Abstract

Write your abstract here...

Sammendrag

Skriv sammendrag her...

Preface

Write your preface here...

Acknowledgements (optional)

Table of Contents

List of Tables

List of Figures

Abbreviations

Symbol = definition

Chapter 1

Introduction

This chapter introduces the research performed and its results.

1.1 Background and Motivation

1.2 Background, Motivation and Problem Outline

1.3 Research Context

The research was conducted as my Master's thesis at the department of Computer and Information Science at the Norwegian University of Science and Technology. The research task was formulated by Odd Erik Gundersen, my supervisor, and is a continuation of previous work by Gundersen (2015) presented at 3DOR2015[1].

1.4 Hypothesis, Objectives and Research Questions

Underlying this thesis is the hypothesis that; *the documentation provided in experimental publications at AI conferences is not good enough to consider the experiments reproducible.*

Objective 1 *Evaluate the reproducibility of accepted papers to AI conferences.*

RQ1 What is the state of reproducibility at AI conferences?

Objective 2 *Recommend practices that could be adopted to aid the reproducibility of conference papers.*

RQ2 What is generally missing from AI papers to support reproducibility?

[1] `http://vc.ee.duth.gr/3DOR2015/`

RQ3 What can ease the documentation of missing information from conference papers?

1.5 Research Approach

1.6 Research Contributions

”We examine the common practices and challenges we see in recent OSN research, from which we propose a set of recommendations for the benefit of OSN researchers in all disciplines.”

C1: A survey of experimental research papers from AI conferences.

C2: An indication of the state of reproducibility at AI conferences.

C3: An approach to measure the reproducibility of AI conference papers.

1.7 Thesis Structure

Chapter 2

Reproducible Research

2.1 Terminology

2.2 Reproducible research

Reproducible research was coined as a term in Claerbout & Karrenbach (1992) as the ability to recreate published figures and results from their data, parameters and programs. Claerbout and his colleagues began publishing CD-ROMs containing the text of their books as interactive documents where figures could be rebuilt from its original data and code. With the growth of the Internet and the world-wide-web, Buckheit & Donoho (1995) began distributing a software package called WaveLab[1] as freeware. WaveLab contained the software environment necessary to reproduce figures and results from their papers, developed in part due to Jon Claerbout's recommendations for really reproducible computational research. Buckheit & Donoho (1995) emphasized the idea that code and data for the process behind a presentation of results is required with a slogan condensing Claerbout's ideas:

> *"An article about computational science in a scientific publication is* not *the scholarship itself, it is merely* advertising *of the scholarship. The actual scholarship is the complete set of instructions which generated the figures."*

Peng et al. (2006) propose reproducible research as a minimum standard in epidemiologic research, stating that full replication is not always feasible. They define replication as *"multiple independent investigators using independent data, analytical methods, laboratories, and instruments."* Criteria for research to be reproducible is defined as making analytical data, code underlying any principal results and the software environment available, and accompanied by adequate documentation to enable repeating the original and similar analyses. This is an important distinction, as reproducible research by this definition allows *"1) verifying published findings, 2) conducting alternative analyses of the*

[1] `https://statweb.stanford.edu/~wavelab/`

same data, 3) eliminating uninformed criticisms that do not stand up to existing data, and 4) expediting the interchange of ideas among investigators." (Peng et al. 2006) It does not, however, address issues prior to data analysis such as study design and data generation. As discussed in Vandewalle et al. (2007), open access benchmark problems and data sets fit well in with the ideas and concepts of reproducible research. Notably, benchmarks for competitions do no allow the designers to choose the test set in a biased way, potentially addressing some of the issues not covered by reproducible research.

Stodden V. C. 2010 Reproducible research: Addressing the need for data and code sharing in computational science yale law school roundtable on data and code sharing. Drummond 2012 Reproducible Research: a Dissenting Opinion

(Dror G. Feitelson. 2015. From Repeatability to Reproducibility and Corroboration. SIGOPS Oper. Syst. Rev. 49, 1 (January 2015), 3-11. DOI: http://dx.doi.org/10.1145/2723872.2723875)

2.3 Empirical Studies into Reproducibility

Collberg et. al. (2014) Repeatability and Benefaction in Computer Systems Research http://reproducibility.cs.arizona.edu/

J. Kovacevic, "How to Encourage and Publish Reproducible Research," 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, Honolulu, HI, 2007, pp. IV-1273-IV-1276. https://doi.org/10.1109/ICASSP.2007.367309 - "Reproducible research (RR) refers to the idea that in computational sciences, the ultimate product is not a published paper but rather the entire environment used to produce the results in the paper (data, software, etc.)."

P. Vandewalle, J. Kovacevic and M. Vetterli, "Reproducible research in signal processing," in IEEE Signal Processing Magazine, vol. 26, no. 3, pp. 37-47, May 2009. https://doi.org/10.1109/MSP.2009.932122

Victoria C. Stodden, 2010, The Scientific Method in Practice: Reproducibility in the Computational Sciences, Columbia University Academic Commons, http://hdl.handle.net/10022/AC:P:1141

2.4 Observations

Reproducible research provides readers the ability to verify findings presented in a publication rather than blindly trusting the results shown. Additionally, the more detailed documentation of the research allows thorough comparisons to corroborating research and attempts to independently implement experiments and methods. Thereby facilitating further research through more robust research and further corroboration of findings. Our approach to reproducible research focus on openness and transparency in methods and experiments, to allow review and discussion of research.

