

# Answering Image Riddles using Vision and Reasoning through Probabilistic Soft Logic

Somak Aditya      Yezhou Yang      Chitta Baral  
 Arizona State University, Tempe, AZ  
 {saditya1, yz.yang, chitta}@asu.edu

Yiannis Aloimonos  
 University of Maryland, College Park  
 yiannis@cs.umd.edu

## Abstract

*In this work, we explore a genre of puzzles (“image riddles”) which involves a set of images and a question. Answering these puzzles require both capabilities involving visual detection (including object, activity recognition) and, knowledge-based or commonsense reasoning. We compile a dataset of over 3k riddles where each riddle consists of 4 images and a groundtruth answer. The annotations are validated using crowd-sourced evaluation. We also define an automatic evaluation metric to track future progress. Our task bears similarity with the commonly known IQ tasks such as analogy solving, sequence filling that are often used to test intelligence.*

*We develop a Probabilistic Reasoning-based approach that utilizes probabilistic commonsense knowledge to answer these riddles with a reasonable accuracy. We demonstrate the results of our approach using both automatic and human evaluations. Our approach achieves some promising results for these riddles and provides a strong baseline for future attempts. We make the entire dataset and related materials publicly available to the community in ImageRiddle Website (<http://bit.ly/22f9A1a>).*

## 1. Introduction



Figure 1. An Image Riddle Example. Question: “What word connects these images?” .

A key component of computer vision is understanding of images and it comes up in various tasks such as image captioning and visual question answering (VQA). In this paper, we propose a new task of “image riddles” which requires deeper and conceptual understanding of images. In this task a set of images are provided and one needs to find a concept

(described in words) that is invoked by all the images in that set. Often the common concept is not something that even a human can observe in her first glance but can come up with after some thought about the images. Hence the word “riddle” in the phrase “image riddles”. Figure 1 shows an example of an image riddle. The images individually connect to multiple concepts such as: *outdoors, nature, trees, road, forest, rainfall, waterfall, statue, rope, mosque* etc. On further thought, the common concept that emerges for this example is “fall”. Here, the first image represents the fall season (*concept*). There is a “waterfall” (*region*) in the second image. In the third image, it shows “rainfall” (*concept*) and the fourth image depicts that a statue is “fall”ing (*action/event*). The word “Fall” is invoked by all the images as it shows logical connections to objects, regions, actions or concepts specific to each image.

In addition, the answer also connects the most significant<sup>1</sup> aspects of the images. Other possible answers like “nature” or “outdoors” do not demonstrate such properties. They are too general. In essence, image riddles is a challenging task that not only tests our ability to detect visual items in a set of images, but also tests our knowledge and our ability to think and reason.

Based on the above analysis, we argue that a system should have the following capabilities to answer Image Riddles appropriately: i) the ability to *detect* and locate the objects, regions, and their properties; ii) the ability to recognize *actions*; iii) the ability to *infer* concepts from the detected words; and iv) the ability to rank a concept (described in words) based on its relative appropriateness; in other words, the ability to *reason* with and *process* background or commonsense knowledge about the semantic similarity and relations between words and phrases. These capabilities, in fact, are also desired of any automated system that aims to understand a scene and answer questions about it. For example, in VQA dataset [1], “Does this man have children”, “Is this a vegetarian Pizza?” are some such examples, where one needs explicit commonsense knowledge.

<sup>1</sup>Formally, an aspect is as significant as the specificity of the information it contains.

These riddles can be thought of as a visual counterpart to IQ test question types such as sequence filling ( $x_1, x_2, x_3, ?$ ) and analogy solving ( $x_1 : y_1 :: x_2 : ?$ )<sup>2</sup> where one needs to find commonalities between items. This task is different from traditional VQA, as in VQA the queries provide some clues regarding what to look for in the image in question. Most riddles in this task require both superior detection and reasoning capabilities, whereas a large percentage (of questions) of the traditional VQA dataset tests system’s detection capabilities. This task differs from both VQA and Captioning in that this task requires analysis of multiple images. While video analysis may require analysis of multiple images, this task of “image riddles” focuses on analysis of seemingly different images.

Hence, this task of Image Riddles is simple to explain; shares similarities with well-known and pre-defined types of IQ questions and it requires a combination of vision and reasoning capabilities. In this paper, we introduce a promising approach in tackling the problem.

In our approach, we first use state-of-the-art Image Classification techniques [21] to get the top identified class-labels from each image. Given these probabilistic detections, we use the knowledge of connections and relations of these words to infer a set of most probable words (or phrases). We use ConceptNet 5 [15] as the source of commonsense and background knowledge that encodes the relations between words and short phrases using a structured graph. Note, the possible range of candidates are the entire **vocabulary** of ConceptNet 5 (roughly 0.2 million). For representation and reasoning with this huge probabilistic knowledge we use the Probabilistic Soft Logic (PSL) [10, 2] framework<sup>3</sup>. Given the inferred words for each image, we then infer the final set of answers for each riddle.

Our **contributions** are threefold: i) we introduce the 3K Image Riddles Dataset; ii) we present a probabilistic reasoning approach to solve the riddles with reasonable accuracy; iii) our reasoning module inputs detected words (a closed set of class-labels) and *logically* infers all relevant concepts (belonging to a much larger vocabulary).

## 2. Related Work

The problem of Image Riddles has some similarities to the genre of topic modeling [3] and Zero-shot Learning [13]. However, this dataset imposes a few unique challenges: i) the possible set of target labels is the entire Natural Language vocabulary; ii) each image, when grouped with different set of images can map to a different label; iii) almost all the target labels in the dataset are unique (3k examples with 3k class-labels). These challenges make it hard

<sup>2</sup>Examples are: word analogy tasks (male : female :: king : ?); numeric sequence filling tasks: (1, 2, 3, 5, ?).

<sup>3</sup>PSL is shown to be a powerful framework for high-level Computer Vision tasks like Activity Detection [16].

to directly adopt topic model-based or Zero-shot learning-based approaches.

Our work is also related to the field of **Visual Question Answering**. Very recently, researchers spent a significant amount of efforts on both creating datasets and proposing new models [1, 18, 6, 17]. Interestingly both [1] and [6] adapted MS-COCO [14] images and created an open domain dataset with human generated questions and answers. Both [18] and [6] use recurrent networks to encode the sentence and output the answer.

Even though some questions from [1] and [6] are very challenging which actually require logical reasoning in order to answer correctly, popular approaches are still hoping to learn the direct signal-to-signal mapping from image and question to its answer, given a large enough annotated data. The necessity of common-sense reasoning could be easily neglected. Here we introduce the new Image Riddle problem which is 1) a well-defined cognitively challenging task that requires both vision and reasoning capability, 2) it is impossible to model the problem as direct signal-to-signal mapping, due to the data sparsity and 3) system’s performance could still be bench-marked automatically for comparison. All these qualities make our Image Riddle dataset a good testbed for vision and reasoning research.

## 3. Background

In this Section, we briefly introduce the different techniques and Knowledge Sources used in our system.

### 3.1. Probabilistic Soft Logic (PSL)

PSL is a recently proposed framework for Probabilistic Logic [10, 2]. A PSL model is defined using a set of weighted if-then rules in first-order logic.

Let  $C = (C_1, \dots, C_m)$  be such a collection where each  $C_j$  is a disjunction of literals, where each literal is a variable  $y_i$  or its negation  $\neg y_i$ , where  $y_i \in \mathcal{Y}$ . Let  $I_j^+$  (resp.  $I_j^-$ ) be the set of indices of the variables that are not negated (resp. negated) in  $C_j$ . Each  $C_j$  can be written as:

$$w_j : \wedge_{i \in I_j^-} y_i \rightarrow \vee_{i \in I_j^+} y_i \quad (1)$$

or equivalently,  $w_j : \vee_{i \in I_j^-} (\neg y_i) \vee \vee_{i \in I_j^+} y_i$ . Each rule  $C_j$  is associated with a non-negative weight  $w_j$ . PSL relaxes the boolean truth values of each ground atom  $a$  (constant term or predicate with all variables replaced by constants) to the the interval  $[0, 1]$ , denoted by  $I(a)$ . To compute soft truth values for logical formulas, Lukasiewicz relaxation [11] of conjunctions ( $\wedge$ ), disjunctions ( $\vee$ ) and negations ( $\neg$ ) is used :

$$\begin{aligned} I(l_1 \wedge l_2) &= \max\{0, I(l_1) + I(l_2) - 1\} \\ I(l_1 \vee l_2) &= \min\{1, I(l_1) + I(l_2)\} \\ I(\neg l_1) &= 1 - I(l_1) \end{aligned} \quad (2)$$

