Twitter Sentiment Detection via Ensemble Classification Using Averaged Confidence Scores

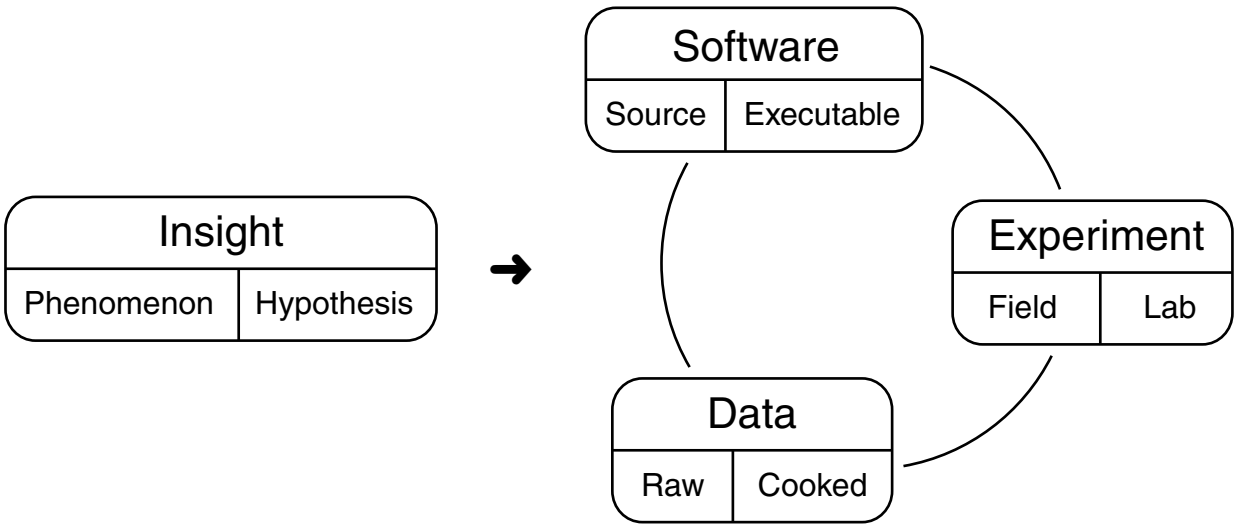
Matthias Hagen Martin Potthast Michel Büchner Benno Stein

Bauhaus-Universität Weimar

[www.webis.de]

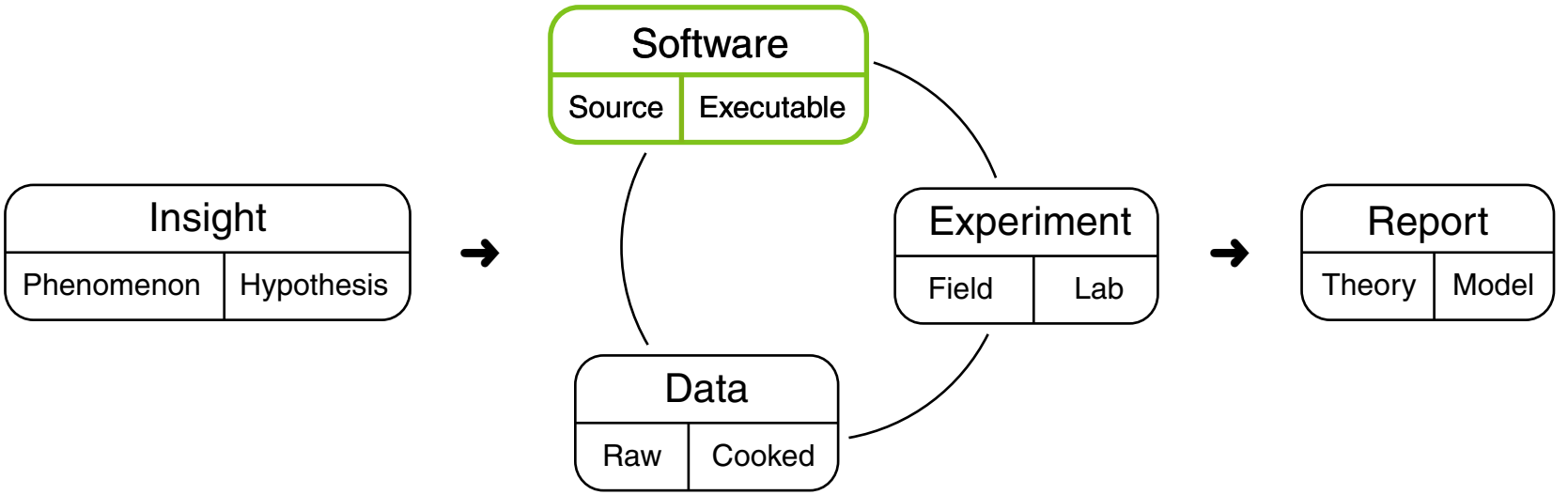
Reproducibility in Computer Science

Reproducibility in Computer Science



- General goal: algorithmic automation of certain tasks
- Evaluations establish success at doing so
- Both rely on pieces of software

Reproducibility in Computer Science



- General goal: algorithmic automation of certain tasks
- Evaluations establish success at doing so
- Both rely on pieces of software

- Reports frequently lack information for “painless” reproduction
- Assets used during modeling and evaluation are frequently not published

