PAN

Uncovering Plagiarism, Authorship, and Social Software Misuse

Bauhaus-Universität Weimar	Martin Potthast, Tim Gollub, Matthias Hagen, Anna Beyer, Matthias Busse, Martin Tippmann, Johannes Kiesel, and Benno Stein
Universitat Politècnica de València	Parth Gupta and Paolo Rosso
University of the Aegean	Efstathios Stamatatos
Autoritas Consulting	Francisco Rangel
Duquesne University	Patrick Juola
Bar-Ilan University	Moshe Koppel
University of Lugano	Giacomo Inches
Illinois Institute of Technology	Shlomo Argamon

PAN

Uncovering Plagiarism, Authorship, and Social Software Misuse

Outline

- Author Profiling
- Author Identification
- Plagiarism Detection
- Software Submissions
- Summary

CLEF 2013 Valencia

upf. Universitat Pompeu Fabra Barcelona

iula Forensiclab

MARIE CURIE ACTIONS

WiQ-Ei

τιρα

The PAN Competition

PAN is a network around digital text forensics.

Mission

- ❑ Foster research and development in our tasks
- ❑ Push the limits of evaluating them
- ❑ Improve methodology for lab-style evaluations

Tasks

- ❑ Author Profiling (new in 2013)
- ❑ Author Identification
- ❑ Plagiarism Detection

Software Submissions

- ❑ Instead of run submissions (i.e., software output on a given input)
- ❑ Improves sustainability, replicability, and reproducibility
- ❑ Increases participant engagement
- ❑ Allows for cross-year evaluations

Author Profiling

- Given a document, what are its author’s demographics?