In PSL, the ground atoms are considered as random variables and the distribution is modeled using **Hinge-Loss**

**Markov Random Field**, which is defined as follows:

**Definition 3.1.** Let  $\mathbf{y}$  and  $\mathbf{x}$  be two vectors of  $n$  and  $n'$  random variables respectively, over the domain  $D = [0, 1]^{n+n'}$ . The feasible set  $\tilde{D}$  is a subset of  $D$ , defined as:

$$\tilde{D} = \{(\mathbf{y}, \mathbf{x}) \in D \mid \begin{array}{l} c_k(\mathbf{y}, \mathbf{x}) = 0, \forall k \in \mathcal{E} \\ c_k(\mathbf{y}, \mathbf{x}) \leq 0, \forall k \in \mathcal{I} \end{array}\}$$

where  $c = (c_1, \dots, c_r)$  are linear constraint functions associated with the index sets  $\mathcal{E}$  and  $\mathcal{I}$  denoting equality and inequality constraints. A *Hinge-Loss Markov Random Field*  $\mathbb{P}$  is a probability density, defined as: if  $(\mathbf{y}, \mathbf{x}) \notin \tilde{D}$ , then  $\mathbb{P}(\mathbf{y}|\mathbf{x}) = 0$ ; if  $(\mathbf{y}, \mathbf{x}) \in \tilde{D}$ , then:

$$\mathbb{P}(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{w}, \mathbf{x})} \exp(-f_{\mathbf{w}}(\mathbf{y}, \mathbf{x})) \quad (3)$$

where  $Z(\mathbf{w}, \mathbf{x}) = \int_{\mathbf{y} \mid (\mathbf{y}, \mathbf{x}) \in \tilde{D}} \exp(-f_{\mathbf{w}}(\mathbf{y}, \mathbf{x})) d\mathbf{y}$ .

The hinge-loss energy function  $f_{\mathbf{w}}$  is defined as:  $f_{\mathbf{w}}(\mathbf{y}, \mathbf{x}) = \sum_{j=1}^m w_j (\max\{l_j(\mathbf{y}, \mathbf{x}), 0\})^{p_j}$ , where  $w_j$ 's are non-negative free parameters and  $l_j(\mathbf{y}, \mathbf{x})$  are linear constraints over  $\mathbf{y}, \mathbf{x}$  and  $p_j = \{1, 2\}$ .

The final inference objective of HL-MRF is:

$$\mathbb{P}(\mathbf{y}|\mathbf{x}) \equiv \arg \min_{\mathbf{y} \in [0, 1]^n} \sum_{j=1}^m w_j (\max\{l_j(\mathbf{y}, \mathbf{x}), 0\})^{p_j} \quad (4)$$

In PSL, each logical rule  $C_j$  in the database  $\mathcal{C}$  is used to define  $l_j(\mathbf{y}, \mathbf{x})$  i.e. the linear constraints over  $(\mathbf{y}, \mathbf{x})$ . Given a set of weighted logical formulas, PSL builds a graphical model defining a probability distribution over the continuous space of values of the random variables in the model.

The final optimization problem is defined in terms of “distance to satisfaction”. For each rule  $C_j \in \mathcal{C}$  this distance to satisfaction is measured using the term  $w_j \times \max\{1 - \sum_{i \in I_j^+} y_i - \sum_{i \in I_j^-} (1 - y_i), 0\}$ . This encodes the penalty to the system if a rule is not satisfied. The final optimization problem becomes:

$$\arg \min_{\mathbf{y} \in [0, 1]^n} \sum_{C_j \in \mathcal{C}} w_j \max\left\{1 - \sum_{i \in I_j^+} y_i - \sum_{i \in I_j^-} (1 - y_i), 0\right\} \quad (5)$$

## 3.2. ConceptNet

ConceptNet [22], is a multilingual Knowledge Graph, that encodes commonsense knowledge about the world and is built primarily to assist systems that attempts to understand natural language text. The knowledge in ConceptNet is semi-curated. The nodes (called concepts) in the graph are words or short phrases written in natural language. The nodes are connected by edges (called assertions) which are labeled with meaningful relations (selected from a well-defined closed set of relation-labels). For example: (*reptile*, *IsA*, *animal*), (*reptile*, *HasProperty*, *cold blood*) are some edges. Each edge has an associated confidence score. Also, compared to other knowledge-bases like WordNet, YAGO,

NELL [23, 20]; ConceptNet has a more extensive coverage of English language words and phrases. These properties make this Knowledge Graph a perfect source for the required probabilistic commonsense knowledge.

## 3.3. Word2vec

Word2vec uses the theory of distributional semantics<sup>4</sup> to capture word meanings and produce word embeddings (vectors). The pre-trained word-embeddings have been successfully used in numerous Natural Language Processing applications and the induced vector-space is known to capture the graded similarities between words with reasonable accuracy [19]. Throughout the paper, for word2vec-based similarities, we use the 3 Million word-vectors trained on Google-News corpus [19].

## 4. Approach

Given a set of images (in our case four:  $\{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}$ ), the objective is to determine a set of ranked words ( $T$ ) based on how well the word semantically connects these image. In this work, we present an approach that uses Probabilistic Reasoning on top of a probabilistic Knowledge Base (ConceptNet). It also uses additional semantic knowledge of words from Word2vec. Using these knowledge sources, we predict the answers to the riddles.

### 4.1. Outline of our Framework

Algorithm 1. Solving Riddles

---

```

1: procedure UNRIDDLE( $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3, \mathcal{I}_4\}, \mathcal{K}_{cnet}$ )
2:   for  $\mathcal{I}_k \in \mathcal{I}$  do
3:      $\tilde{P}(S_k|\mathcal{I}_k) = \text{getClassLabelsNeuralNetwork}(\mathcal{I}_k)$ .
4:     for  $s \in S_k$  do
5:        $T_s, W_m(s, T_s) = \text{retrieveTargets}(s, \mathcal{K}_{cnet})$ ;  $\triangleright$ 
6:        $W_m(s, t_j) = \text{sim}(s, t_j) \forall t_j \in T_s$ 
7:       end for
8:        $T_k = \text{rankTopTargets}(\tilde{P}(S_k|\mathcal{I}_k), T_{S_k}, W_m)$ ;
9:        $I(\hat{T}_k) = \text{inferConfidenceStageI}(T_k, \tilde{P}(S_k|\mathcal{I}_k))$ .
10:      end for
11:       $I(T) = \text{inferConfidenceStageII}([\hat{T}_k]_{k=1}^4, [\tilde{P}(S_k|\mathcal{I}_k)]_{k=1}^4)$ .
12:    end procedure

```

---

As outlined in algorithm 1, for each image  $\mathcal{I}_k$  (here,  $k \in \{1, \dots, 4\}$ ), we follow three stages to infer related words and phrases: i) Image Classification: we get top class labels and the confidence from Image Classifier ( $S_k, \tilde{P}(S_k|\mathcal{I}_k)$ ), ii) Rank and Retrieve: using these labels and confidence scores, we rank and retrieve top related words from ConceptNet ( $\mathcal{K}_{cnet}$ ), iii) Probabilistic Reasoning and Inference (Stage I): using the labels ( $S_k$ ) and the top related words ( $T_k$ ), we design an inference model to logically infer final set of words ( $\hat{T}_k$ ) for each image. Lastly, we use another probabilistic reasoning model (Stage II) on the combined set of inferred words ( $targets$ ) from all images in a riddle.

<sup>4</sup>The central idea is: “a word is known by the company it keeps”.

This model assigns the final confidence scores on the combined set of targets ( $T$ ). The pipeline followed for each image is depicted with an example in Figure 2.

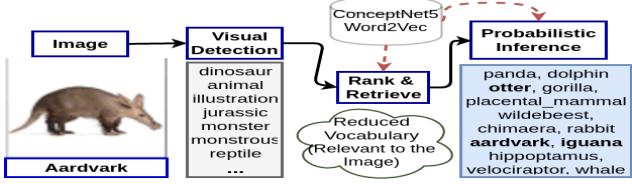


Figure 2. An overview of the framework followed for each Image; demonstrated using an example image of an aardvark (resembles animals such as tapir, ant-eater). We run a similar pipeline for each image and then infer final results using a final Probabilistic Inference Stage (Stage II).

