A categorization scheme for Software Engineering conference papers and its application

Antonia Bertolino[a], Antonello Calabrò[a], Francesca Lonetti[a,*], Eda Marchetti[a], Breno Miranda[a,b]

[a] *ISTI - CNR*
Via Moruzzi 1
56124, Pisa, Italy
[b] *Federal University of Pernambuco*
50740-540, Recife-PE, Brazil

Abstract

Background: In Software Engineering (SE), conference publications have high importance both in effective communication and in academic careers. Researchers actively discuss how a paper should be organized to be accepted in mainstream conferences.
Aiming: This work tackles the problem of generalizing and characterizing the type of papers accepted at SE conferences.
Method: The paper offers a new perspective in the analysis of SE literature: a categorization scheme for SE papers is obtained by merging, extending and revising related proposals from a few existing studies. The categorization scheme is used to classify the papers accepted at three top-tier SE conferences during five years (2012-2016).
Results: While a broader experience is certainly needed for validation and fine-tuning, preliminary outcomes can be observed relative to what problems and topics are addressed, what types of contributions are presented and how they are validated.
Conclusions: The results provide insights to paper writers, paper reviewers and conference organizers in focusing their future efforts, without any intent to provide judgments or authoritative guidelines.

Keywords: Conference, paper categorization, paper type, research contribution, research problem, validation

*Corresponding author. *E-mail address:* francesca.lonetti@isti.cnr.it
Email addresses: antonia.bertolino@isti.cnr.it (Antonia Bertolino), antonello.calabro@isti.cnr.it (Antonello Calabrò), francesca.lonetti@isti.cnr.it (Francesca Lonetti), eda.marchetti@isti.cnr.it (Eda Marchetti), bafm@cin.ufpe.br (Breno Miranda)

Postprint of paper DOI 10.1016/j.jss.2017.11.048 appeared in Journal of Systems and Software

1. Introduction

Writing good papers is an indispensable part of a researcher's activity. It is thanks to the publication of research results that collective knowledge grows and science advances. Writing *good* scientific papers is difficult, though. Within many research fields, this is acknowledged and expert guidance is provided on how to structure a paper content and how to articulate its parts, e.g. [1, 2].

Differently from other disciplines, papers in peer-reviewed conference proceedings constitute an important portion of the SE literature. Very different types of papers are presented at SE conferences: some propose new approaches or theories, other describe empirical studies; some papers focus on industrial experiences, other propose new conceptual frameworks for investigating SE problems; and so on.

However, to date there is not a shared good practice of reporting the various types of SE research, and different writing and reviewing patterns emerge within different conferences. A common taxonomy and more precise guidelines on how to write the different types of articles [3, 4] should be established within the broad SE community. Such taxonomy and guidelines could help conference organizers in more explicitly describing the type of submissions they expect and in clearly reflecting the conference scope in the call for papers. It could also help in defining common criteria for evaluating the different types of submissions, thus making the review process less dependent from the personal expertise and understanding of the reviewers in the SE field. As an example, Wieringa et al. [5] provide a paper classification scheme and evaluation criteria for papers belonging to Requirements Engineering discipline.

In 2003 [6], Shaw noted that researchers in SE had not yet developed well-understood guidelines for paper writing. Her seminal work provides a *minitutorial* with insightful advice on how results in SE research should be reported. Moreover, based on the papers submitted and accepted to one edition of the International Conference on Software Engineering (ICSE), she identified the elements that should form a good SE research paper and the SE research strategies that emerged from the combination of those elements. Shaw's paper is widely referenced and some other authors have analogously conducted literature analyses to identify SE research strategies and paper types [3, 4, 7, 8].

However, to the best of our knowledge there have been no follow up from Shaw's paper[1], neither in terms of reviewing and possibly revising her proposed classification of paper elements, nor in terms of matching the presented guidelines against other sets of SE papers.

The study of research methods and practices is the scope of *meta-research*, or "research on research". Meta-research has now established itself as a scientific discipline that attracts growing interest also thanks to the opportunities of analyses offered by the on-line availability of publications, reports and data.

[1] For completeness we mention that after this paper's submission we have assisted at a presentation given at ICSE 2017 from a work in preparation by Theisen and coauthors [9] that replicates that study on ICSE 2016.

Ioannidis et al. [10] categorize the meta-research discipline into five main thematic areas, namely: Methods, Reporting, Reproducibility, Evaluation, and Incentives, corresponding to *"how to do, report, verify, correct, and reward science"* [10].

This paper contributes to meta-research in SE, focusing on how research in the software engineering (SE) field is or should be reported in the context of scientific conferences. The work includes two parts: first we propose a *paper categorization scheme* aiming at identifying a "paper model" that is comprehensive of the article genres published in the SE field, by extending and revising the types of papers and definitions identified in relevant previous studies of SE literature, specifically [6, 4, 7, 8]. This categorization scheme includes four main dimensions that are: *problem*, *contribution*, *validation* and *topic* and provides a classification for each of them. Examples of papers types according to the proposed classifications are also presented.

Then, we use the proposed categorization scheme to classify papers from three SE conferences, namely: *i)* ICSE that was the subject of Shaw's study [6]; *ii)* the International Conference on Automated Software Engineering (ASE); and *iii)* the Symposium on the Foundations of Software Engineering (FSE). Note that every other year FSE is run jointly with the European Software Engineering Conference (ESEC); in the paper for simplicity we refer to the latter as FSE, intending both FSE and ESEC-FSE depending on the year. Specifically, we study the papers published at ASE, FSE and ICSE in five editions (2012-2016), discuss their classification according to the proposed scheme and comment on possible emerging patterns. We mention that an outline of the scheme is already presented in [11]. This short work, however, only considered ICSE papers.

We believe that examining what types of papers are accepted at the three above conferences against the resulting categorization scheme is informative because they represent three top-tier publication venues in the field that are recognized by the SE community as very exigent in terms of what is required in a research paper. Our study can thus contribute to better understand how research in SE is reported and to possibly identifying gap in research communication.

