Detecting Persuasive Atypicality by Modeling Contextual Compatibility

Meiqi Guo  Rebecca Hwa  Adriana Kovashka
Department of Computer Science, University of Pittsburgh, Pittsburgh PA, USA
meiqi.guo@pitt.edu  {hwa, kovashka}@cs.pitt.edu
https://github.com/MeiqiGuo/ICCV2021-AtypicalityDetection

Abstract

We propose a new approach to detect atypicality in persuasive imagery. Unlike atypicality which has been studied in prior work, persuasive atypicality has a particular purpose to convey meaning, and relies on understanding the common-sense spatial relations of objects. We propose a self-supervised attention-based technique which captures contextual compatibility, and models spatial relations in a precise manner. We further experiment with capturing common sense through the semantics of co-occurring object classes. We verify our approach on a dataset of atypicality in visual advertisements, as well as a second dataset capturing atypicality that has no persuasive intent.

1. Introduction

Visually creative images, such as advertisements or public service announcements, may purposefully contain atypical portrayals of objects as a rhetorical way for attracting viewers’ attention [15]. In the marketing and communications research community, atypicality has gained attention because of its importance to understanding the persuasiveness and rhetoric of visual media [28, 23, 48]. However, detecting this type of atypicality is challenging for intelligent systems. First, atypicality may involve metaphorical object transformations or intentionally surprising composed objects. Second, the atypicality transformation types are diverse and creative. Third, unpacking them may require common-sense reasoning. For example, Fig. 1a is an atypical advertisement for a beverage. It is unusual for a pig to wear a bridal veil even though the pig and veil are both normal objects. The ability to detect this type of purposefully atypical objects and understand their roles in conveying the intent of the image is necessary for an intelligent system to reason about information in persuasive media. In this work, we propose to model implicit knowledge of contextual compatibility in order to detect persuasive atypicality.

Our first hypothesis is that persuasive atypicality can be detected by checking the compatibility between each possibly atypical object and the rest of the image as context. For example, in Fig. 1a, the pig is not compatible with its context (a bridal veil on its head), and the veil is also not compatible with its context — on a pig’s head. We propose an unsupervised approach by using reconstruction losses of masked regions. We expect that a self-supervised model trained on masked region reconstruction could learn enough implicit knowledge of contextual compatibility; this pretrained model may then be used to detect atypical images.

Our second hypothesis is that the interactions between objects and their spatial relative positions play a key role in detecting atypicality. If it were a handled brush instead of a bridal veil over the pig’s head (Fig. 1b), or if the veil were placed at another location instead of on top of the pig’s head (Fig. 1c), the image would no longer be atypical. In order to better interpret object-object spatial interaction, we propose a new method to compute the attention weights between key-query regions of our transformer-based models.

Finally, our third hypothesis is that, for some types of persuasive atypicality, the semantic relation between nearby object classes may offer compatibility clues beyond visual features. In Fig. 1a, knowing that there is a “pig” and a “bridal veil” and their spatial relationship may be helpful to conclude that the image is atypical, instead of knowing exactly what that pig or veil look like. To take advantage of
semantic knowledge learned in language models, we fine-
tune BERT on detected class labels of regions of interest.

Experiments on a recent visual advertising dataset [48] demonstrate the effectiveness of our approaches and support our hypotheses. Our approach outperforms prior approaches for abnormality detection (e.g., a One-Class SVM) by more than 9%. We have also gleaned insights on how different types of persuasive atypicality impact the detection performance. We validate that atypicality transformations involving spatial interactions between objects are better solved by our approaches than baselines. Then we evaluate the generalization of our approaches using an existing dataset of real-scene, non-persuasive atypical images [46].

To understand the labelling requirement of both tasks, we compare our unsupervised approaches of contextual compatibility with supervised models which are trained on the ground-truth labels. We observe very different performances on the two datasets, which reveals that the labelling requirement depends on the training size ratio between supervised and unsupervised methods and the complexity of the atypicality transformations. Lastly, we investigate two possibilities for representing the image context: visual compatibility versus semantic compatibility. Experimental results show that visual features are essential, but the semantic compatibility can help when atypicality transformations feature unusual combination of normal objects.

