Improving Generalizability of ML-enabled Software through Domain Specification

Hamed Barzamini, Mona Rahimi, Murteza Shahzad, Hamed Alhoori
Northern Illinois University
DeKalb, USA
{hbarzamini,mrahimi1,msyed1,alhoori}@niu.edu

ABSTRACT

While the conventional software components implement pre-defined specifications, Machine Learning (ML)-enabled Software Components (MLSC) learn the domain specifications from the training samples. Thus, the MLSC's data-driven and inductive reasoning becomes highly reliant on the quality of the training dataset, which are often arbitrarily collected in ad hoc manners. The random collection of samples leads to a significant gap between the actual specifications of a real-world concept, and the picture that a dataset represents of the concept, reducing MLSC generalizability, particularly in perceptual tasks where understanding the environment is an important factor of accurate prediction.

To fill the gap between the conceptualization of a targeted domain's concept and its visualization in the MLSC dataset, we propose exploiting semantic specification of the concept to identify the concepts' missing variants in the data. We first, semantically specify hard-to-specify targeted domain's concepts and second, refer to the derived specifications to evaluate the diversity and relative completeness of MLSC collected datasets. The systematic augmentation of training datasets, with respect to the semantics of the domain, improves the quality of an arbitrarily collected dataset and potentially yields more reliable models. As a proof of concept, we automatically acquired the existing semantic knowledge for specifying the automotive domain concept *"pedestrian."* Augmenting the state-of-the-art pedestrian datasets accordingly, the evaluations showed that semantic augmentation outperforms brute-force machine learning in satisfying the MLSC accuracy requirements.

CCS CONCEPTS

• **Software and its engineering → Requirements analysis**; **Software reliability**; • **Computing methodologies → Object detection**; **Semantic networks**.

ACM Reference Format:
Hamed Barzamini, Mona Rahimi, Murteza Shahzad, Hamed Alhoori. 2022. Improving Generalizability of ML-enabled Software through Domain Specification. In *1st Conference on AI Engineering - Software Engineering for AI (CAIN'22), May 16–24, 2022, Pittsburgh, PA, USA.* , 12 pages. https://doi.org/10.1145/3522664.3528589

CAIN'22, May 16–24, 2022, Pittsburgh, PA, USA

ACM ISBN 978-1-4503-9275-4/22/05…$15.00
https://doi.org/10.1145/3522664.3528589

1 INTRODUCTION

As deploying ML algorithms in Software Engineering (SE) is rapidly increasing, the domain specifications are transforming from being explicitly articulated in the textual format or formal rules to being implicit within a set of training data, such as images and video frames. Unlike conventional software systems that implement a set of pre-defined "agreed-upon" specifications gathered from stakeholders and customers [22], the MLSC learn and suggest the specifications from collected examples [24]. This characteristic of the MLSC is desirable for programming hard-to-specify concepts for which limited description exists to guide software programmers. For instance, in the automotive domain, many of the advanced automated driving functionalities require software components to perceive the environment. The majority of these functionalities may not be completely specifiable due to the presence of hard-to-specify concepts in the MLSC operating environment [58]. For instance, what is the exact specification for recognizing a potential *pedestrian*? The concept of a pedestrian is hard to specify, as it has various instances with characteristics that are hard to predict (e.g., pedestrians differ from each other in terms of clothing, size, and shape) [49]. Whereas human drivers use their intuition to recognize various instances of pedestrians, the MLSC learn the concept from a limited set of images and video frames of pedestrians in a training dataset. As such MLSC perceive a concept through inductive reasoning, generalizing the common features that the ML model discovers in varying instances of the concept in a collected dataset [5].

However, due to being collected in unsystematic manners, datasets used to train ML models are generally limited in the number and diversity of samples they comprise [20, 45]. For instance, the most recently established datasets in the context of autonomous driving, such as Caltech [13], KITTI [17], CityPersons [74], and EuroCityPerson (ECP) [6], are collected by a vehicle-mounted camera aimlessly navigating rural roads [20]. Unguided collection of pedestrian images may result in an incomplete, unrepresentative, and undiversified dataset, leading to biased models. For example, the inspection of a commonly used pedestrian dataset revealed the lack of images of pedestrians in wheelchairs [44]. A prior research performed a simple cross-dataset evaluation to reveal that the majority of the state-of-the-art pedestrian detectors are biased and, therefore, are vulnerable to small domain shifts [20]. As such, they may perform well on the datasets they have been trained on, yet they perform poorly on unseen datasets. This is because the existing pedestrian detectors are tailored for *some* target datasets, which may not be a complete or fair representation of the actual pedestrians in the operational domain. Without a comprehensive pedestrian dataset that includes a wide variety of the concept's instances, there will be a significant and inevitable misalignment between the specification

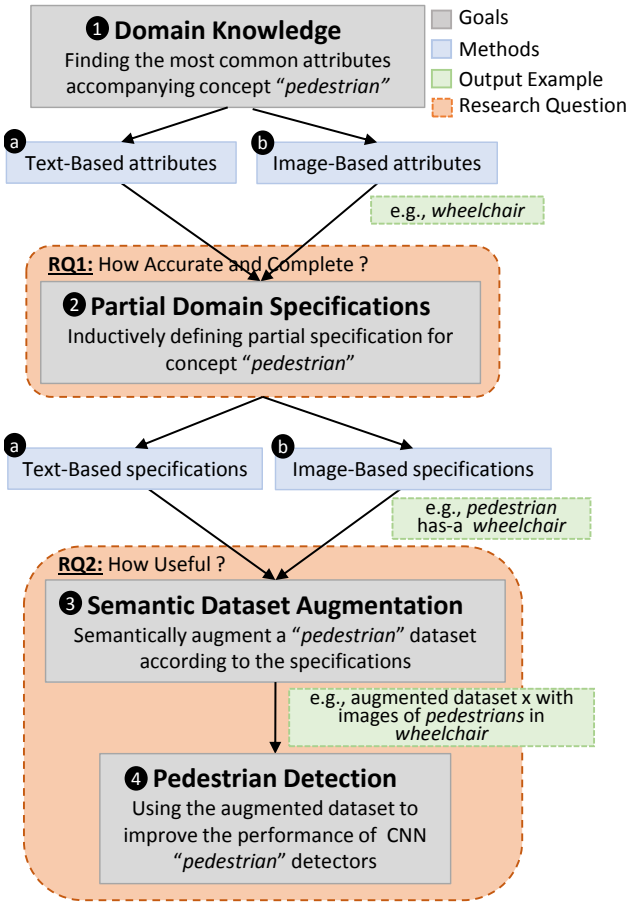

Figure 1: A high-level overview of the approach: (1) Extract domain knowledge (2) to infer partial specifications (3) to be used for dataset semantic augmentation (4) to improve accuracy in state-of-the-art pedestrian detectors.

of a domain's concept that MLSC is designed to detect and what a collected dataset represents as the targeted concept [24, 50].

In this regard, this paper aims to answer a general question:

"Does making use of semantic specifications of a targeted domain's concept improve the concept's representation in MLSC training dataset, to increase their detection accuracy?"

Thus, the goal herein is to **make MLSC better meet domain specifications by augmenting the inductive nature of ML with domain analysis**. The augmentation occurs through incorporating semantic knowledge into MLSC training datasets, which in turn compensates for the missing variants of a concept within the dataset, providing an augmented source of knowledge for ML models. In this regard, we propose an automated approach to formally specify hard-to-specify domain concepts instead of allowing MLSC to learn the specifications solely from a set of arbitrarily collected samples. Referring to the derived specifications, we further validate the presence of the domain specifications in the collected samples, which in turn characterizes the extent to which the dataset contains or lacks features that are important to learn a domain concept. If a specification is not present in a state-of-the-art pedestrian dataset, we augment the dataset with the missing specification. Our experiments verified that systematic selection of images, according to domain specifications, results in a more representative dataset, which in turn, generates more generalizable and accurate MLSC. To the best of our knowledge, **this is the first study to exploit domain specifications to evaluate dataset quality (in terms of semantic completeness) with the purpose of improving MLSC perception accuracy.**

We address this problem in the context of automated driving systems, where the correct description of real-world concepts is critically important for safety reasons, such as correct pedestrian recognition to avoid accidents. In this domain, we focus on the concept of *pedestrian* due to its importance for the perception of autonomous vehicles from the environment. Our contributions are:

- We demonstrate that the SE domain analysis task can be adopted, adapted, and applied to the process of engineering MLSC.
- We demonstrate that by the use of semantic specifications of a targeted domain's concept, we can evaluate, reason, and augment the representation of the concept in MLSC training datasets.
- We propose a novel and generalizable method for deriving partial specifications, where definitive specifications are not feasible, and for exploiting the specifications to assess and improve the semantic completeness of MLSC training dataset.