Reproducibility in Computer Science

Motivation, Incentives, and Barriers to Reproducing and Sharing Software

Personal motivation to reproduce a piece of research:	Bias:
1. to compare it with one's own approach for a given task	high
2. to double-check the results (e.g., to police fraud)	medium
3. to employ it as sub-module of another algorithm	low
4. to complete a library on a given task	low
5. to identify the best approach for application	low

Reproducibility in Computer Science

Motivation, Incentives, and Barriers to Reproducing and Sharing Software

Personal motivation to reproduce a piece of research:	Bias:
1. to compare it with one's own approach for a given task	high
2. to double-check the results (e.g., to police fraud)	medium
3. to employ it as sub-module of another algorithm	low
4. to complete a library on a given task	low
5. to identify the best approach for application	low

Personal incentives to share one's software:

1. to ensure optimal performance in evaluations
2. to build trust

3.-5. to foster adoption in research and practice

Reproducibility in Computer Science

Motivation, Incentives, and Barriers to Reproducing and Sharing Software

Personal motivation to reproduce a piece of research:	Bias:
1. to compare it with one's own approach for a given task	high
2. to double-check the results (e.g., to police fraud)	medium
3. to employ it as sub-module of another algorithm	low
4. to complete a library on a given task	low
5. to identify the best approach for application	low

Personal incentives to share one's software:

1. to ensure optimal performance in evaluations
2. to build trust

3.-5. to foster adoption in research and practice

Top barriers to sharing software [Stodden 2010]:	(n=134)
❑ The time it takes to clean up and document for release	77.78%
❑ Dealing with questions from user about the code / software	51.85%
❑ Supporting others without getting credit / acknowledgement	44.78%
❑ Patents or IP constraints	40.00%

Reproducibility in Computer Science

Related Work from [Gollub et al. 2012]

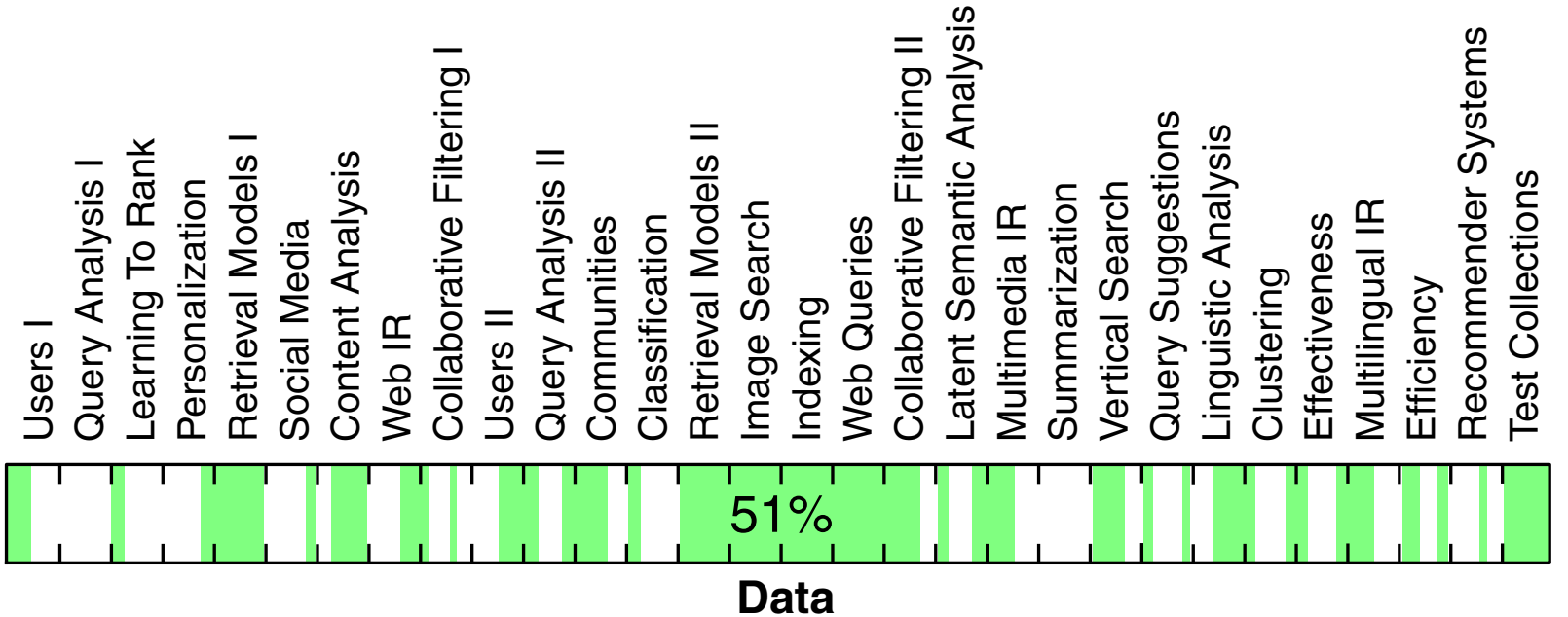


Image source: Gollub et al. 2012

- SIGIR 2011: 108 full papers, grouped by conference session
- Papers were analyzed regarding claims of availability, no attempts were made at downloading the mentioned assets

Reproducibility in Computer Science

Related Work from [Gollub et al. 2012]

