DeepIQ: A Human-Inspired AI System for Solving IQ Test Problems

Jacek Mańdziuk, **Adam Żychowski**

Faculty of Mathematics and Information Science, Warsaw University of Technology, Poland
{j.mandziuk, a.zychowski}@mini.pw.edu.pl

15 July 2019

Motivation

- check algorithms' ability to solve problems challenging for people - IQ tests
- computational intelligence measurement (*Psychometric AI*) vs people intelligence metric
- designing system universally applicable for different types of IQ tasks
- gap in the machine learning literature

Raven's Matrices

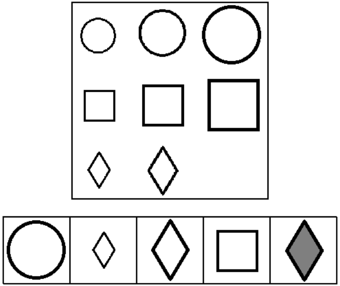
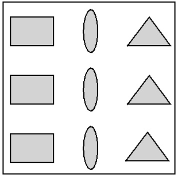


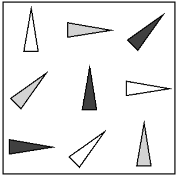
Figure: An Raven's Matrix example.

- the most popular IQ test
- independent of nationality, age, knowledge and language
- well-established method, widely researched by psychologists and used, for instance, in *Mensa* qualification tests

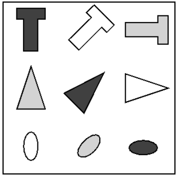
Raven's Matrices types



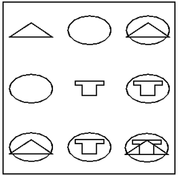
(a) RM with one relation



(b) RM with two relations

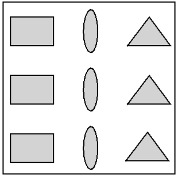


(c) RM with three relations

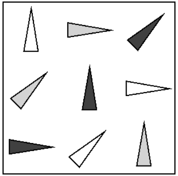


(d) RM with logic relation

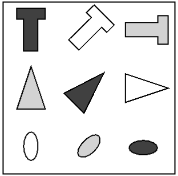
Raven's Matrices types



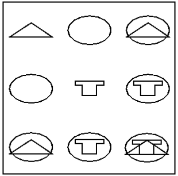
(a) RM with one relation



(b) RM with two relations



(c) RM with three relations



(d) RM with logic relation

System overview

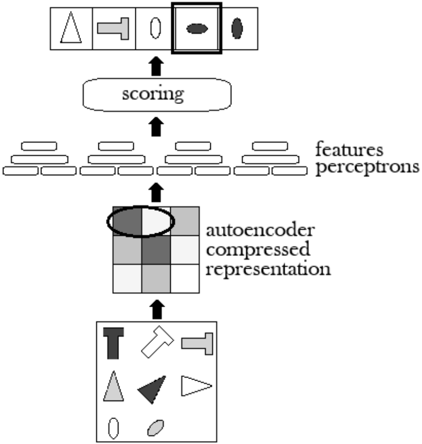


Figure: *DeepIQ* system overview.

Three main system components:

- a deep autoencoder which provides compressed representation of individual RM cells
- four feature related multi-layer perceptrons
- a scoring module

Deep autoencoder

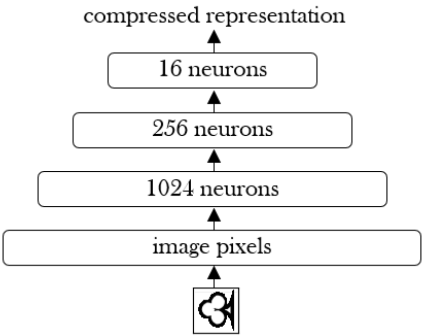


Figure: Deep autoencoder architecture. Its last layer provides a compressed version (16 numbers) of the input image.

