

Finding Common Root Causes in Autotest Results (Applied Machine Learning techniques)

Allan P. Engsig-Karup¹, Steffen Holmslykke¹, Søren Lindegaard Grubov²

Introduction

As a software provider, SimCorp needs to handle a large amount of failing autotests across many installations world-wide during an implementation period. Such failing auto tests of SimCorp Dimension installations are handled manually and individually every day as a part of continuous quality assurance. Even though the tests are different, many of these tests fail with a common root cause due to a shared flow. If the failed tests with the same root cause can be grouped together, the amount of duplicate work could be reduced and the number of uniquely handled root causes could be increased leading to better resources spent on handling such errors.

Research Question

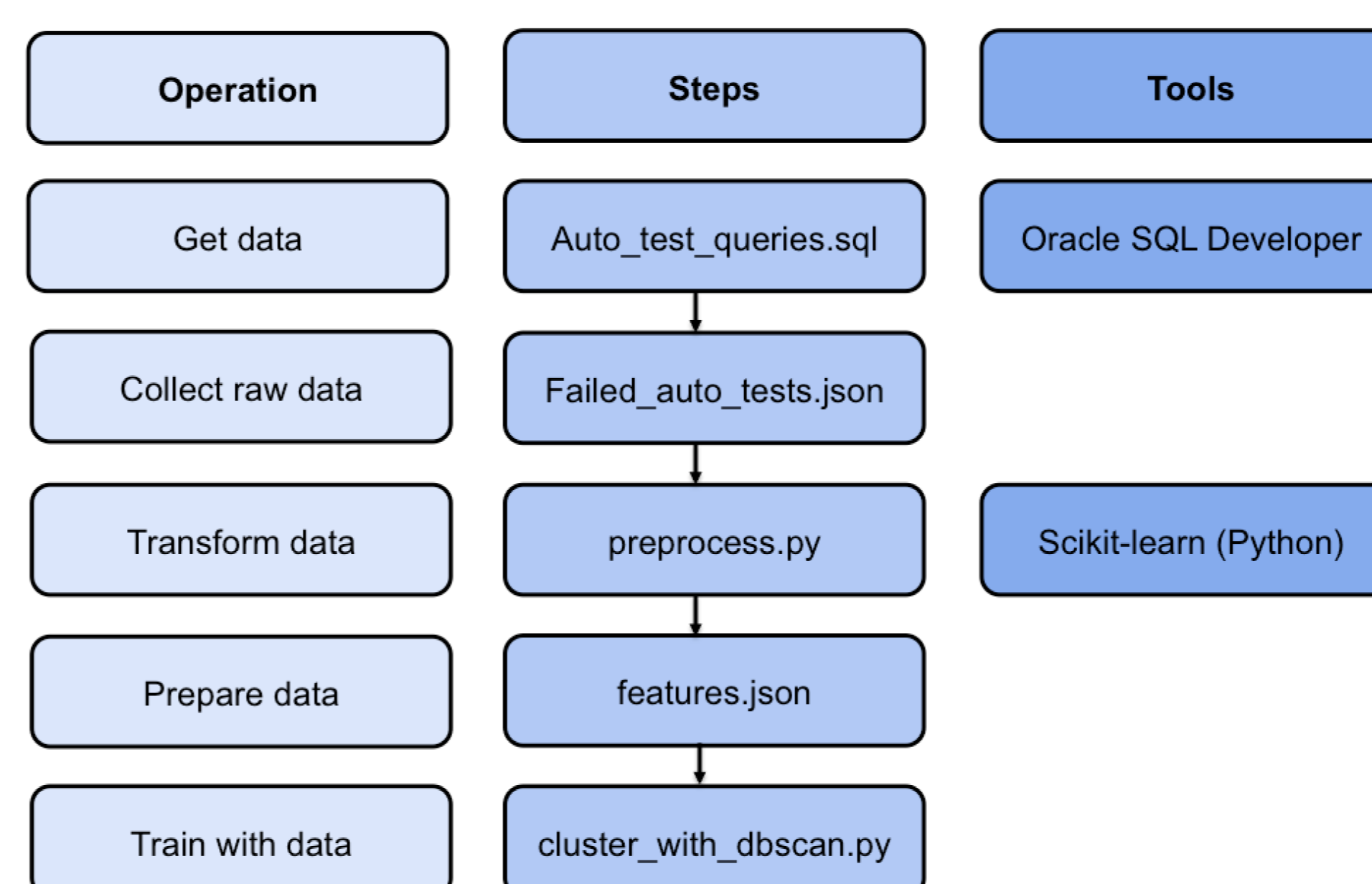
Can we automatically detect different types of errors in the Auto Test results and group these together?