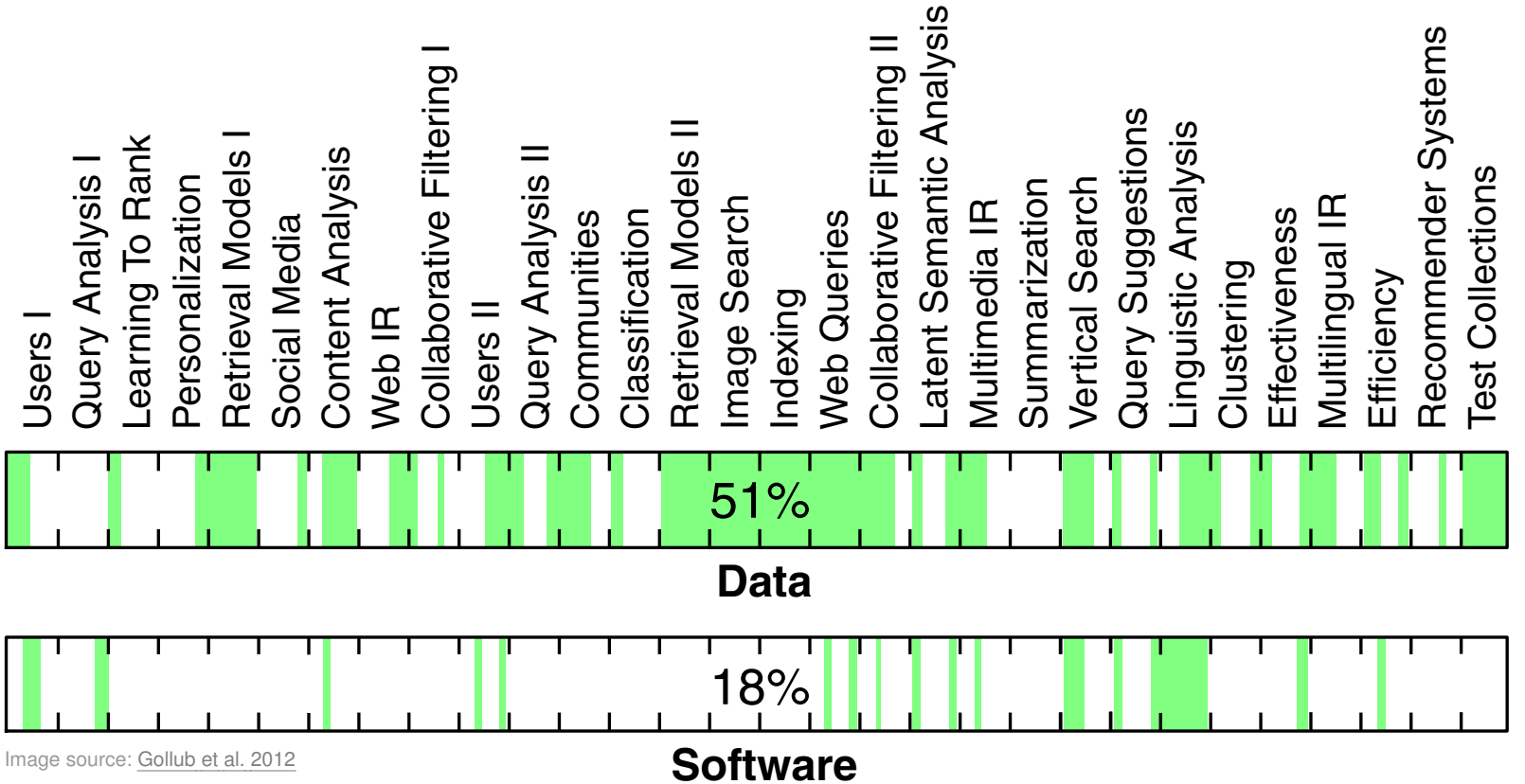


Image source: Gollub et al. 2012

- SIGIR 2011: 108 full papers, grouped by conference session
- Papers were analyzed regarding claims of availability, no attempts were made at downloading the mentioned assets

Reproducibility in Computer Science

Related Work from [Collberg et al. 2015]

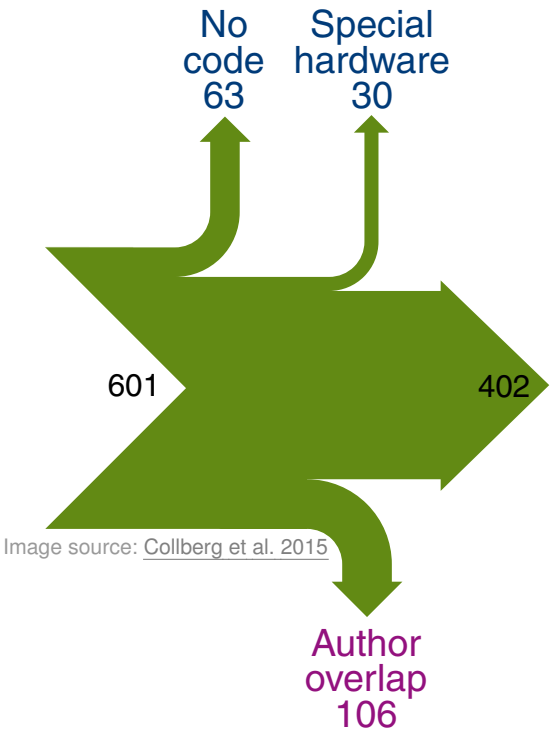


Image source: Collberg et al. 2015

- Papers sampled from 8 ACM conferences, and 5 journals

Reproducibility in Computer Science

Related Work from [Collberg et al. 2015]