## 4.2. Image Classification

Neural Networks trained on ample source of images and numerous image classes has been very effective. Studies have found that convolutional neural networks (CNN) can produce near human level image classification accuracy [12], and related work has been used in various visual recognition tasks such as scene labeling [5] and object recognition [7]. To exploit these advances, we use the state-of-the-art class detections provided by the Clarifai API [21] and the Deep Residual Network Architecture by [8] (using the trained ResNet-200 model). For each image ( $\mathcal{I}_k$ ) we use top 20 detections ( $S_k$ ). Let us call these detections as *seeds*. An example is provided in the Figure 2. Each detection is accompanied with the classifier’s confidence score ( $\tilde{P}(S_k|\mathcal{I}_k)$ ).

## 4.3. Rank and Retrieve Related Words

Our goal is to logically infer words or phrases that represent (higher or lower-level) concepts that can best explain the co-existence of the *seeds* in a scene. Say, for “hand” and “care”, implied words could be “massage”, “ill”, “ache” etc. For “transportation” and “sit”, implied words/phrases could be “sit in bus”, “sit in plane” etc. The reader might be inclined to infer other concepts. However, to “infer” is to derive “logical” conclusions. Hence, we prefer the concepts which shares strong explainable connections with the seed-words.

A logical choice would be traversing a knowledge-graph like ConceptNet and find the common reachable nodes from these *seeds*. As this is computationally quite infeasible, we use the association-space matrix representation of ConceptNet, where the words are represented as vectors. The similarity between two words approximately embodies the strength of the connection over all paths connecting the two words in the graph. We get the top similar words for each *seed*, approximating the reachable nodes.

### 4.3.1 Retrieve Related Words For a Seed

**Visual Similarity:** We observe that, for objects, the ConceptNet-similarity gives a poor result (See Table 1). So, we define a metric called **visual similarity**. Let us call the similar words as *targets*. In this metric, we represent the seed and the target as vectors. To define the dimensions, for each *seed*, we use a set of relations (HasA, HasProperty, PartOf and MemberOf). We query ConceptNet to get the related words (say,  $W_1, W_2, W_3, \dots$ ) under such relations for the seed-word and its superclasses. Each of these relation-word pairs (i.e.  $HasA-W_1, HasA-W_2, PartOf-W_3, \dots$ ) becomes a separate dimension. The values for the seed-vector are the weights assigned to the assertions. For each *target*, we query ConceptNet and populate the target-vector using the edge-weights for the dimensions defined by the seed-vector.

To get the top words using visual similarity, we use the cosine similarity of the seed-vector and the target-vector to re-rank the top 10000 retrieved similar target-words using ConceptNet-similarity. For abstract seed-words, we do not get any such relations and we use the ConceptNet similarity directly. Table 1 shows the top similar words using

ConceptNet	Visual Similarity	word2vec
man, merby, misandrous, philandry, male\_human, dirty\_pig, manthyose, date\_woman,guyliner,manslut	priest, uncle, guy, geezer, bloke, pope, bouncer, ecologist, cupid, fella	women, men, males, mens, boys, man, female, teenagers,girls,ladies

Table 1. Top 10 similar Words for “Men”. More in appendix.

ConceptNet, word2vec and visual-similarity for the word “men”. Moreover, the ranked list based on visual-similarity ranks *boy*, *chap*, *husband*, *godfather*, *male\_person*, *male* in the ranks 16 to 22.

**Formulation:** For each seed ( $s$ ), we get the top words ( $T_s$ ) from ConceptNet using the visual similarity metric and the similarity vector  $W_m(s, T_s)$ . Together for an image, these constitute  $T_{S_k}$  and the matrix  $W_m$ , where  $W_m(s_i, t_j) = sim_{vis}(s_i, t_j) \forall s_i \in S_k, t_j \in T_{S_k}$ . Next we describe the defined similarity metric.

A large percentage of the error in Image Classifiers are due to visually similar (or semantically similar) objects or objects from the same category [9]. In such cases, we use this visual similarity metric to predict the possible visually similar objects and then use an inference model to infer the actual object.

### 4.3.2 Rank Targets

We use  $\tilde{P}(S_k|\mathcal{I}_k)$  as an approximate vector representation for the image, in which the seed-words are the dimensions. The columns of  $W_m$  provides vector representations for the target words ( $t \in T_{S_k}$ ) in the space. We calculate cosine similarities for each target with such a image-vector and then re-rank the targets. We consider the top  $\theta_{\#t}$  targets and we call it  $T_k$ .

## 4.4. Probabilistic Reasoning and Inference

### 4.4.1 PSL Inference Stage I

Given a set of candidate *targets*  $\mathbf{T}_k$  and a set of weighted seeds ( $\mathbf{S}_k, \tilde{P}(\mathbf{S}_k | \mathcal{I}_k)$ ), we build an inference model to infer a set of most probable *targets* ( $\hat{\mathbf{T}}_k$ ). We model the joint distribution using PSL as this formalism adopts Markov Random Field which obeys the properties of Gibbs Distribution. In addition, a PSL model is declared using rules. Given the final answer from the system, the set of satisfied (grounded) rules show the logical connections between the detected words and the final answer, which demonstrates the system's explainability.

The PSL model can be best depicted as an Undirected Graphical Model involving *seeds* and *targets*, as given in Figure 3.

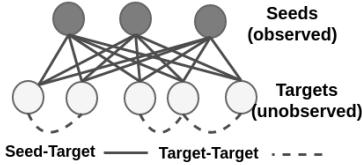


Figure 3. Joint Modeling of seeds and targets, depicted as a Undirected Graphical Model. We define the seed-target and target-target potentials using PSL rules. We connect each seed to each target and the potential depends on their similarity and the target's popularity bias. We connect each target to  $\theta_{t-t}$  (1 or 2) maximally similar targets. The potential depends on their similarity.

**Formulation:** Using PSL, we add two sets of rules: i) to define seed-target potentials, we add rules of the form  $wt_{ij} : s_{ik} \rightarrow t_{jk}$  for each word  $s_{ik} \in \mathbf{S}_k$  and target  $t_{jk} \in \mathbf{T}_k$ ; ii) to define target-target potentials, for each target  $t_{jk}$ , we take the most similar  $\theta_{t-t}$  targets ( $T_j^{max}$ ). For each target  $t_{jk}$  and each  $t_{mk} \in T_j^{max}$ , we add two rules  $wt_{jm} : t_{jk} \rightarrow t_{mk}$  and  $wt_{jm} : t_{mk} \rightarrow t_{jk}$ . Next, we describe the choices in detail.

i) From the perspective of optimization, the rule  $wt_{ij} : s_{ik} \rightarrow t_{jk}$  adds the term  $wt_{ij} * max\{I(s_{ik}) - I(t_{jk}), 0\}$  to the objective. This means that if confidence score of the target  $t_{jk}$  is not greater than  $I(s_{ik})$  (i.e.  $\tilde{P}(\mathbf{S}_k | \mathcal{I}_k)$ ), then the rule is not satisfied and we penalize the model by  $wt_{ij}$  times the difference between the confidence scores. We add the above rule for seeds and targets for which the combined weighted similarity exceeds certain threshold  $\theta_{sim, psl1}$ .

We encode the commonsense knowledge of words and phrases obtained from different knowledge sources into the weights of these rules  $wt_{ij}$ . Both the knowledge sources are considered because ConceptNet embodies commonsense knowledge and word2vec encodes word-meanings. It is also important that the inference model is not biased towards more popular targets (i.e. abstract words or words too commonly used/detected in corpus). We compute eigenvector centrality score ( $\mathbb{C}(.)$ ) for each word in the context of ConceptNet (a network of words and phrases). Higher  $\mathbb{C}(.)$

indicates higher connectivity of a word in the graph. This yields a higher similarity score to many words and might give an unfair bias to this *target* in the inference model. Hence, the higher the  $\mathbb{C}(.)$ , the word provides less specific information for an image. Hence, the weight becomes

$$wt_{ij} = \theta_{\alpha_1} * sim_{cn}(s_{ik}, t_{jk}) + \theta_{\alpha_2} * sim_{w2v}(s_{ik}, t_{jk}) + 1/\mathbb{C}(t_{jk}), \quad (6)$$

where  $sim_{cn}(., .)$  is the normalized ConceptNet-based similarity.  $sim_{w2v}(., .)$  is the normalized word2vec similarity of two words and  $\mathbb{C}(.)$  is the eigenvector-centrality score of the argument in the ConceptNet matrix.

ii) To model dependencies among the targets, we observe that if two concepts  $t_1$  and  $t_2$  are very similar in meaning, then a system that infer  $t_1$  should infer  $t_2$  too, given the same set of observed words. Therefore, the two rules  $wt_{jm} : t_{jk} \rightarrow t_{mk}$  and  $wt_{jm} : t_{mk} \rightarrow t_{jk}$  are designed to force the confidence values of  $t_{jk}$  and  $t_{mk}$  to be as close to each other as possible.  $wt_{jm}$  is the same as Equation 6 without the penalty for popularity.