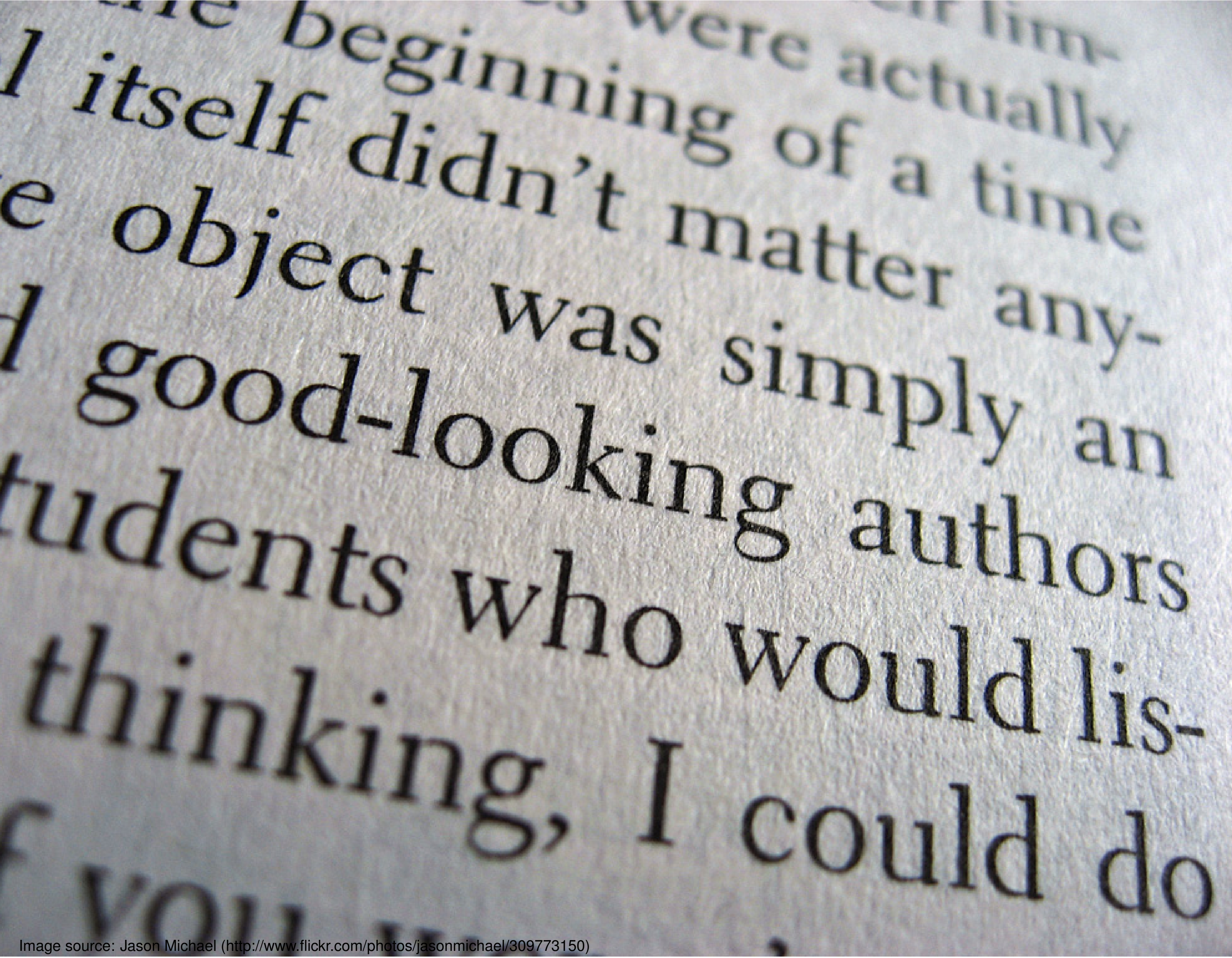
Corpus

- Genre: social media
- Languages: English, Spanish
- Size: 346 100 authors
- Annotations: age, gender

Selected results

- 21 softwares submitted
- Gender difficult to be discriminated, somewhat better in Spanish
- Age correctly detected in about 2/3 of cases

Award from the ForensicLab of the Universitat Pompeu Fabra



eir lim-
s were actually
he beginning of a time
l itself didn't matter any-
e object was simply an
d good-looking authors
tudents who would lis-
thinking, I could do
f you

Image source: Jason Michael (http://www.flickr.com/photos/jasonmichael/30973150)

Author Identification

- Given a document, who wrote it?

Corpus

- Genres: non-fiction writing, short fiction, news
- Languages: English, Spanish, Greek
- Size: 120 cases
- Annotations: authorship

Selected results

- 18 softwares submitted
- Greek more difficult than English and Spanish
- Balancing performance in all languages with a single approach difficult
- Meta-model competitive to participants, but does not dominate

Plagiarism Detection

- Given a document, is it an original?

Corpus

- Genre: web, news
- Language: English
- Size: 10000 suspicious documents
- Annotations: reused text passages, obfuscation

Selected results

- 19 softwares submitted
- Advanced evaluation framework for web-scale retrieval
- Different retrieval paradigms open up trade-off between costs and recall
- Summary plagiarism most difficult to be detected
- First-time cross-year evaluation; first steps toward all-time evaluation

Software Submissions

Software Submissions

Challenges ➜ Approaches

1. Environment diversity ➜ virtualization

   Support a wide variety of programming languages and operating systems.

2. Executing untrusted software ➜ virtualization

   Better be safe than sorry when executing binaries from a third party.

3. Data leakage ➜ sandboxing

   Prevent data leaking by running software in a secured environment.

4. Error handling ➜ unit testing

   Streamline the development round-trips for fixing execution errors.

5. Responsibility ➜ staged submissions

   Incentivice participants to submit early.

6. Execution cost ➜ provide hardware or raise usage fees

   We provided four servers each hosting up to 20 virtual machines.