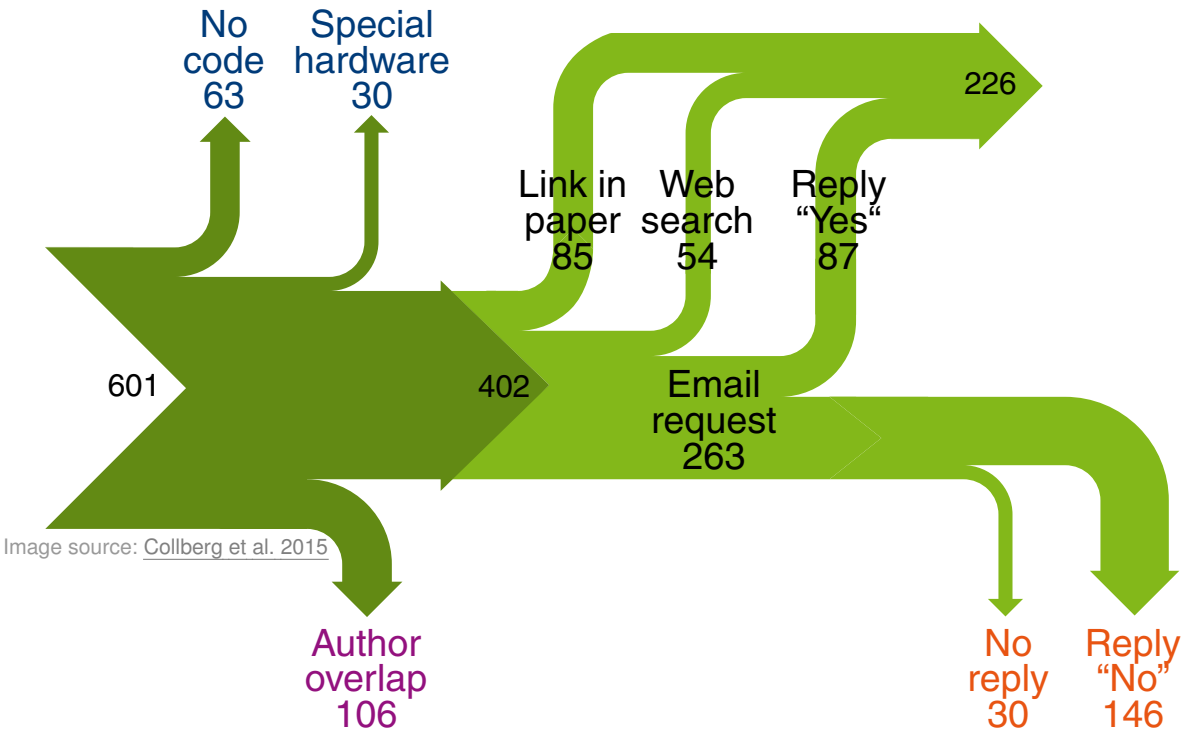
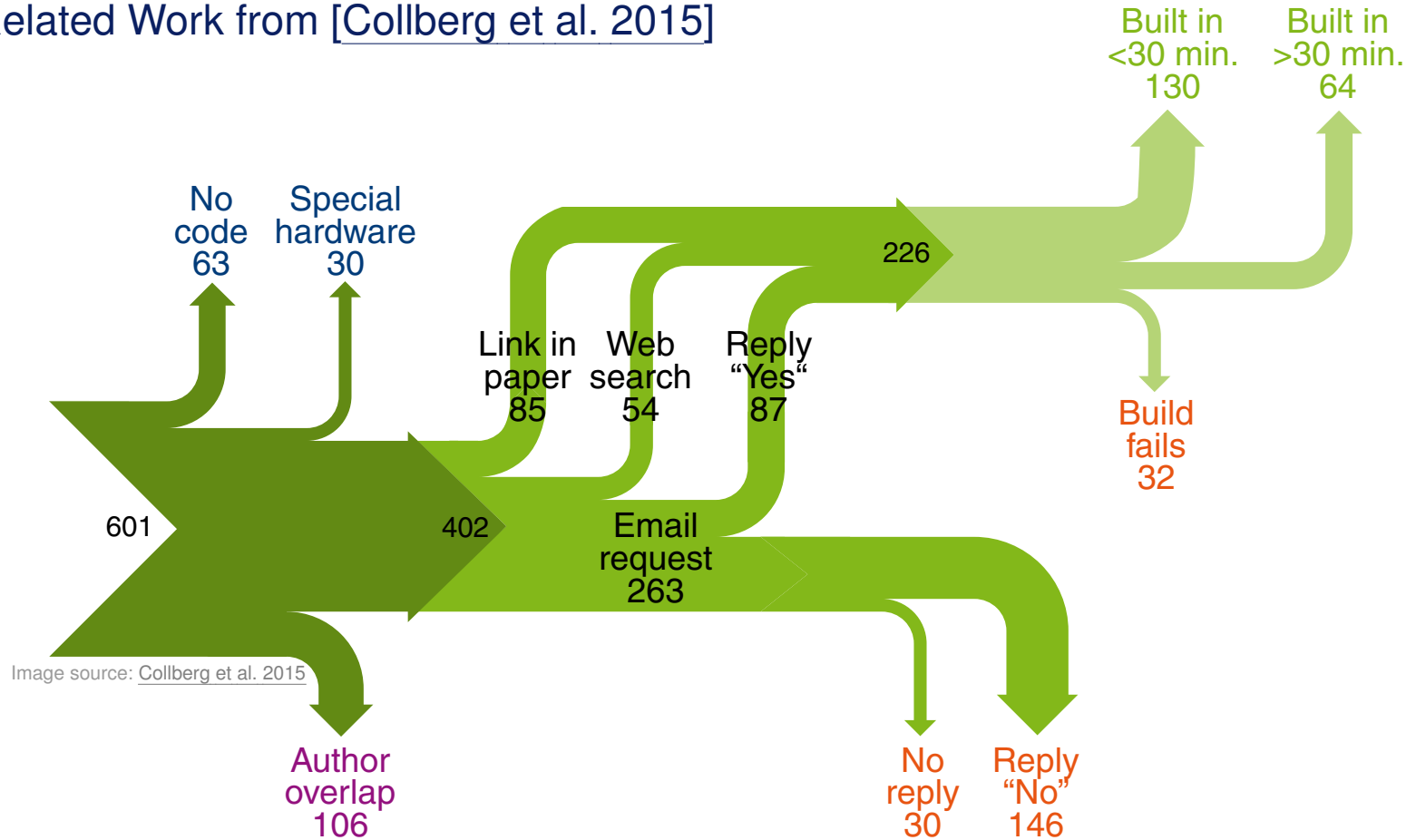