The combined PSL model inference objective becomes:

$$\begin{aligned} \arg \min_{I(\mathbf{T}_k) \in [0,1]^{|T_k|}} & \sum_{s_{ik} \in \mathbf{S}_k} \sum_{t_{jk} \in \mathbf{T}_k} wt_{ij} \max\{I(s_{ik}) - I(t_{jk}), 0\} + \\ & \sum_{t_{jk} \in \mathbf{T}_k} \sum_{t_{mk} \in T_j^{max}} wt_{jm} \left\{ \max\{I(t_{mk}) - I(t_{jk}), 0\} + \right. \\ & \left. \max\{I(t_{jk}) - I(t_{mk}), 0\} \right\}. \end{aligned}$$

To let the targets compete against each other, we add a constraint on the sum of the confidence scores of the targets i.e.  $\sum_{j:t_{jk} \in \mathbf{T}_k} I(t_{jk}) \leq \theta_{sum1}$ . Here  $\theta_{sum1} \in \{1, 2\}$  and  $I(t_{jk}) \in [0, 1]$ . As a result of this model, we get an inferred reduced set of targets  $[\hat{\mathbf{T}}_k]_{k=1}^4$ .

### 4.4.2 PSL Inference Stage II

To learn the most probable set of common targets jointly, we consider the *targets* and the *seeds* from all images together. Assume that the *seeds* and the *targets* are nodes in a knowledge-graph. Then, the most appropriate target-nodes should observe similar properties as an appropriate answer to the riddle: i) a target-node should be connected to the high-weight seeds in an image i.e. should relate to the important aspects of the image; ii) a target-node should be connected to seeds from all images.

**Formulation:** Here, we use the rules  $wt_{ij} : s_{ik} \rightarrow t_{jk}$  for each word  $s_{ik} \in \mathbf{S}_k$  and target  $t_{jk} \in \hat{\mathbf{T}}_k$  for all  $k \in \{1, 2, \dots, 4\}$ . To let the set of targets compete against each other, we add the constraint  $\sum_{k=1}^4 \sum_{j:t_{jk} \in \hat{\mathbf{T}}_k} I(t_{jk}) \leq \theta_{sum2}$ . Here  $\theta_{sum2} = 1$  and  $I(t_{jk}) \in [0, 1]$ .

To minimize the penalty for each rule, the optimal solution will try to maximize the confidence score of  $t_{jk}$ . To

minimize the overall penalty, it should maximize the confidence scores of those targets which will satisfy most of the rules (or rules with maximum total weight). As the summation of confidence scores is bounded, only a few top inferred targets should have non-zero confidence.

## 5. Experiments and Results

In this section, we provide the results of the validation experiments of the newly introduced Image Riddle dataset, followed by empirical evaluation of the proposed approach against vision-only baselines.

### 5.1. Dataset Validation and Analysis

We have collected a set of 3333 riddles from the internet (puzzle websites). Each riddle has 4 images ( $66 \times 66$ , 6KB in size) and a groundtruth label associated with it. To verify the groundtruth answers, we define the metrics: i) “correctness” - how correct and appropriate the answers are, and ii) “difficulty” - how difficult are the riddles. We conduct an Amazon Mechanical Turker-based evaluation. We ask them to rate the correctness from 1-6<sup>5</sup>. The “difficulty” is rated from 1-7<sup>6</sup>. According to the Turkers, the mean correctness rating is 4.4 (with Standard Deviation 1.5). The “difficulty” ratings show the following distribution: toddler (0.27%), younger child (8.96%), older child (30.3%), teenager (36.7%), adult (19%), linguist (3.6%), no-one (0.64%). In short, the average age to answer the riddles seems to be closer to **13-17 yrs**. Also, few of these (4.2%) riddles seem to be incredibly hard. Interestingly, the average age perceived reported for the recently proposed VQA dataset [1] is **8.92 yrs**. Although, this experiment measures “the turkers’ perception of the required age”, one can conclude that the riddles are comparably harder.

### 5.2. System Evaluation

The presented approach suggests the following hypotheses that requires empirical tests: I) the proposed approach (and their variants) attain reasonable accuracy in solving the riddles; II) the individual stages of the framework improves the final inference accuracy of the answers. In addition, we also experiment to observe the effect of using commercial classification methods like Clarifai against a published state-of-the-art Image Classification method.

<sup>5</sup>1: Completely gibberish, incorrect, 2: relates to one image, 3 and 4: connects two and three images respectively, 5: connects all 4 images, but could be a better answer, 6: connects all images and an appropriate answer.

<sup>6</sup>These gradings are adopted from VQA AMT instructions [1]. 1: A toddler can solve it (ages:3-4), 2: A younger child can solve it (ages:5-8), 3: A older child can solve it (ages:9-12), 4: A teenager can solve it (ages:13-17), 5: An adult can solve it (ages:18+), 6: Only a Linguist (one who has above-average knowledge about English words and the language in general) can solve it, 7: No-one can solve it.

#### 5.2.1 Systems

We propose several variations of the proposed approach and compare them with a simple vision-only baseline (hypothesis I). We introduce an additional Bias-Correction stage after the Image Classification, which aims to re-weight the detected seeds using additional information from other images. The variations then, are created to test the effects of varying the Bias-Correction stage and the effects of the individual stages of the framework on the final accuracy (hypothesis II). We also vary the initial Image Classification Method (Clarifai, Deep Residual Network).

**Bias-Correction:** We experimented with two variations:

i) greedy bias-correction and ii) no bias-correction. We follow the intuition that the re-weighting of the seeds of one image can be influenced by the others<sup>7</sup>. To this end, we develop the “GreedyUnRiddler” (**GUR**) approach. In this approach, we consider all of the images together to dictate the new weight of each seed. Take image  $\mathcal{I}_k$  for example. To re-weight seeds in  $S_k$ , we calculate the weights using the following equation:  $\tilde{W}(s_k) = \frac{\sum_{j \in 1 \dots 4} \text{sim}_{\text{cosine}}(V_{s_k,j}, V_j)}{4.0}$ .  $V_j$  is vector of the weights assigned  $\tilde{P}(S_j | \mathcal{I}_j)$  i.e. confidence scores of each seed in the image. Each element of  $V_{s_k,j}[i]$  is the ConceptNet-similarity score between the seed  $s_k$  and  $s_{i,j}$  i.e. the  $i^{th}$  seed of the  $j^{th}$  image. The re-weighted seeds ( $S_k, \tilde{W}(S_k)$ ) of an image are then passed through the rest of the pipeline to infer the final answers.

In the original pipeline (“UnRiddler”, in short **UR**), we just normalize the weights of the seeds and pass on to the next stage. We experiment with another variation (called BiasedUnRiddler or **BUR**), the results of which are included in appendix, as **GUR** achieves the best results.

**Effect of Stages:** We observe the accuracy after each stage in the pipeline (**VB**: Upto Bias Correction, **RR**: Upto Rank and Retrieve stage, **All**: The entire Pipeline). For **VB**, we use the normalized weighted seeds, get the weighted centroid vector over the word2vec embeddings of the seeds for each image. Then we obtain the mean vector over these centroids. The top similar words from the word2vec vocabulary to this mean vector, constitutes the final answers. For **RR**, we get the mean vector over the top predicted targets for all images. Again, the most similar words from the word2vec vocabulary constitutes the answers.

**Baseline:** We create Vision-only Baselines. We directly use the class-labels and the confidence scores predicted using a Neural Network-based Classifier. For each image, we calculate the weighted centroid of the word2vec embeddings of these labels and the mean of these centroids for the 4 images. For the automatic evaluation we use this centroid and for the human evaluation, we use the most similar word to this vector, from the word2vec vocabulary. The Baseline

<sup>7</sup>A person would often skim through all the images at one go and will try to come up with the aspects that needs more attention.

performances are listed in Table 2 in the **VB+UR** cells.

### 5.2.2 Experiment I: Automatic Evaluation

We evaluate the performance of the proposed approach on the 3333 Image Riddles dataset using both automatic and Amazon Mechanical Turk (AMT)-based evaluations.

As an evaluation metric, we use word2vec similarity measure. An answer to a riddle may have several semantically similar answers. Hence it is reasonable to use such a metric. For each riddle, we calculate the maximum similarity between the groundtruth and top 10 detections from an approach. To calculate phrase similarities, we use `n_similarity` method of the `gensim.models.word2vec` package. The average of such maximum similarities is reported in percentage form.