2. Related Work

To set our work, which focuses on images with persuasive intent, in the context of the broader atypicality detection literature, we present an overview of prior efforts. Moreover, because our approach for capturing spatial relationships is based on self-attention, we also review related work on transformers and self-attention in computer vision, as well as self-supervised learning through masking.

Atypicality Detection. Prior work focuses on detecting atypical objects in real-world images. Bergmann et al. developed unsupervised methods for detecting diverse defects such as scratches, dents, contaminations, and structural changes [4]. Wang et al. detect atypical objects through Gaussian Process models based on the distribution of object detection scores in different regions of interest [46]. Choi et al. detect out-of-context objects and scenes by a graphical model and show that physical support relationships between objects are an important clue [8]. Saleh et al. classify anomalies in images in three categories (object-centric, context-centric and scene-centric) and build a generative model from visual attributes in regular images [38]. Most prior studies investigate atypicality that is (1) physically created in the real world, rather than generated with computer graphics; and (2) predominantly accidental and certainly not aiming to convey meaning or persuade an audience to take a certain action. One exception is the work of Ye et al. [48] on interpreting the visual rhetoric in advertisements. Because ad images are intentionally designed by experts to create an association in viewers’ minds, many atypical objects in their dataset cannot appear in the real world (e.g., a kiwi inside an apple). Moreover, those objects are diverse and not limited to a specific set of categories, as they are in Wang et al.’s work [46], in which atypical objects are all from PASCAL VOC [12].

Ye et al. devised a taxonomy of atypicality based on object transformations but only supervisedly trained a basic VGG16 model to detect atypicality as a whole, not per category. The eight categories they defined are:

1. **Texture Replacement 1 (TR1)**: Objects’ texture borrowed from another object, e.g., kiwi inside apple, Fig. 2a.
2. **Texture Replacement 2 (TR2)**: Texture created by combining several small objects, e.g., owl from beans, 2b.
3. **Object Inside Object (OIO)**, e.g., auto racing in car, 2c.
4. **Object w/ Missing Part (OMP)**, woman w/o mouth, 2d.
5. **Combination of Parts (CP)**: Object composed by parts from different objects, e.g., deer head with hand horn, 2e.
6. **Solid Deformed Object (SDO)**, e.g., human arm bent, 2f.
7. **Liquid Deformed Object (LDO)**, e.g., beer as player, 2g.
8. **Object Replacement (OR)**: The whole object appearing in the context normally associated with another, e.g., cigarettes placed in the context where bullets occur, 2h.

Our work closely examines the relationship between our proposed models and each persuasive atypicality category.

Transformers in Computer Vision. Transformers were first introduced by Vaswani et al. as a new network architecture based on attention mechanisms for machine translation [44]. Transformers can perform a variety of tasks by computing scores solely based on self-attention layers,
without the need for expensive and non-parallelizable recurrence. Recently, transformers have been demonstrated as an effective architecture in many problems in natural language processing [9, 34], speech processing [25, 42], computer vision [30, 6, 11] and vision-language tasks [43, 24, 20, 40, 50, 32, 22, 7]. Since the transformer architecture is permutation-invariant, a positional encoding is necessary to provide the order information of the sequential input. For work which represents the image by a set of regions of interest, a common way is to embed the bounding-box coordinates of each region and potentially the fraction of image area covered [7, 43, 24]. For pixel-level representation, Carion et al. explore sinusoidal embeddings based on the absolute position and a learnt positional encoding of pixels [6]. However, experimentation in machine translation [39] and music generation [16] suggested that using relative positional embeddings results in significantly better accuracy. Adding the absolute positional encoding to the inputs, as done in [11, 6, 43], is not always sufficient. Explicitly modeling relative position information separately from other inputs (e.g., features) extends the self-attention mechanism to efficiently consider spatial relationship between each query-key pair [39, 35, 3, 51]. Ramachandran et al. [35] and Bello et al. [3] define 2D relative position embeddings by the relative distance between the position of the query and key pixel. Our approach follows the idea of Ramachandran et al. except that our relative position embedding is at the region level. Our objective is to model spatial relationships between objects, thus a pixel-level representation does not make sense. Besides, we can add overlapping area information to the relative spatial feature between two regions, which a pixel-level representation cannot. Kant et al. also consider relative spatial relationship between object regions, but they transform spatial relationship into twelve categories and then apply the adjacency matrices as an additional attention mask on their base model architecture [17]. Therefore, they only consider the relative spatial direction and ignore the concrete relative distance between pairwise objects, which loses essential information compared to our method. Another weakness is their spatial relationship categories do not have full coverage, e.g., the spatial relationship between two non-overlapped objects far from each other is ignored. We are not aware of any prior work that performs atypicality detection with any type of transformer, nor with the relative-spatial transformer we propose.

