among Wordnet synsets and ConeptNet concepts, which allows the link of each object/attribute/predicate in Visual Genome to a ConceptNet concept, and the first-order logic statements generation, the algorithms to generate the input data for logic tensor networks. Both the alignment process and the statements generation are presented in a specific section, composed of the methodology employed and the experimental results obtained.

4.1 Entity Alignment

Entity alignment is a process that aims to identify entities from different knowledge graphs that describe the same real-world thing. In this thesis, the knowledge bases utilized for the alignment are Visual Genome (in particular, only the objects, attributes and relationships synsets are utilized), ConceptNet, which focuses on common sense knowledge, and Numberbatch embeddings. Then, the point of the entity alignment is to associate each Visual Genome's object, predicate, and attribute to a ConceptNet concept. This process is executed to take advantage of the ConceptNet and Numberbatch knowledge, which can be leveraged to build FOL statements for Logic Tensor Network.

4.1.1 Methodology

For each Visual Genome entity, the entity alignment algorithm has been made with two main points:

- to have an aligned concept highly similar to the WordNet synsets associated with the entity;

- to have a precisely aligned concept (that is, the aligned concept should be quite better than the second, the third, the fourth and the fifth best alignments, also at the cost of dropping potential good alignments).

In particular, the second objective has been reached using a Part Of Speech filter: using "spAcy", an open-source library for natural language processing in Python, for each Visual Genome element the part-of-speech tag for synsets and concepts are obtained, and the concepts with part-of-speech different from the synsets lemmas one are filtered out. This caused a great restriction on the pool of admissible concepts, but at the same time, the better concept found has a high similarity and is far better than the other ones.

In particular, the algorithm for the alignment is the following:

1. The objects, predicates and attributes data frames are loaded;

2. For each object, predicate and attribute data, all the synsets (if they exist) are queried in Wordnet to retrieve the respective lemmas;

3. Each lemma is adapted to a format compatible with ConceptNet concepts (such as "/c/en/dog") and the Part-Of-Speech of the first synset is stored;

4. Conceptnet Numberbatch embeddings are used to obtain the embedding of each concept found in point 2 and, for each data, the mean value between all the concepts is calculated;

5. Considering all the lemmas embeddings previously calculated, we find the 20 most similar concepts (using the cosine similarity as metric) to the mean value calculated in the previous step and we select those we have the same Part Of Speech found in step 3;

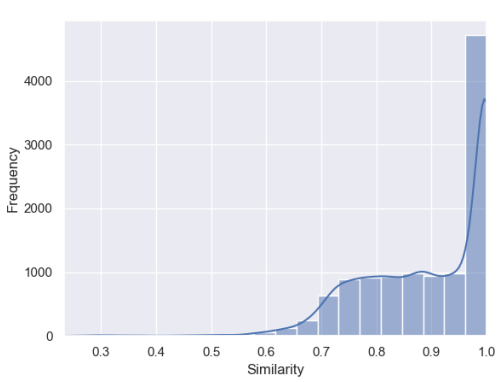
6. If the top 5 similarities are greater than 0.9, we consider that concept as the one that represents the entire dataset;

7. Instead, if some of the top 5 similarities are lower than 0.9 we search through the entire ConceptNet Numberbatch embeddings the remaining concepts, looking for the ones with the highest cosine similarity with respect to the mean value found in step 3 and with the same Part-Of-Speech found in step 2;

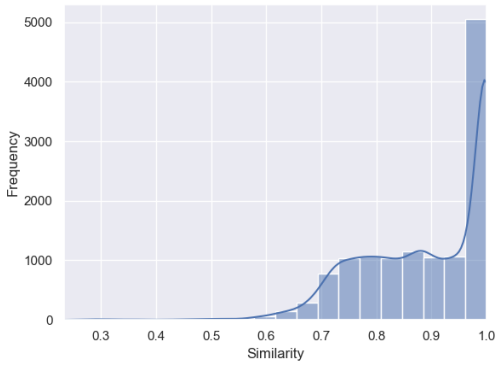
8. The concept with the highest similarity is the concept aligned to the entity.

4.1.2 Experimental Results

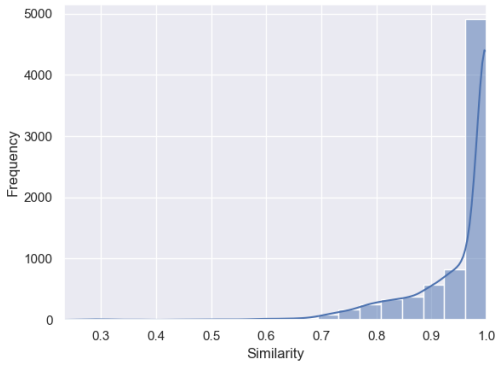
The results of the alignment algorithm are shown in Figure 4.1: the first thing that can be noticed is the bias towards the last bin in the histogram. The reason is quite obvious: since the embedding for each entity is obtained by using all the synsets lemmas, if there is a unique lemma associated with an entity the algorithm will select the concept obtained by that lemma as the best alignment, and the similarity value will be 1.



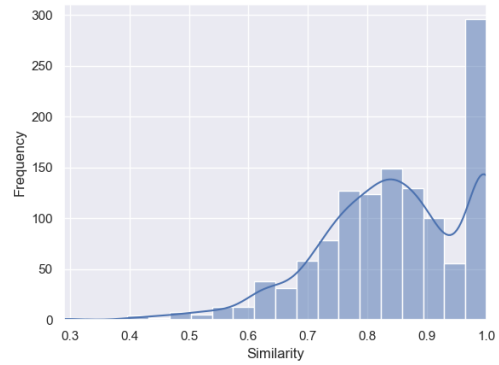
(a) Distribution of the best similarity for objects



(b) Distribution of the best similarity for attributes



(c) Distribution of the best similarity for objects and attributes



(d) Distribution of the best similarity for predicates

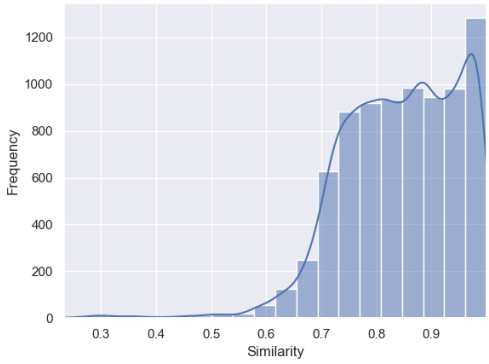
Figure 4.1: Distribution of the best similarity

Moreover, further elements can be noticed by looking at figure 4.2, where the alignments with similarity 1 have been excluded. First of all, objects and attributes similarities tend to have an almost uniform distribution concentrated between the similarity values of 0.7 and 1, with a maximal frequency of 1000. This indicates

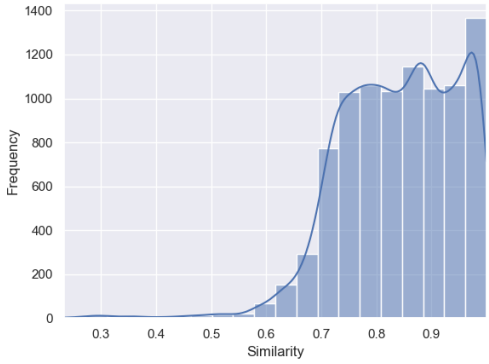
that the objects and attributes have a precise and less varied meaning, and then the alignment is less ambiguous.

Instead, looking at Figure 4.2c, the plot appears as a long tail distribution, indicating that merging objects and attributes introduces a noise factor that isn't noticeable from the figure 4.1c.

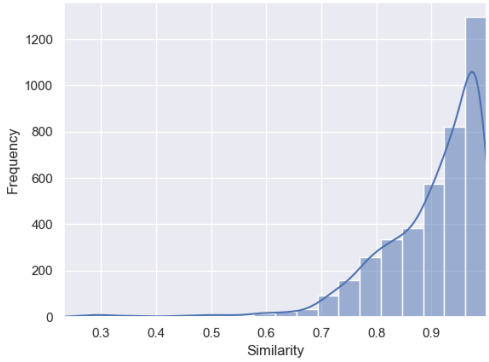
Finally, the similarity for predicates (in figure 4.2d is normally distributed, with a maximal frequency value of 140. This shows how the predicates alignment is more ambiguous than the objects and attributes, and they are more generally applicable across the Visual Genome dataset.



