# Acknowledgements

# Abstract

Contents

[Acknowledgements 1](#_Toc87348956)

[Abstract 2](#_Toc87348957)

[1 Introduction 4](#_Toc87348958)

[1.1 Overview 4](#_Toc87348959)

[1.2 Problem statement 5](#_Toc87348960)

[1.3 Objectives of the research 5](#_Toc87348961)

[2 Literature review 6](#_Toc87348962)

[2.1 The role of Software in Scientific research 6](#_Toc87348963)

[2.1.1 Introduction 6](#_Toc87348964)

[2.1.2 General roles of software in a research 6](#_Toc87348965)

[2.1.3 Concrete roles of software in a research 7](#_Toc87348966)

[2.2 Literature review on classification of software usage purpose 8](#_Toc87348967)

[3 Training Data -SoMeSci 8](#_Toc87348968)

[3.1 Preparation of Training Data – Extension of SoMeSci 8](#_Toc87348969)

[4 Feature selection 9](#_Toc87348970)

[5 Selection of Classifiers and Training 10](#_Toc87348971)

[6 Classification and optimization of Training Models 11](#_Toc87348972)

[7 Evaluation and visualization of Results 12](#_Toc87348973)

[8 Summary of outcomes 13](#_Toc87348974)

[9 Future work 14](#_Toc87348975)

# Introduction

## Overview

* Analysis of research papers can give a lot of insights about the dependency of resources used.
* In a scientific research different kinds of input resources are used. One of such input is a software.
* Used resources in a research are typically mentioned in a citation. Citation practices to references are matured and various citation styles exist. Even if principles for formal citation of a software has already been put out, most scientists are not properly citing resources.
* Surprisingly, sometimes researchers do not mention the type of software they used entirely or mention it with a rather vague abbreviation and just talk about the results they have obtained.
* As long as software is mentioned using formal methods, like RRID, it is possible to perform citation analysis using regular expressions which can be constructed to capture the pattern of citation.
* Though regular expression based analysis can give basic insights about the software citation it has limitations because:
  + Not so many authors use formal citation of software, like RRIDs
  + Even if scientists use formal citations, they may fail to properly follow the guidelines. For example, some authors tend to ignore the RRID-part and that creates an ambiguity by it self that it is not possible to know weather the author is actually making a software citation or it is completely something else.
  + Rule based method fails to capture context information and ignores dependencies. It is not possible to be sure about the authors intention whether or not using a software citation.
* However pattern based analysis, like using regX, is not suitable to extract information about software citation especially when a software mention statement lacks any form of formality where the information is concealed in a natural language description.
* Being able to automatically extract information enables us to know about what type of software is being used for what purpose. In addition it is also important whenever it is needed to answer questions like:
  + Which software is trending right now in scientific research?
  + What type of software (purpose) is being used frequently in a specific area of research? (To determine the most common technique researchers follow when trying to solve a given research problem in a given domain )
  + Which software is producing a lot of data output ? ( one might ask to link data with the software )
  + Who developed a software ? ( one might ask to give credits to the developers)
* Previous attempts to automatically extract information about the software use purpose was constrained mainly because of lack of ground truth data. But this time, with the advent of SoMeSci, it is possible to do so.

## Problem statement

## Objectives of the research

This work has the following objectives:

* To extend SoMeSci with a manual annotation of purpose of software usage.
* To find a classifier and optimize results.

# Literature review

## The role of Software in Scientific research

### Introduction

There is a clear distinction between a scientific software and regular software [4]. A *scientific software* often is developed by researchers who have domain specific knowledge [3] as part of their research and such a software is intended to help understand the research problem [2]. In contrast, a *regular software* is refers to any software that does not heavily rely on domain specific knowledge for its development and can be done by any self-taught software enthusiast. In this paper the term software refers to a scientific software.

These days the use of software in a research is very popular and software plays a very important role as important as any physical Lab-apparatus of the classical research era [3 , 4]. For instance, according to a survey conducted in 2008 [2, 6] :

* nearly 45% of scientists spend more time developing a software as part of their research work than five years ago.
* 38% of researchers spend at least 20% of their time developing a software.
* over 90% of scientists agree that software is important for their research and
* nearly 70% claim that their research directly depends on a use of a software.

Usually a software is considered as an input resource for a scientific investigation where a readily available software is used for tasks in a research. However, in reality, many scientists develop software or write a code as part of their research over the recent years [1].

### General roles of software in a research

One of the most critical role of a software in a scientific investigation is that *a software dictates result(s) of a given research and the quality of the research outcome* [2]. The role of a software is so significant in determining research results that a retrospective discovery of an error in a code of a software, specifically a logic error, renders the entire outcome of a research useless [5]. Following such events several retractions of research papers have been observed in the past [3, 6, 8].

The other very critical role of a software in a research is that it supports *reproducibility or repeatability of a research outcome*. Quality software produces a consistent and repeatable research output [9]. It also makes it easy for validation of the research results by other experimenters using the same software. A list of best practices [3, 9] from software engineering assist the development of quality software. Reproducibility also requires a proper citation of software and its attributes, like version, developer etc.) to enable disambiguation of the software. Proper data citation refers to a citation practice that follows data citation principles [10].

A software also plays a crucial role in a research in terms of *provenance*. A software indicates origins of results of a given research work and help other researchers to understand how a given result is produced and find out the foundation of the central conclusion of a given research.

### Concrete roles of software in a research

More concrete but general examples for roles of a software in a research are modelling , simulation and data analysis [6]. For instance, the Large Hadron Collider (LHC) facility at CERN uses a software consisting of over five million lines of code for collection and analysis of terabytes of data [4].