**Self-supervised Learning.** Self-supervised learning through masked or next-token prediction is a commonly-used method for language modeling in natural language processing [9, 34]. In computer vision, methods exist to learn visual representations through pretext tasks, e.g., via colorization [49, 19, 45], jigsaw puzzles [29, 10], inpainting [31], instance discrimination [47], or even pretext-invariant objectives [26]. Prior work demonstrates the effectiveness of these visual representations for transfer learning [13]. Representations can also be learned by predicting context in a multi-modal setting [43, 24, 41, 5, 27]. Our work follows Tan et al.’s method by using masked object feature regression for learning visual representations [43], but Tan et al. operate in a cross-modal setting, while we operate in a visual one. To our knowledge, we are the first to use self-supervised learning based on context prediction for detecting image atypicality.

### 3. Approach

We define atypicality detection as a binary classification task: for a given image \(I\), our model aims to predict whether \(I\) is atypical or not. We first present our unsupervised atypicality detection system, which leverages masked region reconstruction as the pretext task, and learns implicit knowledge of contextual compatibility from large-scale unlabeled data. The reconstruction losses of masked regions are the clue for predicting atypicality of a test image. We then introduce our Relative-Spatial Transformer which extends the self-attention layer to explicitly model relative position information separately from visual features.

#### 3.1. Masked Region Reconstruction

Fig. 3a shows an overview of our approach. An image \(I\) is represented by a set of regions \(R = \{(v_1, p_1), (v_2, p_2), \ldots, (v_n, p_n)\}\), where \(v_i\) could be region \(i\)’s visual feature vector, pixel matrix, class labels, etc., and \(p_i\) is the positional information. Our hypothesis is that if an image is atypical, the objects appearing in it would not be compatible with each other, thus it would be hard to reconstruct a masked region from image context. We first pre-train a model to reconstruct a region from context using normal cases, then use it to detect atypicality in new test images.

For the pre-training process, we take inspiration from masked language modeling (e.g., BERT [9]) and cross-modality representation learning (e.g., LXMERT [43]). The model is trained to reconstruct the masked regions given the remaining regions, on many general, normal images (which could potentially contain a small proportion of atypical cases). Different from BERT or LXMERT, which aims to learn a language or visual-language representation, our model aims to learn the common co-occurrences and typical spatial relationship between objects.

At test time, we mask each region in the image and compute the reconstruction loss. We compute the average loss of all regions as a clue for predicting atypicality. We use average rather than maximum loss because if an image is atypical, the masked region reconstruction loss is high not only when an atypical object is masked, but also when its surrounding object is masked since it is also hard to reconstruct a normal object from an atypical context.
3.2. Relative-Spatial Transformer

Our model extends the transformer architecture [44]. A common input representation to a transformer for computer vision tasks is the summation of the visual embedding and the positional embedding of the region [6, 43, 11]. However, this technique has two weaknesses. First, when computing attention weights with these input vectors, the visual feature and positional information share the same projection weight without any distinction, therefore the model cannot flexibly adjust the importance of region visual and position. Second, the positional embedding represents the absolute coordinate of the region, however, it is the relative spatial relationship between the masked and the context region which matters for detecting atypicality (e.g. is the veil above or below the pig?). In order to overcome both weaknesses, we propose the Relative-Spatial Transformer which (1) computes the visual-visual interaction and visual-position interaction separately, and (2) is shift-invariant, similar to convolutions but unlike a standard transformer.