This research is fundamentally different but complementing to the work in the computer vision domain whose primary focus is on improving the MLSC robustness through pixel-level transformations of the existing images in the dataset, such as occlusion, rotation, and translation. Here we instead focus on **improving the MLSC generalizability of a targeted domain's concept by searching through an *external* semantic space of prior knowledge to incorporate the missing variances of the concept into the dataset.** Instead of addressing visualization concerns, such as incorporating visible and invisible noise into adversarial examples, we fill the gap between the conceptualization of the targeted concept and its visualization in the dataset.

Section 2 describes the proposed approach for mining relevant terms from the existing knowledge sources. Section 3 explains the procedure to derive partial specifications from the terms. Section 4 evaluates completeness, accuracy, and usefulness of specifications. Finally, Sections 5-8 describe potential applications of our work, related work, threats to validity, and conclusion and future work.

2 GATHERING DOMAIN KNOWLEDGE

(Figure 1- ❶): The use of domain knowledge can play a significant role in improving the quality and efficacy of software development process. For this reason, several studies in the SE domain have previously sought to extract domain knowledge for various concepts from the existing domain documents [8, 19, 65]. Several studies have gone further by capturing the retrieved information in the form of a semantic web or ontology [11, 32]. However, some concepts are inherently difficult to explicitly delineate, yet most humans have an *intuition* of what they refer to. One primary challenge in gathering domain knowledge for the hard-to-specify concepts is the lack of domain documents. For instance, *pedestrian* is a socially constructed concept for which no relatively complete domain document exists. Although there are a few general domain semantic

webs that include a limited specification of the term pedestrian, such as WordNet [38], they fail to adequately capture all varying instances of the concept in sufficient detail. For example, WordNet defines a pedestrian as a "person who travels by foot" and associates the word with the terms *walker* and *footer*. However, this definition is limited given that it excludes, for example, pedestrians riding a bike, roller-skating, or using a wheelchair. It also fails to describe a pedestrian's appearance in terms of attributes, such as clothing and posture.

To tackle the challenge of limited documents, we adopted two complementing approaches to gather the existing domain knowledge for the concept *pedestrian*: textual-based and visual-based. Both of these methods acquire domain knowledge through identifying a set of important attributes associated with the concept. However, they are different in terms of the sources they refer to gain semantic information, as well as in the processing methods they apply to the source artifacts, due to the artifacts type differences. In the first approach, existing human knowledge was our source of reference, whereas, in the second, we combined a set of benchmark pedestrian datasets to form a *super dataset* as a point of reference. To summarize, we first searched the existing textual sources to extract information about the *pedestrian* concept and then extended the list of attributes through processing visual sources, including images and video frames of different-looking pedestrians.

Figure 1 represents an overview of our proposed approach. We initially collect important attributes of a potential *pedestrian* from both sources (step 1). We further infer a set of partial specifications of a potential pedestrian in the form of simply structured relationships between the extracted attributes and a pedestrian (step 2). With reference to the derived specifications, we augment a state-of-the-art pedestrian dataset for missing specifications (step 3). We compare the accuracy of the two commonly used pedestrian detectors once before the augmentation and again after both random and semantic dataset augmentation (step 4).

Our primary question is the applicability of the approach-derived specifications (indicating the applicability of domain semantics) for improving MLSC generalizability (RQ_2). Yet, our evaluation consists of two distinct phases designed to evaluate the efficacy of our approach through addressing two key research questions:

- **RQ1: Accuracy and Completeness:** How accurately and completely can the proposed approach establish partial specifications of a hard-to-specify domain concept?
- **RQ2: Usefulness:** How useful are the established specifications for semantic augmentation of datasets to improve MLSC generalizability?

2.1 Textual Knowledge

(Figure 1- ❶ ⓐ): To exploit human knowledge as a reference, we searched online repositories of American and British English books. We selected Google books repository containing about 40 million book titles with 155 billion words from American English and 34 billion words from British English, published between 1500 and 2019 [15]. Mining the repository, we selected terms that frequently appeared immediately before and immediately after the term *pedestrian*, as well as those that appeared up to four terms apart. This step resulted in 1,329 initial and 265 final distinct terms accompanying the term pedestrian. Further, to remove the less relevant

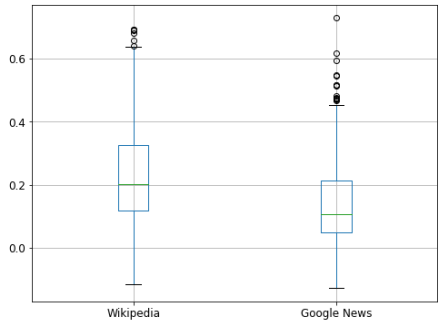

Figure 2: Box plot of similarity scores for terms retrieved from Wikipedia and Google News.

terms, we used semantic similarity as a second filter. We first selected two public and widely used corpora, namely Wikipedia and Google News, built on a substantial number of words, 400,000 and 3 billion, respectively. Using Gensim library [47], we transformed each corpus into a set of Word2vec models, whereby each unique word in the corpus was assigned a corresponding vector in the space [37]. Word vectors were positioned in the vector space such that words that share common contexts in the corpus were located close to one another in the space. Second, querying this implemented two-layer neural network model for each final term, a score between -1 and 1 was assigned to each term, representing their cosine similarity to the term pedestrian, according to Wikipedia and Google News corpora. Finally, we selected terms only with similarity scores within the upper quartile (75th percentile) of all the terms' similarity scores. As such, the cutoff value appeared to be 0.3258 and 0.2137 for Wikipedia and Google News, respectively. Figure 2 represents the box plots for both corpora, Wikipedia, and Google News similarity scores. After removing the out-of-the-range terms, 135 distinct terms remained.

Table 1: Terms extracted from textual and visual sources with their *similarity* and *average of confidence* scores.

Src.	Terms	Similarity Score(Txt.) Ave. Confidence(Img.)
Text	bicyclist, bike, jaywalking, motorist, wheelchair, blind, handicapped, ...	0.616, 0.530, 0.416, 0.407, 0.513, 0.449, 0.300, ...
Image	sidewalk, street, jacket, bag, head, building, tree, ...	0.173, 0.328, 0.248, 0.304, 0.166, 0.342, 0.240,...

The upper section of Table 1 represents a few examples of the final remaining terms with their associated similarity scores. Among all the terms, several terms explicitly described a potential pedestrian with respect to a prospective adjective (e.g., blind or handicapped) or a possible accompanying object (e.g., wheelchair or bike). However, certain terms did not necessarily specify a particular pedestrian, but provided useful information regarding the context in which the instances of the concept appear. For example, the terms *sidewalk*, *stairs*, and *safety* do not particularly describe a

pedestrian, yet the first two terms specify a place where a potential pedestrian may appear, and the last one determines a desired requirement for pedestrians. There were a few terms that neither described potential pedestrians nor provided descriptive information directly relevant to pedestrians. The majority of these terms instead described the context factors, such as *driver*, *collision*, and *accident*. The next section explains our method to place the more meaningful terms in a descriptive format.

We additionally mined a wide range of other existing textual sources, such as knowledge graphs, dictionaries, glossaries, and encyclopedias for terms with relatively more "*known*" relationships to the term *pedestrian*, as well as sources that contained more frequently updated content, such as social media and news feeds. However, to avoid evaluation bias, we did not use these sources for domain specification inference and instead reserved this data to evaluate RQ_1 further, discussed in detail in Section 4. A complete list of the terms extracted from all sources of knowledge and the other artifacts of this paper can be accessed from our publicly available online repository[1].

