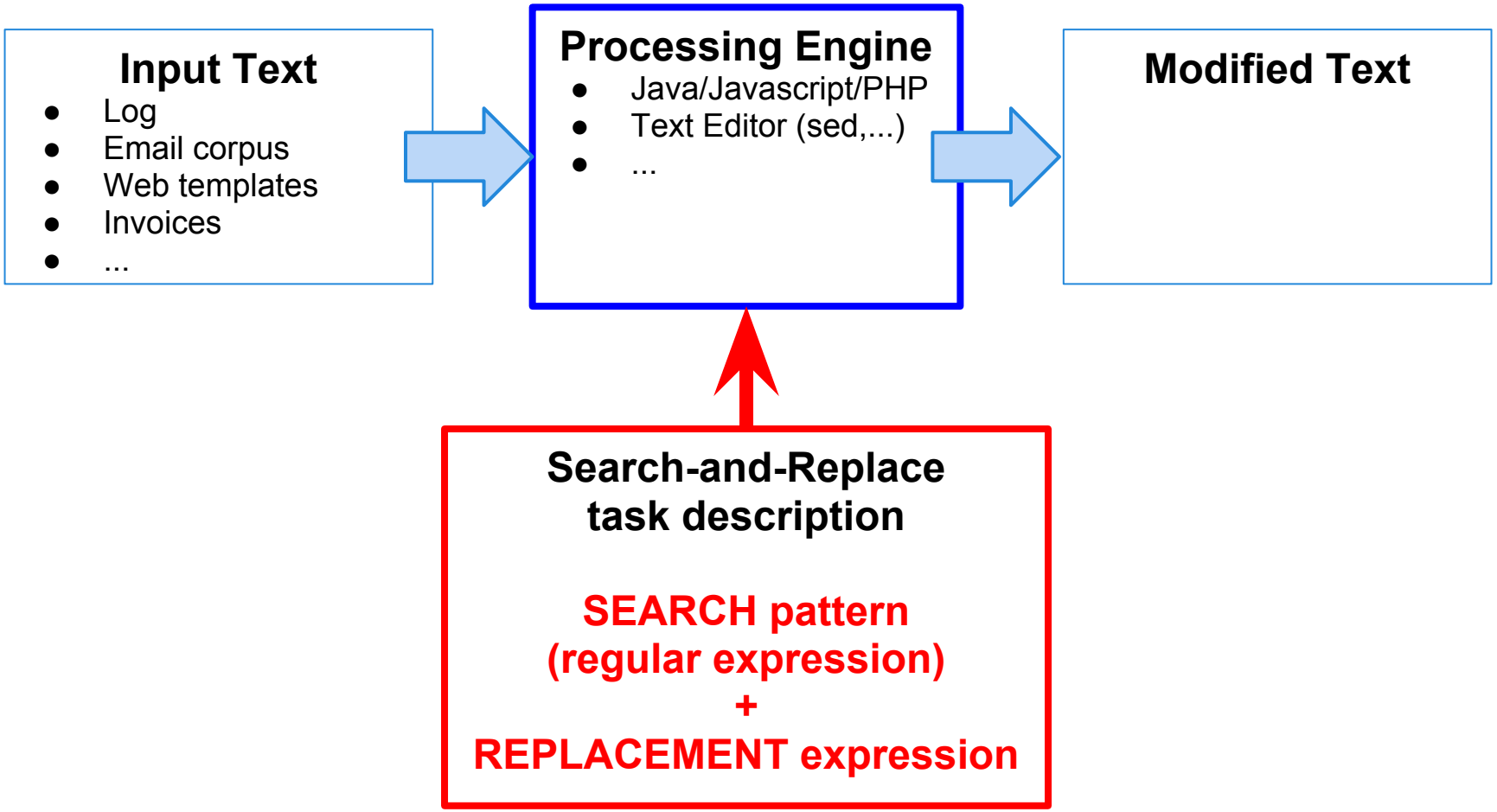
Automatic String Replace by Examples

A. De Lorenzo, E. Medvet, A. Bartoli
University of Trieste, Italy

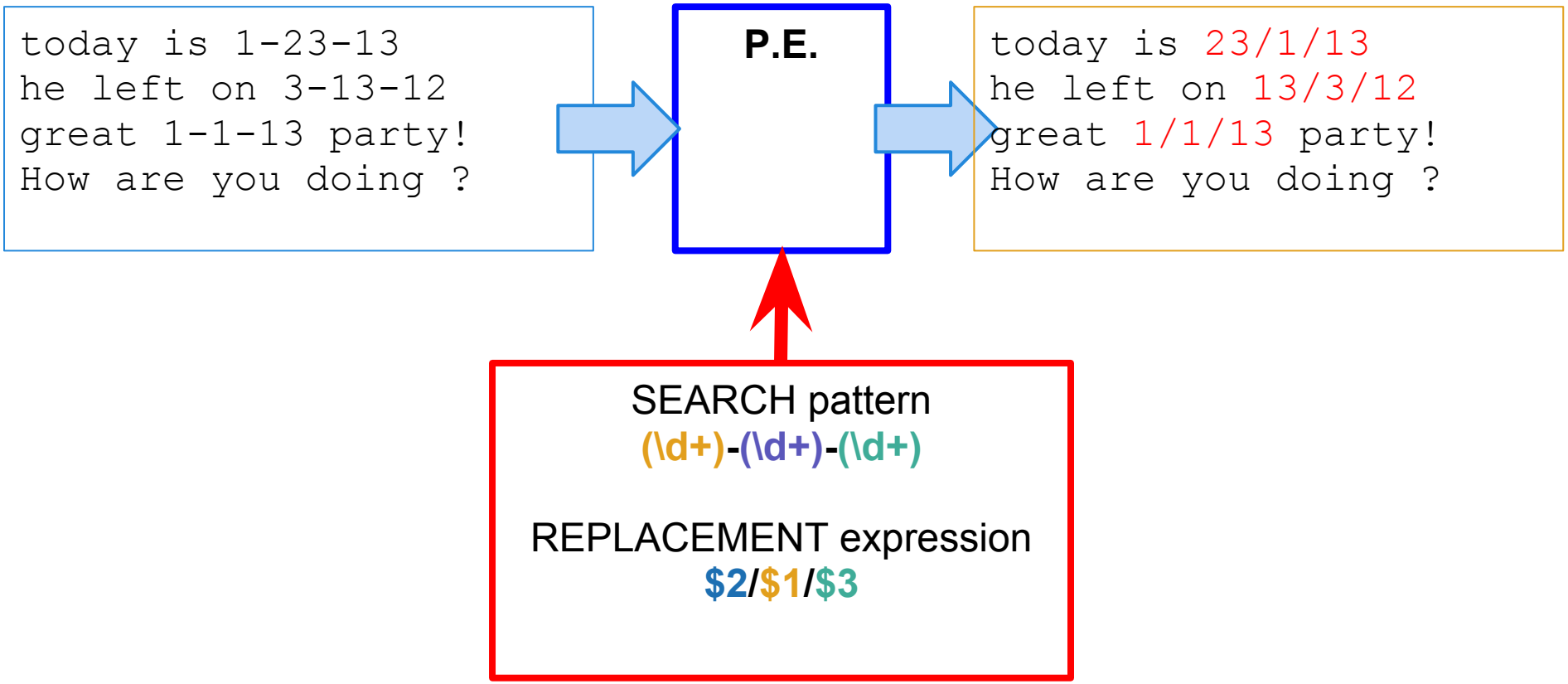
Genetic and Evolutionary Computation Conference

Amsterdam, The Netherlands
July 06-10, 2013

Text Search-and-Replace



Simple Example: Date Format Change



In practice

(@\w\w)\w+

$1xxxx

(\d+)-(\d+)-(\d+)