The Relative-Spatial Transformer (RST) follows the same architecture as the transformer (T) of [44] except for a new way for computing the multi-head self-attention layer, as shown in Fig. 3. The attention weight of the query region \( i \) and key region \( j \) is computed as:

\[
A_{i,j}^{rel} = W_q^T W_{k,V} V_j + W_q^T W_{k,P} P_{j-i}
\]

where \( V_i \) and \( V_j \) are visual features of regions \( i \) and \( j \); \( W_q \) is the projection weight of the query region visual feature; \( W_{k,V} \) and \( W_{k,P} \) are respectively the key region’s projection weights of visual features and relative positions; and \( P_{j-i} \) is the relative position of region \( j \) with respect to region \( i \). The first term computes the interaction between the query and key visual content; the second term computes the interaction between the query visual content and the relative position of the key region. The summation of both terms shows the importance of the key region to the query region. Then we compute the normalized attention weight \( \alpha_{i,j}^{rel} \) as a softmax layer over \( A_{i,j}^{rel} \) for all possible key regions. The last hidden layer of region \( i \) is computed as:

\[
h_i = \sum_j \alpha_{i,j}^{rel} W_v V_j
\]

where \( W_v \) projects the value region’s visual feature.

The reconstruction loss of the masked region \( i \) is computed as the mean squared error (i.e. squared L2 norm) between the input visual feature \( v_i \) and the last hidden layer \( h_i \) of the encoder:

\[
L_i = ||v_i - h_i||_2^2
\]

For computing the relative position of \( j \) with respect to \( i \), we compute the x-axis and y-axis distance of the top-left and bottom-right corners of the two bounding boxes:

\[
P_{j-i} = [x_j^t - x_i^t, y_j^t - y_i^t, x_j^b - x_i^t, y_j^b - y_i^t]
\]

where \((x_i^t, y_i^t)\) is the coordinate of the left-top corner of region \( i \); \((x_j^t, y_j^t)\) is the coordinate of the right-bottom corner of region \( i \); similarly with region \( j \). We also explore adding Intersection-over-Union area between region \( i \) and \( j \) as an additional relative positional feature.

4. Experiments

In the subsequent experiments, we first evaluate our contextual compatibility modeling approach on the intent- and persuasion-driven atypicality in the Ads dataset [48]. Our
experiments show that Relative-Spatial Attention leads to an improvement across a diverse array of atypicality sub-categories. Then we test the generalization of our approach on real-world, non-persuasive atypical images, by detecting atypicality within each object class, on a dataset we refer to as the Single-Object dataset [46]. To understand the labelling requirement of each task, we compare our unsupervised contextual compatibility approaches with supervised models trained on the atypical/typical labels. When considering the different possibilities for representing the image context, we compare visual versus semantic compatibility.

4.1. Setup

Input Representations. We use Faster R-CNN [36] pre-trained on Visual Genome [18] for extracting the visual features [2]. Faster R-CNN itself uses ResNet-101 [14] pre-trained for classification on ImageNet [37]. We take the features of each detected object as the visual representation of the corresponding region. We select a fixed number of objects (36) by sorting detections by confidence score. Each region is represented by its bounding-box coordinates and its 2048-dimensional region-of-interest (RoI) features.

Self-supervised Training and Testing. Following BERT [9], we mask 15% of regions in each sequence at random during training. All masked regions are replaced by a trainable vector with the same dimension as the RoI feature. The spatial information of the masked region is given. We use a batch size of 128 and train for 20 epochs with learning rate of 1e-3. For testing, we mask one region given. We use a batch size of 128 and train for 20 epochs.