Problem Setting

Daily error logs are produced across all SimCorp Dimension installations. These error logs are simple text files that contains information from the batch jobs that are executed to test the installations. These logs contains information about the execution and the errors that have resulted. However, the errors reported are described with textual information that may not be uniquely classified immediately, and therefore it can be difficult to describe root cause without warranting further investigations into the code. For example, it may be that due to one test failing, then subsequently a number of other tests also fails and in such cases the root cause is the first test that failed. Also, it is known how the correct output should look like for a given batch job that are executed successfully and therefore it is possible to identify the textual information about each error that have resulted.

Machine Learning Workflow

We employ the scientific python framework SciKit-Learn [2] to process the raw data and transform it into a structured data set using a bag of words model for identifying patterns in the error messages.



Error Log Data

```

00 00 Hhmmccs Error reading file: "T:\Batch Jobs\Test Cases\Alternative Investments\BFSupport\Files\6.1\TEST_BATCH_EVENTS\Create Transaction.csv"
00 00 Hhmmccs The process cannot access the file "T:\Batch Jobs\Test Cases\Alternative Investments\BFSupport\Files\6.1\TEST_BATCH_EVENTS\Create Transaction.csv" because it is being used by another process.
00 00 Hhmmccs Parameters:
00 00 Hhmmccs Import file = T:\Batch Jobs\Test Cases\Alternative Investments\BFSupport\Files\6.1\TEST_BATCH_EVENTS\Create Transaction.csv
00 00 Hhmmccs Result:
00 00 Hhmmccs 8 of 8 records() ok
00 00 Hhmmccs Batch job "423W TR_CASH" completed.
00 00 Hhmmccs Batch job "423W TR_CASH" started.
00 00 Hhmmccs Filter ("423W TR_CASH") has been loaded.
00 00 Hhmmccs Parameters:
00 00 Hhmmccs Import file = T:\Batch Jobs\Test Cases\Alternative Investments\BFSupport\Files\6.1\TEST_BATCH_EVENTS\Create Transaction.csv
00 00 Hhmmccs Result:
00 00 Hhmmccs 28 of 28 records() ok
00 00 Hhmmccs Batch job "423W TR_CASH" completed.
00 00 Hhmmccs Batch job "423W TR_CASH" started.
00 00 Hhmmccs Data format setup ("423W TR_CASH") has been loaded.
00 00 Hhmmccs Parameters:
00 00 Hhmmccs Import file = T:\Batch Jobs\Test Cases\Alternative Investments\BFSupport\Files\6.1\TEST_BATCH_EVENTS\Create Transaction.csv
00 00 Hhmmccs Result:
00 00 Hhmmccs 6 of 6 records() ok
00 00 Hhmmccs Batch job "423W TR_CASH" completed.
00 00 Hhmmccs Batch job "423W TR_CASH" started.
00 00 Hhmmccs Data format setup ("423W TR_CASH") has been loaded.
00 00 Hhmmccs Parameters:
00 00 Hhmmccs Import file = T:\Batch Jobs\Test Cases\Alternative Investments\BFSupport\Files\6.1\TEST_BATCH_EVENTS\Create Transaction.csv
00 00 Hhmmccs Result:
00 00 Hhmmccs 18 of 18 records() ok
00 00 Hhmmccs Batch job "423W TR_CASH" completed.
00 00 Hhmmccs Batch job "423W TR_CASH" started.
  