Software Submissions

The 2013 Experience

- Entire lab accepts software submissions
- 62 virtual machines requested and provisioned
- 47 softwares installed, prepared for execution, and submitted by participants
- Testing and round-trips to fix errors
- Managed execution and evaluation using TIRA

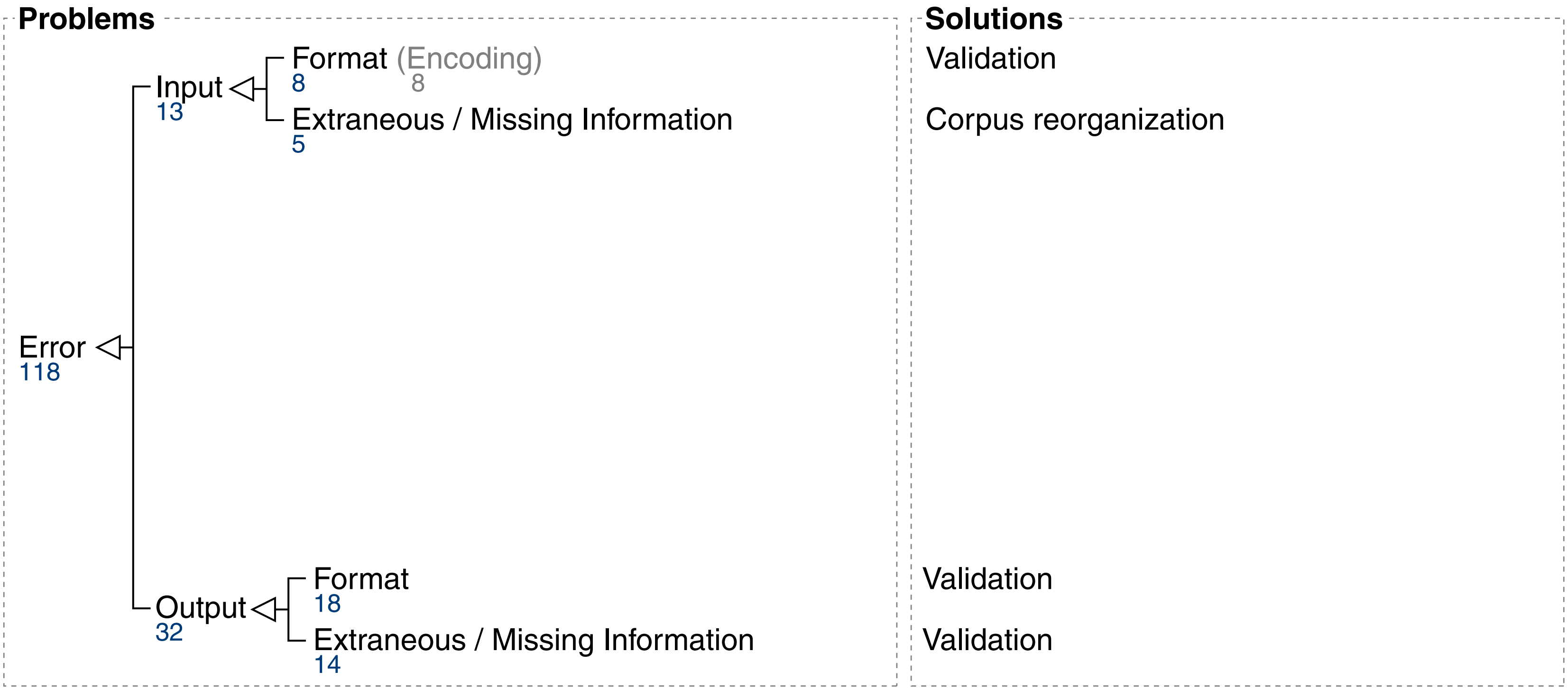
The 2012 Experience

- One task accepts software submissions
- 10 softwares submitted
- Manual preparation for execution by us
- Testing and round-trips to fix errors
- Managed execution and evaluation using TIRA

[tira.webis.de] [demo]

