RETHINKING INFORMATION

Small steps towards a big agenda

Branislav Rovan

Comenius University, Bratislava

February 9, 2010
CONTENTS

- Why to Rethink
- Usefulness of Information - a case study
- Some thoughts and questions
**INFORMATION – What is it?**

I get it from TV, radio, friend, ...

An answer to my enquiry

... 

Formal treatment (measuring its amount, learning, ...)

Philosophy
INFORMATION - many facets

- Shannon’s formalisation
  - stemming from the need to transmit data
  - and to minimize transmission time, ...
  - $\rightarrow$ amount (size) of information

- Not adequate anymore
  - We need to capture and study new ways of using and processing information.
INFORMATION - many facets

- Usefulness, Usability, Relevance, Availability, Impact, Timeliness, Trust, Privacy, Stability, ...
INFORMATION - many facets

- Usefulness, Usability, Relevance, Availability, Impact, Timeliness, Trust, Privacy, Stability, ...

- For What, For Whom, ...
We shall present a case study enabling to make the notion of usefulness more precise and develop a framework that may lead to a better understanding of the notion of information.
Why to Rethink

- Usefulness of Information - a case study

Some thoughts and questions
USEFULNESS – an emerging notion

Use (additional) information to do more or better.
USEFULNESS – an emerging notion

Use (additional) information to do more or better.

- Ask a wise man (oracle) and solve more problems
Use (additional) information to do more or better.

- Ask a wise man (oracle) and solve more problems
- Learn specifics and solve some problems better
An assistant (advisor) can provide additional information about the input instances thereby enabling simpler solution.

**Graph problems** – the input is planar, bounded treewidth, ...

**Communication in networks** – sense of direction, ...

**Online algorithms** – advice complexity

...
we engage an “advisor”, that provides to the “solver” some information about the input

(e.g., advisor may preprocess the input and solver then obtains some information about the results of the advisor’s computation)

we expect that having the information provided by the advisor, the solver’s task may become easier

we also expect the advisor to be simpler than the original solver required for the task, otherwise the advisor would make the task trivial
Developing notions

Information provided by the assistant is useful in case it is simpler to solve the new problem.

Assistant should be simpler than the original solver.

Measure of complexity of solvers is needed.
Finite Automata case

ababb, aababa, bbababaaba
Finite Automata case

Problem this automaton solves

\[ L = \{ w \in \{a, b\}^* \mid \#_a(w) \mod 3 = 2 \land \#_b(w) \mod 2 = 1 \} \]
Assisted Solution

Branislav Rovan
Lorentz Center, February 9, 2010
Assisted Solution
Additional information on input – odd number of $b$. 
Knowing that $#_b(w) \mod 2 = 1$ it suffices to solve

$L' = \{ w \in \{a, b\}^* \mid #_a(w) \mod 3 = 2 \}$
Finite Automata Case

Measure of complexity – *Number of States*

**Building on techniques of automata decomposition**

Technical details can be found in
the solver is a DFA trying to recognize some regular language
the advisor is also a DFA
we let the solver know some result of the advisor’s computation on the input
  did it accept the input?
  what was the final state?
to obtain nontrivial results we require both the advisor and the solver to be simpler than the minimal DFA for the language recognized
Undecomposable automata exist

i.e., There exist automata for which

any additional information given by a DFA is useless.
Types of Results Obtained for DFA

**Undecomposable automata exist**

i.e., There exist automata for which any additional information given by a DFA is useless.

**Perfectly decomposable automata exist**

i.e., such an $n$ state automaton can be decomposed into a $k$ state solver and $l$ state advisor so that $n = k \cdot l$. 
Almost undecomposable automata exist

i.e., for an \( n \) state automaton each nontrivial decomposition leads to an \( n - 1 \) state solver and an \( n - 1 \) state advisor.

Inbetween cases exist

For each \( r, s, t \) there is an \( n \) state automaton, \( n = r.s + t \), decomposable into \( r + t \) state solver and \( s + t \) state advisor.
Experience

Not easy
despite limiting type of solvers and type of additional information provided.

Even measuring complexity may turn out to be a problem (DPDA case)
Experience – In the AUTOMATA setting

The same information may be at the same time usefull and useless.