2.2 Visual Knowledge

(Figure 1- ❶ⓑ): To further extend the list of concept-accompanying terms, we additionally utilized the existing visual sources of knowledge. We initially referred to the three most commonly used datasets of images and video frames of pedestrians: Caltech [13], CityPersons [74], and EuroCity Persons (ECP) [6]. The pedestrian dataset benchmarks are proposed from the context of autonomous driving. However, these datasets are monotonous, such that they lack diverse scenarios. Hence, we selected two additional large-scale human datasets, CrowdHuman [55] and WiderPerson [75], which unlike pedestrian datasets are not limited to traffic scenarios and include images of people in more generic contexts, such as people in parks, restaurants, and selfies. We added these two datasets to narrow the gap between real-world humans and current pedestrian detection benchmarks. Recent research has shown that cross-dataset evaluation of a model trained on a human dataset produces more accurate predictions than models trained only on pedestrian datasets [20]. Table 2 represents the summary of the datasets. We refer to the combination of the five datasets as the *super dataset*.

To detect terms associated with the classes of *pedestrian* and *person* in the super dataset, we used an *anchor-based object detection* technique related to computer vision and image processing. The process of object detection typically happens through two levels: one involving image classification and the other object localization. While image classification assigns an object to one or multiple existing classes, object localization identifies the location of a potential object through drawing an imaginary surrounding *bounding box* around its extent. To localize an object, the anchor-based object detection algorithm first predicts an object's position in an image by creating numerously fixed and predefined *anchor boxes* around it. Anchor boxes are, therefore, referred to as candidate boxes that a model initially predicts to identify an object's location, size, and shape. Later, the detector calculates probability and other attributes, such as Intersection over Union (IoU), for each anchor box. IoU is an evaluation metric that identifies the overlap of each anchor box with the predicted bounding box by the model. Assessing the calculated attributes, the model refines the initial anchor boxes to finally propose an optimal bounding box for label prediction. For instance, Figure 3 illustrates a few of the detected objects and their ultimate predicted bounding boxes. The application of anchor boxes improves the detector's performance, as the entire image can be processed at once. The object detector additionally estimates a *confidence score*, representing the probability of the box containing the predicted object. Finally, the classification accuracy is computed according to standard metrics such as recall, precision, and mean average precision (mAP) scores with respect to a predefined IoU threshold, specifying the overlap of the predicted boundary with the ground truth for a true positive prediction.

[1]https://github.com/SEFORAI/MLClassifiers

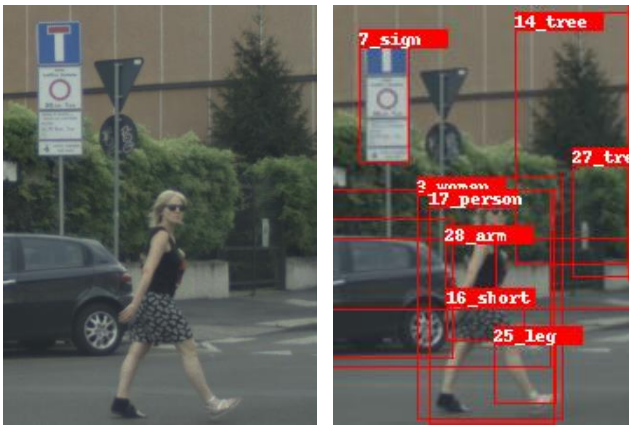

Figure 3: The pre-trained Faster R-CNN detected objects in an image of a pedestrian from the ECP dataset.

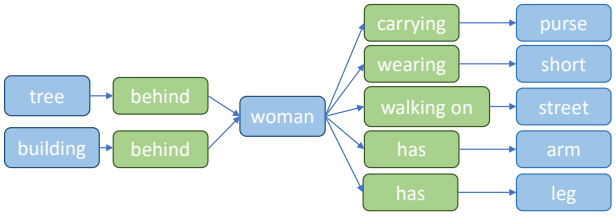

Figure 4: Scene Graph partially drawn from USGG generated tuples for Figure 3. Subjects and objects are illustrated in blue, while predicates are in green.

Following recent work in the computer vision domain [61, 62, 70, 73], we selected a pre-trained anchor-based Faster Region-based Convolutional Neural Network (Faster R-CNN) model [48] as the underlying object detector for the super dataset. Region-based Convolutional Neural Networks (R-CNNs) incorporate a *selective search algorithm* into the network to identify potential regions from an image to label and create bounding boxes. Despite R-CNNs success in object detection, they are typically computationally expensive [48]. To address this problem, Faster R-CNN replaces the *selective search algorithm* with a region proposal network which, in turn, reduces the number of proposed regions generated while ensuring precise object detection [48].

Table 2: Super Dataset statistics

	Caltech(1Frame/Second)	CityPersons	ECP(Day)	CrowdHuman	WiderPerson
Training Images	2,143	2,975	23,892	15,000	8,000
Persons	13,674	19,238	201,323	339,565	287,131
Persons/Image	0.32 (all frames)	6.47	9.2 (all conditions)	22.64	3.2
Distinct Persons	1,273	19,238	201,323	339,565	287,131

Table 3: Details of the detected objects and relationships by Faster-R-CNN and USGG in each dataset of the super dataset.

	Caltech	CityPersons	ECP	CrowdHuman	WiderPerson
Objects (O)	64,290	89,250	716,760	450,000	240,000
Distinct O	92	87	115	147	145
Pedestrians (P)	5,606	10,034	58,216	172,751	116,227
P-Relationships (R)	13,360	38,520	280,100	299,280	159,980
Distinct R	219	264	682	839	513
R/Image	6.23	12.95	11.72	19.95	20

The adopted Faster R-CNN in our approach was equipped with ResNeXt-101-FPN backbone [33, 69], a batch size of 8, and an initial learning rate of 8×10^3, and was trained on the training set of Visual Genome, a large and dense general dataset containing 108,077 images with a detailed description of each image [29]. The dataset contains 75,729 unique objects (labels), and each image has an average of 35 objects, 26 attributes, and 21 pairwise relationships between objects. Analyzing the labels of the dataset, we decided to consider the labels *People*, *Person*, *Woman*, *Man*, *Boy*, *Girl*, *Kid*, *Child*, *Guy*, and *Lady* as potential pedestrians [29].

We used the detector on all images in the super dataset to extract additional terms that are likely to be associated with a potential pedestrian. The network detected a total of 1,560,300 objects in the images. Analyzing the confidence scores associated with the detected objects, we observed a diverse range of numbers, changing from the highest 0.9923 to the lowest 0.0005. Therefore, to generate more accurate final specifications of a potential pedestrian, we removed terms with an average confidence score of 0.15. The remaining objects with the highest confidence scores had an average score of 0.3654. The upper part of Table 3 represents the statistical details of the detected objects, distinct detected objects, and detected pedestrians—including objects with labels people, person, woman, man, boy, girl, guy, and lady—by Faster R-CNN in each dataset. After removing objects with low confidence scores, an average of 26% of the remaining objects were potential pedestrians.

The lower part of Table 1 indicates a few examples of the retrieved terms and their confidence scores. Some terms, similar to the extracted textual terms, do not directly describe potential pedestrians but rather the context in which pedestrians appear, such as building, tree, and stop sign. The following section explains our method for identifying such terms.

3 DERIVING DOMAIN SPECIFICATIONS

(Figure 1-❷): This section describes our method for deriving a set of partial specifications, in the form of simple relationships between the retrieved terms and the term pedestrian. Given previously retrieved textual and visual terms, we partially specified a potential pedestrian as a set of (i) attributes that explicitly describe a potential pedestrian, such as a child or woman; (ii) objects that may accompany a pedestrian, such as a wheelchair or a purse; and (iii) themes that a potential pedestrian may be associated with, such as a walkway and sidewalk. The approaches to derive specifications for the textual and visual terms are discussed separately, as our processing method differed for textual and visual sources.

3.1 Textual Knowledge

(Figure 1- ❷ ⓐ): For each distinct term, an individual rule was inferred. Each rule corresponds to the construction of a single *tuple{subject, predicate, object}*, where the *subject* is always pedestrian, the *predicate* describes a potential relationship between the subject and object, and finally, the *object* represents the term that the *subject* is in relationship with. Therefore, each previously extracted term replaces an *object* in a tuple.