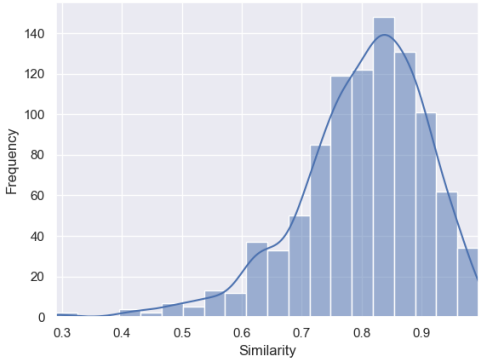
(a) Distribution of the best similarity for objects



(b) Distribution of the best similarity for attributes



(c) Distribution of the best similarity for objects and attributes



(d) Distribution of the best similarity for predicates

Figure 4.2: Distribution of the best similarity (excluding the similarities with value 1)

To further measure the performance of our alignment, 4 performance metrics have been calculated to compare the best concept (that corresponds to the aligned

one) with the top 5 concepts. In particular, the metrics are the following:

1. **Mean Similarity**: the average similarity value for the top 5 aligned concepts. It helps to determine the general closeness of top matches.

2. **Ratio to Mean**: the ratio of the similarity value of the best concept to the mean of the other 4 similarity values. A high ratio to mean indicates that the aligned concept is far better than the others.

3. **Similarity Drop-Off**: calculates the similarity difference between the most and the least similar concept in the top 5 matches. A high drop-off indicates that the best concept is far more aligned than the fifth one.

4. **Difference To Mean**: measures the difference between the similarity value of the best concept and the mean of the other 4 similarity values. If the difference to mean is high, the top score performs better than the others on average.

For each one of these performance measures, the mean, standard deviation, 25th percentile and 75th percentile have been found on the entire dataset. The results are shown in tables 4.1, 4.2, 4.3.

Table 4.1: Objects Alignment Performance

	Mean	Standard Deviation	25th Perc.	75th Perc.
Mean Similarity	0.492	0.087	0.432	0.535
Ratio to Mean	2.47	0.828	1.83	2.99
Similarity Drop-Off	0.562	0.156	0.457	0.690
Difference To Mean	0.500	0.174	0.367	0.644

Table 4.2: Attributes Alignment Performance

	Mean	Standard Deviation	25th Perc.	75th Perc.
Mean Similarity	0.492	0.087	0.432	0.536
Ratio to Mean	2.45	0.823	1.81	2.96
Similarity Drop-Off	0.558	0.156	0.451	0.685
Difference To Mean	0.493	0.174	0.358	0.640

Table 4.3: Predicates Alignment Performance

	Mean	Standard Deviation	25th Perc.	75th Perc.
Mean Similarity	0.489	0.095	0.424	0.533
Ratio to Mean	2.26	0.665	1.78	2.67
Similarity Drop-Off	0.495	0.155	0.395	0.604
Difference To Mean	0.445	0.162	0.338	0.557

In general, the performance measures in tables 4.1, 4.2 and 4.3 show similar values, indicating that the relationship between the similarity of the best concept and the other 4 similarities is similar for objects, attributes and predicates.

The mean similarity is medially around the value of 0.49, with a standard deviation of 0.09, indicating that all the entities have a not-so-high mean similarity value, on average; moreover, the standard deviation of the mean similarity is quite low, suggesting that the mean similarity is concentrated around its mean value of 0.49.

Proceeding with the analysis, the similarity drop-off has a mean of 0.56 for objects and attributes and 0.495 for predicates, with a standard deviation of 0.156 in all the cases; the value of the mean is high, indicating that the top score is far more aligned than the last one. In addition, as for the mean similarity, the standard deviation is low, indicating a drop-off quite packed around its mean value.

Concerning the ratio to mean, it is quite higher than 1 on average, showing that the first value is far better than the others. In this case, the variance is higher than the preceding ones, but looking at the 25th percentile and the 75th percentile we can notice that the ratio to main tends to remain quite high (generally rather higher than 1), suggesting that the best concept performs far better than the others.

Finally, the difference to mean is medially around 0.49 and has a low standard deviation, confirming what we said in the analysis of the ratio to mean metric; the difference to mean also gives a numerical characterization of the difference between the top alignment and the others, that is 0.49 on average.

In conclusion, the first value seems highly similar to the mean embedding found in step 3 of the alignment algorithm, suggesting that the aligned concept is precise and with high similarity on average. Instead, the other 4 concepts have quite lower similarity, indicating that a lot of potentially good alignments have been excluded.

4.2 FOL Statements Generation

This section focuses on how FOL predicates have been built to be used through the Logic Tensor Network framework. Papers that have been very useful for my work, particularly in leveraging entity hierarchies to generate range and domain first-order logic predicates, have been [4] e [5].

4.2.1 Methodology

The first thing done has been building various types of FOL statements using IsA, $CapableOf$, $NotCapableOf$, and $Synonym$ relations from ConceptNet (mentioned in the previous chapter). The following legend is used:

- $\mathcal{O}, \mathcal{P}$**:** sets of objects/attributes and predicates in Visual Genome;
- $\mathcal{O}'$**:** set of objects/attributes classes (chosen among their hypernyms);
- $\mathcal{O}_x^+$**:** set of hypernyms of the object/attribute x (they are found using the IsA relation in Conceptnet);
- $\mathcal{P}'$**:** the set of predicates classes (for each of which we would like to build range and domain constraints);
- $\mathcal{L}_{\mathcal{O}}$**:** set of objects/ attributes labels;
- $\mathcal{L}_{\mathcal{P}}$**:** set of predicates labels;
- $\mathcal{L}_{\mathcal{O}'}$**:** set of labels of objects/ attributes classes;
- $\mathcal{L}_{\mathcal{P}'}$**:** set of labels of predicates classes;
- $\mathcal{I}_{l_z}$**:** the set of of inverse predicates of z. It contains the labels $l_{z'}$, where z' is an inverse of z;
- $\mathcal{E}_{l_g}$**:** the set of predicates labels $l_{g'}$, where g' is defined as a synonym of the object/attribute/predicate g (through the Synonym relation in ConceptNet and further filtering using a threshold on the similarity among embeddings);
- $\mathcal{N}_{l_g}$**:** the set of predicates labels $l_{g'}$, where g' is defined as very different from the object/attribute/predicate g (obtained through the Antonym relationship in ConceptNet and using a threshold on the similarity among embeddings);
- $\mathcal{PD}_{z'}$**:** the subset of $l_{\mathcal{O}'}$ containing objects/attributes labels that compose the positive domain of the predicate z', obtained by using Visual Genome objects and attributes and Conceptnet relations $CapableOf$, IsA;

- $\mathcal{PR}_{z'}$**:** the subset of $\mathcal{O}'$ containing objects/attributes labels that compose the positive range of the predicate z', obtained by using hypernyms of Visual Genome objects and attributes and Conceptnet relations IsA;
- $\mathcal{ND}_{z'}$**:** the subset of $l_{\mathcal{O}'}$ containing objects/attributes labels that compose the negative domain of the predicate z', obtained by using hypernyms of Visual Genome objects and attributes and Conceptnet relations IsA, $NotCapableOf$;
- $\mathcal{NR}_{z'}$**:** the subset of $\mathcal{O}'$ containing objects/attributes labels that compose the negative domain of the predicate z', obtained by using hypernyms of Visual Genome objects/attributes and Conceptnet relation IsA.

$\mathcal{P}'$ contains a set of predicates for which we would like to build range and domain FOL axioms. We can take them as the most frequent ones in Visual Genome, but we can also take them as all the predicates. Using the notation previously introduced, the first categories of axioms that have been built are the following:

- **Ontological Relationships for Visual Genome Objects and Attributes:** If $x' \in \mathcal{O}^+_{\mathbf{x}} \cap \mathcal{O}',\ l_x \in \mathcal{L}_{\mathcal{O}},\ l_{x'} \in \mathcal{L}_{\mathcal{O}'}$,
  $\forall y \in \mathcal{O},\ (l_x(y) \rightarrow l_{x'}(y))$
  Example:
  $\forall y \in \mathcal{O}$ (c/en/man(y) $\rightarrow$ /c/en/living_creature(y));
- **Ontological Relationships for Visual Genome Predicates:** If $y' \in \mathcal{O}^+_y \cap \mathcal{P}',\ l_z \in \mathcal{L}_{\mathcal{P}},\ l_{z'} \in \mathcal{L}_{\mathcal{P}'}$,
  $\forall x, y \in \mathcal{O},\ (l_z(x, y) \rightarrow l_{z'}(x, y))$
  Example:
  $\forall x, y \in \mathcal{O}$ (/c/en/inside(x,y) $\rightarrow$ /c/en/near(x,y));
- **Inverse Relationships:** For $y' \in \mathcal{I}_{l_y}$,
  $\forall x, y \in \mathcal{O},\ (l_z(x, y) \leftrightarrow l_{z'}(y, x))$
  Example:
  $\forall x, y \in \mathcal{O}$ (/c/en/inside(x,y) $\leftrightarrow$ /c/en/outside(y,x));
- **Equivalence Relationships for Visual Genome predicates:** If $z \in \mathcal{P},\ l_z \in \mathcal{L}_{\mathcal{P}},\ l_{z'} \in \mathcal{E}_{l_z}$,
  $\forall x, y \in \mathcal{O},\ (l_z(x, y) \leftrightarrow l_{z'}(x, y))$
  Example:
  $\forall x, y \in \mathcal{O}$ (fight(x, y) $\leftrightarrow$ battle(x, y));
- **Equivalence Relationships for Visual Genome Objects and Attributes:** If $z \in \mathcal{P},\ l_z \in \mathcal{L}_{\mathcal{P}},\ l_{z'} \in \mathcal{E}_{l_z}$,
  $\forall x \in \mathcal{O},\ (l_z(x) \leftrightarrow l_{z'}(x))$
  Example:
  $\forall z \in \mathcal{O}$ (railroad_track(z) $\leftrightarrow$ railway(z));