Chapter 3

Research Method

The survey designed to investigate the state of reproducibility at AI conferences is based on manually analysing research papers and marking Investigating the reproducibility of research papers has previously been done by attempting to run experiments with Analysing the reproducibility of experiments based on a paper have previously been done

3.1 Literature Survey Design

Advantages of doing a survey - Can be replicated on similar documents or on original documents provided the method is shared and documents are accessible - Can produce a lot of data at a low cost, in a relatively short time compared to attempting full replications of experiments - Allows a larger sample population due to the shorter time necessary to evaluate a paper

Disadvantages - The depth is restricted, does not provide detail on the research topic - Focuses on what can be counted and measured, other aspects may be overlooked

3.1.1 Data requirements

The following variables were recorded for each paper. They are here divided into two sections: directly, and indirectly topic related. The directly topic related variables are categorised into: Experiment documentation, documentation and availability of the experiment; Method documentation, documentation and availability of the method presented; and Open Data, documentation and availability of data used. The indirectly topic related variables are categorised into: Miscellaneous, meta-data describing the research; and Research Transparency, documentation and openness of the research method.

Directly topic related

Experiment documentation : How well documented is the experiment and the environment it was performed in.

Open experiment code Is the code to run the experiment made available?

Hardware specification Is the hardware used during the experiment specified?

Software dependencies Are software dependencies listed?

Experiment set-up Is the set-up for the experiment described? Are hyper-parameters used during the experiment specified?

Evaluation criteria Are the criteria used to evaluate the method described?

/* Explain variables and relate to best practices!!! */

Directly topic related : Is source code or data open for the experiment and method? Is the method documented? Is the experiment documented? etc.

Indirectly topic related Research transparency (hypothesis, predictions...) Author affiliation (uni/industry/both) Novel research? Conference view on supplementary material Theoretical / Experimental research

Possible analysis patterns

1. reproducibility related to author affiliation
2. reproducibility related to conference view on supplementary material?
3. reproducibility related to publishing year (improvement over time?)

3.1.2 Data generation method

Data will be generated by evaluating conference papers openly published in proceedings from two instalments of two different conferences. Physical copies can be ordered from the conferences, but all accepted papers are freely available on-line. This makes them easily available, and unobtrusive to obtain. Additionally, it allows other researchers to scrutinize the research based on original material.

The conferences investigated were the International Joint Conference on Artificial Intelligence (IJCAI) and the AAAI Conference on Artificial Intelligence (AAAI), specifically IJCAI-2013 and -2016, and AAAI-2014 and -2016. From these four instalments there are a combined population of 1910 accepted papers. A sample size of 100 from each conference was selected, restricting the necessary time to conduct the survey. Confidence intervals are reported in table 3.1. Probabilistic random sampling of each conference separately was done, as documented in Appendix A[1].

Sampling frame: accepted papers at IJCAI-13, -16 and AAAI-14 and -16 (can be seen in repo files for sample generation) Sampling technique: probabilistic random sampling of each conference separately. "Probability sampling, as its name suggests, means that the sample has been chosen because the researcher believes that there is a high probability that the sample of respondents (or events) chosen are representative of the overall population being studied. That is, they form a representative cross-section of the overall population." Oates p.96

[1]The sampling procedure is also available in a Jupyter notebook here: `https://github.com/sidgek/msoppgave`

Sample size: 100 for each conference, restricts the necessary time to conduct the survey while still providing informative accuracy ranges when considering previous research (cite?)

Conference	Population Size	Sample Size	Confidence Interval
AAAI 2014	398	100	8.49
AAAI 2016	548	100	8.87
IJCAI 2013	413	100	8.54
IJCAI 2016	551	100	8.87
Combined	1910	400	4.36

Table 3.1: Confidence intervals of survey sample populations given a 50/50 yes/no split with confidence level of 95%. (`https://www.surveysystem.com/sscalc.htm`)

The four conferences cover several disciplines within AI, and there may be differences within the disciplines. Between the conferences, however, it is assumed that the populations are not significantly different. This is based on the conferences covering the same disciplines at large, and employing blind peer review for acceptance. None of the conferences are vocal about open source or reproducible research, though the AAAI conferences allow non-reviewed supplemental material provided the documentation relevant for any claims is present in the paper itself. The confidence interval for each conference is reported in table 3.1, assuming that the populations are similar, a combined confidence interval is reduced to 4.36%.

3.2 Evaluation Procedure

- Step by step 'instructions' - Sampling documentation - Evaluation documentation - Example evaluations (variable X: "Exhibit A" covers, "Exhibit B" is not enough)

3.3 Limitations of the Survey

- Evaluation bias (modification of variables) - Sample inconsistency for IJCAI-13 (50 papers) - Not an actual attempt at reproducing experiments, researcher's view that discussion of a variable is missing?