Image source: Collberg et al. 2015

- Papers sampled from 8 ACM conferences, and 5 journals
- About 21% of applicable papers contained link to code

Reproducibility in Computer Science

Related Work from [Collberg et al. 2015]



- Papers sampled from 8 ACM conferences, and 5 journals
- About 21% of applicable papers contained link to code
- Code successfully built for about 48% of applicable papers

Our Reproducibility Study

Notebooks from a Shared Task

Our Reproducibility Study

Notebooks from a Shared Task

What we did:

- Reproduced three selected Tweet sentiment classifiers from SemEval 2013
- Trained an ensemble classifier

Our Reproducibility Study

Notebooks from a Shared Task

What we did:

- Reproduced three selected Tweet sentiment classifiers from SemEval 2013
- Trained an ensemble classifier

Why particularly this task?

- Related investigations, where this task came up
- Focus on software, since data sets and performance measures are fixed

Why notebooks?

- Not all shared task notebooks are flawless
- Shared task results are frequently cited

Our Reproducibility Study

Notebooks from a Shared Task

What we did:

- Reproduced three selected Tweet sentiment classifiers from SemEval 2013
- Trained an ensemble classifier

Why particularly this task?

- Related investigations, where this task came up
- Focus on software, since data sets and performance measures are fixed

Why notebooks?

- Not all shared task notebooks are flawless
- Shared task results are frequently cited

Reproducibility approach:

- Create low bias situation toward originals
- Reproduce rather than replicate
- Maximize performance, where possible

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility

- Replicability: same input and same method ⇒ same output

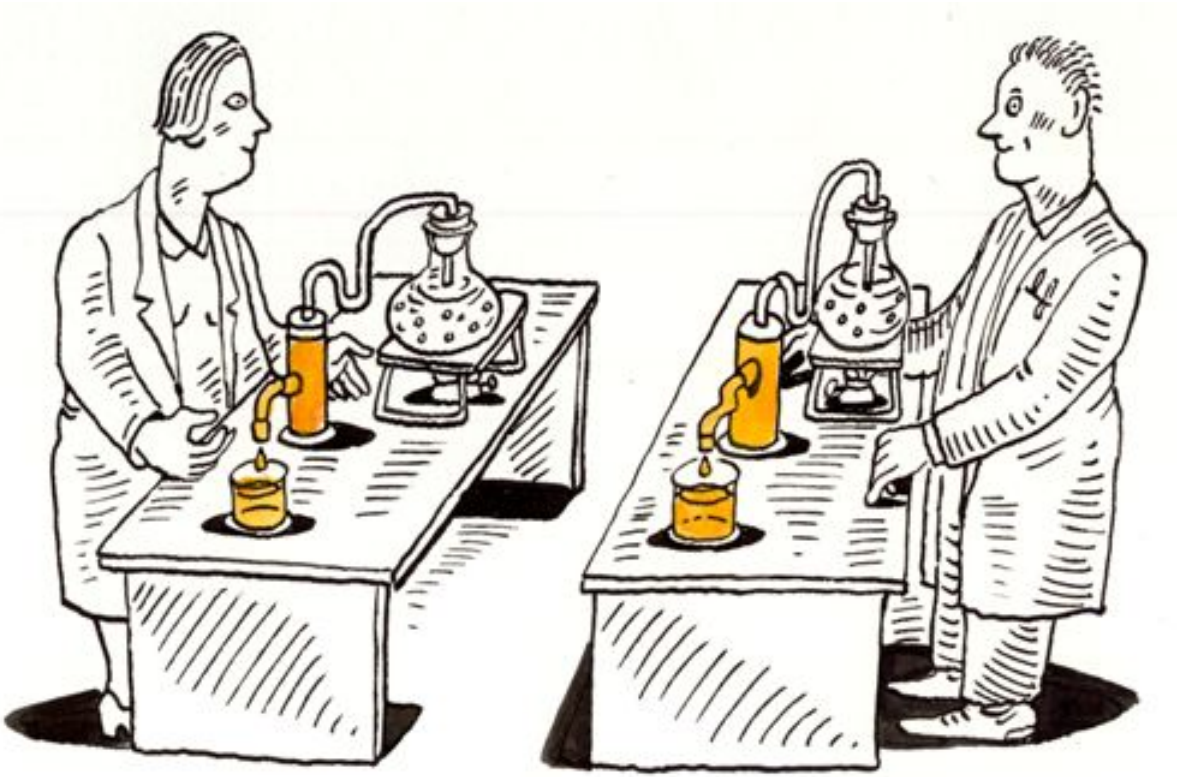


Image source: F1000Research.com, 2014

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility

- Replicability: same input and same method ⇒ same output
- Reproducibility: similar input and equivalent method ⇒ comparable output