The main contributions can be summarized as:

- a categorization scheme for SE conference papers that evolves and merges those presented in [6, 7, 4, 8];
- the classification of the ASE, FSE and ICSE papers published in five editions according to the proposed scheme;
- a discussion of patterns and trends emerging from the study, and also conclusions on future work needed to improve and refine the scheme.

The paper is structured as follows: in the next section we overview related work. In Section 3 we put this work in context by introducing the notion of paper type or *genre*; then, in Section 4 we describe the categorization scheme. In Section 5, we present the application of the proposed categorization scheme

to the papers of ASE, FSE and ICSE. Results are reported in Section 6 and interesting findings are further discussed in Section 7 that also concludes the paper.

2. Related work

Our work performs a secondary study of SE research, to identify paper types and patterns. Most closely related work include:

Secondary studies addressing similar goals. In searching recent related work, we made a quasi-systematic search by snowballing forward and backward from Shaw's minitutorial, but did not find many relevant papers. More interestingly, Glass and coauthors in [7] report findings from the study of a sample of articles published from 1995 to 1999 in six journals. Similarly to us, they used a top-down approach for categorization, among other things, of topics, research approach and research method. However, their scheme is more general and addresses three computing disciplines: SE, Information Systems and Computer Science. The goals and approach of the study are similar to ours, however the examined corpus and the categorization scheme are different.

More recently, Montesi and Lago [4] also propose a classification of SE paper types. Differently from us, they derive a paper type classification mainly based on the call for papers of major SE conferences, the relevant papers, the SE journals included in the Journal Citation Reports and the instructions to authors of other relevant journals not included in the list of the Journal Citation Reports.

In [3] Stol and Fitzgerald observe that the field of SE lacks a holistic view of a more complete spectrum of research methods, beyond those for empirical research that have drawn increasing attention. They thus introduce a framework for positioning SE research strategies (adapting the "circumplex" model originally proposed for analysis of behavioral systems[12]), however they do not consider paper types, but research strategies.

Finally, Wieringa et al. [5] propose a paper classification that differently from our proposal specifically targets Requirement Engineering papers. Moreover, their goal is to define a set of evaluation criteria for different paper classes.

Secondary studies addressing different goals. Several studies restrict the analysis of research strategies to more specific types of papers. For example, Sjoeberg and coauthors [13] survey SE papers in nine journals and three conferences (including ICSE) but with the aim of characterizing only controlled experiments. Zelkowitz and Wallace [8] make a classification of SE papers of a journal, a magazine and ICSE proceedings limited to three different years. Differently from our proposal, they classify papers only according to the type of SE experimentation.

In [14], Zannier and coauthors performed an empirical study to assess whether quantity and quality of empirical evaluations conducted within ICSE papers had improved along the years. They compared a stratified random sample (of 5%) of papers in the periods (1975-1990) and (1991-2005): they found that the quantity of empirical evaluation has grown, but the soundness of such evaluation has

not grown at same pace. We also examine papers according to what validation is included, however our goal is limited to paper classification. At ICSE 2009, Ghezzi gave a keynote providing his reflections on 40+ years of SE research, which also included an analysis of ICSE papers (slides are available from [15]). Although some of the findings overlap with our study, his main focus was on impact of research, which we do not discuss here.

Montesi and Mackenzie Owen [16] examine how conference papers are extended for publication in a journal.

A different thread of studies include [17] and [18]. The authors of [17] investigated the turnover of PC compositions and papers publication by PC members in six SE conferences (including ICSE). Later Vasilescu and coauthors [18] extend that paper by proposing a wider collection of metrics to assess the "health" of SE conferences. Such studies lay out a framework for SE scientometrics, i.e. their goal is measuring and analyzing SE research, and not classifying the produced paper types.

Papers providing writing guidelines rather than reporting results from literature analysis. (The former would be said to be normative papers, in contrast with secondary studies discussed above that are descriptive). The literature includes many entries of such kind, although most of them are not peer-reviewed papers, and many provide general advices that apply to any scientific field, like the rules in [19] meant for students who have yet to write their first paper. An advanced search on Google Scholar for papers including the exact phrase "How to write a research paper" in the title found 44 papers, many of which from medicine related disciplines. The widely referenced paper from Shaw [6] is a good example in this set, although it is not only a tutorial but also includes a literature study limited to ICSE 2002 proceedings.

3. About paper type

As we discuss in the next sections, our study addresses the following main research question: *What paper types are accepted at Software Engineering conferences?* Similarly to [4], by type of paper, or more formally "genre", we refer here to the "form, content, and communicative purpose"[4] of a written report summarizing some research activity. Study of types of articles produced in scientific research is relatively young, and has addressed specific aspects. A noteworthy example is Swales' identification of four rhetorical moves in the Create A Research Space (CARS) Model for introductions in scientific articles [20].

Holmes [21] defines a genre as *a class of texts characterized by a specific communicative function that tends to produce distinctive structural patterns.* The rationale of a genre *shapes the schematic structure of the discourse and influences and constrains choice of content and style. ... Exemplars of a genre exhibit various patterns of similarity in terms of structure, style, content and intended audience* (quote from [20], p.58). As genres are established bottom-up by the community to which the communication is addressed, we believe that

a study of how paper types emerge may be useful to guide the writing and selection of articles.

Miller [22] states that *for the student, genres serve as keys to understanding how to participate in the actions of a community.* We believe this applies also to researchers willing to publish in a new community. There does not exist a standard classification of paper genres for SE research field. We hence identified a classification of research papers that could fit the generally accepted practice of paper writing of SE researchers. To this purpose, we looked in two directions. On the one side, we referred to the most authoritative source in terms of what papers are invited by a conference, i.e., the Call for papers of the conference itself. We looked in particular to information related to the type of submissions that are welcomed and hints provided. On the other side, we made a survey of literature in order to identify earlier relevant studies proposing a classification of papers in software engineering. The result from this preliminary work is the paper categorization scheme defined in the following section.