```

Differences between error log data and expected log data

```

00 00 Hhmmccs There are 2 records in 'Error Messages' for this calculation.
00 00 Hhmmccs Batch job "APT PORT CAL" completed.
00 00 Hhmmccs Batch job "APT PC COMP" started.
00 00 Hhmmccs Portfolio calculation ("APT") has been loaded.
00 00 Hhmmccs Comparison report:
00 00 Hhmmccs 1 calculation(s) have been checked.
00 00 Hhmmccs Portfolio calculation: "APT"
00 00 Hhmmccs Calculation name: For Apt batch job
00 00 Hhmmccs The calculation results are identical.
00 00 Hhmmccs Batch job "APT PC COMP" completed.
00 00 Hhmmccs Risk measurement ("APT ABS") has been loaded.
00 00 Hhmmccs Parameters:
00 00 Hhmmccs From date = dd-mm-yyyy
00 00 Hhmmccs To date = dd-mm-yyyy
00 00 Hhmmccs Data is incomplete for following securities.
00 00 Hhmmccs ORL_WAR_EQ_US
00 00 Hhmmccs ORL_WAR_USD
00 00 Hhmmccs For more information see the error log.
00 00 Hhmmccs Proxy definition rules will be applied if defined else "Defa
00 00 Hhmmccs Would you like to proceed with External model calculate
00 00 Hhmmccs There are 8 errors in the risk measurement execution. (
00 00 Hhmmccs Batch job "APT RM ABS" completed.
00 00 Hhmmccs Batch job "APT COMP ABS" started.
00 00 Hhmmccs Risk measurement ("APT ABS") has been loaded.
00 00 Hhmmccs Comparison report:
00 00 Hhmmccs 1 calculation(s) have been checked.
00 00 Hhmmccs Risk measurement: "APT ABS"
00 00 Hhmmccs Risk measurement name: "absolute tree, simple setup, b
00 00 Hhmmccs The calculation results are identical.
00 00 Hhmmccs Batch job "APT COMP ABS" completed.
00 00 Hhmmccs Batch job "APT RM REL 1" started.
00 00 Hhmmccs A model "WorldBondsLocal(USD)" used for the given Por
00 00 Hhmmccs A model "WorldBondsLocal(USD)" used for the given To
  
```

Feature Engineering

To define a feature vector that can be used for training using DBSCAN [1] to automatically detect clusters, we can use a Bag of Words model / N-gram model. The chosen feature set is here defined in terms of letters (unigrams), from which we can determine the frequency count of each letter and use the resulting vector as a 'fingerprint' of the error messages.