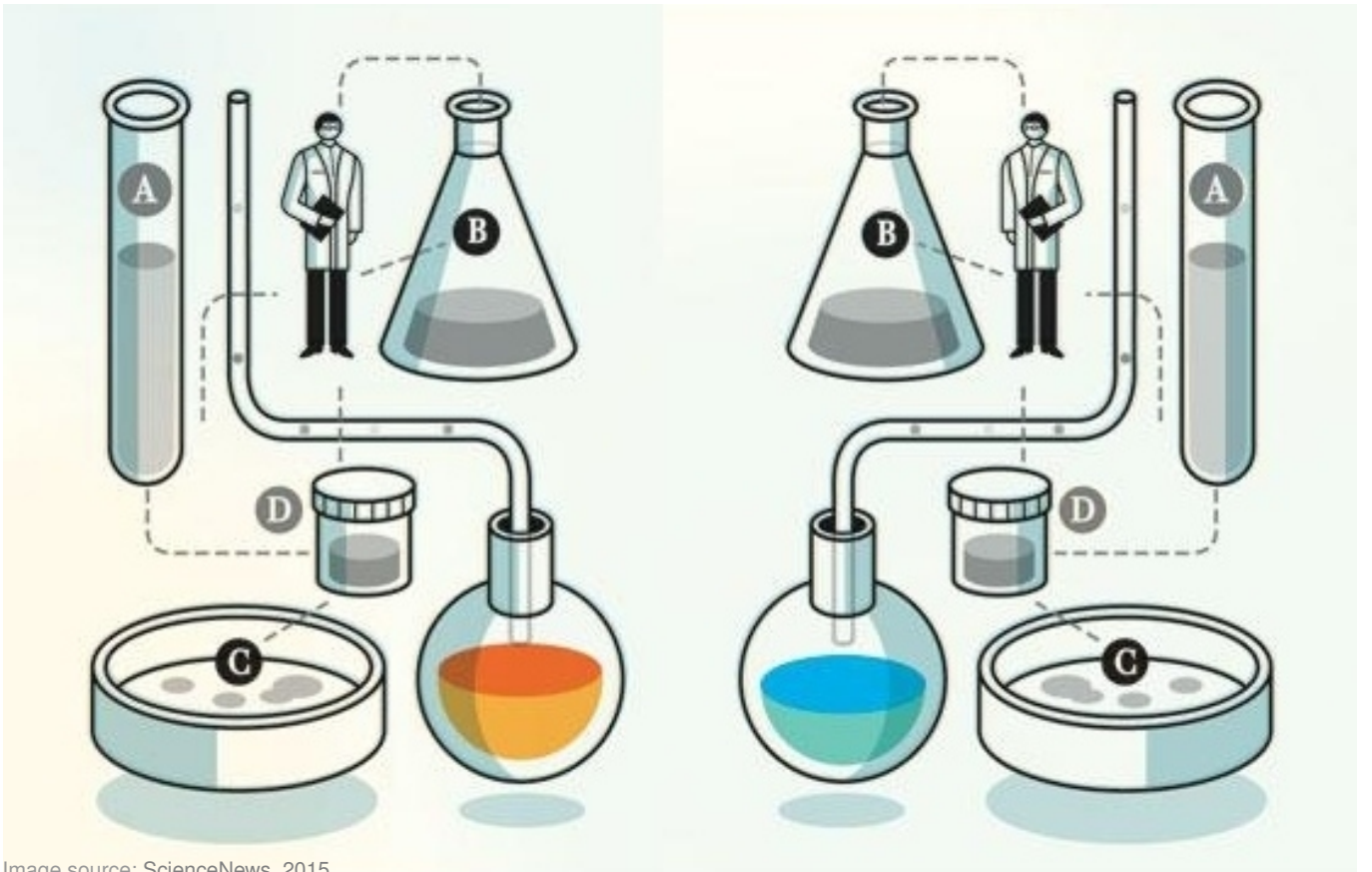


Image source: ScienceNews, 2015

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility

- Replicability: same input and same method $\Rightarrow$ same output
- Reproducibility: similar input and equivalent method $\Rightarrow$ comparable output
- What should we do in case of failure?

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility

- Replicability: same input and same method $\Rightarrow$ same output
- Reproducibility: similar input and equivalent method $\Rightarrow$ comparable output
- What should we do in case of failure?