Table 1: Categorization scheme dimensions for SE papers

Dimension	Description
Problem	The issue the paper would like to solve
Contribution	The main research presented in the paper
Validation	The evidence of the paper validity
Topic	The main topic of the paper

4. Paper Categorization

In this section we provide a paper categorization scheme useful for analyzing the Software Engineering literature. We do this based on available secondary studies of Software Engineering research about paper types and patterns. More precisely, our categorization scheme is the result of merging, extending and revising the types of papers identified in relevant previous studies of Software Engineering literature, specifically [4, 6, 7, 8] and in the call for papers of the main SE conferences. We systematically considered all types and elements included in the above sources, following an iterative approach, such that when new elements or inconsistencies were found, we progressively improved the categorization scheme so to come up with clearer and coherent definitions. Two of the authors led this process by deriving and refining the proposed scheme. The latter underwent the review of the other authors at two major releases. After the revision of the second release, a common final form was agreed among all the authors.

By analyzing the previously proposed classifications we observed that:

- in Shaw's minitutorial [6] papers were classified according to "Type of software engineering questions", "Type of software engineering results" and "Type of software engineering research validation";

Table 2: Overview of the proposed categorization scheme

Dimension	Sub-dimensions
Problem (Table 3)	Development method Analysis method Specific instance Generalization or characterization Feasibility study or exploration
Contribution (Table 4)	Theoretical Technological Empirical Perspectival
Validation (Table 5)	Analysis Evaluation Experience Example No Validation
Topic	See examples in Table 6

- in Glass et al. [7] papers are categorized into "topic", "research approach" and "research methods";
- Montesi and Lago [4] proposed instead a list of "Type of article", among which "Empirical research reports" and "Papers oriented towards practice" are further articulated in more sub-choices;
- Zelkowitz and Wallace [8] classified papers according to a taxonomy of software engineering experimentation.

Finally, among the calls for papers of the main SE conferences, we found that ICSE and ASE (Automated Software Engineering) defined different paper *categories*, with the aim to guide the authors in preparing their submissions and help the reviewers in evaluating them.

Following the above described process, the mentioned proposals have been unified into a multi-dimension categorization scheme; this also required to harmonize the different categorization terminologies. In particular, to the three main dimensions proposed in Shaw[6], we added the "topic" one as in [7], deriving a four dimension categorization scheme as in Table 1. Thus the four categorization dimensions have been identified as:

- *problem*: what issue the paper would like to solve or the question the paper would like to answer. This includes the "Type of software engineering questions" of [6];

Table 3: Groups for problem in SE papers

Problem	Description
Development method	The paper investigates methods or tools for (better) doing/creating/modifying/ evolving/automating/maintaining a software artifact
Analysis method	The paper investigates methods or tools for (better) analyzing/evaluating/ measuring the quality/correctness of a software artifact
Specific instance	The paper investigates how it can (better) design/implement/maintain/adapt/ evaluate/analyze some particular system, specific practice, or other instance of a software artifact
Generalization or characterization	The paper investigates how to generalize or provide important characteristics/ varieties of a software engineering process/technology/method/phenomenon
Feasibility study or exploration	The paper explores software engineering aspects in a completely new way or from a novel prospective

- *contribution*: what is the main result in software engineering research presented in the paper. This integrates the "Type of software engineering results" of [6], the "Type of article" classification of [4], the "research approach" and"research methods" of [7], the three experimentation categories of [8] and finally the paper *categories* of the main SE conferences call for papers such as those of the recent editions of ICSE and ASE;

- *validation*: what evidence the paper shows so that the contribution is valid. This refers to "Type of software engineering research validation" of [6];

- *topic*: what is the main topic the paper addresses. This is in line with the "topic" of [7].

Aiming to further detail our categorization scheme, the items and the examples of [4, 6, 7, 8] classifications have been incorporated as instances of the four main dimensions. An overview of the resulting classification scheme is in Table 2. For each dimension, more details are provided in the following. In addition, for the first three dimensions, a non-exhaustive list of exemplar instances is given so to better characterize them.

4.1. Problem

Software engineering is defined as *the systematic application of scientific and technological knowledge, methods, and experience to the design, implementation, testing, and documentation of software.*[23]. SE papers thus address research questions and challenges about software design, development and evaluation.

We classified the problems that papers try to solve into five groups: *Development method*, *Analysis method*, *Specific instance*, *Generalization or characterization*, *Feasibility study or exploration*. These correspond to those in [6]. In Table 3, an informal definition of each of them is provided.

As an example, the papers belonging to the *Development method* group focus on the identification of features useful for improving the design, implementation, maintenance and automation during the different phases of the software development. The papers of the *Analysis method* group target approaches for software evaluation, such as testing or verification and validation, as well as quality analysis including performance metrics. The third group of problem includes papers dealing with improvement of an existing software artifact or comparative evaluation of specific approaches. Papers generalizing specific aspects of the software engineering process belong to the *Generalization or characterization* group while papers focusing on a new research field belong to the last group.

4.2. Contribution

The classification of papers *contribution* leverages on the seminal work of [4, 6, 7, 8] and integrates the recent categories depicted in the main SE conferences.

Examples are the categories listed in ICSE and ASE Call for papers [2]. The identified types of contribution are: *Theoretical*, *Technological*, *Empirical* and *Perspectival*. Table 4 provides an informal definition of each of them. Aiming at clarifying the informal definition, for each type of contribution, some examples are provided.

Table 4: Groups for contribution in SE papers

Contribution	Description
Theoretical	The paper relies on theoretical assumptions and presents an analytical model or methodology to solve a problem
Technological	A paper in which the main contribution is of technical nature
Empirical	The paper presents an empirical predictive study based on observed data
Perspectival	The paper presents a novel perspective on a specific research field or part of it

Specifically, a paper providing a *Theoretical* contribution can deal with: i) conceptual, grounded or mathematical theory; ii) procedure, algorithm, design pattern or programming paradigm that can be applied to different software development process phases; iii) techniques for bug prediction, dynamic and static analysis, reliability analysis; iv) model or model transformation; v) requirement elicitation method; vi) structure or taxonomy for a problem area; vii) protocol analysis.

Examples of *Technological* contribution are papers describing: i) tools and prototypes; ii) infrastructures and frameworks; or iii) modeling languages.