Usefulness of information depends on the recipient.
Why to Rethink

Usefulness of Information - a case study

Some thoughts and questions
Information Processing – steps needed
Information Processing – steps needed

- “Create” information
  Representing, encoding, … ’real world’ objects
Information Processing – steps needed

- "Create" information
  Representing, encoding, ... ‘real world’ objects

- Find efficient ways to manipulate it
  algorithms, datastructures, ..., (presentation as well)
Information Processing – steps needed

- “Create” information
  Representing, encoding, … ’real world’ objects

- Find efficient ways to manipulate it
  algorithms, datastructures, …., (presentation as well)

- Make sure you did what you intended to do
  specify and prove
Information Processing – steps needed

- “Create” information
  Representing, encoding, … ’real world’ objects

- Find efficient ways to manipulate it
  algorithms, datastructures, …, (presentation as well)

- Make sure you did what you intended to do
  specify and prove

- Make it really work
  implementation
• “Create” information
  Representing, encoding, … ’real world’ objects
• Find efficient ways to manipulate it
  algorithms, datastructures, …, (presentation as well)
• Make sure you did what you intended to do
  specify and prove
• Make it really work
  implementation
'Creating' Information

1. Take a 'real world' object

2. Select some of its (important) features

3. Put them in a suitable form
'Creating’ Information

1. **Take a ’real world’ object**
   having some purpose in mind

2. **Select some of its (important) features**
   may be different for different purposes

3. **Put them in a suitable form**
   to suite manipulation envisaged
’Creating’ Information

1. Take a ’real world’ object having some purpose in mind
2. Select some of its (important) features may be different for different purposes
3. Put them in a suitable form to suite manipulation envisaged

’Real world’ object → Model
'Creating' Information – an Elephant Example

1. **Take a 'real world' object** – an Elephant to be transported
2. **Select some of its (important) features**
   Weight and Dimensions
3. **Put them in a suitable form**
   Object: Weight 2 tons, Dim 3x2x3 meters
'Creating' Information – an Elephant Example

1. Take a 'real world' object – an Elephant to be transported
2. Select some of its (important) features
   - Weight and Dimensions
3. Put them in a suitable form
   - Object: Weight 2 tons, Dim 3x2x3 meters

The forwarding company may come with an open platform truck, a crane and a set of steel ropes for loading 'the object' onto the truck.
INFORMATION – What is it?
INFORMATION – What is it?

INFORMATION is a MODEL of reality created for a PURPOSE.
INFORMATION – (some) Key pieces

- **Purpose** (plan) for handling an object
- **Model** (a particular form of the object)
- **Communicate** (to receiver)
- **Implement** via a Performing agent (receiver)
Questions

**Purpose** – how to specify? New language needed?

**Model** – It can be in a ’better form (I1 is better formed than I2) ’? To compare we probably need ’a purpose of information’ and/or receiver?

**Communicate** – Information has to be ’understood’ by the receiver. Are information and receiver ’compatible’?

**Receiver** – computational power, complexity restrictions, task, … Verifiability of fulfilling the purpose?
We may need to look for a 'compulsory structure' of information (similar to 'object class' and/or 'object instance' in object oriented programming).
We may need to look for a 'compulsory structure' of information (similar to 'object class' and/or 'object instance' in object oriented programming).

No one would consider a piece of computer code found on the net useful unless it is 'properly packaged' (description of what it does, how it communicates with the environment, ...). Should it be the same for information?
Blueprinting Information

What should be in the blueprint?

Some items are clear

author, purpose, 'body of information', intended receiver

Hints for other items follow.
Quality of Information

- The same information may be more useful if it is 'better organized' (e.g., a sorted list of some records as opposed to a chaotic pile).

- Information from a 'trusted' or more 'knowledgable' source is usually considered 'better'.

- Should the information sources be traceable? (e.g., building an 'ancestor tree' may reveal wrong/unreliable information that is perpetuated/amplified even attached higher 'quality mark' if used by a 'more reliable' source).
People are usually proud of the piece of computer code they produced. Could/should the same be true for information? Would requirement to provide ‘meta information’ in the ‘packaged information’ be acceptable?

Profiting from giving false or misleading information is punishable in certain areas. Should it be always punishable?
Green Peace

We separate waste to help our environment. Throwing waste around is considered wrong (even punishable).

We are willing to pay for getting rid of waste and keeping our environment clean.
We separate waste to help our environment. Throwing waste around is considered wrong (even punishable).

We are willing to pay for getting rid of waste and keeping our environment clean.

What makes Information an exception?
Greed

Accumulating things
Greed

Accumulating things

Do we need to store all information just because we can? (it is costly, ...).
No one stores all the worn out socks ...

Be reasonable!
(Would it make a difference if all the kitchen utensils of the past were preserved?)
Information we need

Usefull, easy to use, trustworthy, accurate, accessible, ...
Information we need

Usefull, easy to use, trustworthy, accurate, accessible, ...

Let us

– increase our understanding

– design a framework for information of our needs.
THANK YOU FOR YOUR ATTENTION

ACKNOWLEDGEMENTS

My students:
A. Dénešová, P. Gaži, M. Králik, P. Labath, L. Steskal

The Lorentz Center