- **Negative (or Mutual Exclusivity) Relationships between Objects:** If $x, x' \in \mathcal{O}',\ l_x \in \mathcal{N}_{l_{x'}}$,
  $\forall y \in \mathcal{O},\ (\neg l_x(y) \vee \neg l_{x'}(y))$
  Example:
  $\forall z \in \mathcal{O}$ (¬/c/en/food(x) ∨ ¬/c/en/tool(x));

- **Negative (or Mutual Exclusivity) Relationships between Predicates:** if $z, z' \in \mathcal{P},\ l_z \in \mathcal{N}_{l_{z'}}$,
  $\forall x, y \in \mathcal{O},\ (\neg l_z(x, y) \vee \neg l_{z'}(x, y))$
  Example:
  $\forall x, y \in \mathcal{O}$ (¬/c/en/sit(x,y) ∨ ¬/c/en/walk(x,y));

- **Positive Domain Relationships:** If $z' \in \mathcal{P}'$,
  $\forall x, y \in \mathcal{O},\ (l_{z'}(x, y) \rightarrow \bigvee_{l_{x'} \in \mathcal{PD}_{z'}}\ l_{x'}(x))$
  Example:
  $\forall x, y \in \mathcal{O}$ (/c/en/wear(x,y) → /c/en/biped(x) ∨ /c/en/person(x) ∨ /c/en/being(x) ∨ /c/en/animal(x));

- **Positive Range Relationships:** If $z' \in \mathcal{P}'$,
  $\forall x, y \in \mathcal{O},\ (l_{z'}(x, y) \rightarrow \bigvee_{l_{y'} \in \mathcal{PR}_{z'}}\ l_{y'}(y))$
  Example:
  $\forall x, y \in \mathcal{O}$ (/c/en/wear(x,y) → /c/en/surface(y) ∨ /c/en/tool(y) ∨ /c/en/-physical_object(y) ∨ /c/en/substance(y));

- **Negative Domain Relationships:** If $z' \in \mathcal{P}'$,
  $\forall x, y \in \mathcal{O},\ (l_{z'}(x, y) \rightarrow \bigwedge_{l_{x'} \in \mathcal{ND}_{z'}} \neg l_{x'}(x))$
  Example:
  $\forall x, y \in \mathcal{O}$ (/c/en/wear(x,y) → ¬/c/en/heavier_than_air(x));

- **Negative Range Relationships:** If $z' \in \mathcal{P}'$,
  $\forall x, y \in \mathcal{O},\ (l_{z'}(x, y) \leftrightarrow \bigwedge_{l_{y'} \in \mathcal{NR}_{z'}} \neg l_{y'}(y))$.

Now, for each z' in $\mathcal{P}'$, the set $\mathcal{PD}_{z'}$ (using Visual Genome triples) is built through the following algorithm:

1. The set $\mathcal{S}_{l_{z'}}$ of the semantically similar predicates to z' is built: firstly we add $l_{z'}$ in $\mathcal{S}_{l_{z'}}$; then we find all the labels l_z, for $z \in l_{\mathcal{P}}$, which are linked to $l_{z'}$ through the Synonym relation in ConceptNet (that is, we search for the $l_z \in l_{\mathcal{P}}$ such that Conceptnet triples of the form l_z "Synonym" $l_{z'}$ and $l_{z'}$ "Synonym" l_z exist, and we add all the l_z found in the set $\mathcal{S}_{l_{z'}}$);

2. For each $l_z \in \mathcal{S}_{l_{z'}}$ all the couples subject-predicates $\langle x, z \rangle$ are searched in Visual Genome, and the labels l_x of those subjects are added to the set $\mathcal{PD}_{z'}$;

3. For each $l_z \in \mathcal{S}_{l_{z'}}$ all the triples of the form l_x "CapableOf" l_z are extracted from ConceptNet and added to the set $\mathcal{PD}_{z'}$;

4. For each $l_z \in \mathcal{PD}_{z'}$, all the hypernyms of l_z (obtained through the ConceptNet relationship IsA) are added to $\mathcal{PD}_{z'}$;

5. The frequency of each $l_z \in \mathcal{PD}_{z'}$ is calculated and the label representing the 0.9-quantile is stored ;

6. The set $\mathcal{PD}_{z'}$ is filtered maintaining only the elements in $\mathcal{O}'$ and with a frequency higher than the threshold (the 0.9-quantile) calculated in step 5.

The set $PR_{z'}$ is built with a similar algorithm:

1. The same as before

2. All the couples objects-predicates $\langle y, z \rangle$ where $l_z \in \mathcal{S}_{l_{z'}}$ are found and the objects labels l_z are added to the set $\mathcal{PR}_{z'}$

3. The same as before

4. For each $l_z \in \mathcal{PR}_{z'}$, all the hypernyms of l_z (obtained through the ConceptNet relationship IsA) are added to $\mathcal{PR}_{z'}$;

5. The frequency of each $l_z \in \mathcal{PR}_{z'}$ is calculated and the label associated with the 0.9-quantile frequency value (that is, all labels of the hypernyms are associated with a frequency value that counts how many times they are present among the Visual Genome objects/attributes; after that, all the hypernyms are ordered in a descending way using those frequency values and the hypernym label associated to the frequency value below which 90% of the other frequencies fall) is stored to be used in the following step;

6. The set $\mathcal{PR}_{z'}$ is filtered maintaining only the elements in $\mathcal{O}'$ and with a frequency higher than the threshold (the 0.9-quantile) calculated in step 5.

Instead, about the set $ND_{z'}$, it is built with the following algorithm:

1. The set $\mathcal{S}_{l_{z'}}$ of the semantically similar predicates to z' is built: firstly we add $l_{z'}$ in $\mathcal{S}_{l_{z'}}$; then we find all the labels $l_z \in \mathcal{P}$ which are linked to $l_{z'}$ through the Synonym relation in ConceptNet (that is, we search for the $l_z \in \mathcal{P}$ such that Conceptnet triples of the form l_z Synonym to $l_{z'}$, $l_{z'}$ Synonym to l_z exist, and we add all the l_z found in the set $\mathcal{S}_{l_{z'}}$);

2. For each $l_z \in \mathcal{S}_{l_{z'}}$ all the triples of the form l_x $NotCapableOf$ l_z are extracted from ConceptNet and the l_x are added to the set $\mathcal{ND}_{z'}$;

3. For each $l_z \in \mathcal{ND}_{z'}$, the labels of all the hypernyms of l_z (obtained through the ConceptNet relationship IsA) are added to $\mathcal{ND}_{z'}$;

4. The frequency of each label in $\mathcal{ND}_{z'}$ is calculated, and the label associated with the 0.9-quantile frequency value (that is, all labels of the hypernyms are associated with a frequency value that counts how many times they are present among the Visual Genome objects/attributes; after that, all the hypernyms are ordered in a descending way using those frequency values and the hypernym label associated to the frequency value below which 90% of the other frequencies fall) is stored to be used in the following step;

5. The set $\mathcal{ND}_{z'}$ is filtered maintaining only the elements with a frequency higher than the threshold found in the previous step (the 0.9-quantile) and contained in $l_{\mathcal{O}'}$ calculated in step 4.

Moreover, for each $z' \in \mathcal{P}'$, a way to populate the negative domain and range sets is by adding elements $l_x, x \in O'$ that are semantically distant (in terms of similarity among embeddings or leveraging the *Antonym* relationship in ConceptNet) respectively from the ones in $\mathcal{PD}_{z'}$ and $\mathcal{PR}_{z'}$. Anyway, this has not been experimented with in the thesis.

In conclusion, the construction of $l_{\mathcal{O}'}$ has high importance since all the axioms generated depend on it. By analyzing the frequencies of all the hypernyms of all objects and attributes, the main challenges are the presence of similar elements and the presence of useless ones. To address these problems without resorting to manual selection, the following strategy has been applied: among the set of all the hypernyms of objects/attributes, the frequency of each label is calculated, and a filtering procedure similar to the one made in steps 4 and 5 of the preceding algorithms is applied. This has been done to discard the labels of objects/attributes with a frequency in Visual Genome under the frequency of the label associated with the 0.9-quantile frequency value. Moreover, among them, only those with a corresponding Numberbatch embedding have been chosen. This process generated a set $l_{\mathcal{O}'}$ composed of 83 elements.