Papers that can be classified as *Empirical* contribution are those targeting: i) controlled experiment; ii) case study, field study or observational study; iii) survey of professionals/practitioners through questionnaires or interview; iv) lesson learned reports.

Finally, papers classified as *Perspectival* can deal with: i) systematic literature review; ii) interdisciplinary study or exploratory study; iii) structure or taxonomy of a problem area; iv) historical perspective.

4.3. Validation

On the basis of [6], the identified types of validation of software engineering papers are: *Analysis*, *Evaluation*, *Experience*, *Example*, *No_ Validation*. Table 5 presents an informal definition of each of them.

[2] The CFPs are available at http://2015.icse-conferences.org/submission-guidelines?id=16 and http://www.ase2017.org/calls respectively.

Table 5: Groups for validation in SE papers

Validation	Description
Analysis	The paper presents a rigorous/convincing analysis of the contribution
Evaluation	The paper presents evaluation of the contribution in specific (replicable) situations
Experience	The paper presents evidences supporting the contribution by using third party result collection
Example	The paper presents illustrations derived from practical situation
No_Validation	The paper does not present any of the above mentioned validations

For instance, the validation through *Analysis* can be performed by providing proofs or complexity analysis or run-time analysis. It can also be provided by applying the contribution to a controlled situation or collecting statistically significant results and data.

Examples of validation through *Evaluation* are: i) a description of phenomena of interest involving the paper contribution; ii) the assessment of the benefits/innovation of the paper contribution; iii) the description of a feasibility study or pilot project involving the paper contribution; iv) the generation of results that fit actual data.

The peculiarity of validation through *Experience* is that it can be performed exploiting third party data/results, therefore some examples are: i) a narrative description of the correctness/usefulness/effectiveness of the paper contribution; ii) a collection of (statistical) data about the contribution in (real) specific cases; a comparison of software systems/artifacts in actual use.

Moreover, validation through *Example* can be performed as: i) a narrative description of how the paper contribution works; ii) presenting the use of the paper contribution in a simplified (real) situation; iii) using toy or textbook example for motivating the paper contribution.

Finally example of *No_ Validation* are persuasion and blatant assertion.

4.4. Topic

This category identifies the research topics addressed by the specific SE conference to be classified.

Concerning this category we note that over the years, the research scope of SE conferences has been extended and revised, so to include at each edition the most recent interests and open issues across the full spectrum of software engineering. Moreover, the list of topics addressed by each SE conference may be specialized to reflect some peculiarities of the conference research domain.

For these reasons, the topic category of our model is not statically defined, but needs to be instantiated each time to reflect the most recent call for papers and also possibly the specific list of topics belonging to the SE conference of interest. In our study, to classify the papers from the target conferences (ASE[3], FSE[4], and ICSE[5]), we use their recent list of topics as reported in Table 6.

5. Application of the Categorization scheme

In this second part of the paper, we apply the categorization scheme presented in Section 4 to classify papers published from some main SE conferences, as detailed in the following sections. The main goal of such study is to show how and if the scheme can be useful for secondary studies; however, as a secondary goal, we also aim at understanding the categories of accepted papers at some relevant SE conferences. On the contrary, evaluating or cross-comparing the papers accepted at the three selected conferences is not a goal of this study.

5.1. Research Questions

Precisely, our study aims at answering the RQ: *What types of paper are accepted at top-tier SE conferences?* With reference to the proposed paper categorization scheme, we can articulate this question into more detailed sub-questions, as follows:

RQ1 What topics are mostly investigated?
RQ2 What problems are mainly addressed?
RQ3 What is the main type of contribution?
RQ4 What is the main type of validation?
RQ5 What are, if any, specific patterns or trends?

5.2. Selected Conferences

As said in the introduction, among the many conferences within the broad area of Software Engineering we applied our classification scheme to three of them: (i) the International Conference on Automated Software Engineering (ASE); (ii) the International Symposium on the Foundations of Software Engineering (FSE); and (iii) the International Conference on Software Engineering (ICSE). These conferences were chosen for two reasons: they are three well established conferences commonly acknowledged as high-quality venues, and they have a broad scope, covering all facets of SE research.

By the time we carried out this study, ASE is on its 31st edition. The first conference in the series was held in 1986 and, at the time, it was known as *Knowledge-Based Software Assistant (KBSA)*. Between 1991 and 1996 the

[3]Call for papers at: `http://ase2016.org/cfp.html`
[4]Call for papers at: `http://www.cs.ucdavis.edu/fse2016/calls/research-papers`
[5]Call for papers at: `http://2016.icse.cs.txstate.edu/researchTrack`

Table 6: Research Topics from the CfP for ASE, FSE, and ICSE 2016

ASE	FSE	ICSE
Automated reasoning techniques	Architecture and design	Agile software development
Component-based systems	Aspect-orientation	Autonomic and (self-)adaptive systems
Computer-supported cooperative work	Autonomic computing and (self-)adaptive systems	Cloud computing
Configuration management	Big data	Component-based software engineering
Data mining for software engineering	Cloud computing	Configuration management and deployment
Domain modeling and meta-modeling	Components, services, and middleware	Cooperative, distributed, and collaborative software engineering
Empirical software engineering	Computer-supported cooperative work	Cyber physical systems
Human-computer interaction	Configuration management and deployment	Debugging, fault localization, and repair
Knowledge acquisition and management	Crowdsourcing	Dependability, safety, and reliability
Maintenance and evolution	Debugging	Embedded software
Model-driven development	Dependability, safety, and reliability	Empirical software engineering
Model transformations	Development tools and environments	End-user software engineering
Program synthesis & transformations	Distributed, parallel, and concurrent software	Formal methods
Modeling language semantics	Education	Green and sustainable technologies
Open systems development	Embedded and real-time software	Human factors and social aspects of software engineering
Program comprehension	Empirical software engineering	Human-computer interaction
Re-engineering	End-user software engineering	Middleware, frameworks, and APIs
Requirements engineering	Formal methods	Mining software engineering repositories
Specification languages	Green computing	Mobile applications
Software analysis	Human and social factors in software engineering	Model-driven engineering
Software architecture and design	Human-computer interaction	Parallel, distributed, and concurrent systems
Software product line engineering	Knowledge based software engineering	Performance
Software visualization	Mobile, ubiquitous, and pervasive software	Probabilistic systems
Testing, verification, and validation	Model-driven software engineering	Program analysis
	Patterns and frameworks	Program comprehension
	Policy and ethics	Program synthesis
	Processes and workflows	Programming languages
	Program analysis	Recommendation systems
	Program comprehension and visualization	Refactoring
	Program synthesis	Requirements engineering
	Programming languages	Reverse engineering
	Refactoring	Search-based software engineering
	Requirements engineering	Security, privacy and trust
	Reverse engineering	Software architecture
	Safety-critical systems	Software economics and metrics
	Scientific computing	Software evolution and maintenance
	Search-based software engineering	Software modeling and design
	Security and privacy	Software process
	Software economics and metrics	Software product lines
	Software evolution and maintenance	Software reuse
	Software product lines	Software services
	Software reuse	Software testing
	Software services	Software visualization
	Specification and verification	Specification and modeling languages
	Testing	Tools and environments
	Traceability	Traceability
	Web-based software	Ubiquitous/pervasive software systems
		Validation and verification