Image source: The Economist, 2013

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility vs. Improvability

- Replicability: same input and same method $\Rightarrow$ same output
- Reproducibility: similar input and equivalent method $\Rightarrow$ comparable output
- Improvability: same input and better method $\Rightarrow$ better output
  (may include improvisation)

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility vs. Improvability

- Replicability: same input and same method $\Rightarrow$ same output
- Reproducibility: similar input and equivalent method $\Rightarrow$ comparable output
- Improvability: same input and better method $\Rightarrow$ better output
  (may include improvisation)

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility vs. Improvability

- Replicability: same input and same method $\Rightarrow$ same output
- Reproducibility: similar input and equivalent method $\Rightarrow$ comparable output
- Improvability: same input and better method $\Rightarrow$ better output
  (may include improvisation)

- Order of degrees of freedom:

$$\text{improve} \succ \text{reproduce} \succ \text{replicate}$$

- What shared asset helps:

$$\text{source code} \succ \text{library or API} \succ \text{demo or web service}$$

An Ongoing Debate in Science Reproducibility

Replicability vs. Reproducibility vs. Improvability

- Replicability: same input and same method $\Rightarrow$ same output
- Reproducibility: similar input and equivalent method $\Rightarrow$ comparable output
- Improvability: same input and better method $\Rightarrow$ better output
  (may include improvisation)

- Order of degrees of freedom:

$$\text{improve} \succ \text{reproduce} \succ \text{replicate}$$

- What shared asset helps:

$$\text{source code} \succ \text{library or API} \succ \text{demo or web service}$$

- Possible code of conduct for reproducing research:
  1. Try to improve it (e.g., by adding your own expertise and experience)
  2. Try to reproduce it with variations (e.g., different domains of application)
  3. As a last resort, replicate it, following the original to the letter

Our Reproducibility Study

Remarks on Reproducing the Selected Appraoches

Our Reproducibility Study

Remarks on Reproducing the Selected Appraoches

Selection criteria:

- High performance at SemEval 2013: NRC-Canada, GU-MLT-LT, KLUE
- Complementary approaches (i.e., not simply the top three)

Our Reproducibility Study

Remarks on Reproducing the Selected Appraoches

Selection criteria:

- High performance at SemEval 2013: NRC-Canada, GU-MLT-LT, KLUE
- Complementary approaches (i.e., not simply the top three)

Notable improvments / improvisations:

- All: feature descriptions generally very terse
- All: unification of Tweet normalization procedures
- All: L2-regularized logistic regression instead of original learning algorithms
- All: different parameter settings per approach
- KLUE: creation of our own emoticon polarity dictionary
- KLUE: unification from word frequency to Boolean occurrence

Our Reproducibility Study

Remarks on Reproducing the Selected Appraoches

Selection criteria:

- High performance at SemEval 2013: NRC-Canada, GU-MLT-LT, KLUE
- Complementary approaches (i.e., not simply the top three)

Notable improvments / improvisations:

- All: feature descriptions generally very terse
- All: unification of Tweet normalization procedures
- All: L2-regularized logistic regression instead of original learning algorithms
- All: different parameter settings per approach
- KLUE: creation of our own emoticon polarity dictionary
- KLUE: unification from word frequency to Boolean occurrence

Performance comparison:

Team	Original SemEval 2013	Reimplementation	Delta
NRC-Canada	69.02	69.44	+0.42
GU-MLT-LT	65.27	67.27	+2.00
KLUE	63.06	67.05	+3.99

Our Reproducibility Study

Performance in the Context of SemEval

SemEval 2013	
Team	F1
Our ensemble	**71.09**
NRC-Canada	69.02
GU-MLT-LT	65.27
teragram	64.86
BOUNCE	63.53
KLUE	63.06
AMI&ERIC	62.55
FBM	61.17
AVAYA	60.84
SAIL	60.14
27 more ...	

SemEval 2014	
Team	F1
TeamX	70.96
coooolll	70.14
RTRGO	69.95
NRC-Canada	69.85
Our ensemble	**69.79**
TUGAS	69.00
CISUC KIS	67.95
SAIL	67.77
Swiss-Chocolate	67.54
Synalp-Empathic	67.43
40 more ...	