4.2.2 Experimental Results

It has been possible to generate the first-order logic statements discussed in the previous section using the preceding algorithms. In particular, very strict thresholds have been chosen to have solid first-order logic axioms, which count is shown in Figure 4.3.

To have statistical measurements of the generated first-order logic axioms and to understand the link between each different type of axiom and each different triple (characterized by image_id, synsets, names, bounding box information of subjects,

objects and relationships), a statistical analysis has been executed. Firstly, the distribution of the number of axioms has been plotted for each type of first-order logic axiom (in Figure 4.3): the most numerous axioms are the ontological ones, followed by the positive domain, positive range and the negative ones; after that, we have the positive domain using $CapableOf$ axioms, the equivalence axioms and the negative domain using $NotCapableOf$ axioms.

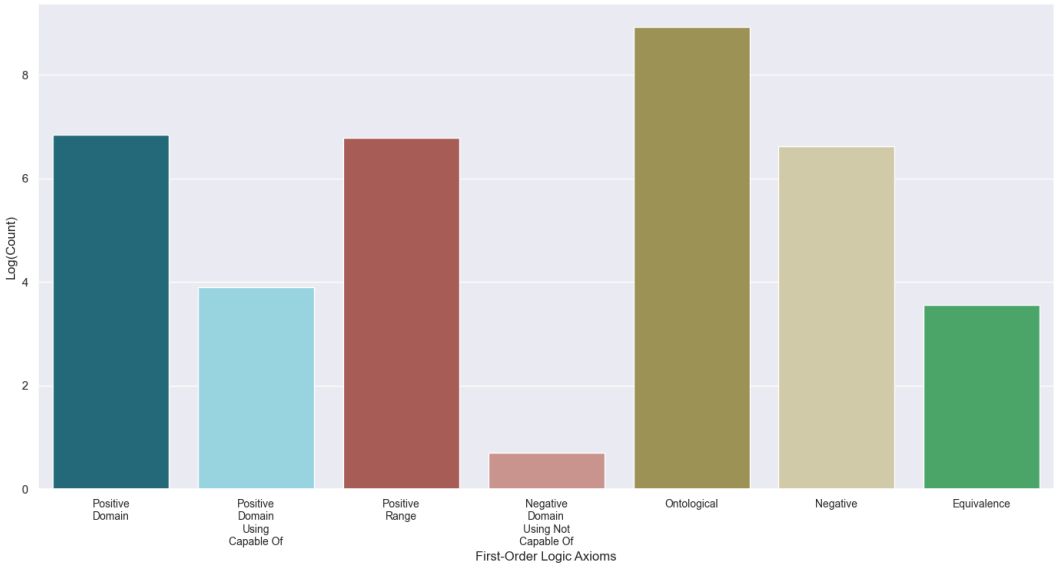


Figure 4.3: Axioms Count

In addition, to have a less granular view of the first-order logic statements, the distribution of the number of axioms in each different triple has been plotted for each different type of fol axiom.

In this case, when is said that an axiom is present in a triple it means that at least one among subject, predicate and object corresponds to a term of the FOL axiom (or has a hypernym corresponding to a term of the FOL axiom). Moreover, if an axiom is present in more than one among subject, object and predicate, it is counted only once.

Concerning the construction of the set $\mathcal{PD}$ (as said before), in the statistical analysis the positive domain FOL axioms generated using ConceptNet (in Figure 4.4c) have been separated from the ones generated using Visual Genome (in Figure 4.4a). For the plots, the following approaches approaches have been applied:

- **Distribution of the positive domain axioms count:** given all the positive

domain axioms generated using only Visual Genome triples and the relationship IsA of ConceptNet (that is, without using the Conceptnet relationship $CapableOf$), the x-axis represents their count in each different triple, and the y-axis shows the frequency in Visual Genome of those elements;

- **Distribution of the positive domain axioms count (obtained using the $CapableOf$ relationship in ConceptNet):** given all the positive domain axioms generated using the $CapableOf$ relationship in ConceptNet, the x-axis represents their count in each different triple, and the y-axis shows the frequency in Visual Genome of those elements;

- **Distribution of the positive range axioms count:** given all the positive range axioms generated as said in the previous section, the x-axis represents their count in each different triple, and the y-axis shows the frequency in Visual Genome of those elements;

- **Distribution of the negative domain axioms count (obtained using the $NotCapableOf$ relationship in ConceptNet):** given all the negative domain axioms generated using the $NotCapableOf$ relationship in ConceptNet, the x-axis represents their count in each different triple, and the y-axis shows the frequency in Visual Genome of those counts;

- **Distribution of the Negative Axioms Count:** given all the negative axioms generated (for predicates and objects/attributes hypernyms), the x-axis represents their count in each different triple (mixing both the ones generated from antonyms and the ones generated through embedding), and the y-axis shows the frequency in Visual Genome of those counts;

- **Distribution of the Ontological Axioms Count:** given all the ontological axioms generated (for objects, attributes and objects/attributes hypernyms), the x-axis represents their count in each different triple, and the y-axis shows the frequency in Visual Genome of the elements in the x-axis;

- **Distribution of the Equivalence Axioms Count:** given all the equivalence axioms generated for predicates, objects, attributes and objects/attributes hypernyms, the x-axis represents their count in each different triple, and the y-axis shows the frequency in Visual Genome of those counts;

- **Distribution of the Total Axioms Count:** given all the axioms generated for predicates, objects, attributes and objects/attributes hypernyms, the x-axis representstheir count in each different triple, and the y-axis shows the frequency in Visual Genome of those counts;

Table 4.4: Statistics of First-Order Logic Axioms Count For Triples

	Mean	Standard Deviation	25th Perc.	75th Perc.
PD	258.27	257.29	2.00	549.00
PR	218.09	187.27	3.00	396.00
PD - $CapableOf$	10.13	9.62	1.000	18.00
ND - $NotCapableOf$	0.47	0.64	0.00	1.00
Negative	121.59	124.71	3.00	225.00
Ontological	8.19	9.20	0.00	13.00
Equivalence	0.01	0.09	0.00	0.00
Total	616.74	556.49	8.00	1164.00

In general, from Figure 4.4 and 4.5 can be noticed that, for each type of axiom, the first bin is generally the most numerous in the histogram. This means the axioms generated for a single triple tend to be far lower than the total number of axioms.

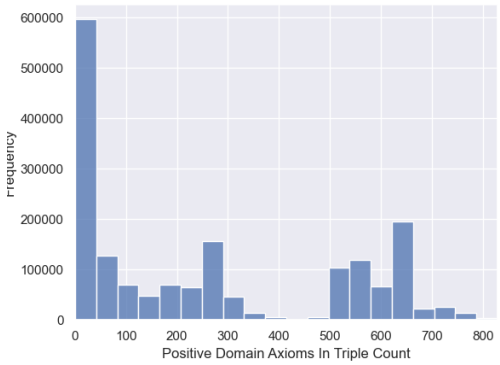
Entering specifically in the plots in Figure 4.4, 4.5 and in the table 4.4, in the case of positive domain axioms the mean value in triples is 258 and the standard deviation value is 257 axioms. This indicates that, despite the first bin size is numerous, the number of positive domain axioms is quite high. Looking at Figure 4.4a, the axioms count tends to be in the range of 0-300 or 500-700, but not in the range 300-500, which explains why both mean and standard deviation have high values.

About the positive range axioms in Figure 4.4b, the plot shows that the mean is quite similar to the previous plot (proportionally to the maximal value of 700 axioms), but, differently from the previous plot, the distribution is concentrated in the range $100 - 500$ and there is not an empty central range (as happens for the range $300 - 500$ in the previous plot).

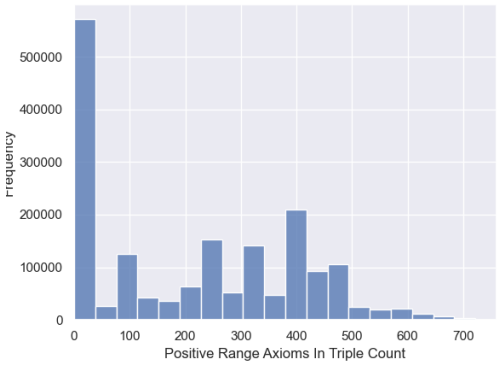
Moreover, looking at the axioms generated using ConceptNet (Figure 4.4c, the first thing that can be noticed is that they are far less numerous (in particular the negative domain ones) than the axioms generated with the Visual Genome data. Anyway, the ones present in Visual Genome have high frequency, suggesting that working with the most frequent hypernyms allows taking full advantage of a low number of axioms too.

About the negative axioms in Figure 4.5a, the plot shows characteristics similar to the positive domain and range axioms counts in Figures 4.4a and 4.4b, with high cardinality of axioms and mean similar to the standard deviation (around the value of 120).