conference was known as *Knowledge-Based Software Engineering (KBSE)*, and since 1997 it has been renamed to Automated Software Engineering.

The first FSE conference was held on 1993 and, since 1997, in odd-numbered years it is jointly held with the *European Software Engineering Conference (ESEC)*. Its 24th edition was held in 2016.

The first ICSE conference in the series was held in 1975 and it was called *1st National Conference on Software Engineering*. In the following year it was renamed to International Conference on Software Engineering and its 38th edition was held in 2016.

For the three conferences we analyzed all the papers published in five editions (from 2012 to 2016) and selected the "full papers" (for the purpose of this study we qualify as full those papers that are longer than 6 pages) in the technical research track. For the case of ASE, the "New Ideas" track was not considered due to the number of pages of the papers published in that track. Analogously, for FSE the papers from both "Industrial" and "New Ideas" track would not qualify as full papers according to our selection criterion.

The final number of papers analyzed in our classification schema was 244, 296, and 436 for ASE, FSE, and ICSE, respectively (a total of 976 papers).

5.3. Classification Process

Consistently with similar studies such as [6], we classified the papers on the basis of their title, abstract and keywords. As a byproduct of the study, we were thus also able to answer an additional research question: *Do abstracts of ASE, FSE and ICSE papers carry sufficient information for understanding the paper type?* (we answer to such question in the Discussion section).

Similarly to [7] we adopted the following classification process:

i) each paper was randomly assigned to two of the authors, who performed an independent coding for each of the four dimensions of the categorization presented in Tables 3, 4, 5, and 6. For the first three dimensions only one group could be selected: where a paper might be assigned to different groups, we asked anyhow to select one *main* group. This decision is motivated by our aim of better capturing the typology of papers studied. We understand that every paper might actually include aspects of more than one dimension. For example if we consider "contribution", a hypothetical paper might include a theoretical part that is implemented into a technological solution, which is then empirically evaluated, providing perspectival conclusions. Including for a paper as this all possible contributions then would provide a flat and little informative classification. Instead, we try to decide for each paper what is the *predominating* one among various possible contributions. It is worth noting that this decision however refers to our application of the scheme on ASE, FSE and ICSE, but the scheme *per se* does not prevent to select more than one group. In the case of *topic* dimension, each author could instead select up to three values, considering that the main contribution of the paper might address more than one topic.

ii) following the individual codings, one of the remaining authors compared the two groups for each categorization dimension, and in case of inconsistencies,

acted as an arbiter and proposed the final assignment that was very often (but not always) equal to one of the two assigned values. The final assignments were then agreed among all authors. In the specific case of *topic* dimension, the arbiter chose the common values among those individually selected or made (very few cases) the own final coding in case of disagreement.

Before proceeding to the classification study above described, whose results are reported in the next section, we performed a trial phase with the following objectives:

1. clarifying any possible ambiguity or misunderstanding regarding the paper categorization scheme within the team of authors;
2. tuning the classification process and getting insights to drive the application study.

The trial phase interested a small sample including 165 papers from the whole set of 1654 papers published along the whole ICSE history until 2016, (i.e., a 10% sample). During such *Trial* phase, the applied classification process evidenced some disagreements in how the authors interpreted the dimensions. Along the trial process, for improving the rate of agreement and reducing misunderstanding regarding the paper categorization, we held several workshops among all authors and improved the list of examples as presented in Section 4.

We performed an inter-rater agreement test to measure the degree of agreement among the individual codings in step *i)*. Such a test is able to capture how much consensus exists in the assignments made by the authors. Among the various statistics available to determine inter-rater reliability, we adopted the Fleiss' kappa measure and the results are interpreted according to Landis et al. [24], in which a *kappa* value less than 0 is described as "poor", from 0 to 0.20 is described as "slight", 0.21 to 0.40 as "fair", 0.41 to 0.60 as "moderate", 0.61 to 0.80 as "substantial", and, finally, from 0.81 to 1.00 is described as "almost perfect". For the three conferences, we achieved "moderate" agreement among the individual codings, with *kappa* equal to 0.51, 0.56, and 0.50 for ASE, FSE, and ICSE, respectively. Generally speaking, if there is a lot of disagreement, either the classification scheme is inadequate or the raters need to be re-trained to properly use the scheme. However, as said, concerning the final assignments at step *ii*, these were eventually agreed among the raters and the arbiter. We discuss potential validity threats to the scheme in Section 6.6. We also looked at the inter-rater agreement per each dimension individually, and we observed that we achieved the lowest agreement rates for the "Validation" dimension. One possible explanation for that is the fact that many of the abstracts analyzed do not explicitly state which kind of validation is performed to support the results and claims reported in the paper, as we further discuss in Section 6.

6. Results

In the following we answer to the research questions by reporting the main observations we collected from studying the papers accepted at ASE, FSE and ICSE in the years 2012-2016.