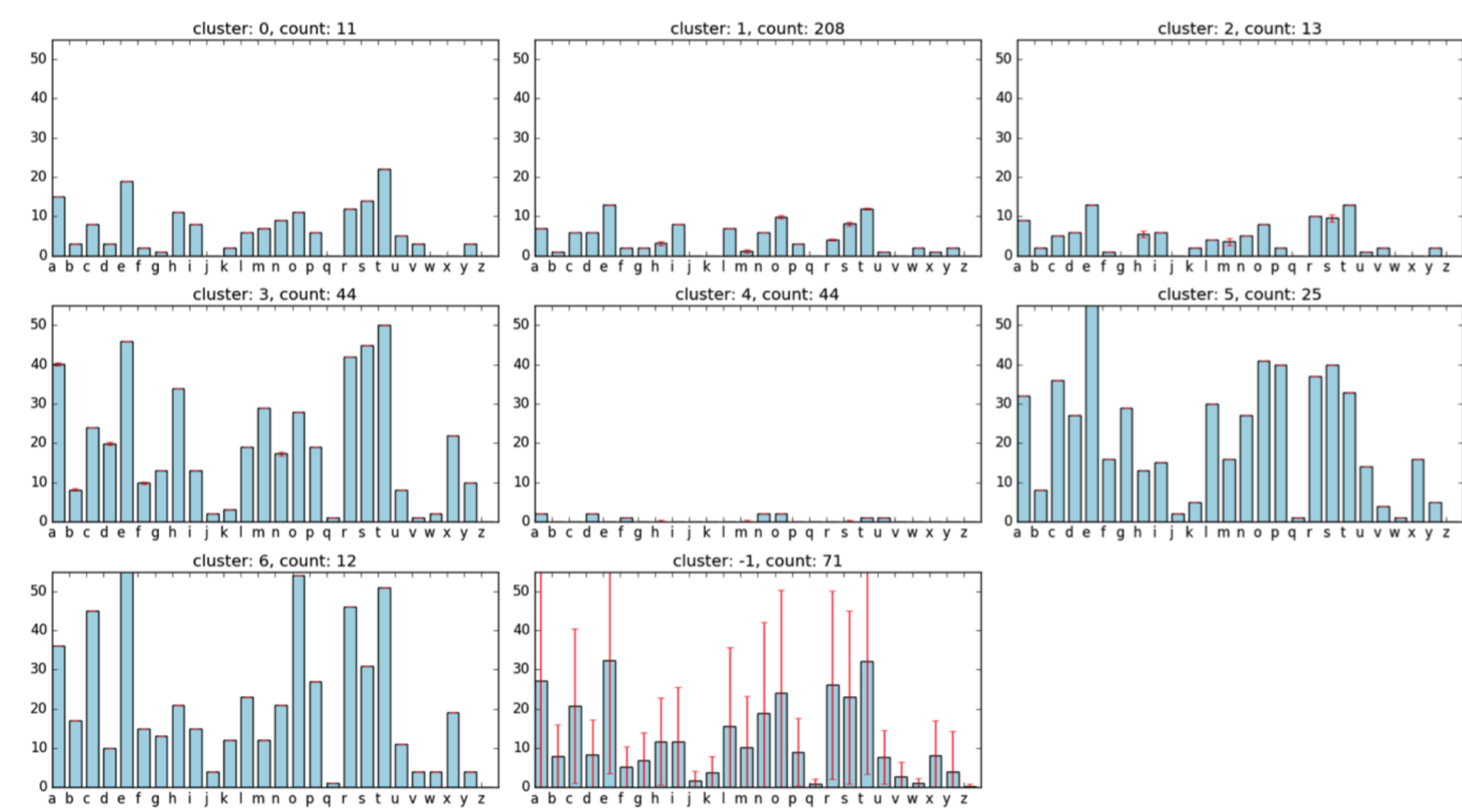
References

[1] ESTER, M., KRIEGLER, H.-P., SANDER, J., AND XU, X. A density-based algorithm for discovering clusters in large spatial databases with noise. AAAI Press, pp. 226–231.

[2] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M., AND DUCHESNAY, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.

Results of Analysis

The results of the unsupervised analysis using DBSCAN on the normalised structured dataset based on sparse feature vectors, shows that we can identify key clusters of common root causes and a number of error messages that cannot immediately be identified with a common root cause.



Legal Disclaimer

The contents of this presentation are for general information and illustrative purposes only and are used at the readers own risk. SimCorp uses all reasonable endeavours to ensure the accuracy of the information. However, SimCorp does not guarantee or warrant the accuracy, completeness, factual correctness, or reliability of any information in this publication and does not accept liability for errors, omissions, inaccuracies, or typographical errors. The views and opinions expressed in this publication are not necessarily those of SimCorp. 2016 SimCorp A/S. All rights reserved. Without limiting rights under copyright, no part of this document may be reproduced, stored in, or introduced into a retrieval system, or transmitted in any form, by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose without the express written permission of SimCorp A/S.

* Corresponding author: aptk@simcorp.com, ¹ SimCorp Technology Labs, SimCorp A/S, Denmark, ² SimCorp A/S, Denmark.