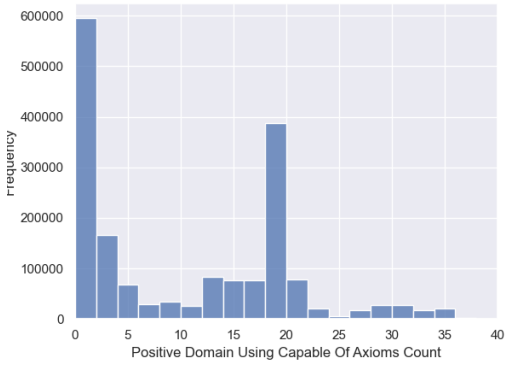
About the equivalence axioms in Figure 4.5c, they seem practically concentrated



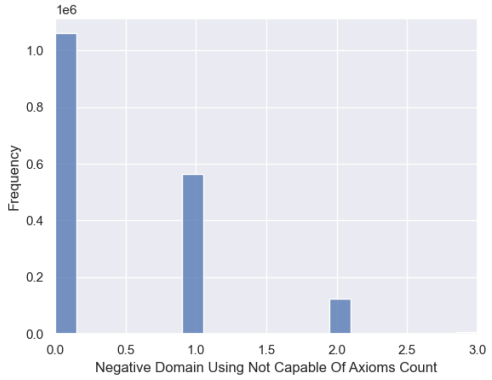
(a) Distribution of the positive domain axioms



(b) Distribution of the positive range axioms



(c) Distribution of the positive domain axioms (obtained using the $CapableOf$ relationship in ConceptNet)



(d) Distribution of the negative domain axioms (obtained using the $NotCapableOf$ relationship in ConceptNet)

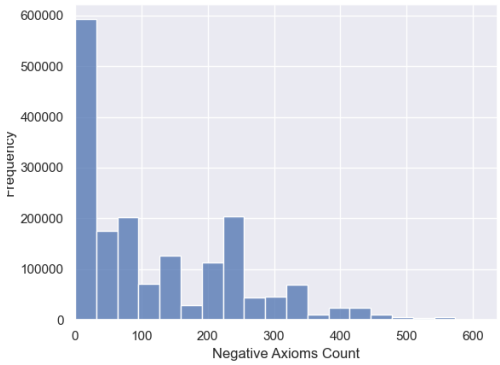
Figure 4.4: Distribution of Axioms in Visual Genome Triples I

only in the first bin, despite their not-so-low cardinality in absolute value (as can be seen in Figure 4.3). Further analysis has to be done.

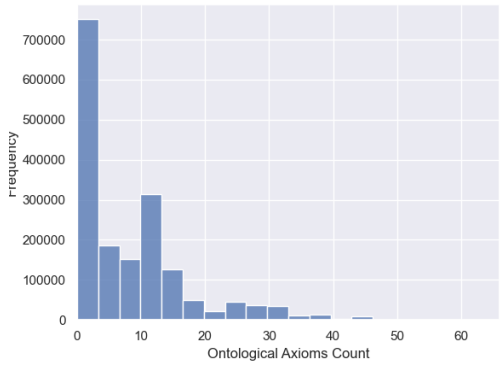
A similar behaviour can be seen in the ontological axioms. In fact, they have high cardinality in absolute value (Figure 4.3), but they have low density in triples, with a mean of 8.19 and a standard deviation of 9.20 ontological axioms per triple.

Finally, the total axioms count plot in Figure 4.5d shows a similar pattern to the previously seen distributions, but here the standard deviation is quite lower (proportionally to the mean value) than the positive domain and range one. Probably, the fact that the axiom types have different sizes helps to reduce the variance in the total axiom count distribution.

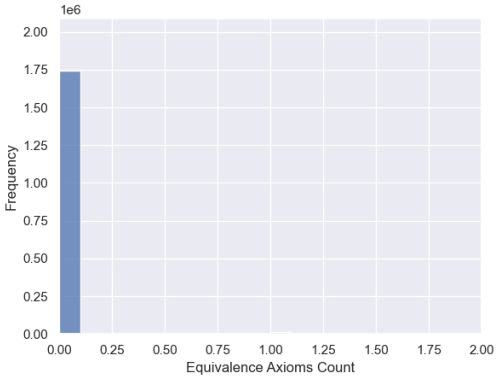
To make further analysis, the plots in Figure 4.6 and 4.7 have been built imposing that the elements in the x-axis should be greater than 0.



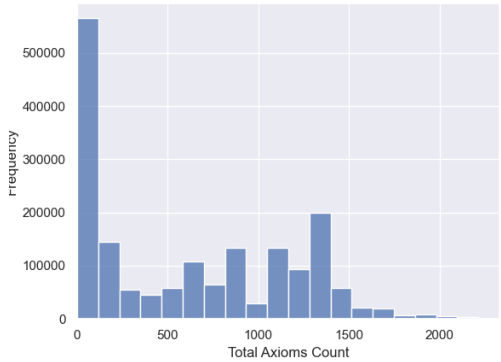
(a) Distribution of the Negative Axioms



(b) Distribution of the Ontological Axioms



(c) Distribution of the Equivalence Axioms



(d) Distribution of the Total Axioms

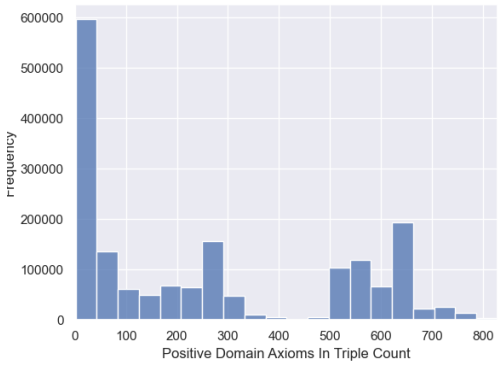
Figure 4.5: Distribution of Axioms in Visual Genome Triples II

As can be seen in Figures 4.6a, 4.6b, 4.6c, 4.6d, the plots of the positive domain, positive range, positive domain using $CapableOf$ and negative domain using $NotCapableOf$ don't show great variations if compared with the respective plots in Figure 4.4.

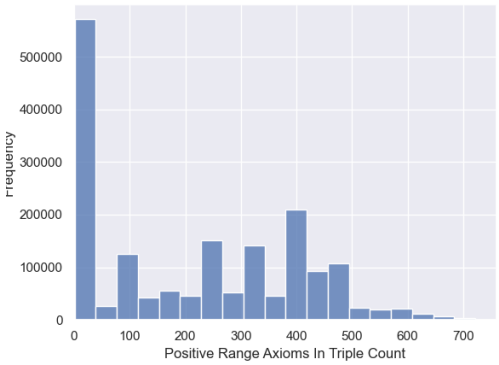
Instead, about the plots of negative domain axioms in Figure 4.7a, ontological axioms (in Figure 4.7b), equivalence axioms(4.7c) and total axioms (4.7d), the maximal value in the y-axis is far diminished, indicating that the count of 0 frequency values (in Figures 4.5a, 4.5b, 4.5c, 4.5d) are far more dominant than the other frequency values for these categories of axioms.

Another part of the analysis is the count of axioms for each different image. First of all, each image contains more than 19 triples on average.

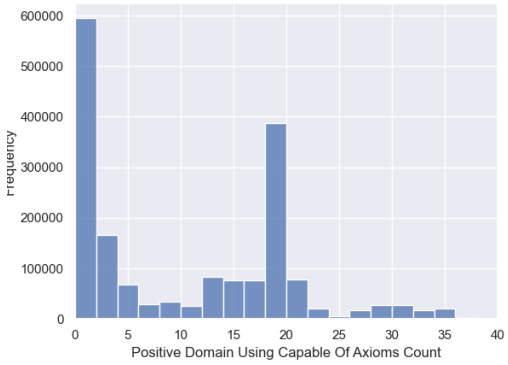
Looking at the tables 4.5 and 4.4 and comparing the mean and the standard deviation values for the domain, range and negative axioms (the first 5 rows in the tables), the mean in 4.5 tends to be more than the double of the mean in 4.4,



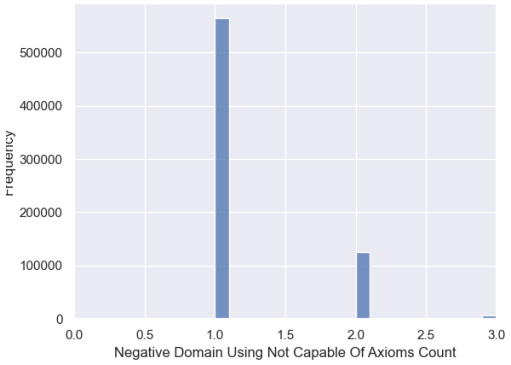
(a) Distribution of the positive domain axioms