Our Reproducibility Study

Performance in the Context of SemEval

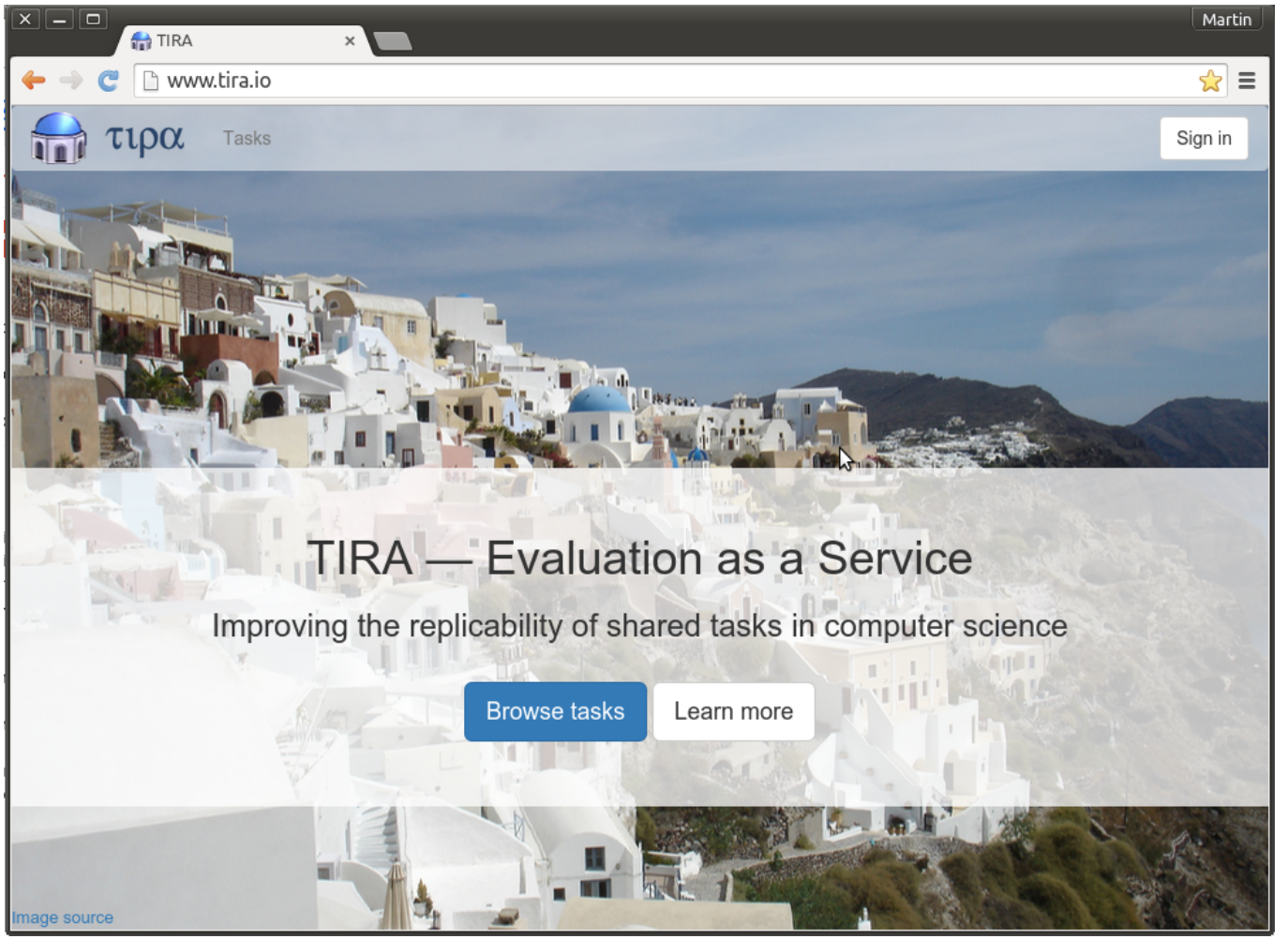
SemEval 2013	
Team	F1
Our ensemble	**71.09**
NRC-Canada	69.02
GU-MLT-LT	65.27
teragram	64.86
BOUNCE	63.53
KLUE	63.06
AMI&ERIC	62.55
FBM	61.17
AVAYA	60.84
SAIL	60.14
27 more ...	

SemEval 2014	
Team	F1
TeamX	70.96
coooolll	70.14
RTRGO	69.95
NRC-Canada	69.85
Our ensemble	**69.79**
TUGAS	69.00
CISUC KIS	67.95
SAIL	67.77
Swiss-Chocolate	67.54
Synalp-Empathic	67.43
40 more ...	

SemEval 2015	
Team	F1
Our ensemble	**64.84**
unitn	64.59
Isislif	64.27
INESC-ID	64.17
Splusplus	63.73
wxiaoac	63.00
IOA	62.62
Swiss-Chocolate	62.61
CLaC-SentiPipe	62.00
TwitterHawk	61.99
30 more ...	

- Adding TeamX pushes our ensemble to the top in 2015
- Refer to [Hagen et al. 2015] for details
- Task organizers should predict ensemble performance as a baseline

Evaluation as a Service using TIRA



[www.tira.io]

Summary & Conclusion

Summary:

- State-of-the-art Twitter sentiment detection approaches reproducible
- Our code is publicly available at GitHub: http://www.github.com/webis-de
- Neither of the existing approaches maximizes performance

Summary & Conclusion

Summary:

- State-of-the-art Twitter sentiment detection approaches reproducible
- Our code is publicly available at GitHub: http://www.github.com/webis-de
- Neither of the existing approaches maximizes performance

Take-home messages:

- Computer science can tackle reproducibility at a fundamental level
- Replicability vs. reproducibility lacks a third dimension: improvability
- Reproducibility should incorporate personal expertise and experience
- Sharing software may greatly improve aspects of reproducibility

Summary & Conclusion

Summary:

- State-of-the-art Twitter sentiment detection approaches reproducible
- Our code is publicly available at GitHub: http://www.github.com/webis-de
- Neither of the existing approaches maximizes performance

Take-home messages:

- Computer science can tackle reproducibility at a fundamental level
- Replicability vs. reproducibility lacks a third dimension: improvability
- Reproducibility should incorporate personal expertise and experience
- Sharing software may greatly improve aspects of reproducibility

Open questions ahead:

- What are the most worthy targets?
- What constitutes impact in reproducing science?
- Will sharing software become the norm?

Summary & Conclusion

Summary:

- State-of-the-art Twitter sentiment detection approaches reproducible
- Our code is publicly available at GitHub: http://www.github.com/webis-de
- Neither of the existing approaches maximizes performance

Take-home messages:

- Computer science can tackle reproducibility at a fundamental level
- Replicability vs. reproducibility lacks a third dimension: improvability
- Reproducibility should incorporate personal expertise and experience
- Sharing software may greatly improve aspects of reproducibility

Open questions ahead:

- What are the most worthy targets?
- What constitutes impact in reproducing science?
- Will sharing software become the norm?

Thank you for your attention!