Baseline Models. We consider two baselines, Auto-encoder and One-Class SVM, since they are standard methods for detecting abnormality and outliers [1, 21]. For the Auto-encoder, we implement the same encoder as DCGAN’s discriminator and DC-GAN’s generator as the decoder [33], using the hyperparameters in [33]. The loss is L2 error between input and generated images. However, we make an interesting observation that atypicality relates to image complexity in a potentially counter-intuitive way: We found strong correlation between atypical images and relatively plain backgrounds, likely because ad designers of atypical images want to make sure the image is plain enough for the audience to notice the atypicality. Images with uniform background are more easily reconstructed while images with plenty of objects are harder. Further, images with more pixels tend to contain more information to be compressed and reconstructed. To ensure the auto-encoder captures atypicality rather than complexity, we need to normalize for image complexity. We first prepossess all images by resizing them to a fixed number of pixels (64*64). We also measure image complexity (IC) as the average of horizontal and vertical gradient of pixels (IC = avg(I_x^2 + I_y^2) where I_x and I_y are respectively the horizontal and vertical gradient). Then we divide the auto-encoder reconstruction loss by IC. In addition, to force the auto-encoder model to learn an effective encoder and decoder, we limit the dimension of the middle hidden layer to 2048 which is much smaller than the input image dimension (3*64*64). For the One-Class SVM model, we represent each image by the average of its 36 RoI feature vectors. Then we fit the One-Class SVM model with default settings on the training images.

4.2. Unsupervised persuasive atypicality detection

Data. We evaluate our method on a dataset of advertisement images where atypicality is creative and has a purpose to convince an audience to take a certain action [48]. The Ads dataset contains in total 64,832 ad images and the authors annotated 3,928 of them for the atypicality task. Since each image is annotated by one or multiple annotators, we set a rule for deciding the atypical/typical label if annotators do not agree with each other. In particular, we consider an ad atypical if any annotator labels it as atypical. We use the ifany rule because some atypical cases are subtle, subjective or need background knowledge, thus any annotator providing the atypical label is cause to believe the image is not quite typical. Under this labeling rule, there are 2,285 atypical ads and 1,643 typical ads. For the self-supervised training, we use all ads except for those 3,928 with atypicality labels. For supervised training and for testing, we randomly split the 3,928 atypical/typical images using a 7:1:2 ratio for train:val:test sets. Note we did not use any training data from the atypicality dataset for our unsupervised methods, to make them fairly comparable to supervised methods. All methods are evaluated on the exact same test set.

Results. The experimental results of our unsupervised contextual compatibility approaches are shown in the upper part (unsup) of Tab. 1. To gain insights on the impact of different types of persuasive atypicality on the detection result, we also report the model performance on the eight atypicality categories separately, as defined in the Ads dataset (Sec. 2). Experimental results in Tab. 1 show that our approaches significantly outperform baseline models overall (MICRO AVE) and for CP, OR, Others (with p-
value < 0.1). While Transformer (T) is an existing architecture, and Relative-Spatial Transformer (RST) is our new design, neither has been used for atypicality detection before. T outperforms the simpler baselines significantly, but RST achieves the best results overall. By looking into each category, RST leads to an improvement across a diverse array of atypicality types. In particular, the improvement of RST over T is large for TR1, TR2 and OIO where atypicality mainly comes from unusual spatial relationship between normal objects (these categories involve object compositions). These results demonstrate that our approach of checking for contextual compatibility is effective for detecting persuasive atypicality and our proposed RST architecture does capture object-object spatial relationships well. OMP is the only atypicality category for which the baseline model (One-Class SVM) is better than ours. This is because this type of atypicality only comes from a single object without any complex interaction with surrounding objects.