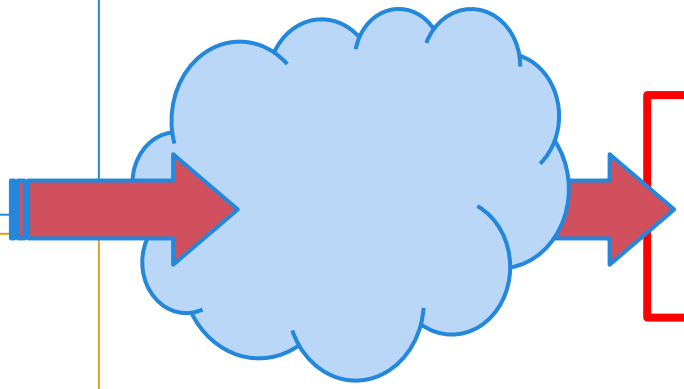
$2/$1/$3

- Search-and-replace tasks must be described **by hand**
- Requires **technically-savvy** users
- Often difficult to **debug**

Wouldn't it be nice if...

```
today is 1-23-13
he left on 3-13-12
great 1-1-13 party!
How are you doing ?
```

```
today is 23/1/13
he left on 13/3/12
great 1/1/13 party!
How are you doing ?
```

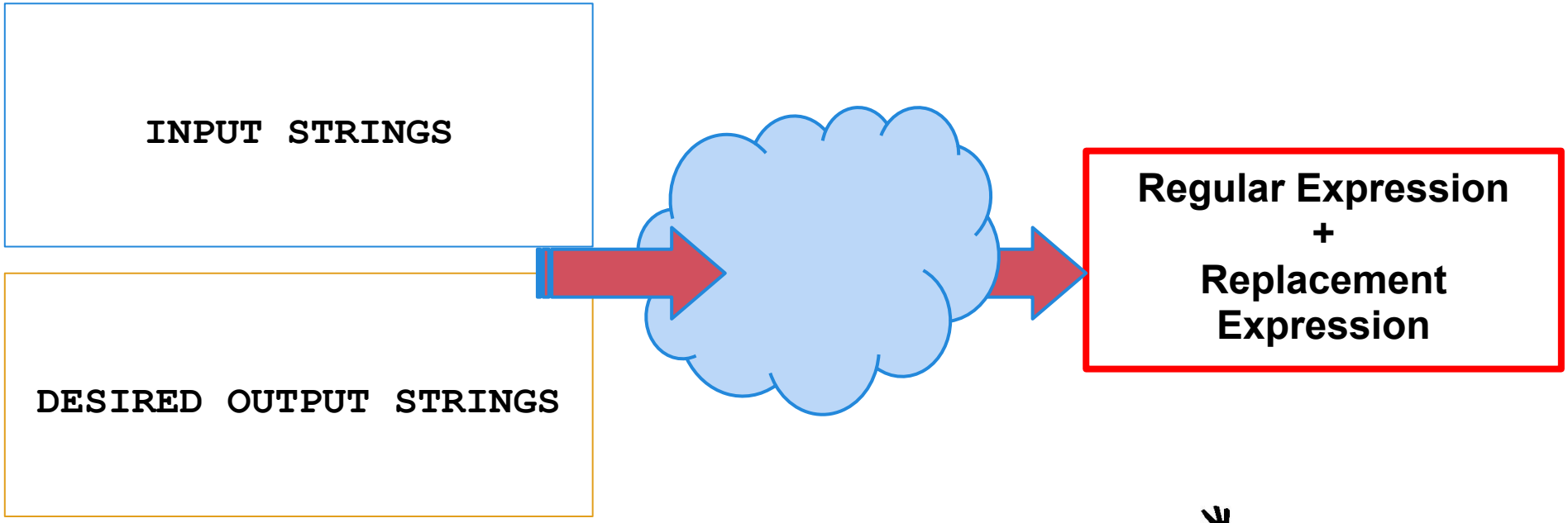


(\d+)-(\d+)-(\d+)

$2/$1/$3

- Search-and-Replace tasks could be described merely by a few **examples ?**

Our work



- Completely **AUTOMATIC**
- No similar proposals
- *Of course, with limitations / constraints...*

How it works (in a nutshell)

1. Identify a **context** where changes have to be confined
   (genetic programming)

```
...
Bartoli is ok @Bartoli
...
```

out of context *in context*

```
...
Bartoli is ok @Baxxxx
...
```

2. Build the **replacement expression** r
   (deterministically)

3. Generate the **regular expression** s
   (genetic programming)

1. *Identify a **context** where changes have to be confined*

Phase 1: Remark

- We do **not** require that the user specifies a context explicitly

```
...
Bartoli is ok @Bartoli
...
```

```
...
Bartoli is ok @Baxxxx
...
```