Chapter 4

Results and Analysis

Results are presented in the following chapter. The chapter is separated into sections in line with the categories defined in section 3.1.1, in the following order: miscellaneous, research transparency, method documentation, experiment documentation, and open data. At the end a section on researcher error is included.

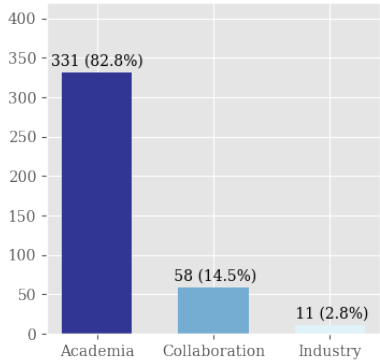
4.1 Miscellaneous

Variables in the miscellaneous category describe the research and include the following variables: affiliation, conference, research type, result outcome, and third-party citation. The data for each variable except conference can be seen in figure 4.1. The conference distribution is seen in table 4.1.

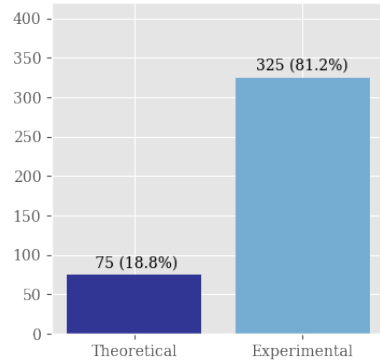
Note that the Third-party citation data in figure 4.1d and Result outcome data in figure 4.1c should not be used for analysis due to researcher error as discussed in section 4.6, but is presented for completeness.

Conference	Papers
AAAI 14	100
AAAI 16	100
IJCAI 13	100
IJCAI 16	100

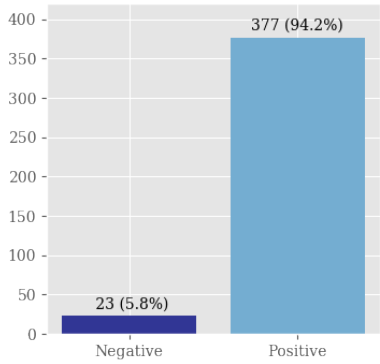
Table 4.1: Distribution of papers between conferences.



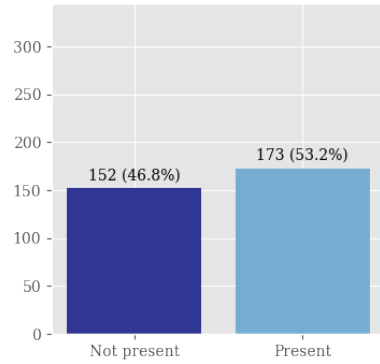
(a) Affiliation



(b) Research type



(c) Result outcome

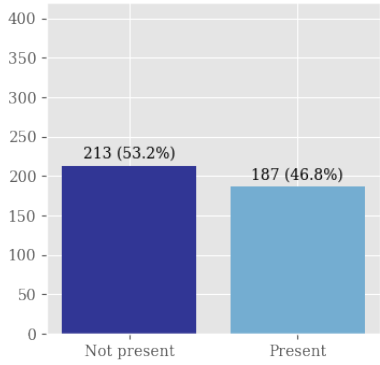


(d) Third-party citations

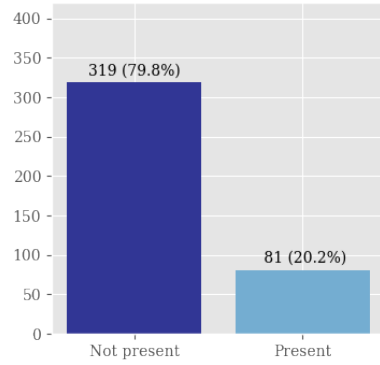
Figure 4.1: Summary of miscellaneous data for all 400 papers. Note that for Third-party citation only the 325 experimental papers are relevant, which accounts for the lower values on the left axis.

4.2 Research Transparency

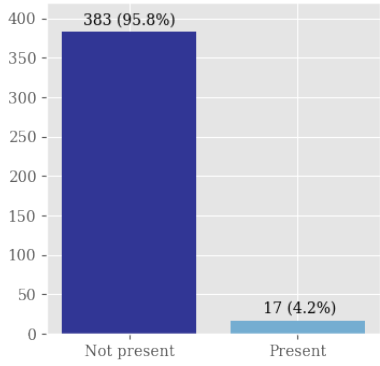
Research transparency variables describe how well the research method is documented. This includes explicit mentions of: contribution, research goal or objective, hypothesis, prediction, problem description, research method, and research question. The distributions for each variable can be seen in figure 4.2 and 4.3.