**Error analysis.** We qualitatively show several cases where the One-Class SVM fails (Fig. 4a - d) or both the baseline and our models fail (Fig. 4e - h). One-Class SVM fails when atypicality involves composition of normal objects (e.g., cream on top of alcohol bottle), while our transformer models (especially RST) detect this atypicality by learning context via self-supervised training and show large gains. However, our model fails to capture metaphoric similarity: Fig. 4e and 4f look typical at first, but shade versus puma, surfboard versus brand make them atypical. It also fails to interpret symbolic meanings: vodka is held like a microphone by Hitler who is a symbol of power in Fig. 4g. Thus, typicality judgment requires more fine-grained visual features, and knowledge of historical figures.

**Ablation.** To see the impact of the number of transformer blocks (model depth), we conduct an ablation study on the layer number ($L$). Considering the variation of relation position features, we add an additional feature, Intersection-over-Union area (IoU), to the previous relative coordinates. Results are shown in Tab. 2. We find that the deeper Transformer greatly improves over a shallow Transformer, while Relative-Spatial Transformers are less sensitive to depth. In addition, we observe that a shallow RS Transformer is competitive against a deep Transformer, suggesting that the proposed RS Transformer is more efficient. We also observe that adding the area overlap (IoU) feature slightly improves performance.

### 4.3. Performance on non-persuasive atypicality

**Data.** We investigate how our approach generalizes to non-persuasive atypicality, where the unusual object itself is the main source of atypicality and there is no need to consider the complex spatial relationship between objects for predicting atypicality. For answering this question, we use Wang et al.’s Single-Object dataset [46] for evaluation. Their dataset contains 20,420 regular/unusual images belonging to 20 classes. Different from the Ads dataset with various atypicality transformations, each image in the Single-Object dataset has only one main object which is atypical or not.

**Definition.** Let $C$ denote a given object category, with $C = C^r \cup C^u$ and $C^r \cap C^u = \emptyset$, where $C^r$ and $C^u$ respec-
tively means the set of regular images or unusual images in category $C$. The task is to determine, for any test image $I \in C$, whether $I \in C^u$. In other words, our task is to detect atypical images from each single object class. This is different from Wang et al.’s [46] problem setting: their task aims to determine, for any test image $I \in C \cup O^r$ (with $O^r$ denoting regular images not containing the object in category $C$), if $I \in C^u$. We formulate the task in a different way because our focus is to evaluate our methods on atypicality detection within a single object class and we only use $C^r$ as our training data. In contrast, [46] also train object detectors on $C^r \cup O^r$ then use the object detection scores for predicting atypicality in $C$. In conclusion, our method performance is not directly comparable to theirs since neither our training nor test set includes $O^r$.

We use the Auto-encoder model as a baseline. Given that each image only contains one main object, we do not normalize the auto-encoder loss with image complexity as we did for the Ads Dataset. We follow the same split as Wang et al., dividing $C^r$ into training ($C^r_{\text{train}}$) and test set ($C^r_{\text{test}}$). For each object category $C$, our models and the baseline are trained on $C^r_{\text{train}}$ and evaluated on $C^u \cup C^r_{\text{test}}$.

The upper part (unsup) of Tab. 3 shows the results. We use the same evaluation metric, Average Precision, as Wang et al. Different from what we observe with the Ads dataset, Transformer is generally more effective than the RS Transformer here. The reason is that RST does not capture useful information for predicting atypicality since the Single-Object dataset has little object-object spatial relationship as the atypicality source. Moreover, some learnt interactions between objects by RST might be noisy because of over-fitting with only hundreds of training samples (as shown in Tab. 4). For this task, a standard attention mechanism with regions’ absolute position as input can handle those single atypical objects well, and Transformer achieves comparable results to those shown in [46] (mAP of 0.90).⁵ In conclusion, our unsupervised approach by checking for contextual compatibility works well not only on persuasively creative images with complex atypicality transformation, but also on single-object images. As expected, RST is not beneficial for detecting non-persuasive atypicality of single object.