	GUR		UR		
	3.3k	2.8k	3.3k	2.8k	
Clarifai	VB	65.3	65.36	65	65.3 <sup>†</sup>
	RR	65.9	65.73	65.9	65.7
	All	<b>68.8*</b>	<b>68.7</b>	68.5	<b>68.57</b>
ResNet	VB	66.8	66.4	68.3	68 <sup>†</sup>
	RR	66.3	66.2	67	66.7
	All	68.2	68.2	<b>68.53</b>	68.2

Table 2. Accuracy on the Image Riddle Dataset. Pipeline variants (VB, RR and All) are combined with Bias-Correction stage variants (GUR, UR). All values are in percentage form. (\*- Best, † - Baselines).

$\theta_{\#t}$	Number of Targets	2500
$\theta_{\alpha_1}$	ConceptNet-similarity Weight	1
$\theta_{\alpha_2}$	word2vec-similarity weight	4
$\theta_{t-t}$	Number of maximum similar Targets	1
$\theta_{sim,psl1}$	Seed-target similarity Threshold	0.8
$\theta_{sum1}$	Sum of confidence scores in Stage I	2

Table 3. A List of parameters  $\theta$  used in the approach

To select the parameters in the parameter vector  $\theta$ , We employed a random search on the parameter-space over first 500 riddles over 500 combinations. The final set of parameters used and their values are tabulated in Table 3.

Each of the stage-variants (VB, RR and All) are combined with different variations of the Bias-Correction stage (GUR and UR respectively). The accuracies on all are listed in Table 2. We provide our experimental results on this 3333 riddles and 2833 riddles (barring 500 riddles we used for the parameter search).

### 5.2.3 Experiment II: Human Evaluation

We conduct an AMT-based comparative evaluation of the results of the proposed approach (GUR+All using Clarifai) and two vision-only baselines. We define two metrics: i) “correctness” and ii) “intelligence”. Turkers are presented

with a scenario: *We have three separate robots that attempted to answer this riddle. You have to rate the answer based on the correctness and the degree of intelligence (explainability) shown through the answer..* The correctness is defined as before. In addition, turkers are asked to rate intelligence in a scale of 1-4<sup>8</sup>. We plot the the percentage of total riddles per each value of correctness and intelligence in Figure 4. In these histograms plots, we expect a increase in the rightmost buckets for the more “correct” and “intelligent” systems.

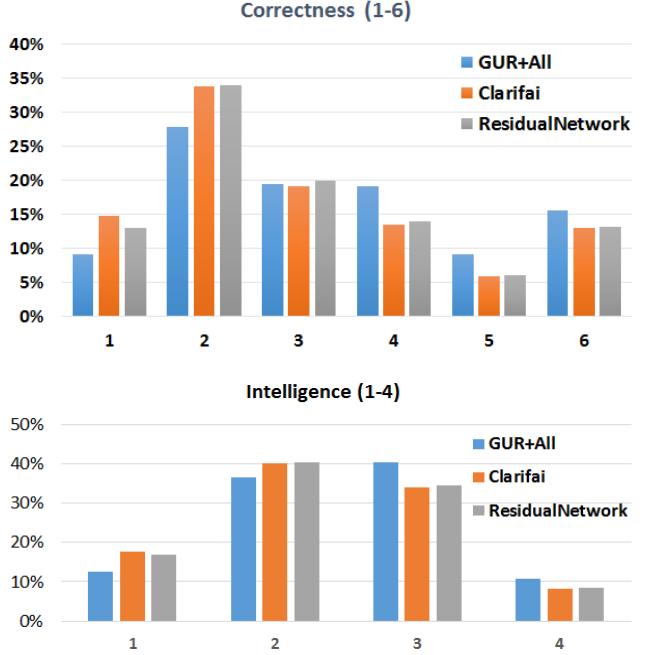


Figure 4. AMT Results of The GUR+All (our), Clarifai (baseline 1) and ResidualNet (baseline 2) approaches. Correctness Means are:  $2.6 \pm 1.4$ ,  $2.4 \pm 1.45$ ,  $2.3 \pm 1.4$ . For Intelligence:  $2.2 \pm 0.87$ ,  $2 \pm 0.87$ ,  $1.8 \pm 0.8$

### 5.2.4 Analysis

Experiment I shows that the GUR variant (**GUR+All** in Table 2) achieves the best results in terms of word2vec-based accuracy. Similar trend is reflected in the AMT-based evaluations (Figure 4). Our system has increased the percentage of puzzles for the rightmost bins i.e. produces more “correct” and “intelligent” answers for more number of puzzles. The word2vec-based accuracy puts the performance of ResNet baseline close to that of the GUR variant. However, as evident from Figure 4, the AMT evaluation of the correctness shows clearly that the ResNet baseline lags in predicting meaningful answers. Experiment II also includes what the turkers think about the intelligence of the systems that tried to solve the puzzles. This also puts the GUR variant at the top. The above two experiments empirically show

<sup>8</sup>1: Not intelligent, 2: Moderately Intelligent, 3: Intelligent, 4: Very Intelligent.

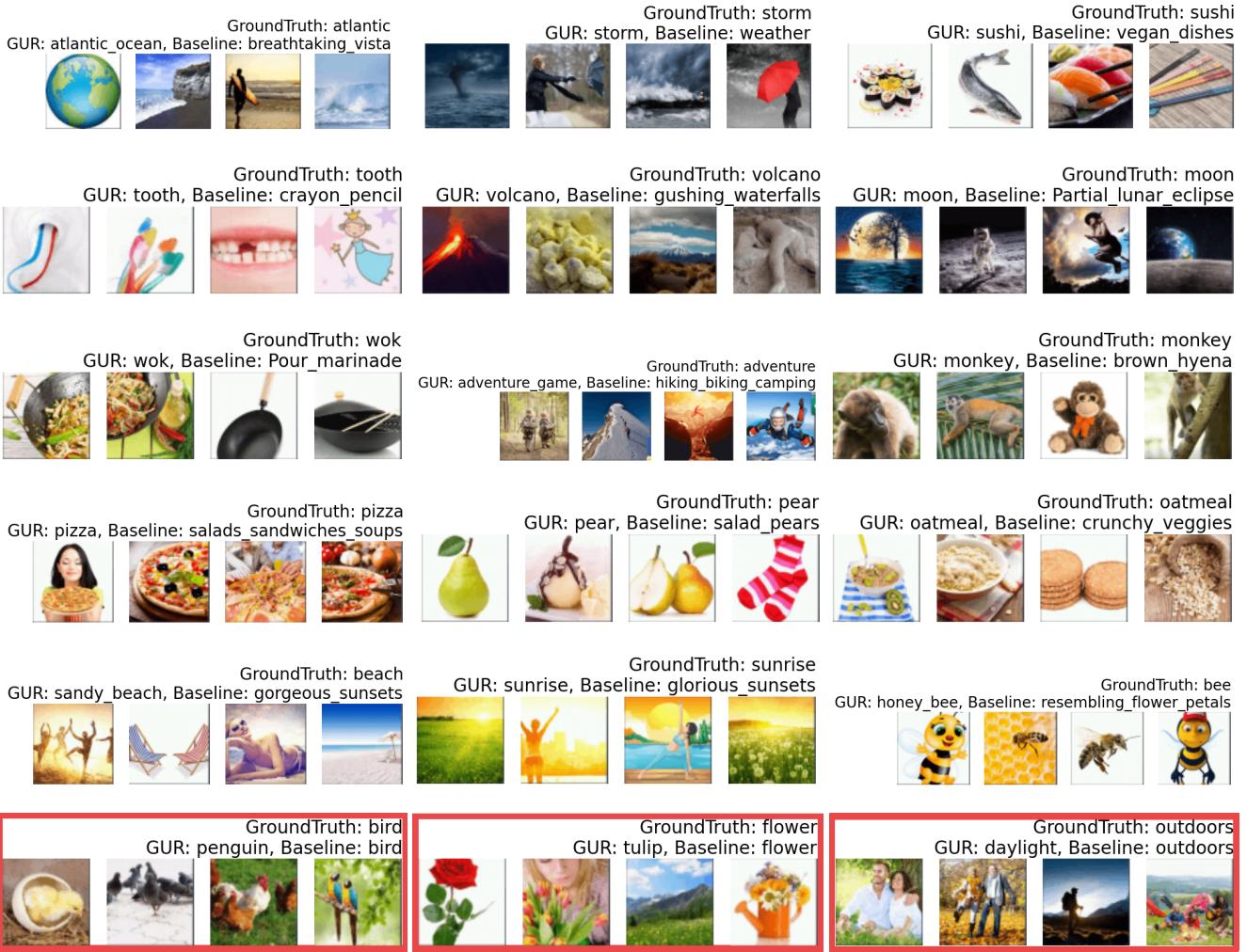


Figure 5. Positive and Negative (in red) results of the “GUR” approach (**GUR+All** variant) on some of the riddles. The groundtruth labels, closest label among top 10 from GUR and the Clarifai baseline are provided for all images. For more results, check Appendix and the ImageRiddle website (<http://bit.ly/1Rj4tFc>).

that our approach achieves a reasonable accuracy in solving the riddles (Hypothesis I). In table 2, we observe how the accuracy varies after each stage of the pipeline (hypothesis II). The table shows a jump in the accuracy after the RR stage, which leads us to believe the primary improvement of our approach is attributed to the Probabilistic Reasoning model. We also provide our detailed results for the “GUR” approach using a few riddles in Figure 5.