(b) Distribution of the positive range axioms



(c) Distribution of the positive domain axioms(obtained using the $CapableOf$ relationship in ConceptNet)

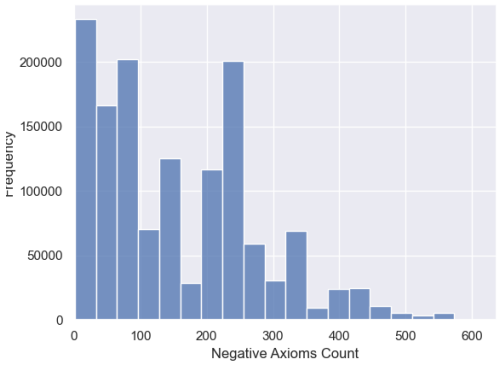


(d) Distribution of the negative domain axioms (obtained using the $NotCapableOf$ relationship in ConceptNet)

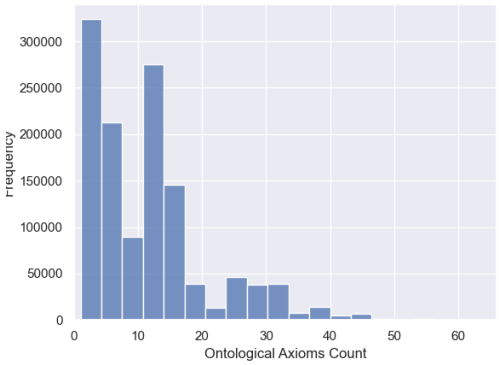
Figure 4.6: Distribution of Axioms in Visual Genome Triples III

while the standard deviation results are similar. This means that the triples in the same image have many common domain, range and negative axioms. This is supported by the fact that the average number of triples for images is 19, while the mean is only two times the mean in Table 4.4). This can be explained by the fact that the constructed range and domain axioms are very dense in the single images and triples, but at the same time they are not too general to take into account the differences between images.

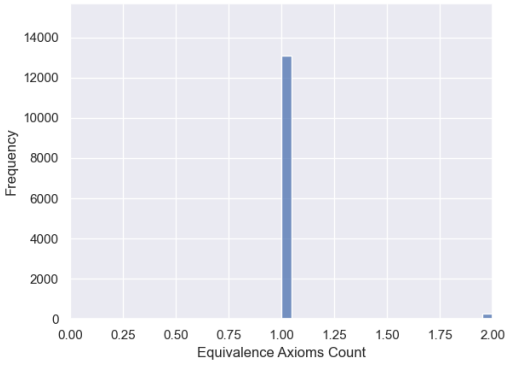
A different discourse has to be made for the ontological and equivalence axioms: they have a mean value of respectively 8.2 and 0.01 in Table 4.4, and 35.54 and 0.08 in Table 4.5. Moreover, contrary to the previous types of axioms and looking at the figures 4.9b and 4.9c, these axioms are concentrated in the first bins. Together with the fact that the mean value in Table 4.5 is far distant from the total count of



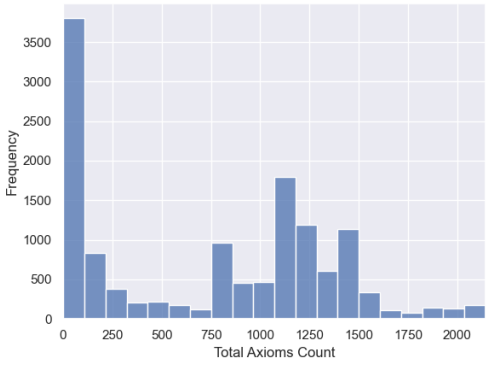
(a) Distribution of the Negative Axioms



(b) Distribution of the Ontological Axioms



(c) Distribution of the Equivalence Axioms



(d) Distribution of the Total Axioms

Figure 4.7: Distribution of Axioms in Visual Genome Triples IV

ontological and equivalence axioms (in Figure 4.3), this highlights they are far more specific than the previous types of axioms, making their average values become higher with respect to the average value in triples.

Finally, the total axioms behave similarly to the domain, range and negative axioms, since these types of axioms are far more numerous than the ontological and equivalence ones.

To make a better understanding of the generated statements, an image from Visual Genome has been randomly chosen (Figure 4.10), and a subsection of axioms referring to the image has been shown (the total number of axioms generated for the Figure 4.10 are 2266).

- **Positive Domain Axioms:**
  - /c/en/transport(x, y) $\rightarrow$ /c/en/biped(x) $\vee$ /c/en/person(x) $\vee$ /c/en/being(x) $\vee$ /c/en/animal(x),

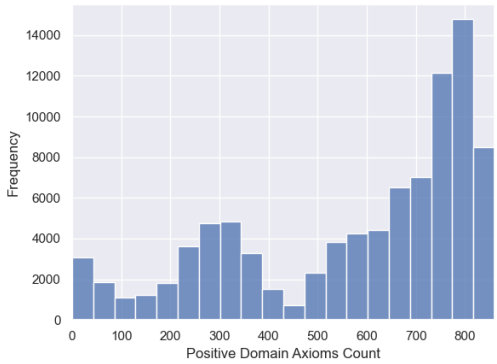
Table 4.5: First-Order Logic Axioms Count For Images

	Mean	Standard Deviation	25th Perc.	75th Perc.
PD	572.42	243.30	351.00	774.00
PR	518.96	181.20	417.00	660.00
PD - $CapableOf$	24.57	10.48	18.00	34.00
ND - $NotCapableOf$	0.97	0.65	1.00	1.00
Negative	355.64	172.27	233.00	494.00
Ontological	35.54	24.65	16.00	50.00
Equivalence	0.08	0.28	0.00	0.00
Total	1507.38	590.76	1154.00	1976.00

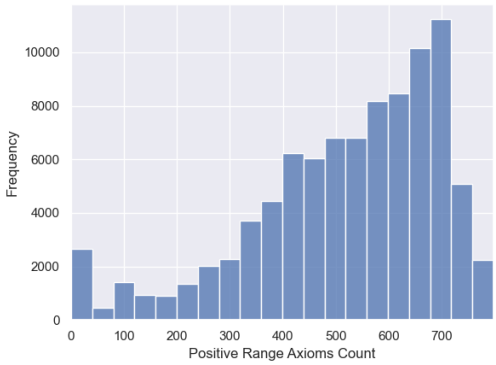
  - /c/en/wear(x, y) → /c/en/biped(x) ∨ /c/en/person(x) ∨ /c/en/being(x) ∨ /c/en/animal(x),
  - /c/en/along(x, y) → /c/en/anything(x) ∨ /c/en/mechanism(x) ∨ /c/en/-physical_object(x) ∨ /c/en/artifact(x),

- **Positive Range Axioms:**
  - /c/en/next(x, y) → /c/en/surface(y) ∨ /c/en/tool(y) ∨ /c/en/physical_object(y) ∨ /c/en/substance(y),
  - /c/en/wear(x, y) → /c/en/normally(y) ∨ /c/en/man_made_object(y) ∨ /c/en/machine(y) ∨ /c/en/artifact(y),
  - /c/en/along(x, y) → /c/en/small_building(y) ∨ /c/en/structure(y) ∨ /c/en/where_people_live(y) ∨ /c/en/tv_show(y),

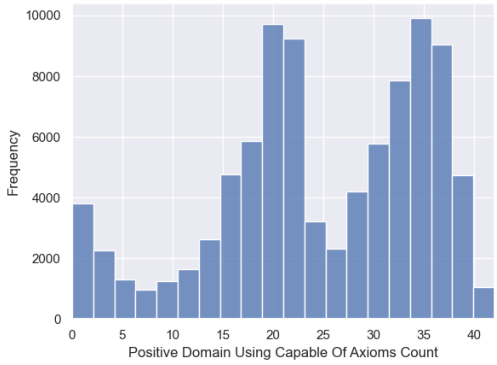
- **Positive Domain Using $CapableOf$ Axioms:**
  - /c/en/transport(x, y) → /c/en/artifact(x) ∨ /c/en/machine(x) ∨ /c/en/-physical_object(x) ∨ /c/en/vehicle(x),
- **Negative Domain using $NotCapableOf$:**
  - /c/en/laugh(x, y) → ¬/c/en/substance(x) ∧ ¬/c/en/food(x) ∧ ¬/c/en/fuel(x) ∧ ¬/c/en/good(x),
- **Ontological Axioms:**
  - /c/en/man(z) → /c/en/being(z),



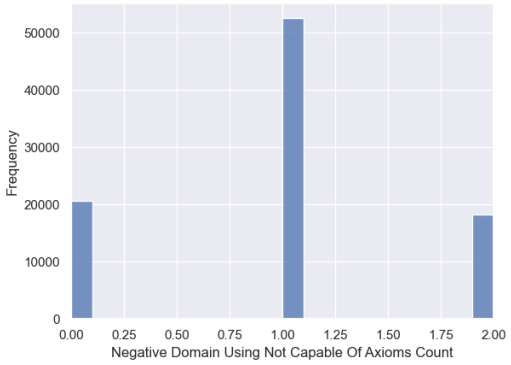
(a) Distribution of the positive domain axioms