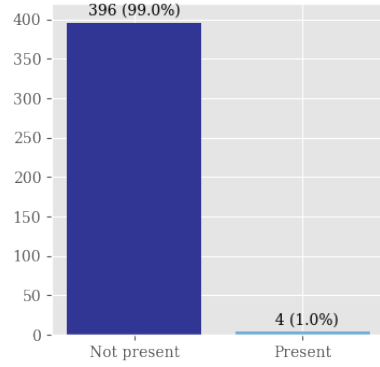
(a) Contribution



(b) Goal or objective

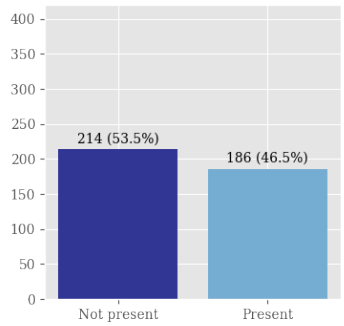


(c) Hypothesis

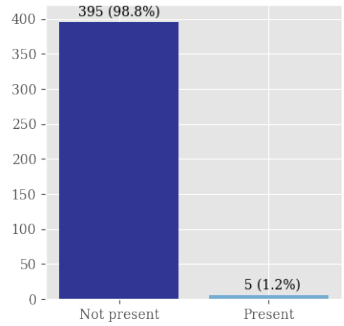


(d) Prediction

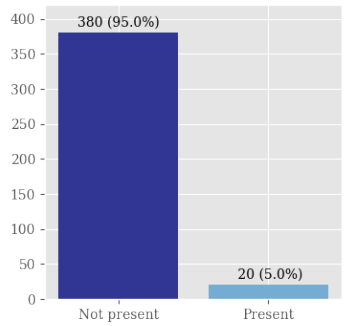
Figure 4.2: Summary of data on research transparency. A term is *Present* if it is explicitly mentioned in a paper. These variables are applicable to all 400 papers.



(a) Problem description



(b) Research method

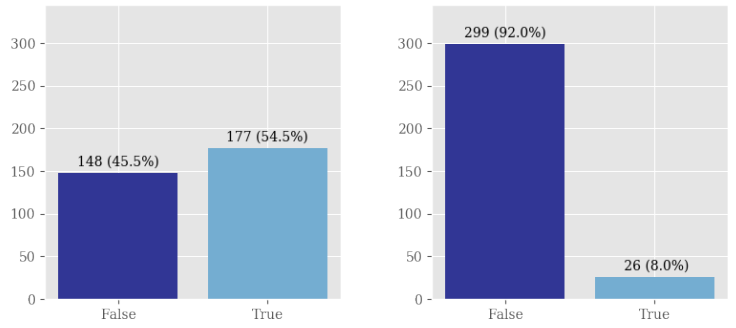


(c) Research question

Figure 4.3: Continued summary of data on research transparency. A term is *Present* if it is explicitly mentioned in a paper. These variables are applicable to all 400 papers.

4.3 Method Documentation

The method documentation category investigates the availability of the method under investigation through the pseudocode, and open source code variables. Only the 325 experimental papers are relevant for these variables, as seen by the lower values on the left axis compared to the transparency data. The data is summarised in figure 4.4.



(a) Pseudocode

(b) Open source code

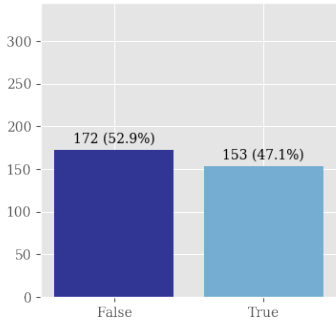
Figure 4.4: Summary of data for the method documentation category. These variables are only applicable to the 325 experimental research papers.

4.4 Experiment Documentation

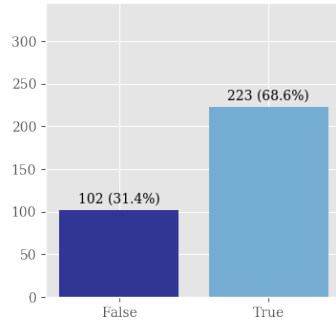
Experiment documentation variables relate to how well the experiment is documented and if it is made available. The following variables are included: evaluation criteria, experiment set-up, hardware specification, open experiment code and software dependencies. A summary of the data can be seen in figure 4.5. As in the method documentation category, only the experimental papers are relevant.

4.5 Open Data

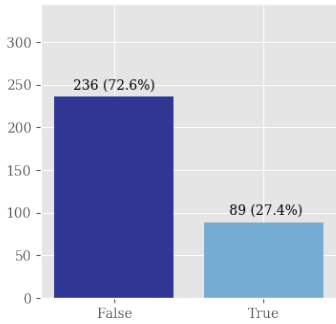
Open Data relates to the availability of data used during an experiment and documentation of dataset splits. The following variables are included: training data, validation data, test data, and results data. Figure 4.6 summarizes the results for experimental papers.