## 6. Conclusion and Future Works

In this work, we presented a Probabilistic Reasoning based approach to solve a new class of image puzzles, called “Image Riddles”. We have collected over 3k such riddles. Crowd-sourced evaluation of the dataset demonstrates the validity of the annotations and the nature of the difficulty of the riddles. We empirically show that our approach

improves on vision-only baselines and provides a stronger baseline for future attempts.

The task of “Image Riddles” is equivalent to conventional IQ test questions such as analogy solving, sequence filling; which are often used to test human intelligence. This task of “Image Riddles” is also in line with the current trend of VQA datasets which require visual recognition and reasoning capabilities. However, it focuses more on the combination of both vision and reasoning capabilities. In addition to the task, the proposed approach introduces a novel inference model to infer related words (from a large vocabulary) given class labels (from a smaller set), using semantic knowledge of words. This method is general in terms of its applications. Systems such as [24], which use a collection of high-level concepts to boost VQA performance; can benefit from this approach.

## References

- [1] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *International Conference on Computer Vision (ICCV)*, 2015. 1, 2, 6
- [2] S. Bach, B. Huang, B. London, and L. Getoor. Hinge-loss markov random fields: Convex inference for structured prediction. *arXiv preprint arXiv:1309.6813*, 2013. 2
- [3] D. M. Blei. Probabilistic topic models. *Commun. ACM*, 55(4):77–84, Apr. 2012. 2
- [4] M. Brysbaert, A. B. Warriner, and V. Kuperman. Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior research methods*, 46(3):904–911, 2014. 10
- [5] C. Farabet, C. Couprie, L. Najman, and Y. LeCun. Learning hierarchical features for scene labeling. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(8):1915–1929, 2013. 4
- [6] H. Gao, J. Mao, J. Zhou, Z. Huang, L. Wang, and W. Xu. Are you talking to a machine? dataset and methods for multilingual image question answering. *arXiv preprint arXiv:1505.05612*, 2015. 2
- [7] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 580–587. IEEE, 2014. 4
- [8] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *arXiv preprint arXiv:1512.03385*, 2015. 4
- [9] D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In *European conference on computer vision*, pages 340–353. Springer, 2012. 4
- [10] A. Kimmig, S. Bach, M. Broeckeler, B. Huang, and L. Getoor. A short introduction to probabilistic soft logic. In *Proceedings of the NIPS Workshop on Probabilistic Programming: Foundations and Applications*, pages 1–4, 2012. 2
- [11] G. Klir and B. Yuan. Fuzzy sets and fuzzy logic: theory and applications. 1995. 2
- [12] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 4
- [13] H. Larochelle, D. Erhan, Y. Bengio, U. D. Montral, and M. Qubec. Zero-data learning of new tasks. In *In AAAI*, 2008. 2
- [14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014*, pages 740–755. Springer, 2014. 2
- [15] H. Liu and P. Singh. Conceptnet - a practical commonsense reasoning tool-kit. *BT Technology Journal*, 22(4):211–226, Oct. 2004. 2
- [16] B. London, S. Khamis, S. Bach, B. Huang, L. Getoor, and L. Davis. Collective activity detection using hinge-loss markov random fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 566–571, 2013. 2
- [17] L. Ma, Z. Lu, and H. Li. Learning to answer questions from image using convolutional neural network. *arXiv preprint arXiv:1506.00333*, 2015. 2
- [18] M. Malinowski, M. Rohrbach, and M. Fritz. Ask your neurons: A neural-based approach to answering questions about images. *arXiv preprint arXiv:1505.01121*, 2015. 2
- [19] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. 3
- [20] T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves, and J. Welling. Never-ending learning. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, 2015. 3
- [21] G. Sood. *clarifai: R Client for the Clarifai API*, 2015. R package version 0.2. 2, 4
- [22] R. Speer and C. Havasi. Representing general relational knowledge in conceptnet 5. 2012. 3
- [23] F. M. Suchanek, G. Kasneci, and G. Weikum. Yago: A core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web, WWW '07*, pages 697–706, New York, NY, USA, 2007. ACM. 3
- [24] Q. Wu, C. Shen, L. Liu, A. Dick, and A. v. d. Hengel. What value do explicit high level concepts have in vision to language problems? *arXiv preprint arXiv:1506.01144*, 2015. 8

## Appendices

### A. BiasedUnRiddler (BUR): A Variation of the BiasCorrection Stage

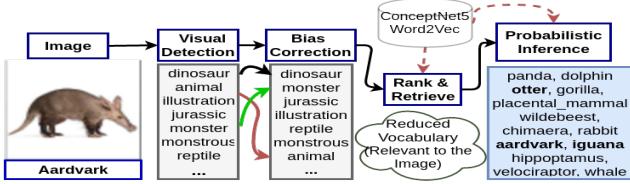


Figure 6. Clarifai detections and results from different stages for the aardvark image (for **BUR** variant).

In Figure 6: *dinosaur*, *animal* and *reptile* all provide evidence that the image has an animal. Only the word *dinosaur* indicates what kind of animal is in the image. The other words do not add any additional information. Some high-confidence detections also provide erroneous abstract information. Here, the labels *monstrous*, *monster* are some such detections. Hence, the objective is to re-weight the seeds so that: i) the more specific seed-words should have higher weight than the ones which provide *similar* but more general information; ii) the seeds that are too frequently used or detected in corpus, should be given lower weights.

**Specificity and Popularity:** We compute eigenvector centrality score (*ECS*) for each word in the context of ConceptNet. Higher *ECS* indicates higher connectivity and yields a higher similarity score to many words and might give an unfair bias to this *seed* (and words implied by this *seed*) in the inference model. Hence, the higher the *ECS*, the word provides less specific information for an image. Additionally, we use the concreteness rating (*CR*) from [4]. In this paper, the top 39955 frequent English words are rated from the scale of 1 (very abstract) to 5 (very concrete). For example, the mean ratings for *monster*, *animal* and *dinosaur* are 3.72, 4.61 and 4.87 respectively.

**Problem Formulation:** We formulate the problem as a resource flow problem on a graph. The directed graph  $G$  is constructed in the following way: we order the *seeds* based on decreasing centrality scores (*CS*). We compute *CS* as:

$$CS = (ECS + (-CR))/2, \quad (7)$$

where we normalize *ECS* and  $-CR$  to the scale of 0 to 1. For each seed  $u$ , we check the immediate next node  $v$  and add an edge  $(u, v)$  if the (ConceptNet-based) similarity between  $u$  and  $v$  is greater than  $\theta_{sim,ss}$ <sup>9</sup>. If in this iteration, a node  $v$  is not added in  $G$ , we get the most recent predecessor  $u$  for which the similarity exceed  $\theta_{sim,ss}$  and add  $(u, v)$ . The idea is that if a word  $u$  is more abstract than  $v$  and if they are quite similar in terms of conceptual similarity, then

<sup>9</sup> $\theta$  denotes the set of parameters used in the model.

word  $v$  provides similar but more specific information than word  $u$ . Each node has a resource  $\tilde{P}(u|\mathcal{I}_k)$ , the confidence assigned by the Neural Network. If there is an edge from the node, some of this resource should be sent along this edge until for all edges  $(u, v) \in G$ ,  $w_v$  becomes greater than  $w_u$ . We formulate the problem as a Linear Optimization problem:

$$\begin{aligned} & \text{minimize}_{w=(w_1, \dots, w_{|S_k|})} \sum_{(u,v) \in G} \max\{w_u - w_v, 0\} \\ & \text{subject to} \quad \sum_{s \in S_k} w_s = \sum_{s_k \in S_k} \tilde{P}(s_k|\mathcal{I}_k) \\ & \quad w_u = \tilde{P}(u|\mathcal{I}_k), u \notin G \\ & \quad w_u \geq 0.5 \tilde{P}(u|\mathcal{I}_k), \forall u \in G \end{aligned}$$

To limit the resource a node  $u$  can send, we limit the final minimum value by  $0.5 \tilde{P}(u|\mathcal{I}_k)$ . The solution provides us with the necessary weights for the set of seeds  $S_k$  in  $\mathcal{I}_k$ . We normalize these weights and get  $\tilde{W}(S_k)$ .