Software Submissions

Error Analysis



Problems

Error 118

Input 13

Format 8 (Encoding) 8

Extraneous / Missing Information 5

Output 32

Format 18

Extraneous / Missing Information 14

Solutions

Validation

Corpus reorganization

Validation

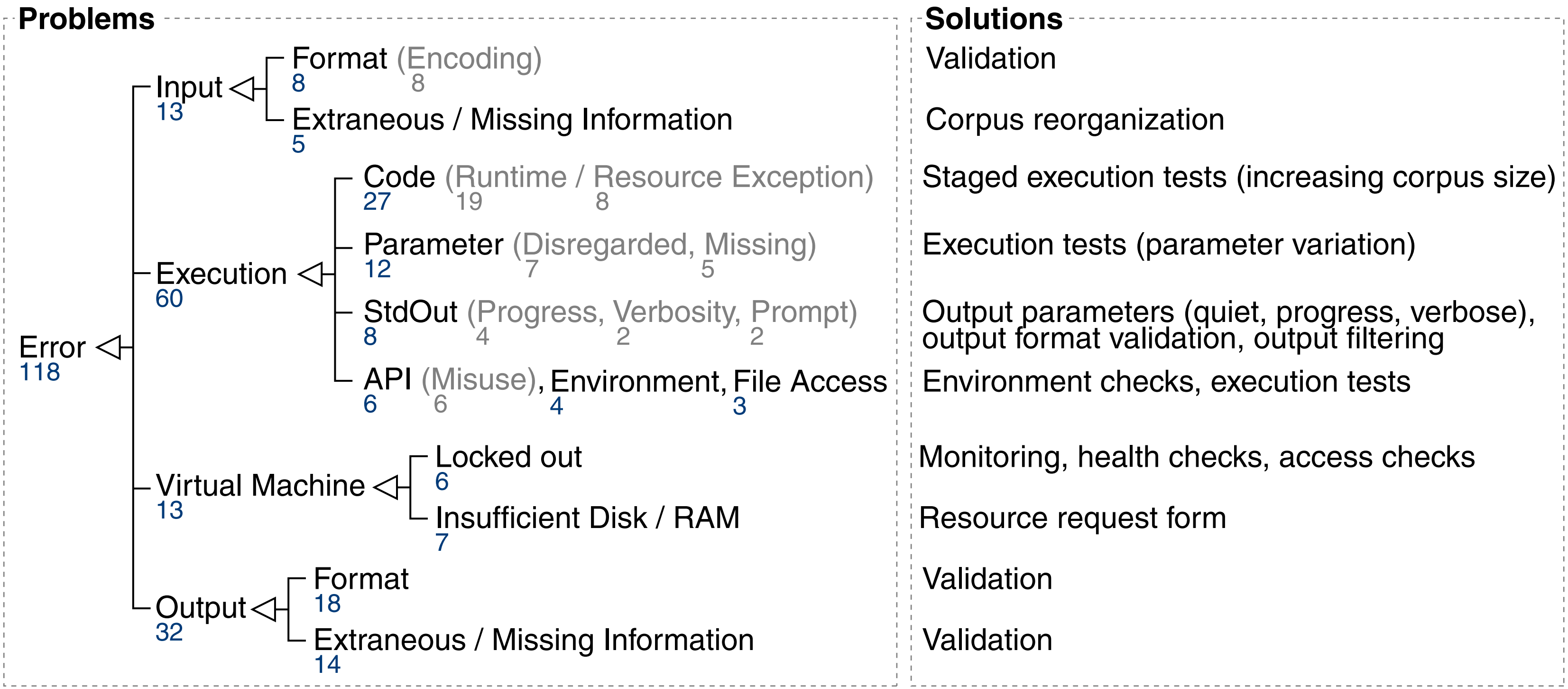
Validation

- 1493 mails exchanged in 392 conversations
- 39 of 46 teams experienced at least one error, 26 at least two, 1 team 10
- No one panicked
- Staged submissions helped resolve errors early on