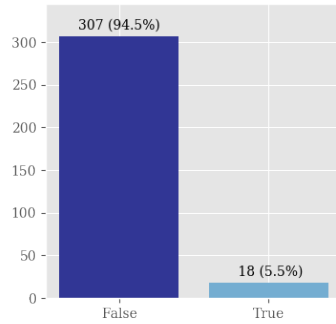
(a) Evaluation criteria



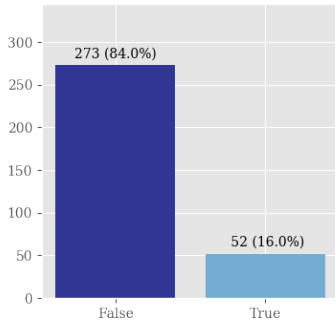
(b) Experiment set-up



(c) Hardware specification

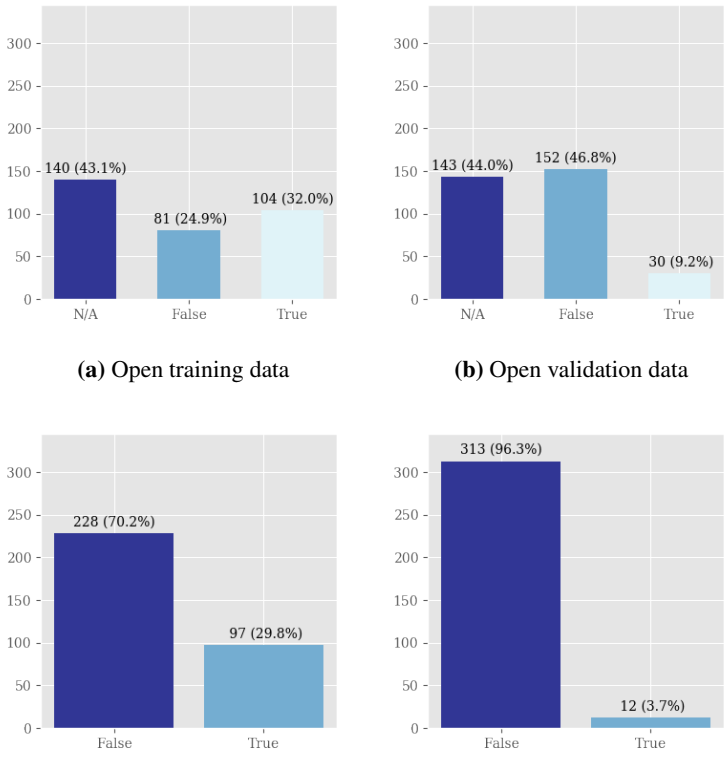


(d) Open experiment code



(e) Software dependencies

Figure 4.5: Summary of data from the experiment documentation category. These variables are only applicable to the 325 experimental research papers.



(a) Open training data

(b) Open validation data

(c) Open test data

(d) Open results data

Figure 4.6: Summary of the open data category. These variables are only applicable to the 325 experimental research papers.

4.6 Researcher Error

The data for Third-party citations and Result outcome have been excluded from further analysis due to inaccurate data and researcher error, respectively.

For Third-party citations, the intent was to record citations of software and data used for an experiment. This intent was to investigate if public datasets and published source code is cited when used by other researchers. For the most part, the papers noted with *Present* in figure 4.1d show correct citations to public datasets. However, it is difficult to say what portion of the *Not present* papers have used public datasets or source code, or only mentioned them by name without a correct citation.

Result outcome (figure 4.1c) was erroneously recorded as a positive result, instead of a notion of the novelty of the research. This would be any paper that presents confirmation of a hypothesis, or where the wording of their findings present a solution or improvement to something. Since very few papers include a hypothesis in the first place, the data for this variable is excluded from any further analysis.

4.7 Patterns of Analysis Revisited

Author affiliation

Conference view on supplemental material

Changes over time

Chapter 5

Discussion

Anonymous publication of source code and data along with papers (blind review)

Hunold and Trff 2013: Add a description of how to reproduce the findings in a publication

ACM TOMS: Independent replication review http://toms.acm.org/replicated-computational-results.cfm

Chapter 6

Conclusion

Bibliography

Buckheit, J. B. & Donoho, D. L. (1995), Wavelab and reproducible research, *in* 'Wavelets and Statistics', Springer New York, New York, NY, pp. 55–81.

Claerbout, J. F. & Karrenbach, M. (1992), Electronic documents give reproducible research a new meaning, *in* 'SEG Technical Program Expanded Abstracts 1992', Society of Exploration Geophysicists, pp. 601–604.

Gundersen, O. E. (2015), Towards Scientific Benchmarks: On Increasing the Credibility of Benchmarks, *in* I. Pratikakis, M. Spagnuolo, T. Theoharis, L. V. Gool & R. Veltkamp, eds, 'Eurographics Workshop on 3D Object Retrieval', The Eurographics Association.

Peng, R. D., Dominici, F. & Zeger, S. L. (2006), 'Reproducible epidemiologic research', *American Journal of Epidemiology* **163**(9), 783.

Vandewalle, J., Suykens, J., Moor, B. D. & Lendasse, A. (2007), State-of-the-art and evolution in public data sets and competitions for system identification, time series prediction and pattern recognition, *in* '2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07', Vol. 4, pp. IV–1269–IV–1272.

Appendix

Appendix A: Population selection

paper_selection

June 10, 2017

0.1 Sampling of papers from conferences

This Jupyter notebook shows the procedure used to select a sub-sample of the accepted papers from each conference.