To replace predicates, we initially explored retrieving verbs from sentences that contained the pair of object and pedestrian as the subject. For this, we applied part-of-speech tagging using the Python library for Natural Language Processing (NLP), namely Natural Language Toolkit (NLTK) [35]. We first wrote a Python wrapper script to use the NLP tagger to parse and tag the verb in each sentence with subject-object pair. However, after merging all the predicates which belonged to the same pair, we found that the majority of pairs had many potential predicates. For instance, in one record, we had *pedestrian* as the *subject*, *freeway* as the *object*, and {*killed*, *hit*, *run*, *closed*, *cross*, *identified*, *filmed*, *dies*, *struck*, *call*, *crossing*, *backed*, *injured*, *forcing*} as a set of *predicates*.

Thus only for representation purposes, we further limited the predicates to three less specific categories of *is-a, has-a*, and *does* relations. The *predicate* between the *subject* pedestrian and the *object* is selected according to the grammatical structure of the *object*, applying part-of-speech tagging. The *is-a* predicate is selected for objects with the role of a noun, *has-a* is placed between the subject and objects with an adjective tag, and *does* is placed between the subject and objects with all other roles. For instance, if wheelchair was retrieved as an object, with the part of speech determined to be a noun, then the predicate to be placed between the subject pedestrian and the object wheelchair is selected as *has-a*, to form the rule *"pedestrian has-a wheelchair"*. As another example, the retrieval of the terms *handicapped* and *jaywalking* functioning as an adjective and verb, respectively, resulted in the addition of two new rules: *"pedestrian is-a handicapped"*, and *"pedestrian does jaywalking"*. The

Table 4: A few examples of the derived specifications in form of *Subject-Predicate-Object* tuple

Source	Spec.	Subject	Predicate	Object
Textual	S_{1-3}	pedestrian	has-a	{bike, wheelchair, safety...}
	S_{4-7}	pedestrian	does	{jaywalking, walking, crossing...}
	S_{8-10}	pedestrian	is-a	{handicapped, blind, drunk...}
Visual	S_{10-20}	pedestrian	is	{people, person, woman, man, boy, girl, kid, child, guy, and lady}
	S_{21-26}	pedestrian	wearing	{pant, shirt, shoe, jean, shorts, jacket...}
	S_{27-32}	pedestrian	carrying	{bag, umbrella, surfboard, skateboard, jacket, paper...}
	S_{33-41}	pedestrian	has	{head, leg, arm, face hair, mouth, nose, neck, eye...}
	S_{42-46}	pedestrian	walking on	{sidewalk, street, track, railing, snow...}
	S_{47-50}	pedestrian	riding	{horse, bike, skateboard, motorcycle...}

full list of predicates is available on our online repository in the form of an *.owl* document.

This process resulted in 42 *has-a*, 11 *does*, and 4 *is-a* relationships. The rest of the terms with minor roles, such as adverb and preposition, were removed as they did not specifically describe a potential pedestrian. Among the established relationships, several resulted in meaningful descriptions of a potential pedestrian, while a few yielded less valid phrases. For instance, the terms *bicyclist* and *motorist* are more descriptive of a *pedestrian* with the selection of *is-a* rather than *has-a* as the predicate. However, since their role is determined as a noun, the method derives the specification *"pedestrian has-a {motorist, bicyclist}"*. Another example is the word *parking*, identified as a verb although it may also be referring to the place of parking and therefore, has the role of a noun. The upper section of Table 4 contains a few examples of the established specifications. The complete list is placed in our publicly available repository. In the next subsection, we describe our approach for identifying additional rules.

3.2 Visual Knowledge

(Figure 1- ❷ ⓑ): To infer partial specifications from the extracted terms out of images and video frames, we similarly established a relationship between a detected potential pedestrian (people, person, woman, man, boy, girl, kid, child, guy, and lady) and each term. Although here, for this purpose, we generated a Scene Graph (SG) from each of the 52,010 images in the super dataset. A generated SG from an image provides an abstract structured representation of the image content. Scene Graph Generation (SGG) is a well-researched problem in the computer vision domain [70, 73], and has received increasing attention from the research community due to its use for visual reasoning tasks [71]. SGG targets different semantic levels and describes a scene by extracting relationships between the detected objects. These relationships can be represented by directed edges, that connect two objects in the form of a *subject-predicate-object* phrase, such as *woman-wearing-shorts*.

To generate the scene graphs, we selected a state-of-the-art Unbiased SGG (USGG) framework that uses causal inference [61]. The USGG first builds an initial causal graph for the set of objects in an image, and further removes the mistakenly inferred relationships due to counterfactual causality. The model, therefore, generates relatively less biased results by distinguishing between the main effect and side effects. The USGG framework demonstrated significantly improved predictions over other frameworks, such as Iterative Message Passing [70], MOTIFs [62, 73], and VCTree [29]. The model is previously trained across 75k object categories and 37k predicate categories [61]. We selected this model as it also provides more fine-grained relationships from the ostensibly probable but trivial ones, such as replacing *near* with *behind/in front of*, and *on* with *standing on/walking on/parking on/driving on*. Figure 4 represents a partial scene graph drawn from a subset of the generated tuples by USGG corresponding to the image in Figure 3.

Since USGG also uses Faster R-CNN with the same parameters, trained on the same dataset, Visual Genome, it detected relationships between the same set of objects that we previously extracted in Section 2.2. Applying USGG to the super dataset, a total of 1,083,060 relationships were detected. Among them, we selected 469,709 of the most confident ones (with the average confidence score of 0.15) that contained a potential pedestrian (people, person, woman, man, boy, girl, kid, child, guy, and lady). The lower part of Table 3 represents the number of detected pedestrian-related relationships, distinct relationships, and an average number of detected relationships per image for each dataset, while the bottom part of Table 4 shows a few examples of the derived specifications in the form of *subject-predicate-object* tuples. We manually added specifications 10 to 20, S_{10-20} (the first row) since we considered these labels as a potential pedestrian. The rest of the specifications in the table were established by USGG. The complete set of the derived partial specifications can be found in our publicly available repository.

4 EVALUATION

This section addresses the two research questions specified in Section 2 regarding the relative completeness and accuracy of semantically reasoned specifications and, most importantly, their usefulness in improving generalizability of object detection:

4.1 Completeness and Accuracy (RQ_1)

(Figure 1-RQ1): We do not claim that the proposed approach generates absolute complete or definitive specifications of domain concepts, yet the experiments indicated that the semantic specifications, which the approach derived for the concept pedestrian, effectively served our primary purpose of outlining a hard-to-specify domain concept to enrich the MLSC training dataset:

4.1.1 Completeness: Due to the presence of several sources of uncertainty in MLSC operational domain, such as domain shifts and environmental uncertainties, proving absolute completeness of the inferred specifications in such way that they covers the entire instances of potential pedestrians within any context, seems impossible. Instead here, we provide evidence of the incompleteness

of currently used pedestrian datasets relative to the derived specifications. Table 5 reports the number of specifications inferred from each aforementioned dataset that were missing from the remaining four datasets. As previously shown in Table 3, potentially due to the large size of CrowdHuman and ECP datasets (in terms of number of objects and relations) in comparison to the other datasets, it is somewhat reasonable to observe the highest number of unique specifications in these two datasets. Similarly Caltech and CiyPerson datasets with the lowest number of unique objects and relations contain a lower number of mutually exclusive specifications.

Table 5: Datasets mutually exclusive specifications.

	Caltech	CityP.	ECP	CrowdH.	WiderP.
Unique Spec.	6	9	231	250	43

While the results do not provide evidence for completeness of the approach-inferred specifications, they indicate that randomly collected datasets, formed to serve the same purpose, are incomplete relative to each other, supporting our work of addressing such incompleteness. This is appropriate to remind the readers that the overall performance of ML algorithms depends on the extent to which a dataset represents the original distribution rather than its size. Therefore, unsystematically increasing the size of a training dataset—in this case for instance training with the super dataset—does not necessarily improve a model's generalizability, overfits the models, and reduces computing performance in large models [2, 12, 34, 43]. To demonstrate this, Section 4.2.2 carries out experiments, comparing models performance trained on randomly augmented datasets and on semantically augmented datasets.