(b) Distribution of the positive range axioms



(c) Distribution of the positive domain axioms(obtained using the *CapableOf* relationship in ConceptNet)



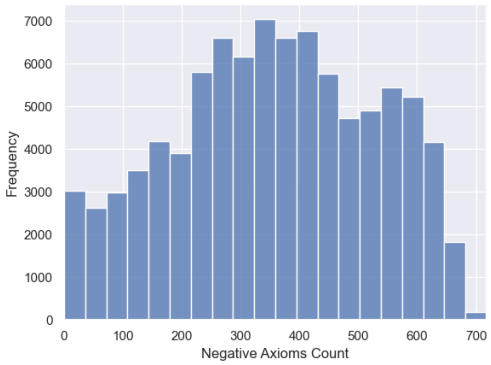
(d) Distribution of the negative domain axioms (obtained using the *NotCapableOf* relationship in ConceptNet)

Figure 4.8: Distribution of axioms in Visual Genome for each image I

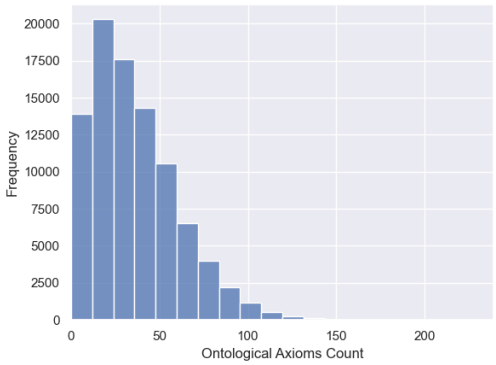
  - /c/en/road(z) → /c/en/surface(z),
  - /c/en/car(z) → /c/en/physical_object(z),

- **Negative Axioms:**
  - ¬/c/en/preditor(x) ∨ ¬/c/en/physical_object(x),
  - ¬/c/en/individual_item(x) ∨ ¬/c/en/large_building(x),
  - ¬/c/en/wear(x,y) ∨ ¬/c/en/behind(x,y).

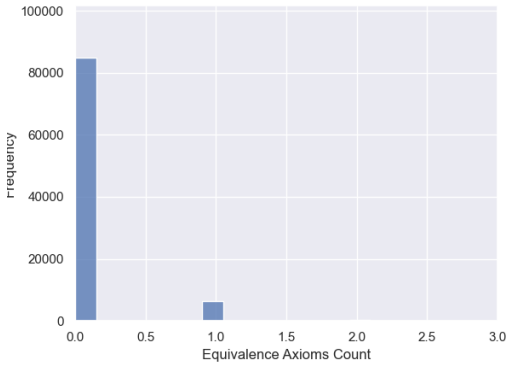
To better understand the potentiality of these constraints, here are the preceding range and domain constraints, transformed by integrating the ontological axioms directly in the domain and range axioms (that is, by mapping each hypernym



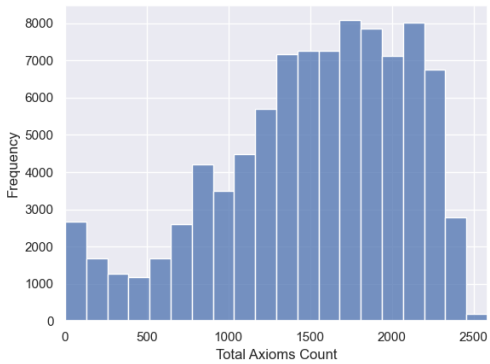
(a) Distribution of the Negative Axioms



(b) Distribution of the Ontological Axioms



(c) Distribution of the Equivalence Axioms



(d) Distribution of the Total Axioms

Figure 4.9: Distribution of Axioms in Visual Genome for Image II

Figure 4.10: Image From Visual Genome

label in each range or domain axiom to the label of those objects present in the same images and which are hyponym of the substituted hypernym), and filtered on axioms that contain at least a predicate present in the image:

- **Positive domain axioms**:
  - /c/en/wear(x, y) → /c/en/man(x),
  - /c/en/have(x, y) → /c/en/car(x) ∨ /c/en/van(x) ∨ /c/en/building(x),
  - /c/en/behind(x, y) → /c/en/van(x) ∨ /c/en/car(x),
  - /c/en/along(x, y) → /c/en/bicycle(x) ∨ /c/en/van(x) ∨ /c/en/car(x),
  - /c/en/about(x, y) → /c/en/building(x) ∨ /c/en/window(x),
  - /c/en/in(x, y) → /c/en/man(x),
  - /c/en/next(x, y) → /c/en/tree(x),
  - /c/en/by(x, y) → /c/en/tree(x).
- **Positive range axioms**:
  - c/en/wear(x, y) → /c/en/car(y),
  - /c/en/have(x, y) → /c/en/building(y) ∨ /c/en/window(y),
  - /c/en/transport(x, y) → /c/en/car(y) ∨ /c/en/building(y),
  - /c/en/along(x, y) → /c/en/building(y),
  - /c/en/in(x, y) → /c/en/man(y) ∨ /c/en/tree(y),
  - /c/en/next(x, y) → /c/en/road(y) ∨ /c/en/car(y) ∨ /c/en/building(y).
- **Positive domain using $CapableOf$ axioms:**
  - /c/en/transport(x, y) → /c/en/car(x).

Chapter 5

Future Works

This chapter illustrates an overview of the next logical steps towards fully integrating the generated axioms into the proposed neuro-symbolic framework.

The first one regards building negative domain and range constraints using Visual Genome triples. As said in Section 4.2.1, for each predicate $z' \in \mathcal{P}'$, it could be done by leveraging the labels in the sets $\mathcal{PD}_{z'}$ and $\mathcal{PR}_{z'}$, relation antonyms, embeddings among concepts and the set $\mathcal{L}_{\mathcal{O}'}$ to build the sets $\mathcal{ND}_{z'}$ and $\mathcal{NR}_{z'}$ (for example, for each $z' \in \mathcal{P}'$, we could build $\mathcal{ND}_{z'}$ with all the elements in $\mathcal{L}_{\mathcal{O}'}$ that are not in $\mathcal{PD}_{z'}$ and lowly similar to the mean embedding of all the elements in $\mathcal{PD}_{z'}$).

Another element could be to build range and domain constraints not only for predicates but also for subject-predicate and object-predicate couples. This in theory allows building more precise though less general axioms.

Another option is to leverage knowledge bases other than ConceptNet. For example, OWL-World presented in paper [5], which contains precise hierarchies and inverse relationships that could be really helpful for this task. But also other bigger knowledge bases such as YAGO or DbPedia could be a great source of external knowledge.

Finally, another way is to build the set $\mathcal{PD}_{z'}, \mathcal{PR}_{z'}, \mathcal{ND}_{z'}, \mathcal{NR}_{z'}$, for each z' in $\mathcal{P}'$, not through a deterministic algorithm, but introducing, for each $x \in \mathcal{O}$ and for each couple subject/object - predicate $\langle x, z' \rangle$, the more general LTN predicates InDomain(x,z') and InRange(x,z'), built by using information from Visual Genome and ConceptNet triples, and ConceptNet embeddings (such as Numberbatch). Despite facing the challenge of adapting the grounding in [4] (the grounding considered in this thesis), these predicates allow to build less and more general FOL statements such as:

- **Positive Domain Construction:**

$$\forall x \in \mathcal{O},\ \forall x' \in \mathcal{O}',\ \forall z \in \mathcal{P},\ \forall z' \in \mathcal{P}' :$$
$$\text{HasKnowledgeBaseLinkForDomain}(x, z)\ \wedge\ \text{SemanticallySimilar}(z', z)$$
$$\wedge\ \text{InHierarchy}(x, x')\ \Longrightarrow\ \text{InDomain}(x', z')$$

- **Positive Range Construction:**

$$\forall x \in \mathcal{O},\ \forall x' \in \mathcal{O}',\ \forall z \in \mathcal{P},\ \forall z' \in \mathcal{P}' :$$
$$\text{HasKnowledgeBaseLinkForRange}(x, z)\ \wedge\ \text{SemanticallySimilar}(z', z)$$
$$\wedge\ \text{InHierarchy}(x, x')\ \Longrightarrow\ \text{InRange}(x', z')$$

- **Negative Domain Construction:**

$$\forall x \in \mathcal{O},\ \forall x' \in \mathcal{O}',\ \forall z \in \mathcal{P},\ \forall z' \in \mathcal{P}' :$$
$$\text{HasKnowledgeBaseLinkForDomain}(x, z)\ \wedge\ \text{SemanticallyAntonym}(z', z)$$
$$\wedge\ \text{InHierarchy}(x, x')\ \Longrightarrow\ \neg\text{InDomain}(x', z')$$