➜ Rigorous unit testing and tools to assist participants in development

Software Submissions

Error Analysis



Problems

- Error 118
  - Input 13
    - Format (Encoding) 8 8
    - Extraneous / Missing Information 5
  - Execution 60
    - Code (Runtime / Resource Exception) 27 19 8
    - Parameter (Disregarded, Missing) 12 7 5
    - StdOut (Progress, Verbosity, Prompt) 8 4 2 2
    - API (Misuse), Environment, File Access 6 6 4 3
  - Virtual Machine 13
    - Locked out 6
    - Insufficient Disk / RAM 7
  - Output 32
    - Format 18
    - Extraneous / Missing Information 14

Solutions

- Validation
- Corpus reorganization
- Staged execution tests (increasing corpus size)
- Execution tests (parameter variation)
- Output parameters (quiet, progress, verbose), output format validation, output filtering
- Environment checks, execution tests
- Monitoring, health checks, access checks
- Resource request form
- Validation
- Validation

❑ 1493 mails exchanged in 392 conversations

❑ 39 of 46 teams experienced at least one error, 26 at least two, 1 team 10

❑ No one panicked

❑ Staged submissions helped resolve errors early on

➜ Rigorous unit testing and tools to assist participants in development

Software Submissions

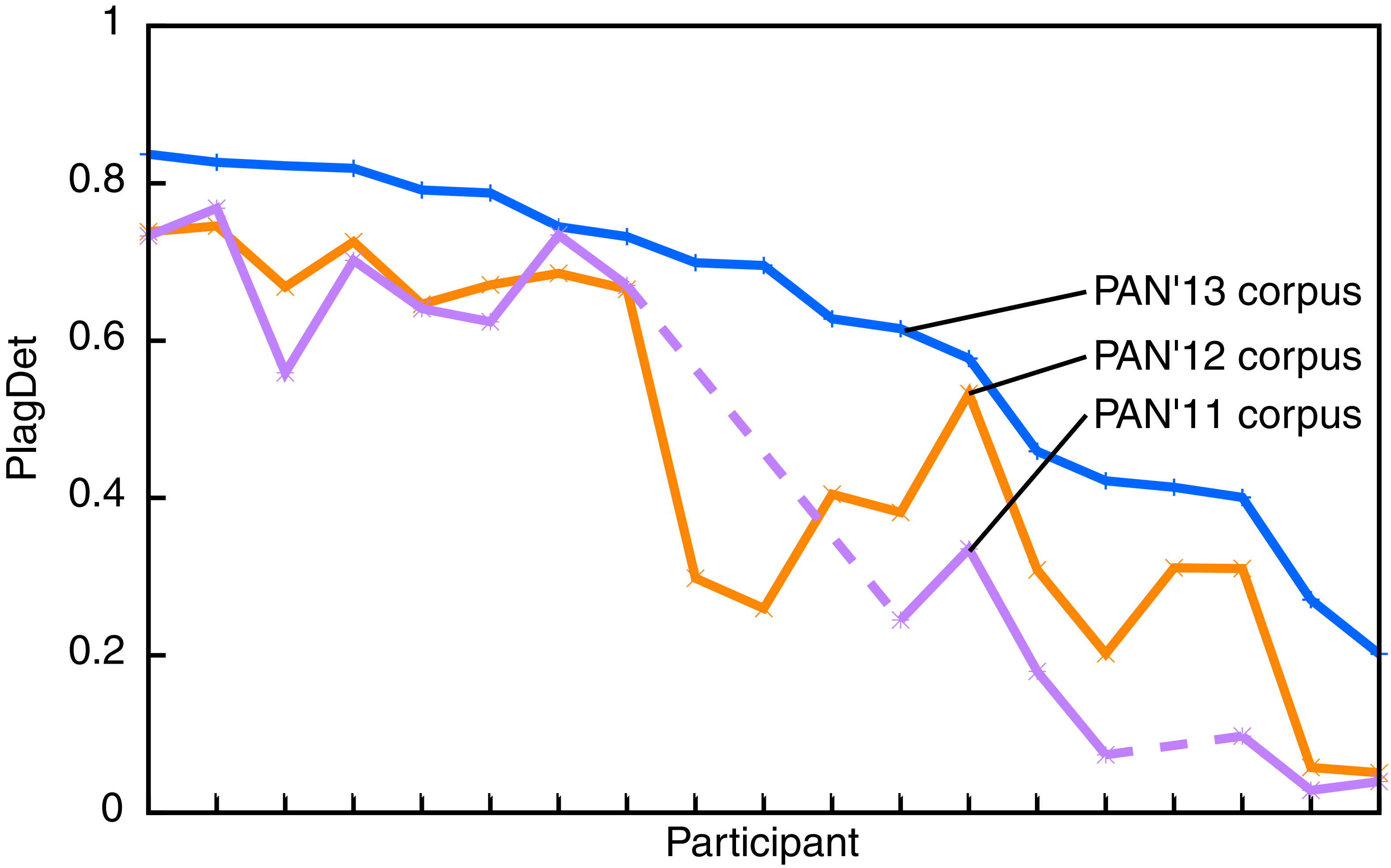
Cross-year Evaluation 2011-2013

Software Submission		PlagDet on PAN Plagiarism Corpus		
Team	Year	2013	2012	2011
Kong	**2012**	0.84	0.74	0.73
Oberreuter	**2012**	0.83	0.75	0.77
R. Torrejón	2013	0.82	0.67	0.56
Kong	2013	0.82	0.73	0.70
Palkovskii	**2012**	0.79	0.65	0.64
R. Torrejón	**2012**	0.79	0.67	0.62
Suchomel	2013	0.74	0.69	0.73
Suchomel	**2012**	0.73	0.67	0.67
Saremi	2013	0.70		
Shrestha	2013	0.70		
Kueppers	**2012**	0.63	0.40	
Palkovskii	2013	0.62	0.38	0.25
Nourian	2013	0.58	0.53	0.34
Sánchez-Vega	**2012**	0.46	0.31	0.18
Baseline		0.42	0.20	0.07
Gillam	**2012**	0.41	0.31	0.10
Gillam	2013	0.40	0.31	0.10
Jayapal	2013	0.27	0.06	0.03
Jayapal	**2012**	0.20	0.05	0.04

Software Submissions

Cross-year Evaluation 2011-2013 (continued)