4.1.2 Accuracy: Given that manually verifying the correctness of specifications for describing pedestrians is highly subjective, we constructed multiple binomial classifiers for detecting potential association between a wide range of terms and the term pedestrian.

The classifiers were trained on knowledge sources that (i) were not previously used for specifications retrieval and (ii) contained relatively *"known"* relationships to the term pedestrian. This allows unbiased testing on unseen data and prevents classifiers bias towards possibly incorrect *unknown* relationships. Considering the two characteristics, we built a training set from endorsed dictionaries, glossaries, and encyclopedias, using Onelook, a web interface to search through 18,955,870 words from 1,061 general dictionaries and glossaries, as well as more specific ones, such as science, technology, and slang [1]. As an additional source, we traversed ConceptNet, a commonly used knowledge graph to reason about associations between words, which contains verified and relatively known relationships [59]. During training, these terms served as positive instances, while negative instances were selected through pairing the term pedestrian with a set of random *non-associated* terms using the same sources and process. These negative instances may, therefore, by chance, contain a small number of positive but yet unknown related pairs. We accepted this "noise" as a characteristic of the larger problem we seek to solve. The process of sample collection resulted in 575 positive and 692 negative samples.

Given the collected positive and negative samples, six classifiers were built (representing different classes of models) to recognize potential associations between a term with the term pedestrian.[2] These classifiers measured several features, both semantic-based, such as cosine similarity, and lexical-based, such as frequency of co-appearance, to learn the existing patterns. The classifiers included Logistic Regression (LR), Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), Support Vector Machines (SVM), and K-Nearest Neighbor (K-NN). LR and NB are both probabilistic classifiers, LR makes a prediction using a logistic function to model the class variable, whereas NB learns how the data was generated given the results. DT uses a tree structure, in which leaves and branches represent class labels and conjunctions of features, while RF chooses the mode of a multitude of DTs decisions as the final label. SVM is another non-probabilistic model that maps the training data to a higher dimension and searches for a hyperplane that separates the classes. Finally, K-NN classifies by a popularity vote of K neighbors.

Table 6: Precision, Recall, F1, and F2 on the testing set

	LR	NB	DT	RF	SVM	K-NN
Precision	0.77	0.73	0.76	0.76	0.76	0.77
Recall	0.77	0.54	0.74	0.76	0.76	0.77
F1	0.77	0.62	0.74	0.76	0.76	0.77
F2	0.64	0.47	0.61	0.63	0.63	0.64

During the testing process, the approach-derived specifications served as the positive samples, while the same number of random terms were selected as the negative ones. Negative testing samples were randomly selected from the same sources, following the same processes from which previously the negative training samples were extracted. Finally merging positive and negative samples as a balanced testing set, the trained classifiers either agreed or disagreed with the association in each sample.

The results are reported in Table 6, using the standard metrics, precision, recall, F_1, and F_2. As reported, the average agreement between the models is 0.75, 0.72, 0.73, and 0.61 in terms of precision, recall, F1, and F2, respectively. Please note, the primary concern here is the accuracy of the established associations (positive samples) and less with the associations that are not established (negative samples) since the positive samples present associations in the approach-generated specifications rather than the negative ones. The intention of training classifiers is thus, to assess the percentage of our established associations (positive instances) that are mutually predicted positive by the classifiers (true positives). As such, in our experiments recall is the fair measurement versus precision, and F2 (favoring recall) is more informative than F1. Yet, we report precision and F1 as factors representing the fitness of the trained model for the entire test set. Additionally, positive testing samples that the classifiers disagreed with, may not necessarily represent incorrectly established associations as our manual post-process investigation verified. These results can possibly be improved through a finer training. However, we did not adjust the initial experiments and reported the preliminary results.

[2]The classifiers were built with the open source Python ML library scikit-learn [42]. The optimal values for hyper-parameters of each individual classifier were found through adopting a method, called *exhaustive grid search*. This method exhaustively implements and evaluates a classifier with all combinations of different parameter values to retain the best combination. As such the final hyper-parameters were set to: LR (penalty=11, C_value=0.2); DT (max_depth=4, min_sample_split=3); RF (criteria=entropy, max_depth=40, min_sample_split=30, n_estimator=20); K-NN (K=180); and finally SVM (gamma='scale', kernel='linear').

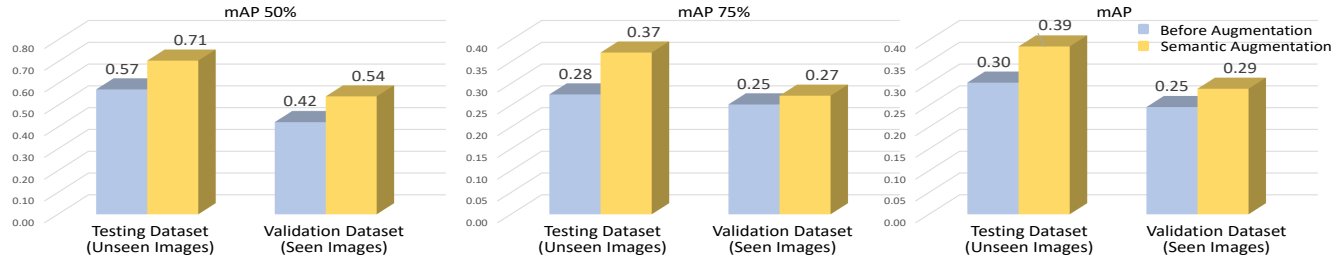

Figure 5: *R*esNet-50 mAP-50%, mAP-75%, mAP on pedestrian classification before and after semantic augmentations.

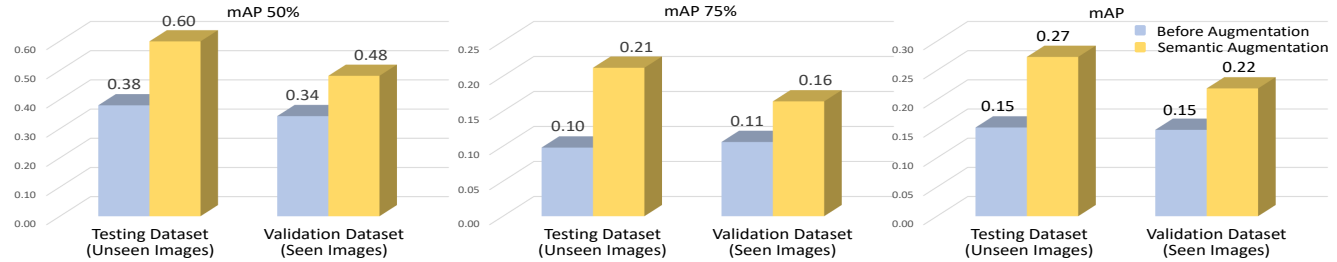

Figure 6: *D*arknet-53 mAP-50%, mAP-75%, mAP on pedestrian classification before and after semantic augmentations.

4.2 Usefulness (RQ_2)

(Figure 1- ❸ & ❹): We conducted two series of experiments to answer RQ_2 regarding usefulness of the derived specifications in improving MLSC prediction accuracy. The experiments differ with respect to the selected benchmarks, as well as the applied detectors:

4.2.1 Semantic Augmentation vs. Before Augmentation: We selected CityPersons as the *base dataset* (to be augmented), since it contained relatively lower number of images within the super dataset and lacked a high number of the derived specifications. The publicly available pedestrian datasets often do not release ground truth associated with the testing set to establish public challenges for training the most accurate model on their test dataset. Therefore, we randomly reserved 15% of the base training data with the purpose of using it as our *testing set* later. This way we were able to perform an unbiased evaluation of the models on unseen images. We refer to the remaining 85% of CityPersons dataset as base dataset (70% and, 15% respectively for training and validation).

We then used the remaining datasets in the super dataset as the supplementary source for the purpose of semantic augmentation. Among the previously derived specifications from each dataset, we identified those which were missing from the base dataset. As such, we retrieved 2,797 associated images with the missing specifications, split the images to 70% for training, 15% for validation and 15% for testing, and added them to the corresponding sets in the base dataset (details shown in Table 7).