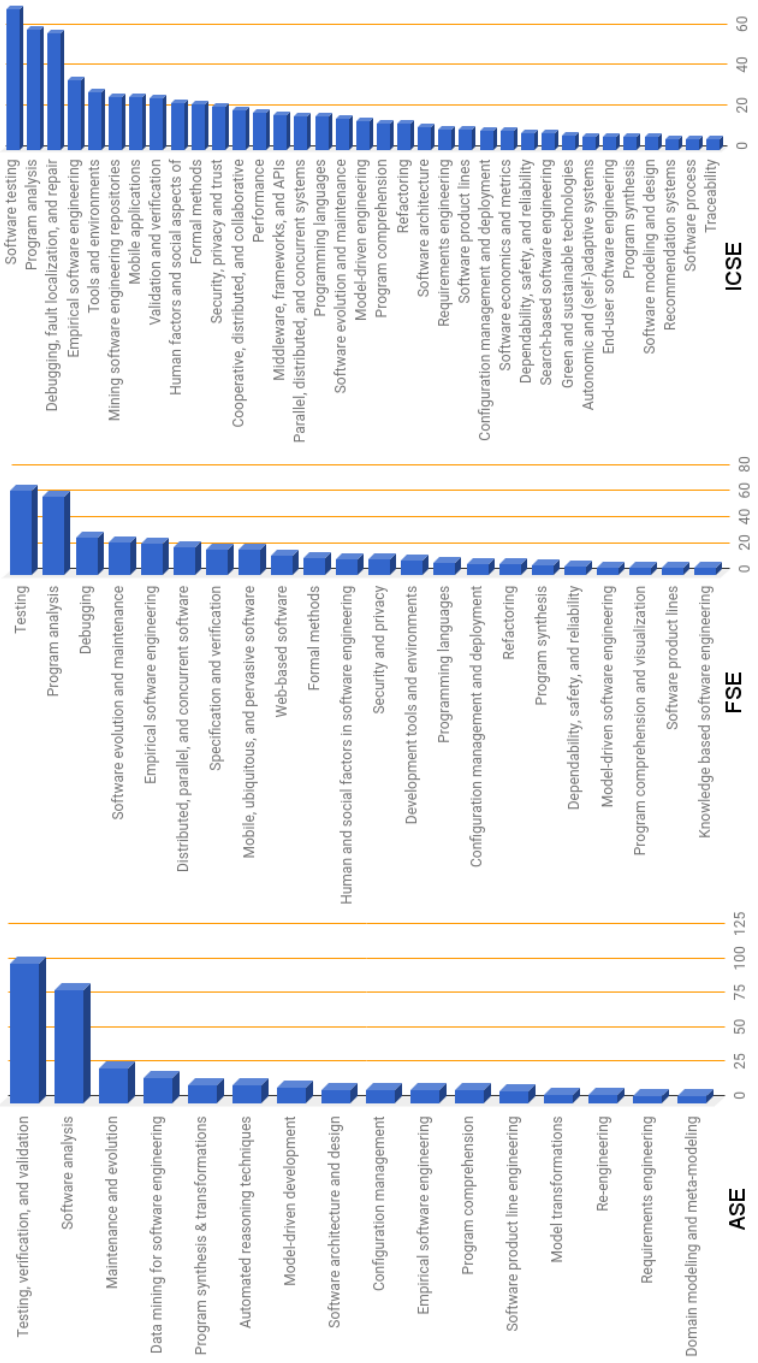
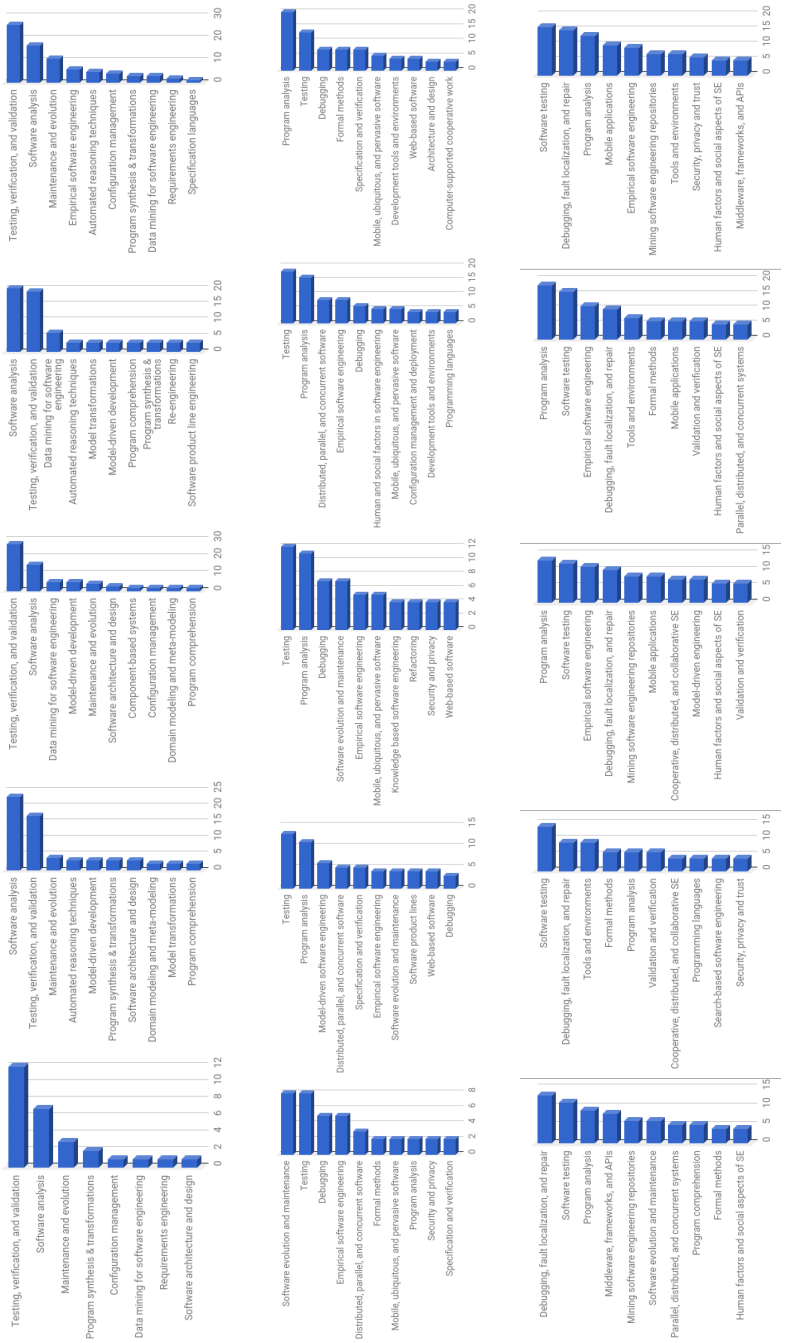