0.1.1 Accepted conference papers

Sampling of papers is based on the listing of accepted papers at the following locations:
AAAI-14 http://www.aaai.org/Library/AAAI/aaai14contents.php
AAAI-16 http://www.aaai.org/Library/AAAI/aaai16contents.php
IJCAI-13 http://ijcai-13.org/program/accepted_papers
IJCAI-16 http://ijcai-16.org/index.php/welcome/view/accepted_papers
These listings were used to generate the files available in the ../data/ folder. Each conference is represented by a textfile containing the papers accepted to the conference's main and special tracks. Each line in the textfiles represent a paper, including its title and the authors. Example:

```
Causality based Propagation History Ranking in Social Networks  Zheng Wang, Chaokun Wang, Jishen
Intervention Strategies for Increasing Engagement in Volunteer-Based Crowdsourcing  Avi Segal, K
```

Papers are available through AAAI Publications for all but IJCAI-16 (at the time of writing):
AAAI-14 http://www.aaai.org/ocs/index.php/AAAI/AAAI14/schedConf/presentations
AAAI-16 http://www.aaai.org/ocs/index.php/AAAI/AAAI16/schedConf/presentations
IJCAI-13 http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/schedConf/presentations
For IJCAI-16, see the proceedings at: http://www.ijcai.org/Proceedings/2016
First, the accepted papers are loaded from files.

```
In [1]: from glob import glob

        accepted_papers = {}
        track_files = glob('../data/accepted*'.format(dir))
        for file in track_files:
            conference = file.split('_')[-1]
            accepted_papers[conference] = []
            with open(file, 'r') as f:
                for line in f:
                    accepted_papers[conference].append(line)
```

The resulting dictionary accepted_papers contains a list of the accepted papers for each conference.

```
In [2]: for conference, papers in sorted(accepted_papers.items()):
            print('{conference} includes {papers} accepted papers.'.format(conference=conference
```

```
aaai-14 includes 398 accepted papers.
aaai-16 includes 548 accepted papers.
ijcai-13 includes 413 accepted papers.
ijcai-16 includes 551 accepted papers.
```

0.1.2 Selection

A sample population of 100 papers is selected from each conference using Python's pseudo-random number module. As per the documentation on random.sample "*The resulting list is in selection order so that all sub-slices will also be valid random samples.*" The seed is set to the unix timestamp for Jan 10 14:46:40 2017 UTC: 1484059600.

```
In [3]: import random
        random.seed(1484059600)

        k = 100
        samples = {}

        # The order is set explicitly due to originally not sorting accepted_papers.items().
        conferences = ['aaai-16', 'aaai-14', 'ijcai-13', 'ijcai-16']

        for conference in conferences:
            samples[conference] = random.sample(accepted_papers[conference], k)
```

Note that when originally generating the samples, the dictionary was iterated by the use of Python 3's dict.items() view. The order is not guaranteed, and I forgot to sort the iteration so repeated runs of the code would generate the same populations. Therefore, the order has to be set explicitly as above to generate the original populations.

The generated random samples are permanently stored to files in the ../data/ directory (Github: https://github.com/sidgek/msoppgave/tree/master/data/.

```
In [4]: for conference, papers in samples.items():
            outputfile = '../data/sampled_{conference}'.format(conference=conference)
            with open(outputfile, 'w') as f:
                for line in papers:
                    f.write(line)
```

Appendix B

Survey data (and population samples)

Appendix C: Analysis code

analysis

June 10, 2017

1 Evaluation analysis

We will be taking a look at the evaluations from the data folder ../data/ (notebook, github).

1.1 Setup

Before looking at the data, a list of imports and the version of libraries used is reported.

```
In [1]: # Built-in python libraries
        import platform
        from glob import glob
        from itertools import chain

        # 3rd-party libraries
        import matplotlib
        import matplotlib.pyplot as plt
        import numpy as np
        import pandas as pd
        import IPython
        from IPython.utils.coloransi import TermColors

        # Print versions.
        print('Python version: {}'.format(platform.python_version()))
        print('IPython version: {}'.format(IPython.__version__))
        print('matplotlib version: {}'.format(matplotlib.__version__))
        print('numpy version: {}'.format(np.__version__))
        print('pandas version: {}'.format(pd.__version__))


        # Initialize the backend for Jupyter
        %matplotlib notebook

        # Set style-sheet to grayscale.
        matplotlib.style.use('ggplot')
        colormap = plt.cm.get_cmap('RdYlBu_r')
        C = [colormap(x/5) for x in range(5)]
        # Set figure font to serif.
```

```
        plt.rcParams['font.family'] = 'serif'

        # Set how many columns to show in tables.
        pd.options.display.max_columns = 50
        pd.options.display.max_rows = 400
        # Set the format to print float values to 3 decimal points.
        pd.options.display.float_format = lambda x: '%.3f' % x
```

```
Python version: 3.6.1
IPython version: 5.3.0
matplotlib version: 2.0.2
numpy version: 1.12.1
pandas version: 0.20.1
```