- We need a way to identify a context **automatically**

1. *Identify a **context** where changes have to be confined*

Phase 1

1. Build an ancillary **example set** composed of:

Input strings

```
I like @GECCO13 conf
RT @MaleLabTs New paper
Bartoli is ok @Bartoli
nothing new here
```

To-be-changed strings

```
CCO13
leLabTs
rtoli
(empty)
```

Constructed automatically from Input Strings + Desired Output Strings

2. Generate a single regex extracting a **superstring** **--->** the context

HOW ?

```
@GECCO13
@MaleLabTs
@Bartoli
(empty)
```

Regex Generation from Examples

- Build upon our GECCO 2012 work
  - Text extraction
  - Generate regex **automatically** from examples (with GP)
  - Available online (`http://regex.inginf.units.it/`)
  - Regex extracts exactly what is specified (i.e., not a superstring)
- Key difference: **fitness**
- Promotes regexes with minimal distance between:
  - string extracted by the **first capturing group** of the candidate
  - to-be-changed string

Phase 2

2. *Build the replacement expression r*

1. For each example, build a replacement expression **deterministically**
   - Identify:
     - context extracted from input string
     - "corresponding" substring in the output string
   - "Play" with them (see the paper for details)

07-14-1789

14/07/1789

$r_k =$ $2/$1/$3

2. Select the replacement expression that occurs most often

Phase 3

3. *Generate the regex s*
   - *Based on examples and r*

1. Take the examples of the overall task

```
today is 1-23-13
he left on 3-13-12
great 1-1-13 party!
```

```
today is 23/1/13
he left on 13/3/12
great 1/1/13 party!
```

2. **Generate** a regex based on these examples and r

- Multiobjective fitness (NSGA II)
  - Minimal difference in number of capturing groups
    - in the candidate
    - occurring in r
  - Minimal distance between
    - string generated by < candidate + r >
    - desired string

Experiments

- Tasks (500 with changes, 500 unchanged)
  - Full tweet anonymization
  - Partial IP anonymization
  - Date format change
  - Phone number change

- Experiments
  - Learning set: 20, 50, 100 (balanced)
  - 5-fold cross-validation

- Performance index: Count error rate

Salient Results

not available for learning

Task	Dataset $\lvert T^t \rvert$	$\lvert T^v \rvert$	$\lvert T^e \rvert$	Overall (ϵ_c %) μ
Twitter anonymization	10	10	980	5.5
	25	25	950	3.1
	50	50	900	2.0
IP partial anonymization	10	10	980	0.5
	25	25	950	0.0
	50	50	900	0.0
Date format change	10	10	980	60.0
	25	25	950	0.0
	50	50	980	0.0
Phone number format change	10	10	980	52.4
	25	25	950	8.2
	50	50	900	6.6

regex learning for mere extraction [4,11] a "difficult" dataset

- With only 25 positive examples:
  - perfect result for two tasks (on this dataset)
  - "very good" for the two other tasks

    (ok...of course...it depends on what we mean by "good"...)

Further experimental remarks

- "Many" solutions with "good" performance (not just lucky individuals)

- Relative performance in validation good predictor for relative performance in testing

- Execution time too high to devise interactive use (at least in the near future)

  But at 1-2$/hour is cheaper than a specialist ?

Task	$\lvert T^t \rvert + \lvert T^v \rvert$	Time (min)
Twitter anonymization	20	1
	25	1
	50	2
IP partial anonymization	20	12
	25	33
	50	43
Date format change	20	18
	25	36
	50	78
Phone number format change	20	35
	25	80
	50	132

Many open issues...of course...

- Remarkable and quite promising exercise (we believe)

- Keep in mind:
  - Results not human competitive (yet ?)
  - Execution time has to be improved

- Some key-but-unanswered questions:
  - How many examples are both "practical" and "adequate" ?
  - How to realize whether the examples are "enough" and "adequate" ?
  - How to characterize tasks that are just hopeless ?

Thanks for your attention

MACHINE LEARNING LAB

http://machinelearning.inginf.units.it