A software can also be used to make predictions from large data sets. For example, historical weather data can be used to make predictions about future temperature variations [4].

In a research scientists may use simulation software that model and mimic the behavior of real world phenomena where the actual implementation of a project is costly in terms of finance and/or a disaster to a life or environment [7].

Further more, sometimes it is not feasible or possible to replicate behavior of some systems [4], such as complex chemical systems, in a real world where the use of a software is inevitable to simulate behaviors of such systems.

Usually researchers work with various types of data. They often perform statistical data analysis on numerical data, image processing on Image data and so on. This implies that results of a given research , provenance, directly depend on the software usage.

Though some areas of research like economics does not heavily rely on software usage, most fields of science heavily rely on the use of various types of software and data to discover knowledge in their research work.

In the scientific publications software can be cited formally or informally. Even though formal citation of software provides more structured syntax, often times researchers mention the type of software they used by simply using a natural language expression. This implies that most of knowledge about software usage is implicit and hidden in a natural language text and requires an automatic extraction of software citations from a text using machine learning techniques.

To do automatic extraction of software citations from the natural language using a machine learning approach, more specifically a supervised machine learning approach, requires a reliable ground truth data, which was not available so far. How ever, recently a new gold standard corpora , SoMeSci, has been introduced and can be used for automatic extraction of software citations from the a natural language text.

## Literature review on classification of software usage purpose

During a research scientists may use several software resources and cite them in a single publication [12]. This indicates that each software has a unique purpose and functionality that serves to explore the research problem and help to find an answer. For instance, a scientist might use one software for modelling the behavior of some system. Then a behavior of that system could then be understood using a data analysis software and so on.

Citation of a given software might include a lot of information about the software itself, its purpose of use, weather or not a researcher is introducing a novel software or using already existing one.

Citation about the purpose of software use in a research would give a contextual information that might help to disambiguate and uniquely identify each software. This would directly assist the effort to automatic extraction and disambiguation of a software resource from scientific papers.

A software can be used for various purposes during a research. One example of software use in a research is to perform Data analysis. Data analysis is an umbrella term that might include inspecting, cleaning, transformation, and modelling a data. The main purpose of data analysis is to extract meaningful information from a data that will help to make businesses operate more effectively [13].

Simulation

Prediction

Calculation

# Training Data -SoMeSci

## Preparation of Training Data – Extension of SoMeSci

# Feature selection

# Selection of Classifiers and Training

# Classification and optimization of Training Models

# Evaluation and visualization of Results

# Summary of outcomes

# Future work

References

[1] C. Goble, "Better Software, Better Research," in IEEE Internet Computing, vol. 18, no. 5, pp. 4-8, Sept.-Oct. 2014, doi: 10.1109/MIC.2014.88.

[2] J. E. Hannay, C. MacLeod, J. Singer, H. P. Langtangen, D. Pfahl and G. Wilson, "How do scientists develop and use scientific software?," 2009 ICSE Workshop on Software Engineering for Computational Science and Engineering, 2009, pp. 1-8, doi: 10.1109/SECSE.2009.5069155.

[3] Wilson G, Aruliah DA, Brown CT, Chue Hong NP, Davis M, Guy RT, et al. (2014) Best Practices for Scientific Computing. PLoS Biol 12(1): e1001745. <https://doi.org/10.1371/journal.pbio.1001745>

[4] Tim Storer. 2017. Bridging the Chasm: A Survey of Software Engineering Practice in Scientific Programming. ACM Comput. Surv. 50, 4, Article 47 (November 2017), 32 pages. <https://doi.org/10.1145/3084225>

[5] Soergel DAW. Rampant software errors may undermine scientific results [version 2; peer review: 2 approved]. F1000Research 2015, 3:303 (<https://doi.org/10.12688/f1000research.5930.2>)

[6] Merali, Z. Computational science: ...Error. Nature 467, 775–777 (2010). <https://doi.org/10.1038/467775a>

[7] Wikipedia contributors, "Simulation software," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Simulation\_software&oldid=1045982192 (accessed October 30, 2021).

[8] Miller G. Scientific publishing. A scientist's nightmare: software problem leads to five retractions. Science. 2006 Dec 22;314(5807):1856-7.doi: 10.1126/science.314.5807.1856. PMID: 17185570.

[9] Jiménez RC, Kuzak M, Alhamdoosh M et al. Four simple recommendations to encourage best practices in research software [version 1; peer review: 3 approved]. F1000Research 2017, 6:876 (<https://doi.org/10.12688/f1000research.11407.1>)

[10] Data Citation Synthesis Group: Joint Declaration of Data Citation Principles. Martone M. (ed.) San Diego CA: FORCE11; 2014 <https://www.force11.org/group/joint-declaration-data-citation-principles-final>

[11] Schindler, D., Bensmann, F., Dietze, S., & Krüger, F. (2021b). SoMeSci-a 5 star open data gold standard knowledge graph of software mentions in scientific articles [accepted]. Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21). <https://doi.org/10.1145/3459637.3482017>

[12] Alice Allen, Peter J. Teuben, and P. Wesley Ryan. 2018. Schroedinger’s Code: A Preliminary Study on Research Source Code Availability and Link Persistence in Astrophysics. The Astrophysical Journal Supplement Series 236, 1 (May 2018), 10. <https://doi.org/10.3847/1538-4365/aab764>

[13] Wikipedia contributors, "Data analysis," Wikipedia, The Free Encyclopedia, https://en.wikipedia.org/w/index.php?title=Data\_analysis&oldid=1052485049 (accessed November 3, 2021).