Feature related perceptrons

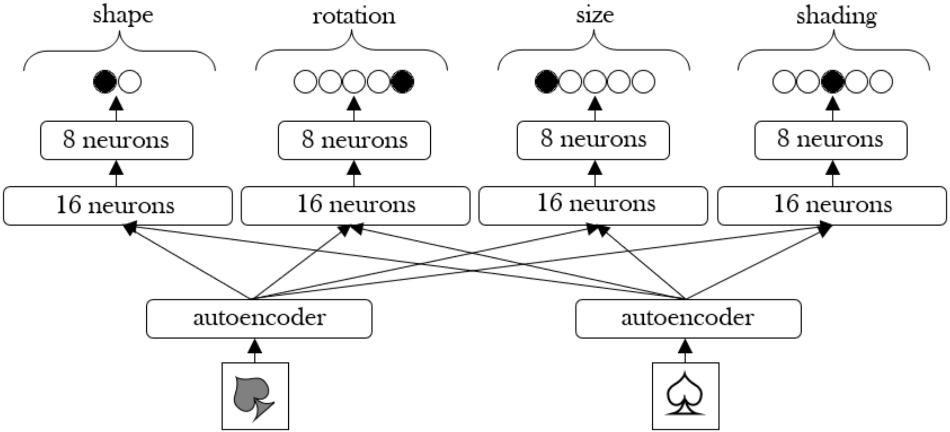
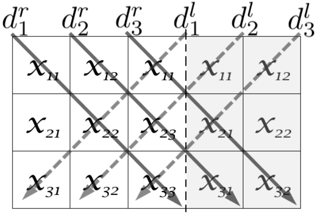


Figure: An architecture of the feature-related module with two example figure images shown in the input. Figures have the same shape and size, differ “moderately” in shading and “significantly” by rotation angle.

Answers scoring module

$s_{a_i} := 0$
if $f^k(x_{11}, x_{12}) = f^k(x_{12}, x_{13})$ and
  $f^k(x_{21}, x_{22}) = f^k(x_{22}, x_{23})$ then
    if $f^k(x_{31}, x_{32}) = f^k(x_{32}, x_{33})$ then
        $s_{a_i} := s_{a_i} + 1$
    else
        $s_{a_i} := s_{a_i} - 1$



$f^k(x_{i_1 j_1}, x_{i_2 j_2})$ - the output neuron with the highest activation value in the k-th feature related MLP ($k \in$ {shape, rotation, shape, shading}) in response to the compressed representation of images at positions (i_1, j_1) and (i_2, j_2) in the RM, respectively. f^k is interpreted as a *distance of feature k* between these two images.

Training procedure

- randomly generated images (15 different shapes, random shading, rotation and size)
- autoencoder trained with 5000 random images
- feature perceptrons trained with 5000 pairs of random images

Tested datasets

Generated tests (G-set)

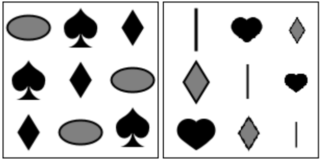


Figure: Sample tests from generated set.

Sandia tests (S-set)

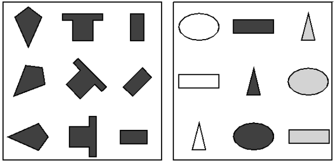


Figure: Sample tests from Sandia set.

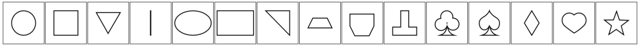


Figure: All shapes in generated set.

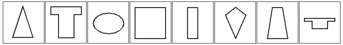


Figure: All shapes in Sandia set.

Results

	TR $\rightarrow$ TS	1 relation	2 relations	3 relations
DeepIQ	G $\rightarrow$ G	73.3% $\pm$ .7%	74.1% $\pm$ .5%	76.0% $\pm$ .6%
DeepIQ	G $\rightarrow$ S	70.2% $\pm$ .4%	71.9% $\pm$ .6%	73.2% $\pm$ .2%
Humans	$\rightarrow$ S	87.0%	72.0%	55.0%

Table: A comparison of the averaged accuracy of *DeepIQ* on the RMs generated by the authors (G-set) and on Sandia RMs (S-set). The last row presents human results on the S-set.