- **Negative Range Construction:**

$$\forall x \in \mathcal{O},\ \forall x' \in \mathcal{O}',\ \forall z \in \mathcal{P},\ \forall z' \in \mathcal{P}' :$$
$$\text{HasKnowledgeBaseLinkForRange}(x, z)\ \wedge\ \text{SemanticallyAntonym}(z', z)$$
$$\wedge\ \text{InHierarchy}(x, x')\ \Longrightarrow\ \neg\text{InRange}(x', z')$$

- **Existence of maximal hypernym in the domain of a predicate:**

$$\forall x \in \mathcal{O},\ \forall x'' \in \mathcal{O}_{x'},\ \forall y \in \mathcal{P},\ \forall y' \in \mathcal{P}',\ \exists x' \in \mathcal{O}_x :$$
$$\text{HasKnowledgeBaseLinkForDomain}(x, z) \wedge \text{SemanticallySimilar}(z', z)\ \Longrightarrow$$
$$\text{InDomain}(x', z') \wedge \neg\text{InDomain}(x'', z')$$

- **Existence of maximal hypernym in the range of a predicate:**

$$\forall x \in \mathcal{O},\ \forall x'' \in \mathcal{O}^+_{x'},\ \forall y \in \mathcal{P},\ \forall y' \in \mathcal{P}',\ \exists x' \in \mathcal{O}^+_x :$$
$$\text{HasKnowledgeBaseLinkForRange}(x, z) \wedge \text{SemanticallySimilar}(z', z)\ \Longrightarrow$$
$$\text{InRange}(x', z') \wedge \neg\text{InRange}(x'', z')$$

Chapter 6

Conclusion

This thesis explored the potential of integrating symbolic reasoning with deep learning through Logic Tensor Networks (LTNs) to enhance the semantic understanding of images. By incorporating structured knowledge from knowledge graphs, prior knowledge in the form of first-order logic statements has been derived automatically and leveraged by logic tensor networks for the task of scene graph generation.

Firstly, the results obtained in the alignment process are accurate, despite the process filtering out many potential good alignments to better value the best embedding found for each object/predicate/attribute.

Concerning the first-order logic axiom generation, the IsA, $CapableOf$, $NotCapableOf$ and $Synonyms$ ConceptNet relationships have been extensively leveraged to build range and domain axioms using general hypernyms and all Visual Genome predicates. These axioms have been demonstrated to be quite dense in Visual Genome images. In addition, ontological axioms are generated to work as the link between specific objects/attributes and the general hypernyms, but they can also be utilized to build range and domain constraints with specific objects/attributes. Finally, the equivalence and negative axioms are less numerous and less general, though are far more precise than the previous ones. The total number of axioms for the image is 1517 on average, which are 572 positive domain axioms, 518 positive range axioms, 24 positive domain using $CapableOf$ axioms, 1 Negative domain using $NotCapableOf$ axioms, 356 negative axioms, 36 ontological axioms, 0 equivalence axioms.

In conclusion, this thesis contributes to the growing field of neuro-symbolic AI, highlighting the viability of using common-sense prior knowledge in combination with Logic Tensor Networks as a bridge between symbolic knowledge and deep learning, enriching scene graph generation. Future works could extend the typology of LTN predicates (such as with the predicates InDomain and InRange) and the categories of axioms (for example, by adding negative domain and range), the number of generated axioms, and the knowledge bases utilized to extract

information.

Bibliography

[1] Luciano Serafini and Artur d'Avila Garcez. *Logic Tensor Networks: Deep Learning and Logical Reasoning from Data and Knowledge*. 2016. arXiv: 1606.04422 [cs.AI]. URL: https://arxiv.org/abs/1606.04422 (cit. on pp. 3, 12).

[2] Ivan Donadello, Luciano Serafini, and Artur S. d'Avila Garcez. «Logic Tensor Networks for Semantic Image Interpretation». In: *CoRR* abs/1705.08968 (2017). arXiv: 1705.08968. URL: http://arxiv.org/abs/1705.08968 (cit. on pp. 3, 12).

[3] Ivan Donadello and Luciano Serafini. «Compensating Supervision Incompleteness with Prior Knowledge in Semantic Image Interpretation». In: *CoRR* abs/1910.00462 (2019). arXiv: 1910.00462. URL: http://arxiv.org/abs/1910.00462 (cit. on pp. 3, 12).

[4] Alessandro Sebastian Russo, Lia Morra, Fabrizio Lamberti, and Paolo Emmanuel Ilario Dimasi. «ESRA: a Neuro-Symbolic Relation Transformer for Autonomous Driving». In: *2024 International Joint Conference on Neural Networks (IJCNN)*. 2024, pp. 1–10. DOI: 10.1109/IJCNN60899.2024.10651426 (cit. on pp. 3, 13, 20, 36).

[5] D. Herron, E. Jiménez-Ruiz, and T. Weyde. «On the Potential of Logic and Reasoning in Neurosymbolic Systems using OWL-based Knowledge Graphs». In: *Neurosymbolic Artificial Intelligence* (Apr. 2024). In Press. URL: https://neurosymbolic-ai-journal.com/paper/potential-logic-and-reasoning-neurosymbolic-systems-using-owl-based-knowledge-graphs-0 (cit. on pp. 3, 20, 36).

[6] Fajar J. Ekaputra et al. «Describing and Organizing Semantic Web and Machine Learning Systems in the SWeMLS-KG». In: *The Semantic Web*. Ed. by Catia Pesquita, Ernesto Jimenez-Ruiz, Jamie McCusker, Daniel Faria, Mauro Dragoni, Anastasia Dimou, Raphael Troncy, and Sven Hertling. Cham: Springer Nature Switzerland, 2023, pp. 372–389. ISBN: 978-3-031-33455-9 (cit. on p. 3).

[7] Molood Barati, Quan Bai, and Qing Liu. «Mining semantic association rules from RDF data». In: *Knowledge-Based Systems* 133 (2017), pp. 183–196. ISSN: 0950-7051. DOI: https://doi.org/10.1016/j.knosys.2017.07.009. URL: https://www.sciencedirect.com/science/article/pii/S0950705117303258 (cit. on p. 3).

[8] Liyang Zhang, Shuaicheng Liu, Donghao Liu, Pengpeng Zeng, Xiangpeng Li, Jingkuan Song, and Lianli Gao. «Rich Visual Knowledge-Based Augmentation Network for Visual Question Answering». In: *IEEE Transactions on Neural Networks and Learning Systems* 32.10 (2021), pp. 4362–4373. DOI: 10.1109/TNNLS.2020.3017530 (cit. on p. 4).

[9] Xirong Li, Shuai Liao, Weiyu Lan, Xiaoyong Du, and Gang Yang. «Zero-shot Image Tagging by Hierarchical Semantic Embedding». In: *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '15. Santiago, Chile: Association for Computing Machinery, 2015, pp. 879–882. ISBN: 9781450336215. DOI: 10.1145/2766462.2767773. URL: https://doi.org/10.1145/2766462.2767773 (cit. on p. 4).

[10] Elaheh Raisi and Stephen H. Bach. «Selecting Auxiliary Data Using Knowledge Graphs for Image Classification with Limited Labels». In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2020, pp. 4026–4031. DOI: 10.1109/CVPRW50498.2020.00473 (cit. on p. 4).

[11] Francesco Giuliari, Geri Skenderi, Marco Cristani, Alessio Del Bue, and Yiming Wang. *Leveraging commonsense for object localisation in partial scenes*. 2022. arXiv: 2211.00562 [cs.CV]. URL: https://arxiv.org/abs/2211.00562 (cit. on p. 4).

[12] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. *Bridging Knowledge Graphs to Generate Scene Graphs*. 2020. arXiv: 2001.02314 [cs.CV]. URL: https://arxiv.org/abs/2001.02314 (cit. on p. 4).

[13] Zhanwen Chen, Saed Rezayi, and Sheng Li. «More Knowledge, Less Bias: Unbiasing Scene Graph Generation with Explicit Ontological Adjustment». In: *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2023, pp. 4012–4021. DOI: 10.1109/WACV56688.2023.00401 (cit. on p. 4).

[14] Ranjay Krishna et al. «Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations». In: *International Journal of Computer Vision* 123 (May 2017). DOI: 10.1007/s11263-016-0981-7 (cit. on p. 5).

[15] George A. Miller. «WordNet: A Lexical Database for English». In: *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*. 1994. URL: https://aclanthology.org/H94-1111 (cit. on p. 7).

[16] Robyn Speer, Joshua Chin, and Catherine Havasi. *ConceptNet 5.5: An Open Multilingual Graph of General Knowledge*. 2018. arXiv: 1612.03975 [cs.CL]. URL: https://arxiv.org/abs/1612.03975 (cit. on pp. 8, 11).