2 The data

First we load the CSV file into a pandas DataFrame, print the amount of samples and take a look at the column headers of the dataset.

```
In [2]: file = '../data/evaluations.csv'

        conversion_dict = {'research_type': lambda x: int(x == 'E')}

        evaluation_data = pd.read_csv(file, sep=',', header=0, index_col=0, converters=conversio

        print('Amount of samples: {}'.format(len(evaluation_data.index)))

        column_headers = evaluation_data.columns.values
        print('\nColumn headers: {}'.format(column_headers))
```

```
Amount of samples: 400

Column headers: ['title' 'research_type' 'result_outcome' 'affiliation'
 'problem_description' 'goal/objective' 'research_method'
 'research_question' 'hypothesis' 'prediction' 'contribution' 'pseudocode'
 'open_source_code' 'open_experiment_code' 'train' 'validation' 'test'
 'results' 'hardware_specification' 'software_dependencies'
 'third_party_citation' 'experiment_setup' 'evaluation_criteria' 'authors'
 'link' 'comments' 'conference']
```

There are 400 samples with 27 columns in total for each sample. However, some columns are not necessary for further analysis: *title, authors, link, comments*. The *comments* column contains short messages such as *"Points to an extended paper"* or *"Links to appendix which links to code"* to give extra information in case an evaluation is unclear. The other three identify which paper was evaluated. These columns are therefore removed from the dataframe.

```
In [3]: evaluation_data.drop(['title', 'link', 'authors', 'comments'], axis=1, inplace=True)
        column_headers = evaluation_data.columns.values
        print('\nColumn headers: {}'.format(column_headers))
```

```
Column headers: ['research_type' 'result_outcome' 'affiliation' 'problem_description'
 'goal/objective' 'research_method' 'research_question' 'hypothesis'
 'prediction' 'contribution' 'pseudocode' 'open_source_code'
 'open_experiment_code' 'train' 'validation' 'test' 'results'
 'hardware_specification' 'software_dependencies' 'third_party_citation'
 'experiment_setup' 'evaluation_criteria' 'conference']
```

The remaining 23 columns can be placed in more clarifying categories. All data is boolean with the value 0 or 1, unless otherwise specified below.

Miscellaneous Variables describing the research
research_type - Experimental (1) or theoretical (0).
result_outcome - Novel research or not.
affiliation - The affiliation of the authors; academia (0), collaboration (1), industry (2).
conference - The conference the paper was accepted to.
third_party_citation - Is third-party source code or data referenced?

Research Transparency How well documented is the research method?
problem_description - The problem the research seeks to solve.
goal/objective - The objective of the research.
research_method - Research method used.
research_question - Research question(s) asked.
hypothesis - Investigated hypothesis.
prediction - Predictions related to the hypothesis.
contribution - Contribution of the research.
Note: The variables under Research Transparency are 1 if explicitly mentioned in the paper, otherwise 0.

Experiment Documentation How well is the experiment documented?
open_experiment_code - Is the experiment code available?
hardware_specification - Hardware used.
software_dependencies - For method or experiment.
experiment_setup - Is the experiment setup described with parameters etc.?
evaluation_criteria - Specification of evaluation criteria.

Method Documentation How well is the method under investigation documented?
pseudocode - Method described in pseudocode.
open_source_code - Is the method code available?

Open Data How well is the data documented, and is it available?
train - Training set specification.
validation - Validation set specification.
test - Test set specification.
results - Raw results data.
Note: If no data is open sourced all will be 0. If data is open source but the sets are not specified train or test will be set to 1 depending on whether the research requires training or not. If the research does not require training, train and validation does not have a value set.

```
In [4]: category_headers = {
            'Miscellaneous': np.append(column_headers[0:3], column_headers[[19, 22]]),
            'Research Transparency': column_headers[3:10],
            'Method Documentation': column_headers[10:12],
            'Open Data': column_headers[13:17],
            'Experiment Documentation': column_headers[[12, 17, 18, 20, 21]]
        }
```

A look at the first two samples of the dataset show the difference between experimental and theoretical papers.

```
In [5]: evaluation_data.head(2)
```

```
Out[5]:        research_type  result_outcome  affiliation  problem_description  \
        index
        1                  1               1            0                    1
        2                  0               1            0                    0

               goal/objective  research_method  research_question  hypothesis  \
        index
        1                   0                0                  0           0
        2                   0                0                  0           0

               prediction  contribution  pseudocode  open_source_code  \
        index
        1               0             1       1.000             0.000
        2               0             0         nan               nan

               open_experiment_code  train  validation  test  results  \
        index
        1                     0.000  1.000       1.000 0.000    0.000
        2                       nan    nan         nan   nan      nan

               hardware_specification  software_dependencies  third_party_citation  \
        index
        1                       0.000                  0.000                 0.000
        2                         nan                    nan                   nan

               experiment_setup  evaluation_criteria conference
        index
        1                 1.000                1.000   IJCAI 16
        2                   nan                  nan   IJCAI 16
```

The first sample is an experimental paper (**research_type=1**) and has values set for all the columns. The second paper, however, is a theoretical paper (**research_type=0**) and only has values set for the *Miscellaneous*, and *Research Transparency* categories, excluding the *third_part_citation* column. Note that the datafile has Experimental noted as E and theoretical noted as T.

Cells with missing values are represented as `NaN` in pandas and can be seen for all the columns exclusive to experimental papers in the second sample above. For experimental papers where

training is not relevant, both the *train* and *validation* columns will show as `NaN`. To add `NaN` to visualisations below, we fill them out with the value -1.