Odd-one-out

- point the *oddest* figure image out of given n images
- while solving RMs relies on detection and quantification of similarities, the odd-one-out tests are focused on feature based differences between figure images.

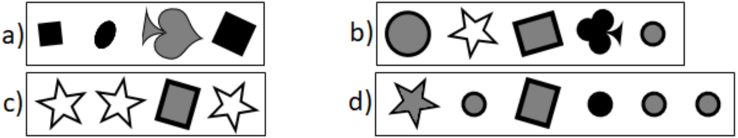


Figure: Example odd-one-out tests.

Odd-one-out results

- the first two modules (deep AE and feature related MLPs) used with no additional training or tuning of any kind
- only the scoring module was adopted to address specificity of odd-one-out problem

		Number of figures (n)					
		4		5		6	
		G-set	S-set	G-set	S-set	G-set	S-set
c	1	$93.2\% \pm .4\%$	$91.5\% \pm 0.2\%$	$93.1\% \pm .2\%$	$91.3\% \pm .1\%$	$93.3\% \pm .4\%$	$91.1\% \pm .2\%$
	2	$95.2\% \pm .2\%$	$91.2\% \pm 0.5\%$	$95.5\% \pm .4\%$	$92.1\% \pm .2\%$	$95.2\% \pm .4\%$	$92.1\% \pm .1\%$
	3	$96.2\% \pm .4\%$	$92.5\% \pm 0.1\%$	$96.4\% \pm .3\%$	$92.7\% \pm .4\%$	$96.3\% \pm .3\%$	$92.9\% \pm .3\%$

Table: *DeepIQ* averaged accuracy on G1-set and S1-set for various problem sizes (n) and numbers of common features in the subset of $n-1$ figures (c).

Summary

- the first two system components (autoencoder and feature networks) are universal and could be applied to other types of IQ test problems
- the system follows *transductive transfer learning* approach (knowledge learned from one data set is successfully used to solve the same problem with data set created with different distribution)
- ability to detect differences (and assess their range) in figure shapes, sizes, rotations and shadings
- approximately human level performance

Questions?

Challenging RM tests

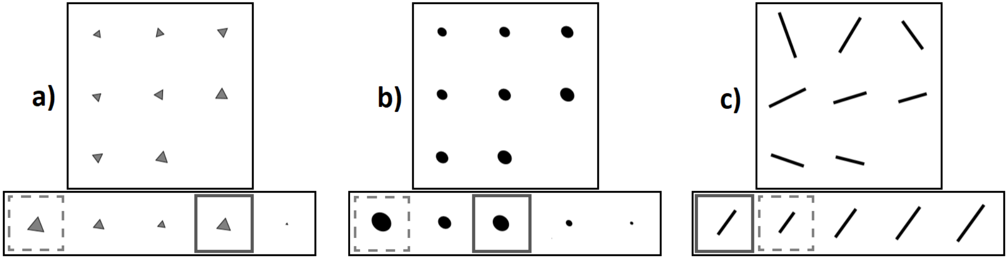


Figure: Examples of challenging RM tests. A correct answer is outlined with a solid line and the one proposed by *DeepIQ* with a dashed line.

Challenging Odd-one-out tests

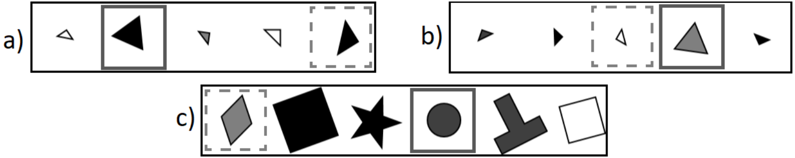


Figure: Examples of challenging odd-one-out tests. A correct answer is outlined with a solid line and the one proposed by *DeepIQ* with a dashed line.