### 4.4. Are supervised labels essential for these tasks?

**Models.** To understand the labelling requirement for detecting atypicality, we compare our unsupervised contextual compatibility approaches with supervised models trained on the atypical/not labels, for both Ads and Single-Object. We use the same Transformer and RS Transformer architectures for fair comparison. We also add a supervised baseline model which is trained only on the RoI features (each image is represented by the average of all regions-of-interest features).⁶ For transformers, the output layer is an average pooling over the last hidden layer followed by a simple 2-layer neural network for predicting the atypicality label. For the RoI baseline, the input image features feed directly to the output layer which is the same 2-layer network.

**Results.** Tab. 1 and Tab. 3 show the comparison of unsupervised and supervised approaches for the Ads and Single-Object datasets, respectively. We find that for Ads, our unsupervised approaches achieve comparable performance to the supervised approaches, which highlights that even with labeling the task is still difficult. This also demonstrates the effectiveness of our proposed contextual compatibility method. When looking into each atypicality category, we observe the unsupervised RS Transformer wins on those atypicality transformations which involve more object-object interaction, e.g. TR1, TR2, OIO, CP. This is expected because RST efficiently learns contextual compat-

---

⁵Even though they are not directly comparable, the inclusion of these earlier results is still informative because we aimed to show our approach produces results in the same ballpark.

⁶The input features are the same as the One-Class SVM baseline. This baseline is conceptually similar to the approach in Ye et al. [48] except that they use VGG16 for extracting the image features.

<table>
<thead>
<tr>
<th>Methods</th>
<th>aeroplane</th>
<th>apple</th>
<th>bicycle</th>
<th>boat</th>
<th>building</th>
<th>bus</th>
<th>car</th>
<th>chair</th>
<th>cow</th>
<th>dining table</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>0.74</td>
<td>0.44</td>
<td>0.58</td>
<td>0.78</td>
<td>0.85</td>
<td>0.55</td>
<td>0.74</td>
<td>0.52</td>
<td>0.52</td>
<td>0.33</td>
</tr>
<tr>
<td>T (ours)</td>
<td>0.93</td>
<td>0.86</td>
<td>0.84</td>
<td>0.90</td>
<td>0.86</td>
<td>0.86</td>
<td>0.92</td>
<td>0.78</td>
<td>0.74</td>
<td>0.90</td>
</tr>
<tr>
<td>RST (ours)</td>
<td>0.90</td>
<td>0.78</td>
<td>0.76</td>
<td>0.88</td>
<td>0.83</td>
<td>0.78</td>
<td>0.87</td>
<td>0.70</td>
<td>0.65</td>
<td>0.82</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Methods</th>
<th>horse</th>
<th>house</th>
<th>motorbike</th>
<th>road</th>
<th>shoes</th>
<th>sofa</th>
<th>street</th>
<th>table lamp</th>
<th>train</th>
<th>tree</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>0.42</td>
<td>0.65</td>
<td>0.40</td>
<td>0.65</td>
<td>0.71</td>
<td>0.64</td>
<td>0.59</td>
<td>0.39</td>
<td>0.45</td>
<td>0.56</td>
<td>0.58</td>
</tr>
<tr>
<td>T (ours)</td>
<td>0.80</td>
<td>0.90</td>
<td>0.71</td>
<td>0.91</td>
<td>0.87</td>
<td>0.83</td>
<td>0.90</td>
<td>0.82</td>
<td>0.77</td>
<td>0.80</td>
<td>0.85</td>
</tr>
<tr>
<td>RST (ours)</td>
<td>0.75</td>
<td>0.90</td>
<td>0.60</td>
<td>0.85</td>
<td>0.76</td>
<td>0.75</td>
<td>0.90</td>
<td>0.62</td>
<td>0.67</td>
<td>0.74</td>
<td>0.78</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Methods</th>
<th>Ads</th>
<th>aeroplane</th>
<th>apple</th>
<th>bicycle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unsupervised</td>
<td>46,757</td>
<td>169</td>
<td>667</td>
<td>268</td>
</tr>
<tr>
<td>Supervised</td>
<td>2,741</td>
<td>189</td>
<td>429</td>
<td>312</td>
</tr>
</tbody>
</table>
We next consider different possibilities for representing the image context, namely checking visual versus semantic compatibility. Our previous experiments use Faster R-CNN RoI features which represent the visual content of the region and then learn compatibility from them. We now consider using the class labels predicted by Faster R-CNN as the semantic features of the region and then we use the same model for learning semantic compatibility.