We then twice trained and validated the formerly introduced object detector, Faster R-CNN with ResNet-50 backbone, once with the base dataset (before augmentation) and once with the augmented dataset using two 16GB Tesla P100 GPUs for 50 epochs. The training process took about 8 hours on the base dataset and 10 hours after the augmentation. We then tested both models on the same testing dataset with about 841 images, containing 15% of the base that we reserved, as well as 15% randomly-selected instances of the superset training dataset. The results are represented in Figure 5 in terms of the standard object detection metric mAP for (i) 50%, (iii) 70%, and (iii) the average over multiple IoU levels. The mAP scores are reported once on the validation set, including the images that were used to tune the model's hyper-parameters during training, and once again, on the testing set.

Table 7: Images Contain Missing Specifications By Datasets

	Train	Val	Test	Total
Caltech	201	44	43	288
ECP	871	168	155	1,194
CrowdHuman	562	121	120	803
WiderPerson	358	77	77	512
Total (used in semantic augmentation)	1,992	410	395	2,797

The first bars (blue) represent the model's performance on the base dataset (before the augmentation), while the second ones (yellow) show the performance after the semantic augmentation. As we hypothesized the model's performance is improved when the neural network is trained with a dataset that is systematically augmented according to the semantics of a domain concept that the model aims to detect. Note a slight increase in accuracy of software systems with safety applications in reality translates to preventing severe damage to people's lives, properties or the environment.

Since specifications were partly derived using ResNeXt-101 (inherited from ResNet), to remove detection bias and assess the approach in a cross-cutting evaluation, we repeated the experiments with YOLOv3 object detector equipped with Darknet-53 backbone [46]. The training and testing datasets, processes, and parameters held the same as the previous experiments. Darknet was trained significantly faster than ResNet50, taking 4 hours before and 5 hours after the augmentation under the same conditions as ResNet. The models mAP scores based on 50%, 75%, and the average IoU are illustrated in Figure 6. In agreement with our objectives, the base mAP scores are improved when the model is re-trained with the missing semantics of the domain.

On a side note, both models' performance improved on the testing set in comparison to the validation set. This provides support for proper training and the models' fitness for coping with unseen

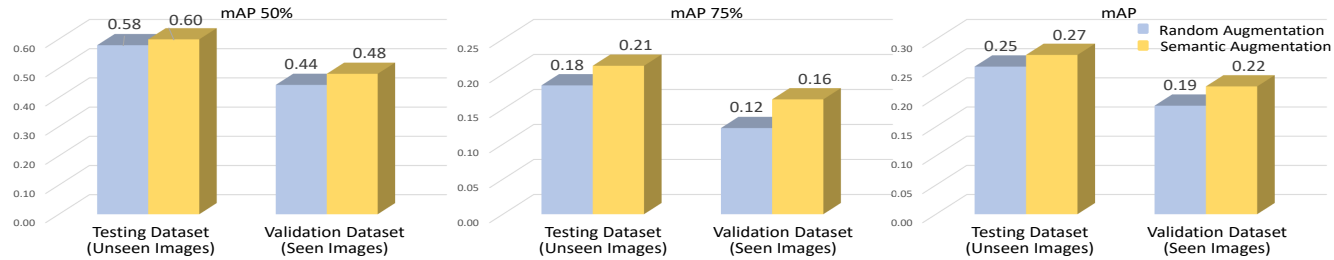

Figure 7: *D*arknet-53 mAP-50%, mAP-75%, mAP on pedestrian classification after random and after semantic augmentations.

data. The improved accuracy from validation to testing sets is not due to training underfitted models as we observe the same pattern for models trained on the base datasets. As shown in Figures 5 and 6 the accuracy generally decreases for higher IoU values, as a correct answer requires more area of overlap between the predicted and ground truth bounding boxes.

4.2.2 Semantic Augmentation vs. (Same-Size) Random Augmentation: To further evaluate the approach, we assessed whether adding randomly-selected additional images (non-systematic increase in the size of the training dataset) also result in a more accurate model. If true, then it became apparent that the detection improvement in the previous experiments was only due to the use of a larger dataset for training, not due to the systematic selection of the specification-guided samples. Hence, once more we trained Darknet with the same base dataset. However here, while the number of the augmenting images were equal to the images in the semantic augmentation, they were rather randomly selected from the superset. Re-training the network with the randomly augmented dataset took about 5 hours. After training, we compared the detector's accuracy on the same testing dataset, used in the previous experiment, before and after the augmentation. The detector's performance is shown in Figure 7. The results support the effectiveness of using semantic specifications of domain concepts in improving the generalizability of pedestrian detection. On a side note, Figures 5, 6, and 7 reported that the re-trained ResNet generally achieved higher accuracy than Darknet in detecting pedestrians on the selected base dataset.

Although data augmentation is shown to improve image classification, its potential has not yet been thoroughly investigated for object detection [77]. However, due to high cost of annotation, data augmentation may be of even greater importance for detection tasks [77]. Therefore this paper investigated the application of systematic and semantic data augmentation on pedestrian detection performance. In this work, we first showed hard-to-specify domain-related concepts can be partially specified by exploiting the existing domain knowledge. We further found that semantic specifications of the domain concept pedestrian is effective for identifying the missing variants of the concept within five commonly-used but unsystematically-collected pedestrian datasets. We also observed that by exploiting and incorporating the derived specifications into the training dataset, we can semantically improve MLSC accuracy in pedestrian detection. Our experiments with CityPerson as the base dataset showed that semantic augmentation improves pedestrian detection accuracy in the ResNet and the Darknet-53 respectively 4 and 7 mAP value, while the best augmentation policy identified in the literature with COCO dataset improves a strong baseline on PASCAL-VOC only by +2.7 mAP [77]. Further, repeating our experiments with the same number of randomly-selected images, we observed about +1 mAP improvement. This paper selected pedestrian as a critical concept in the automotive domain, yet the proposed approach and semantic principles are generalizable to other domain concepts.

5 APPLICATIONS

We foresee further potential applications for this work:

Formal Benchmarks: The presentation of the derived specifications in the form of a semantic web can represent a formal definition of the categories between the concepts, data, and entities and therefore, facilitates effective communication of the extracted information, knowledge sharing and reuse, and provides a machine-readable point of reference for hard-to-specify domain concepts.

Dataset Collection: In the proposed approach, we exploit the learned domain knowledge to compensate (augment) for semantic weaknesses of existing datasets. Given the established benchmarks, one can identify the most important dimensions of a concept's variations within the domain. The identification of deterministic features provides guidance for the process of data collection.

Model Verification: A representative dataset is the underlying condition of a well-trained model, yet does not guarantee that a particular model equally learns all variations of a concept. The specifications can be used to verify if the ML model has adequately learned the concepts variations. A requirement of this application is to interpret ML models in terms of domain-related features.

Safety Assurance: The majority of suggested verification and testing methods in safety standards rely on software specifications being available [50]. Generating a set of pre-defined domain specifications permits verification of ML software components against this list, improving the process of MLSC safety assurance.

Concept Drift: In ML, *concept drift* refers to the phenomena that important attributes of a concept to be predicted will change over time [64]. To address this, one can use the proposed approach to repeatedly extract the most recent body of knowledge, update the stale specifications, and incorporate new images into the dataset.

6 RELATED WORK

Domain Specifications: There have been attempts by different communities to specify requirements for MLSC by creating (1) component-level specifications: to define the behavior of MLSC as a whole with respect to how they address the target applications [54]. However, in such approaches, the implication of the high-level specification to the downstream MLSC development tasks, such as data collection and model selection, is unclear; (2) dataset specifications: since dataset management is critical to the

overall quality of systems with MLSC [26, 27]. However, studies in this area are limited to specific domains; (3) model specifications: based on the particular machine learning algorithms authors aim to define how the theoretical properties should hold during implementation [52, 53]; and (4) development process specifications: to produce consistent results authors proposed that the training procedure be clearly specified to meet the functional safety standards [49]. Thus, the lack of specification challenge remains for *domain-specific concepts* [49].