Figure 1: Most covered topics ASE - FSE - ICSE (2012-2016)



(a) **ASE** (b) **FSE** (c) **ICSE**

Figure 2: Most covered topics per year (2012-2016) for ASE, FSE, and ICSE

6.1. RQ1: What topics are mostly investigated?

The results of the classification of topics for the ASE, FSE and ICSE papers in the years 2012-2016 are summarized in Figure 1. For readability, the figure shows only the topics covered in at least 5 papers. We recall that for each conference we classified the papers against the topics listed in its own call (from 2016), which are summarized in Table 6. Although there are clearly overlaps among the three lists, several differences emerge both in their granularity (ASE lists 24 topics, against the 47 ones of FSE and the 48 ones of ICSE) and in their coverage. So, for example, while the topics "Requirements engineering" or "Empirical software engineering" appear identically in all three lists, the same does not happen for other branches of SE research: in ASE we have one topic named "Testing, verification, and validation", whereas in ICSE these arguments form two separate topics, namely "Software testing", and "Validation and verification"; and in FSE the topics list has an entry named "Testing", and another one "Specification and verification". "Education" and "Crowdsourcing" are present only in FSE topics, while "Cyber physical systems" and "Recommendation systems" are only among ICSE topics. From such considerations, we see that a cross-comparison among the topics covered in the three conferences would not make sense. This is why we report in Figure 1 the most investigated topics separately per each conference. However, even so, one can notice that the highest portion of published papers across the three SE conferences tackles with topics related to software analysis, maintenance and testing: for ASE the 3 top topics are: "Testing, verification and validation", "Software analysis", and "Maintenance and evolution"; for FSE: "Testing", "Program analysis", and "Debugging"; and for ICSE: "Software testing", "Program analysis", and "Debugging, fault localization and repair".

We also report in Figure 2 the frequency of topics per year. We can notice that for the 3 conferences the topic of testing always appears within the top two investigated topics across all five years.

In addition to our manual classification, we also made a check of the topics mostly investigated by an automated textual analysis of the paper titles, in the assumption that these reflect, according to the paper authors themselves, the topics of the paper. We derived word clouds (i.e., weighted lists of the words) from the titles of the papers published at ASE, FSE and ICSE. For creating the word clouds, we cleaned up the paper titles to *i)* remove numbers, punctuation, and special characters; *ii)* remove unnecessary white space; *iii)* convert the text to lower case; and *iv)* remove common stopwords like "*the*", "*a*", "*to*", etc. After that, we applied stemming (using the Porter stemming algorithm [25]) for reducing inflected and derived words to their word stem. For the resulting word cloud this means that words deriving from the same root are counted together (e.g., *test*, *testing*, and *tests*, all map to the word *test*). The resulting clouds for the 50 most used words in the titles of ASE, FSE, and ICSE papers are displayed in Figures 3a, 3b, and 3c, respectively. By visual analysis the three clouds show large overlaps and also high similarity with the topic classification in Figure 1.

(a) **ASE** (b) **FSE** (c) **ICSE**

Figure 3: Word clouds from the titles of the papers published at ASE, FSE, and ICSE from 2012 to 2016

6.2. RQ2: What problems are mainly addressed?

The results of classification of problems addressed by the ASE, FSE and ICSE papers are summarized in Table 7 (rows 2-6). We see that for the three conferences the highest percentage of papers (precisely, 41% of ASE, 42% of FSE and 31% of ICSE) tackles problems related to analysis and evaluation of software artifact (*Analysis method*). Then the second highest problem for ASE (31%) and FSE (23%) papers is related to (*Development method*), whereas for ICSE papers, the second highest tackled problem is *Generalization or characterization* (25%).

Figure 4 details further the distribution of papers published in ASE, FSE and ICSE according to the problem dimension per year.

6.3. RQ3: What is the main type of contribution?

Table 7 shows the results of the classification of contributions addressed by the ASE, FSE and ICSE papers (rows 8-11). The table clearly evidences that the main contribution of ASE, FSE and ICSE papers is *Technological*, while on the other hand only a very few papers provide *Perspectival* contribution. In Figure 5 we depict the percentages of papers for each type of contribution per year.

We performed also an additional analysis matching for each conference the contribution types of the papers against the corresponding researched topics (see Tables 8, 9, and 10). Such kind of tables could be useful to conference organizers for instance to understand whether the papers they accept are skewed towards some typology, and to trigger reflection on if and how different typologies should be sought. For example we can see that in the years 2012-2016, 53 papers out of the 244 we considered for ASE provide a *Technological* contribution addressing the topic of "Software analysis". We can also see that empirical contributions are not present for several of the listed topics. ICSE conference, on the other side, has accepted *Technological*, *Theoretical* and *Empirical* contribution on almost

Table 7: Results for Problem, Contribution and Validation dimensions

Problem	ASE	FSE	ICSE
Development method	31%	23%	19%
Analysis method	41%	42%	31%
Specific instance	15%	10%	24%
Generalization or characterization	12%	20%	25%
Feasibility study or exploration	1%	5%	2%
Contribution			
Theoretical	20%	23%	35%
Technological	68%	52%	40%
Empirical	10%	23%	22%
Perspectival	2%	2%	3%
Validation			
Analysis	20%	20%	17%
Evaluation	56%	54%	60%
Experience	3%	2%	4%
Example	16%	13%	8%
No_Validation	5%	10%	12%