Additionally, we split the experimental papers into a separate dataframe for plotting later.

```
In [6]: evaluation_data = evaluation_data.fillna(-1)
        experimental_data = evaluation_data[evaluation_data.research_type == 1]
```

2.1 Miscellaneous

We start with the miscellaneous category, defining the plot function which will be used for all categories. The only variable not plotted is the `conference` variable, which has its frequencies printed out instead.

Variables describing the research
research_type - Experimental (1) or theoretical (0).
result_outcome - Novel research or not.
affiliation - The affiliation of the authors; academia (0), collaboration (1), industry (2).
conference - The conference the paper was accepted to.
third_party_citation - Is third-party source code or data referenced?

```
In [14]: def plot_full_series(series, title, labels, width=0.4):
             bins=len(labels)
             Y, X = np.histogram(series, bins=bins)
             total_Y = sum(Y)
             fig = plt.figure(figsize=(4,4))
             ax = plt.subplot(111)
             plt.bar(X[:-1], Y, color=C, width=width, axes=ax)
             ax.set_ylim(0, total_Y + 20)
             ax.set_xticks(X[:-1])
             ax.set_xticklabels(labels)
             #ax.set_title(title)

             # Add amount labels to bars
             for y, x in zip(Y, X[:-1]):
                 label = '{:3.0f} ({:.1%})'.format(y, y / total_Y)
                 ax.text(x, y + 5, label, ha='center', va='bottom')
             plt.show()
             fig.savefig('../doc/report/fig/{}'.format(title.replace(' ', '_')))
```

```
In [16]: print(evaluation_data.groupby('conference').size(), end='\n\n')

         plot_full_series(evaluation_data.affiliation, 'Affiliation', ['Academia', 'Collaboratio
         plot_full_series(evaluation_data.research_type, 'Research Type', ['Theoretical', 'Exper
         plot_full_series(evaluation_data.result_outcome, 'Result Outcome', ['Negative', 'Positi
         plot_full_series(experimental_data.third_party_citation, 'Third-party Citation', ['Not
```

```
conference
AAAI 14     100
AAAI 16     100
IJCAI 13    100
```

```
IJCAI 16    100
dtype: int64


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```

2.2 Research Transparency

How well documented is the research method?
problem_description - The problem the research seeks to solve.
goal/objective - The objective of the research.
research_method - Research method used.
research_question - Research question(s) asked.
hypothesis - Investigated hypothesis.
prediction - Predictions related to the hypothesis.
contribution - Contribution of the research.
Note: The variables under Research Transparency are 1 if explicitly mentioned in the paper, otherwise 0.

```
In [15]: plot_full_series(evaluation_data.contribution, 'Contribution', ['Not present', 'Present
         plot_full_series(evaluation_data['goal/objective'], 'Goal or Objective', ['Not present'
         plot_full_series(evaluation_data.hypothesis, 'Hypothesis', ['Not present', 'Present'])
         plot_full_series(evaluation_data.prediction, 'Prediction', ['Not present', 'Present'])
         plot_full_series(evaluation_data.problem_description, 'Problem Description', ['Not pres
         plot_full_series(evaluation_data.research_method, 'Research Method', ['Not present', 'P
         plot_full_series(evaluation_data.research_question, 'Research Question', ['Not present'
```

```
<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>

<IPython.core.display.Javascript object>

<IPython.core.display.HTML object>
```

2.3 Experiment Documentation

How well is the experiment documented?
    evaluation_criteria - Specification of evaluation criteria.
experiment_setup - Is the experiment setup described with parameters etc.?
hardware_specification - Hardware used.
open_experiment_code - Is the experiment code available?
software_dependencies - For method or experiment.

```
In [17]: plot_full_series(experimental_data.evaluation_criteria, 'Evaluation Criteria', ['False'
         plot_full_series(experimental_data.experiment_setup, 'Experiment Setup', ['False', 'Tru
         plot_full_series(experimental_data.hardware_specification, 'Hardware Specification', ['
         plot_full_series(experimental_data.open_experiment_code, 'Open Experiment Code', ['Fals
         plot_full_series(experimental_data.software_dependencies, 'Software Dependencies', ['Fa
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```

2.4 Method Documentation

How well is the method under investigation documented?
    pseudocode - Method described in pseudocode.
open_source_code - Is the method code available?

```
In [22]: plot_full_series(experimental_data.pseudocode, 'Pseudocode', ['False', 'True'])
         plot_full_series(experimental_data.open_source_code, 'Open Source Code', ['False', 'Tru
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```

2.5 Open Data

How well is the data documented, and is it available?
train - Training set specification.
validation - Validation set specification.
test - Test set specification.
results - Raw results data.

```
In [23]: plot_full_series(experimental_data.train, 'Training Data', ['N/A', 'False', 'True'])
         plot_full_series(experimental_data.validation, 'Validation Data', ['N/A', 'False', 'Tru
         plot_full_series(experimental_data.test, 'Test Data', ['False', 'True'])
         plot_full_series(experimental_data.results, 'Results Data', ['False', 'True'])
```

```
<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>


<IPython.core.display.Javascript object>


<IPython.core.display.HTML object>
```