Dataset Augmentation: Data augmentation is an effective technique to alleviate the overfitting problem in training deep networks [21, 23, 30, 31, 57, 68]. A large body of research applies dataset augmentation methods to address the problem of overfitting MLSC and to improve their generalizability. The existing dataset augmentation approaches rely on the assumption that more information can be extracted from the original dataset through augmentations. These augmentations artificially inflate the training dataset size by either data warping or oversampling [56]. Data warping augmentations transform existing images such that their labels are preserved. This encompasses augmentations such as geometric and color transformations [3, 9], random erasing [76], adversarial training [39, 60, 72], and neural style transfer [16]. Oversampling augmentations create synthetic instances and add them to the training set. This includes mixing images, feature space augmentations [28, 63], and generative adversarial networks (GANs) [18, 25]. All the research referenced here focuses on visualization concerns by applying the augmentation to the existing set of images in the dataset. However, if a feature is initially missing from the dataset, these approaches can not be used to add the missing feature to the dataset. As opposed to the existing work, our approach focuses on filling the gap between conceptualization and visualization.

Pedestrian Detection: In the past decade, pedestrian detection has received significant attention, as evidenced by over 2,000 research publications [7]. Most existing pedestrian detection algorithms are either based on a set of handcrafted features or features extracted by deep convolutional neural networks (CNNs) [7]. Before the success of deep CNNs in computer vision tasks, a variety of handcrafted feature descriptors, including SIFT [36], LBP [40], SURF [4], HOG [10], and Haar [66], have been investigated in the context of pedestrian detection. These handcrafted features usually extract color, texture, or edge information from images. More recently, with the success of deep learning in generic object detection, several attempts have been made to apply deep CNN features to pedestrian detection [41, 51]. Studies show that despite significant progress, performance still has much room for improvement. In particular, detection is disappointing at low resolution and for partially occluded pedestrians [14]. Most existing methods focus on pedestrian detection from a specific type of dataset and cannot guarantee that the proposed methods will be generalizable to a significant degree [20, 67]. To that end, our proposed approach focuses particularly on using domain knowledge to improve MLSC generalizability in detecting pedestrians.

7 THREATS TO VALIDITY

Internal Validity When there is a causal relationship between the dependent and the independent type of variable. The level of detail and completeness of the specifications, in addition to their correctness, significantly impacts the MLSC performance. While absolute completeness of the derived specifications is open to question, the results demonstrated relative completeness and improved prediction ability after the augmentation. Moreover, determining completeness of specifications of hard-to-specify concepts, such as pedestrian, is highly inconclusive. Additionally, the inferred textual-based specifications are inconsistent with regard to their informativity. Some specifications provided more useful, detailed, and descriptive information of a pedestrian, whereas others described environment-related attributes. This happened for two main reasons: first, textual terms were extracted out of their context and second, a limited number of predicates were used. These issues did not emerge in the visual datasets since the relationships were inferred in their context and USGG is trained for a wider variety of predicates. **Construct Validity** refers to the type in which the construct of the test is involved in predicting the relationship for the dependent type of variable. We evaluated our approach with a limited number of datasets, detectors, and missing specifications, mainly due to computationally expensive experiments, involving R-CNNs multiple training on images and video frames. However, we minimized this threat through selecting the best performing and most commonly used state-of-the-art datasets and detectors as well the most confident and repeated specifications, extracted from large-scale and wide-reaching datasets. **External Validity** refers to the type where there is a causal relationship between the cause and the effect. The experiments in this paper were carried out only for the domain concept *pedestrian*. While the approach and knowledge sources are generalizable, further experimentation is required to determine whether applying the approach to other concepts improves MLSC accuracy in other domains.

8 CONCLUSION AND FUTURE WORK

We presented a generalizable approach for characterizing the extent to which a dataset lacks subsidiary features of a domain concept *pedestrian*. While most research to date focuses on dataset bias, such that a principal direction is sub-sampling an existing dataset, here we tackled the complex problem of finding elements of a targeted domain concept that are missing from an image dataset.

We exploited the existing knowledge sources to derive a set of partial specifications for a potential domain's concept *pedestrian*, whose perception is crucially important for autonomous driving safety. We first evaluated the accuracy and relative completeness of the established specifications through training multiple classifiers and with respect to benchmark pedestrian datasets. Further, to evaluate the usefulness of the derived specifications, we compared the accuracy of pedestrian detectors before, after a specification-based, and a random-based augmentation of their training dataset. We plan to create a feature model and semantic web to represent the formal definition of domain concepts. The creation of a semantic web helps to better communicate the extracted information, facilitates knowledge sharing and reuse, and provides a point of reference for the hard-to-specify domain concepts. We additionally refer to the established benchmarks to identify and retract those features that seem to be irrelevant to the domain concepts.

9 ACKNOWLEDGMENTS

This work is partially funded by NSF award number of 2124606. This research used resources of ddiLab Laboratory.

REFERENCES

[1] [n.d.]. Onelook Dictionary Search. https://www.onelook.com/. Accessed: 2020-02-08.

[2] Alhanoof Althnian, Duaa AlSaeed, Heyam Al-Baity, Amani Samha, Alanoud Bin Dris, Najla Alzakari, Afnan Abou Elwafa, and Heba Kurdi. 2021. Impact of Dataset Size on Classification Performance: An Empirical Evaluation in the Medical Domain. *Applied Sciences* 11, 2 (2021), 796.

[3] Hessam Bagherinezhad, Maxwell Horton, Mohammad Rastegari, and Ali Farhadi. 2018. Label refinery: Improving imagenet classification through label progression. *arXiv preprint arXiv:1805.02641* (2018).

[4] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. Surf: Speeded up robust features. In *European conference on computer vision*. Springer, 404–417.

[5] Christopher M Bishop. 2006. *Pattern recognition and machine learning*. springer.

[6] Markus Braun, Sebastian Krebs, Fabian Flohr, and Dariu Gavrila. 2019. EuroCity Persons: A Novel Benchmark for Person Detection in Traffic Scenes. *IEEE Trans. Pattern Anal. Mach. Intell.* (Feb. 2019).

[7] Jiale Cao, Yanwei Pang, Jin Xie, Fahad Shahbaz Khan, and Ling Shao. 2020. From Handcrafted to Deep Features for Pedestrian Detection: A Survey. (Oct. 2020). arXiv:2010.00456 [cs.CV]

[8] Jane Cleland-Huang. 2015. Mining domain knowledge [requirements]. *IEEE Software* 32, 3 (2015), 16–19.

[9] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. 2018. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501* (2018).

[10] Navneet Dalal and Bill Triggs. 2005. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, Vol. 1. Ieee, 886–893.

[11] Diego Dermeval, Jéssyka Vilela, Ig Ibert Bittencourt, Jaelson Castro, Seiji Isotani, Patrick Brito, and Alan Silva. 2016. Applications of ontologies in requirements engineering: a systematic review of the literature. *Requirements Engineering* 21, 4 (2016), 405–437.

[12] Josip Djolonga, Jessica Yung, Michael Tschannen, Rob Romijnders, Lucas Beyer, Alexander Kolesnikov, Joan Puigcerver, Matthias Minderer, Alexander D'Amour, Dan Moldovan, et al. 2021. On robustness and transferability of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 16458–16468.

[13] P Dollar, C Wojek, B Schiele, and P Perona. 2009. Pedestrian detection: A benchmark. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*. ieeexplore.ieee.org, 304–311.

[14] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. 2012. Pedestrian detection: an evaluation of the state of the art. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 4 (April 2012), 743–761.

[15] Paul Duguid. 2007. Inheritance and loss? A brief survey of Google Books. *First Monday* (2007).

[16] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2015. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576* (2015).

[17] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. 2013. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* 32, 11 (2013), 1231–1237.

[18] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* (2014).

[19] Jin Guo, Marek Gibiec, and Jane Cleland-Huang. 2017. Tackling the term-mismatch problem in automated trace retrieval. *Empirical Software Engineering* 22, 3 (2017), 1103–1142.

[20] Irtiza Hasan, Shengcai Liao, Jinpeng Li, Saad Ullah Akram, and Ling Shao. 2020. Generalizable Pedestrian Detection: The Elephant In The Room. (March 2020). arXiv:2003.08799 [cs.CV]

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[22] B. C. Hu, R. Salay, K. Czarnecki, M. Rahimi, G. Selim, and M. Chechik. 2020. Towards Requirements Specification for Machine-learned Perception Based on Human Performance. In *2020 IEEE Seventh International Workshop on Artificial Intelligence for Requirements Engineering (AIRE)*. 48–51. https://doi.org/10.1109/AIRE51212.2020.00014

[23] Gao Huang, Zhuang Liu, Geoff Pleiss, Laurens Van Der Maaten, and Kilian Weinberger. 2019. Convolutional networks with dense connectivity. *IEEE transactions on pattern analysis and machine intelligence* (2019).