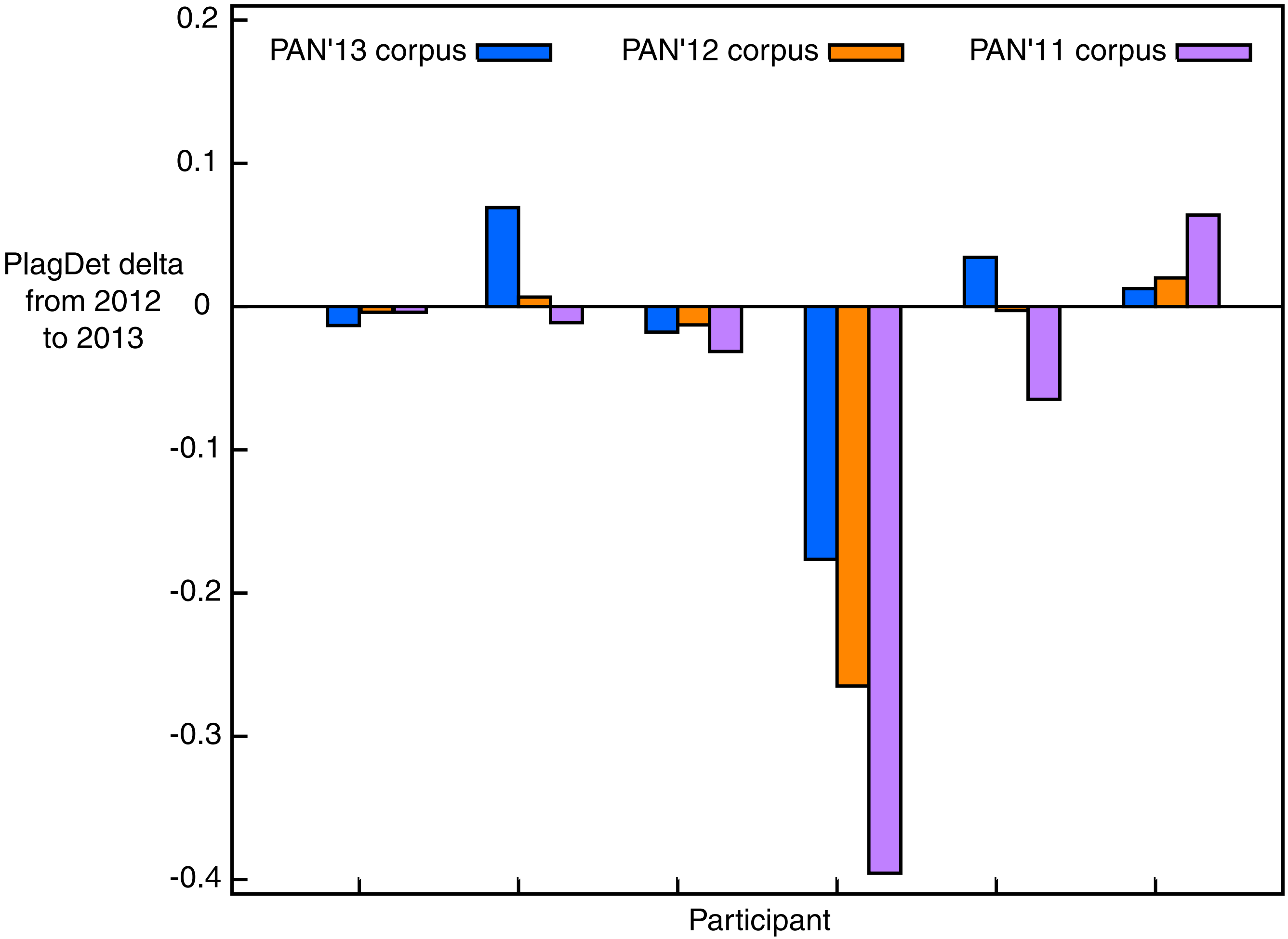
Assessing corpus difficulty



Software Submissions

Cross-year Evaluation 2011-2013 (continued)

Assessing improvments across versions



Summary

Statistics	**ALLC**	**SEPLN**	**FIRE**			**CLEF**			
	2004	2009	2011	2012	2013	2010	2011	2012	2013
Task(s)	1	1	1	1	1	2	3	3	3
Follower		78				151	181	232	286
Registrations	11	21	6	12	16	53	52	68	110
Runs/Software	13	14	6	8	8	27	27	48	58
Notebooks	8	11	6	2	6	22	22	34	47
Attendees	5	18	6	30		25	36	61	

Take-away messages

- Software submissions improve sustainability
- Software submissions allow for re-evaluation
- Software submissions allow for cross-year evaluation
- Software submissions do not discourage participation

Summary

Statistics	ALLC	SEPLN	FIRE			CLEF			
	2004	2009	2011	2012	2013	2010	2011	2012	2013
Task(s)	1	1	1	1	1	2	3	3	3
Follower		78				151	181	232	286
Registrations	11	21	6	12	16	53	52	68	110
Runs/Software	13	14	6	8	8	27	27	48	58
Notebooks	8	11	6	2	6	22	22	34	47
Attendees	5	18	6	30		25	36	61	

Take-away messages

- Software submissions improve sustainability
- Software submissions allow for re-evaluation
- Software submissions allow for cross-year evaluation
- Software submissions do not discourage participation

Thank you for your attention!