### B. Intermediate Results for the “Aardvark” Riddle

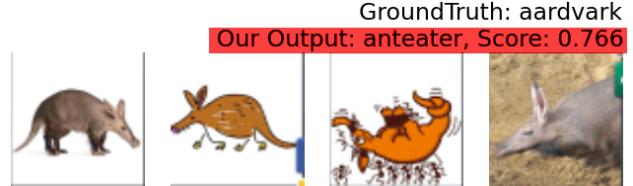


Figure 7. The four different Images for the “aardvark” riddle.

From the four figures in Figure 7, we get the top 20 Clarifai detections as given in the Table 4.

Based on the GUR approach (**GUR+All** in paper), our PSL Stage I outputs probable concepts (words or phrases) depending on the initial set of detected class-labels (*seeds*). They are provided in Table 5. Note that, these are the top *targets* detected from almost 0.2 million possible candidates. Observe the following:

- i) the highlighted detected animals have a few visual features in common, such as *four short legs*, *a visible tail*, *short height* etc.
- ii) the detections from the third image does not at all lead us to an animal and the PSL Stage I still thinks that its a cartoon of sort.
- iii) the detections from second gets affected because of its close relation to the detections from third image and it infers that the image just depicts cartoon.

In the final PSL Stage II however, the model figures out that there is an animal that is common to all these images. This is mainly because *seeds* from the three images *confidently* predict that some animal is present in the images.

Image1	Image2	Image3	Image4
monster	food	fun	rock
jurassic	small	retro	<b>nobody</b>
monstrous	vector	clip	travel
primitive	dinosaur	<b>halloween</b>	<b>water</b>
lizard	wildlife	<b>set</b>	<b>sea</b>
paleontology	cartoon	border	<b>aquatic</b>
vertebrate	nature	messy	outdoors
dinosaur	<b>evolution</b>	ink	<b>sand</b>
creature	reptile	design	<b>beach</b>
wildlife	outline	ornate	bird
nature	cute	decoration	wildlife
<b>evolution</b>	sketch	<b>ornament</b>	<b>biology</b>
reptile	painting	vector	<b>zoology</b>
<b>wild</b>	silhouette	<b>contour</b>	carnivora
horizontal	horizontal	cartoon	nature
illustration	art	cute	horizontal
animal	illustration	silhouette	animal
side view	graphic	art	side view
panoramic	animal	illustration	panoramic
mammal	panoramic	graphic	mammal

Table 4. Top 20 detections from Califai API. The detections that are completely noisy is colored using red. It can be observed that the third image does not give any evidence of an animal present.

Image1	Image2	Image3	Image4
dolphin	graph.toughness	decorative	bison
rhinoceros	cartography	graph.toughness	american_bison
<b>komodo.dragon</b>	color_paint	graph	<b>marsupial</b>
african_elephant	graph	artwork	gibbon
<b>lizard</b>	spectrograph	spectrograph	monotreme
gorilla	revue	kesho_mawashi	moose
crocodile	linear_functional	tapestry	mole
indian_elephant	simulacrum	map	<b>wildebeest</b>
<b>wildebeest</b>	pen_and_ink	arabesque	<b>echidna</b>
elephant	luck_of_draw	sgraffito	turtle
<b>echidna</b>	<b>cartoon</b>	linear_functional	mule_deer
chimaera	camera_lucida	hamiltonian_graph	mongoose
chimpanzee	explode_view	emblazon	tamarin
liger	micrographics	pretty_as_picture	chimpanzee
<b>gecko</b>	hamiltonian_graph	art_deco	wolverine
rabbit	crowd_art	dazzle_camo	prairie_dog
<b>iguana</b>	<b>depiction</b>	ecce_homo	western_gorilla
hippopotamus	echocardiogram	pointillist	<b>anteater</b>
mountain_goat	scenography	pyrography	okapi
loch_ness_monster	linear_perspective	echocardiogram	skunk

Table 5. Top 20 detections per each image from PSL Stage I (GUR).

That is why most of the top detections correspond to animals and animals having certain characteristics in common.

The top detections from PSL Stage II (GUR) are: *monotreme*, *gecko*, *hippopotamus*, *pyrography*, *anteater*, *lizard*, *mule\_deer*, *chimaera*, *liger*, *iguana*, *komodo.dragon*, *echidna*, *turtle*, *art\_deco*, *sgraffito*, *gorilla*, *loch\_ness\_monster*, *prairie\_dog*.

**BUR:** For BUR, PSL Stage I outputs probable concepts (words or phrases) depending on the current set of *seeds*.

Image1	Image2	Image3	Image4
panda	like_paint	hamiltonian_graph	giraffe
dolphin	projective_geometry	graph_toughness	waterbuck
african_forest_elephant	diagram	lacquer	sandy_beach
placental_mammal	line_of_sight	figuration	moose
<b>otter</b>	venn_diagram	war_paint	<b>wildebeest</b>
gorilla	hippocratic_face	graph	skunk
<b>wildebeest</b>	real_number_line	spectrograph	<b>anteater</b>
chimaera	sight_draft	map	<b>echidna</b>
african_savannah_elephant	x_axis	arabesque	bobcat
florida_panther	simulacrum	fall_off_analysis	mule_deer
liger	cartoon	art_collection	bison
rabbit	diagrammatic	statue	pygmy_marmoset
<b>aardvark</b>	camera_lucida	delineate	mongoose
<b>iguana</b>	explode_view	jack_o_lantern	sea_otter
hippopotamus	crowd_art	gussie_up	<b>squirrel_monkey</b>
hadrosaur	lottery	ecce_homo	wolverine
mountain_goat	depiction	pointillist	okapi
panda_bear	concept_design	art_deco	cane_rat
velociraptor	infinity_symbol	pyrography	whale
whale	scenography	scenography	american_bison

Table 6. Top 20 detections per each image from PSL Stage I (IUR).

They are provided in the Table 6. Observe that the individual detections are better compared to GUR<sup>10</sup>.

Final output from PSL Stage II (for BUR) is comparable to that of the GUR approach. The top detections are: *hadrosaur*, *sea\_ottter*, *diagrammatic*, *panda*, *iguana*, *pyrography*, *mule\_deer*, *placental\_mammal*, *liger*, *panda\_bear*, *art\_deco*, *squirrel\_monkey*, *giraffe*, *echidna*, *otter*, *anteater*, *pygmy\_marmoset*, *hippopotamus*.

Here, the set of output mainly contains the concepts (words or phrases) that either represents “animals with some similar visual characteristics to aardvark” or it pertains to “cartoon or art”.

## C. Detailed Accuracy Histograms For Different Variants

In this section, we plot the accuracy histograms for the entire dataset for all the variants (using Clarifai API) of our approach (listed in Table 2 of the paper). We also add the accuracy histograms for variants using BUR approach. The plots are shown in the Figure 8. From the plots, the shift towards greater accuracy is evident as we go along the stages of our pipeline.

## D. Visual Similarity: Additional Results

Additional results for Visual Similarity are provided in Tables 7 and 8.

ConceptNet	Visual Similarity	word2vec
man, merby, misandrous, philandry, male_human, dirty_pig, manlyhose, date_woman,guyliner,manslut	priest, uncle, guy, geezer, bloke, pope, bouncer, ecologist, cupid, fella	women, men, males, mens, boys, man, female, teenagers,girls,ladies

Table 7. Similar Words for “Men”

<sup>10</sup>The output from the PSL Stage I for BUR, is completely independent of the other images. In essence, for each image, we are predicting all relevant concepts from a large vocabulary given a few detections from a small set of class-labels.

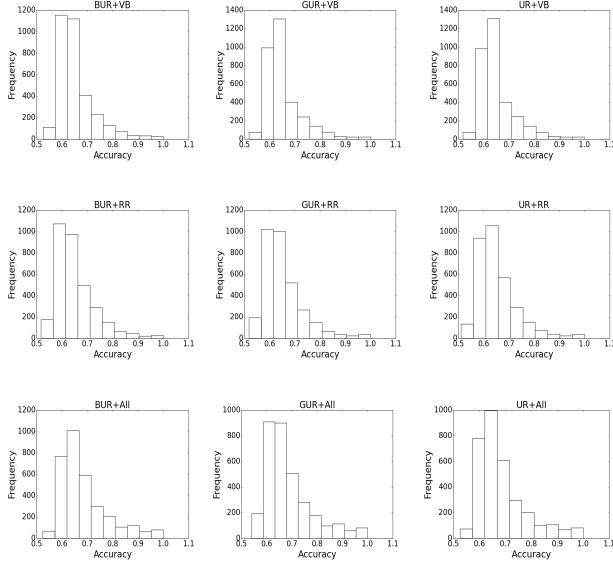


Figure 8. The accuracy histograms of the BUR, GUR and UR approaches (combined with the VB, RR and All stage variants).