[24] Christian Kaestner. 2020. Machine Learning is Requirements Engineering — On the Role of Bugs, Verification, and Validation in Machine Learning. https://medium.com/analytics-vidhya/machine-learning-is-requirements-engineering-8957aee55ef4.

[25] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. 2017. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196* (2017).

[26] Marc Kohli, Ronald Summers, and Jr Geis. 2017. Medical Image Data and Datasets in the Era of Machine Learning. *JDI* 30(4) (2017), 392–399. https://doi.org/10.1007/s10278-017-9976-3

[27] Marc D Kohli, Ronald M Summers, and J Raymond Geis. 2017. Medical image data and datasets in the era of machine learning—whitepaper from the 2016 C-MIMI meeting dataset session. *Journal of Digital Imaging* 30, 4 (2017), 392–399.

[28] Tomohiko Konno and Michiaki Iwazume. 2018. Icing on the cake: An easy and quick post-learnig method you can try after deep learning. *arXiv preprint arXiv:1807.06540* (2018).

[29] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael S Bernstein, and Li Fei-Fei. 2017. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. *Int. J. Comput. Vis.* 123, 1 (May 2017), 32–73.

[30] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).

[31] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* 25 (2012), 1097–1105.

[32] Yonghua Li and Jane Cleland-Huang. 2013. Ontology-based trace retrieval. In *2013 7th International Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE)*. IEEE, 30–36.

[33] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. 2017. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2117–2125.

[34] Trond Linjordet and Krisztian Balog. 2019. Impact of training dataset size on neural answer selection models. In *European Conference on Information Retrieval*. Springer, 828–835.

[35] Edward Loper and Steven Bird. 2002. Nltk: The natural language toolkit. *arXiv preprint cs/0205028* (2002).

[36] David G Lowe. 2004. Distinctive image features from scale-invariant keypoints. *International journal of computer vision* 60, 2 (2004), 91–110.

[37] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[38] George A Miller. 1995. WordNet: a lexical database for English. *Commun. ACM* 38, 11 (1995), 39–41.

[39] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. 2016. Deepfool: a simple and accurate method to fool deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2574–2582.

[40] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. 2002. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence* 24, 7 (2002), 971–987.

[41] Wanli Ouyang and Xiaogang Wang. 2013. Joint deep learning for pedestrian detection. In *Proceedings of the IEEE international conference on computer vision*. 2056–2063.

[42] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *the Journal of machine Learning research* 12 (2011), 2825–2830.

[43] Joseph Prusa, Taghi M Khoshgoftaar, and Naeem Seliya. 2015. The effect of dataset size on training tweet sentiment classifiers. In *2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 96–102.

[44] Mona Rahimi, Jin LC Guo, Sahar Kokaly, and Marsha Chechik. 2019. Toward Requirements Specification for Machine-Learned Components. In *2019 IEEE 27th International Requirements Engineering Conference Workshops (REW)*. IEEE, 241–244.

[45] Amir Rasouli, Iuliia Kotseruba, and John K Tsotsos. 2018. It's Not All About Size: On the Role of Data Properties in Pedestrian Detection. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*. 0–0.

[46] Joseph Redmon and Ali Farhadi. 2018. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* (2018).

[47] Radim Řehřek, Petr Sojka, et al. 2011. Gensim—statistical semantics in python. *Retrieved from genism. org* (2011).

[48] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *arXiv preprint arXiv:1506.01497* (2015).

[49] Rick Salay and Krzysztof Czarnecki. 2018. Using Machine Learning Safely in Automotive Software: An Assessment and Adaption of Software Process Requirements in ISO 26262. *ArXiv* abs/1808.01614 (2018).

[50] Rick Salay and Czarnecki Krzysztof. 2018. Using machine learning safely in automotive software: An assessment and adaption of software process requirements in ISO 26262. *arXiv preprint arXiv:1808.01614* (2018).

[51] Pierre Sermanet, Koray Kavukcuoglu, Soumith Chintala, and Yann LeCun. 2013. Pedestrian detection with unsupervised multi-stage feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3626–3633.

[52] Sanjit A. Seshia et al. 2018. Formal Specification for Deep Neural Networks. In *ATVA'18*. Springer, 20–34.

[53] Sanjit A Seshia, Ankush Desai, Tommaso Dreossi, Daniel J Fremont, Shromona Ghosh, Edward Kim, Sumukh Shivakumar, Marcell Vazquez-Chanlatte, and Xiangyu Yue. 2018. Formal specification for deep neural networks. In *International Symposium on Automated Technology for Verification and Analysis*. Springer, 20–34.

[54] Sanjit A. Seshia and Dorsa Sadigh. 2016. Towards Verified Artificial Intelligence. *ArXiv* abs/1606.08514 (2016).

[55] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. 2018. CrowdHuman: A Benchmark for Detecting Human in a Crowd. (April 2018). arXiv:1805.00123 [cs.CV]

[56] Connor Shorten and Taghi M Khoshgoftaar. 2019. A survey on Image Data Augmentation for Deep Learning. *Journal of Big Data* 6, 1 (July 2019), 60.

[57] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[58] Bernd Spanfelner, Detlev Richter, Susanne Ebel, Ulf Wilhelm, Wolfgang Branz, and Carsten Patz. 2012. Challenges in applying the ISO 26262 for driver assistance systems. *Tagung Fahrerassistenz, München* 15, 16 (2012), 2012.

[59] Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. , 4444–4451 pages. http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14972

[60] Jiawei Su, Danilo Vasconcellos Vargas, and Kouichi Sakurai. 2019. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation* 23, 5 (2019), 828–841.

[61] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. 2020. Unbiased scene graph generation from biased training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. openaccess.thecvf.com, 3716–3725.

[62] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. 2019. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6619–6628.

[63] V Terrance and W Taylor Graham. 2017. Dataset augmentation in feature space. In *Proceedings of the international conference on machine learning (ICML), workshop track*.

[64] Alexey Tsymbal. [n.d.]. The problem of concept drift: definitions and related work. ([n. d.]).

[65] Kunal Verma and Alex Kass. 2008. Requirements analysis tool: A tool for automatically analyzing software requirements documents. In *International semantic web conference*. Springer, 751–763.

[66] Paul Viola and Michael J Jones. 2004. Robust real-time face detection. *International journal of computer vision* 57, 2 (2004), 137–154.

[67] Xiaogang Wang, Meng Wang, and Wei Li. 2013. Scene-specific pedestrian detection for static video surveillance. *IEEE transactions on pattern analysis and machine intelligence* 36, 2 (2013), 361–374.

[68] Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Gao Huang, and Cheng Wu. 2019. Implicit Semantic Data Augmentation for Deep Networks. In *Advances in Neural Information Processing Systems*, H Wallach, H Larochelle, A Beygelzimer, F dAlché-Buc, E Fox, and R Garnett (Eds.), Vol. 32. Curran Associates, Inc., 12635–12644.

[69] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. 2017. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1492–1500.

[70] Danfei Xu, Yuke Zhu, Christopher B Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5410–5419.

[71] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10685–10694.

[72] Michał Zajac, Konrad Zołna, Negar Rostamzadeh, and Pedro O Pinheiro. 2019. Adversarial framing for image and video classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 10077–10078.

[73] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5831–5840.

[74] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. 2017. Citypersons: A diverse dataset for pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. openaccess.thecvf.com, 3213–3221.

[75] Shifeng Zhang, Yiliang Xie, Jun Wan, Hansheng Xia, Stan Z Li, and Guodong Guo. 2019. WiderPerson: A Diverse Dataset for Dense Pedestrian Detection in the Wild. (Sept. 2019). arXiv:1909.12118 [cs.CV]

[76] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2020. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 13001–13008.

[77] Barret Zoph, Ekin D Cubuk, Golnaz Ghiasi, Tsung-Yi Lin, Jonathon Shlens, and Quoc V Le. 2020. Learning data augmentation strategies for object detection. In *European Conference on Computer Vision*. Springer, 566–583.