**Training.** For unsupervised training with transformer-based models, the input is a sequence of class labels with the bounding-box coordinates of regions ordered by the detection confidence score. Similarly with visual features, we mask one (or several during the training) object class label by a [MASK] token in the input, and the model is trained to predict the class label of the masked region. We use the cross-entropy loss for training and testing; the loss is the atypicality signal. Since the input of class labels are discrete textual tokens, we project them through an embedding layer before feeding to the transformer; at the output, we project the last hidden layer of the masked input back to the class label by a decoder which shares the same weight as the embedding layer. We follow the same experimental setting as with the visual features. For supervised training, we fine-tune the pre-trained BERT model (bert-base-uncased)\(^7\) with the sequence of class labels as input. We use batch size 16, leaning rate 3e-5 and 5 epochs, as suggested in [9].

**Results.** Experimental results on the Ads dataset are shown in Tab. 5. We find that checking semantic compatibility (CL) is not as effective as checking the visual compatibility (VF) under the unsupervised setting. Thus, visual features contain more useful information (e.g., the visual features of an atypical apple and a typical apple are different; however, the class label input does not have this information when the atypical apple is correctly detected as "apple" by Faster R-CNN), and only checking the semantic compatibility is not enough for solving this task. However, fine-tuned BERT with predicted class labels slightly outperforms the unsupervised RS Transformer using visual feature input, especially for those categories whose atypical transformations are mainly from unusual combinations of normal objects, such as TR2 and OR, which are well captured by the semantic compatibility.

<table>
<thead>
<tr>
<th>Methods</th>
<th>TR1</th>
<th>TR2</th>
<th>OIO</th>
<th>OMP</th>
<th>CP</th>
<th>SDO</th>
<th>LDO</th>
<th>OR</th>
<th>Others</th>
<th>MICRO AVE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transformer with VF</td>
<td>62.66</td>
<td>60.72</td>
<td>63.07</td>
<td>42.52</td>
<td>69.18</td>
<td>63.71</td>
<td>61.63</td>
<td>64.05</td>
<td>63.68</td>
<td>62.86</td>
</tr>
<tr>
<td>RS Transformer with VF</td>
<td>67.50</td>
<td>68.37</td>
<td>67.31</td>
<td>55.18</td>
<td>71.26</td>
<td>68.67</td>
<td>63.99</td>
<td>61.84</td>
<td>59.68</td>
<td>64.32</td>
</tr>
<tr>
<td>Transformer with CL</td>
<td>51.39</td>
<td>58.28</td>
<td>61.90</td>
<td>41.53</td>
<td>62.76</td>
<td>54.80</td>
<td>60.49</td>
<td>56.09</td>
<td>62.38</td>
<td>57.63</td>
</tr>
<tr>
<td>RS Transformer with CL</td>
<td>54.89</td>
<td>62.30</td>
<td>60.49</td>
<td>47.03</td>
<td>61.47</td>
<td>58.25</td>
<td>53.00</td>
<td>58.28</td>
<td>61.76</td>
<td>58.46</td>
</tr>
<tr>
<td>sup Fine-tuned BERT with CL</td>
<td>62.94</td>
<td>69.59</td>
<td>59.02</td>
<td>56.25</td>
<td>70.74</td>
<td>69.87</td>
<td>61.92</td>
<td>65.05</td>
<td>64.71</td>
<td></td>
</tr>
</tbody>
</table>

Table 5. Comparison of Faster R-CNN RoI visual feature (VF) and predicted class label (CL). AUC for each atypicality category and micro ave are reported, with best AUC per column bolded. AUC for Fine-tuned BERT with CL is bolded if it outperforms all unsup. methods.

---

\(^7\)https://huggingface.co/transformers/model_doc/bert.html
References