ConceptNet	Visual Similarity	word2vec
saurischian, ornithischian, protobird, elephant bird, sauropsid, cassowary, ibis, nightingale, ceratosaurian, auk, vulture	lambeosaurid, lambeosaur, bird, allosauroid, theropod, stegosaur, triceratops, tyrannosaurus_rex, deinonychosaur, dromaeosaurus, brontosaurus	dinosaurs, dino, T_rex, Tyrannosaurus_Rex, T_rex, fossil, triceratops, dinosaur_species, tyrannosaurus_dinos, Tyrannosaurus_rex

Table 8. Similar Words for “Dinosaur”

## E. More Positive and Negative Results

We provide positive and Negative results in Figures 9 and 10 of the “GUR+All” variant of the pipeline. We obtain better results with Clarifai detections rather than Residual Network detections. Based on our observations, one of the key property of the ResidualNetwork confidence score distribution is that there are few detections (1-3) which are given the strongest confidence scores and the other detections have very negligible confidence scores. These top detections are often quite noisy.

For example, for the aardvark image 1, the ResidualNetwork detections are: **triceratops**, **wallaby**, **armadillo**, **hog**, **fox squirrel**, **wild boar**, **kit fox**, **grey fox**, **Indian elephant**, **red fox**, **mongoose**, **Egyptian cat**, **wombat**, **tusker**, **mink**, **Arctic fox**, **toy terrier**, **dugong**, **lion**. Only the first detection has 0.84 score and the rest of the scores are very negligible. For the second, third and fourth images, the top detections are respectively:

1. **pick** (0.236), **ocarina** (0.114), **maraca** (0.091), **chain saw** (0.06), **whistle** (0.03), **can opener** (0.03), **triceratops** (0.02), **muzzle**, **spatula**, **loupe**, **hatchet**, **letter opener**, **thresher**, **rock beauty**, **electric ray**, **tick**, **gong**, **Windsor tie**, **cleaver**, **electric guitar**
2. **jersey** (0.137), **fire screen** (0.129), **sweatshirt** (0.037), **pick** (0.035), **comic book** (0.030), **book jacket**

(0.029), **plate rack**, **throne**, **wall clock**, **face powder**, **binder**, **hair slide**, **velvet**, **puck**, **redbone**.

3. **hog** (0.48), **wallaby** (0.19), **wild boar** (0.10), **Mexican hairless** (0.045), **gazelle** (0.023), **wombat** (0.017), **dhole** (0.016), **hyena** (0.015), **armadillo** (0.009), **ibex**, **hartebeest**, **water buffalo**, **bighorn**, **kit fox**, **mongoose**, **hare**, **wood rabbit**, **warthog**, **mink**, **polecat**.

These predictions show that for the first and fourth image, there are some animals detected with some distant visual similarities. The second and third image has almost no animal mentions. This also shows some very confident detections (such as **triceratops** for the first image) is quite noisy.

In many cases, due to these high-confidence noisy detections, the PSL-based inference system gets biased towards them. Compared to that, Clarifai detections provide quite a few (abstract but) correct detections about different aspects of the image (for example, for 2nd Image, predicts labels related to “cartoon/art” and “animal” both). This seems to be one of the reasons, for which the current framework provide better results for Clarifai Detections. Using Residual Network, the final output from the GUR system for the “aardvark” riddle is: *antelope, prairie\_dog, volcano\_rabbit, marsupial\_lion, peccary, raccoon, pouch\_mammal, rabbit, otter, monotreme, jackrabbit, hippopotamus, moose, tapir, echidna, gorilla*.

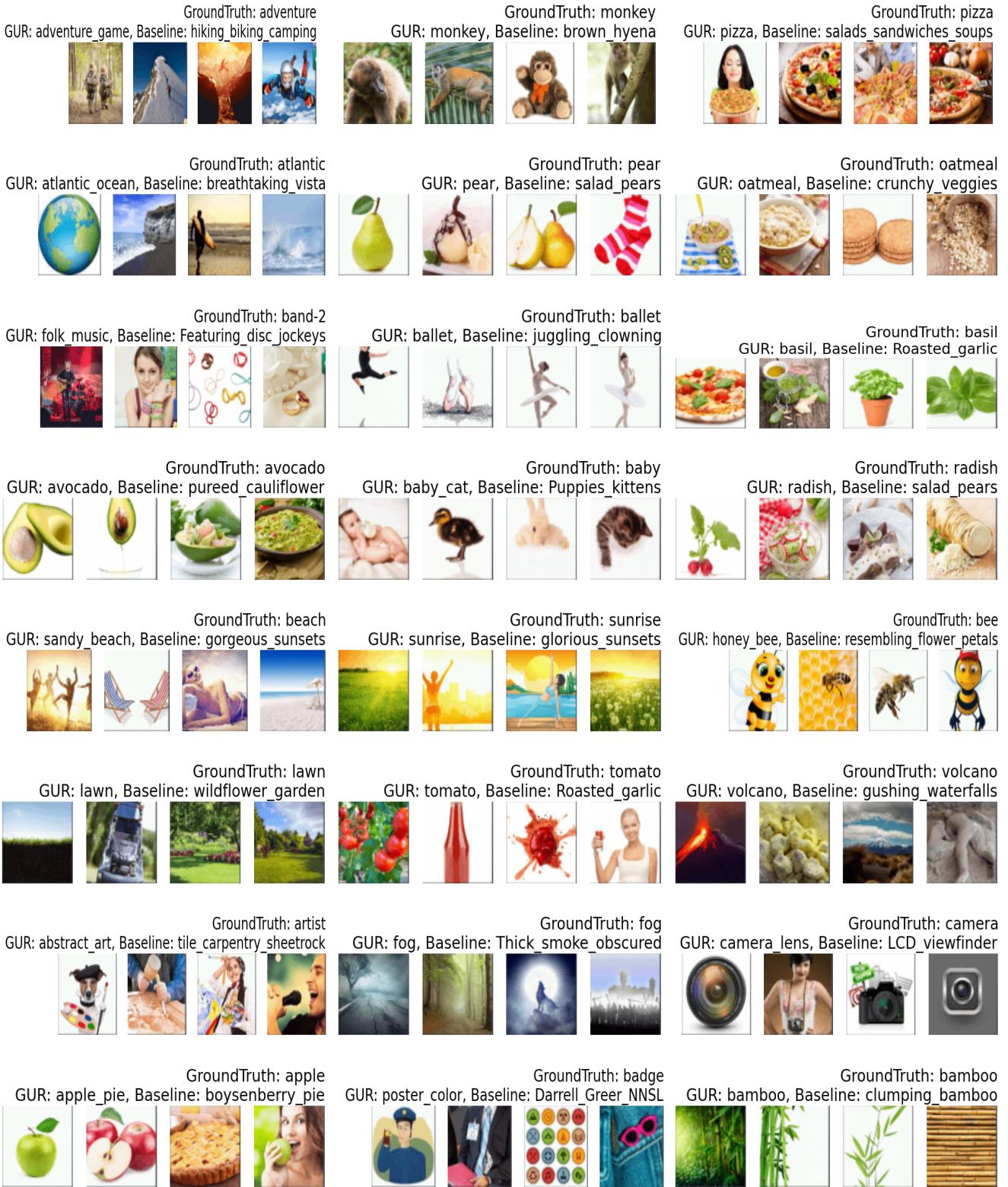


Figure 9. More Positive results from the “GUR” approach on some of the riddles. The groundtruth labels, closest label among top 10 from GUR and the Clarifai baseline are provided for all images. For more results, check <http://bit.ly/1Rj4tFc>.

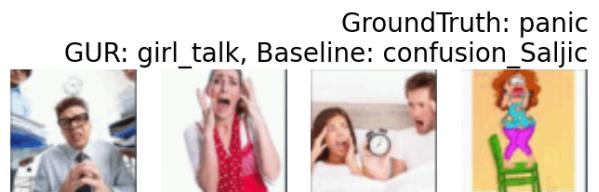


Figure 10. Some Negative results from the “GUR” approach on some of the riddles. The groundtruth labels, closest label among top 10 from GUR and the Clarifai baseline are provided for all images. For more results, check <http://bit.ly/1Rj4tFc>.