Table 8: Matching Contribution to Topic - ASE

Topic	Contribution			
	Technological	*Theoretical*	*Empirical*	*Perspectival*
Software analysis	53	21	8	-
Maintenance and evolution	18	2	3	1
Data mining for software engineering	11	4	3	-
Program synthesis & transformations	10	3	-	-
Model-driven development	9	2	-	-
Automated reasoning techniques	8	3	-	-
Testing, verification, and validation	8	1	1	-
Software architecture and design	7	1	1	-
Configuration management	4	2	1	2
Empirical software engineering	-	-	8	-
Program comprehension	5	1	1	-
Software product line engineering	2	4	-	-
Model transformations	5	1	-	-
Requirements engineering	4	1	-	-
Re-engineering	5	-	-	-
Domain modeling and meta-modeling	3	1	-	-
Modeling language semantics	2	1	-	-
Component-based systems	2	1	-	-
Specification languages	1	-	1	-
Software visualization	2	-	-	-
Open systems development	2	-	-	-
Computer-supported cooperative work	2	-	-	-
Knowledge acquisition and management	1	-	-	-
Human-computer interaction	-	-	1	-

Table 9: Matching Contribution to Topic - FSE

Topic	Contribution			
	Technological	*Theoretical*	*Empirical*	*Perspectival*
Testing	39	11	14	-
Program analysis	36	4	18	2
Debugging	20	4	4	-
Software evolution and maintenance	16	7	2	-
Empirical software engineering	1	23	-	-
Distributed, parallel, and concurrent software	14	2	5	-
Specification and verification	6	1	12	-
Mobile, ubiquitous, and pervasive software	14	4	1	-
Web-based software	16	-	-	-
Formal methods	3	-	11	-
Security and privacy	7	3	2	-
Human and social factors in SE	1	7	-	4
Development tools and environments	9	2	1	-
Refactoring	5	3	1	-
Programming languages	5	2	2	-
Computer-supported cooperative work	4	4	-	1
Configuration management and deployment	6	1	1	-
Program synthesis	4	1	2	-
Architecture and design	4	3	-	-
Knowledge based software engineering	3	2	1	-
Dependability, safety, and reliability	5	-	1	-
Software product lines	2	1	2	-
Program comprehension and visualization	3	1	1	-
Model-driven software engineering	2	-	3	-
Requirements engineering	1	2	1	-
Patterns and frameworks	1	-	3	-
Autonomic computing and (self-)adaptive systems	3	-	1	-
Traceability	-	1	2	-
Software reuse	3	-	-	-
Software economics and metrics	1	2	-	-
Search-based software engineering	2	1	-	-
Reverse engineering	2	-	1	-
Components, services, and middleware	1	1	1	-
Cloud computing	2	1	-	-
Green computing	2	-	-	-
End-user software engineering	-	1	1	-
Education	1	-	1	-
Crowdsourcing	2	-	-	-
Processes and workflows	-	1	-	-
Policy and ethics	1	-	-	-
Human-computer interaction	-	1	-	-
Embedded and real-time software	1	-	-	-

Table 10: Matching Contribution to Topic - ICSE

Topic	Contribution			
	Technological	*Theoretical*	*Empirical*	*Perspectival*
Software testing	27	29	13	–
Program analysis	23	29	5	2
Debugging, fault localization, and repair	11	38	7	1
Empirical software engineering	–	1	28	5
Tools and environments	6	19	2	1
Mining software engineering repositories	13	9	4	–
Mobile applications	7	16	3	–
Validation and verification	14	6	5	–
Human factors and social aspects of SE	2	1	19	1
Formal methods	18	3	1	–
Security, privacy and trust	7	9	5	–
Cooperative, distributed, and collaborative SE	3	3	11	2
Performance	6	9	3	–
Middleware, frameworks,and APIs	6	9	2	–
Parallel, distributed, and concurrent systems	9	6	–	1
Programming languages	2	7	7	–
Software evolution and maintenance	6	7	2	–
Model-driven engineering	10	3	1	–
Program comprehension	5	2	5	1
Refactoring	3	8	2	–
Software architecture	6	3	2	–
Requirements engineering	4	3	3	–

all covered topics, and even the fewer *Perspectival* papers span over several different topics.

Moreover, in Table 11 we match paper contributions to the tackled problems. The table shows that for the 976 considered papers, the problem of *Generalization or characterization* is more often addressed through an *Empirical* contribution (∼66% for ASE, ∼87% for FSE and ∼81% for ICSE). Instead if we consider the problem of *Feasibility study or exploration*, in FSE and ICSE we find mostly *Empirical* contributions (∼57% and ∼40% respectively), while in ASE most contributions (∼67%) were *Technological*. The problems of *Development* and *Analysis* methods and of *Specific instance* have been most often addressed by *Technological* contributions in both ASE and FSE, whereas in ICSE we can see a more even distribution between *Technological* and *Theoretical*.

6.4. RQ4: What is the main type of validation?

Table 7 also presents the results of classification of type of validation for ASE, FSE and ICSE papers (rows 13-17). The collected data evidenced that over the years 2012-2016 for ASE, FSE and ICSE conferences the most frequent type of validation is *Evaluation* with a percentage of 56%, 54% and 60%, respectively. The second is *Analysis* with 20% for both ASE and FSE, and 17% for ICSE. Validation by *Experience* is the most rarely achieved. Figure 6 further details the distribution of published papers according to the validation per year.

In using the scheme, it can be also interesting both for paper authors and for conference organizers to look at the mapping of contribution types vs. problem

tackled, as shown in Table 12, or of contribution types vs. validation, as in Table 13.

6.5. RQ5: What are, if any, specific patterns or trends?

In previous sections, we reported data about pairwise matchings, e.g., how types of contribution map to the problem addressed (Table 11), or what type of validation corresponds to each problem or contribution type (Tables 12 and 13 respectively).

Another interesting study concerns the patterns from elements belonging to three categorization dimensions (*Problem*, *Contribution*, *Validation*), which could provide insights on the most frequent genres of papers at ASE, FSE and ICSE. For this, we derived all the possible triples combining the values of the above three categorization dimensions obtaining 100 possible combinations. We then counted the occurrences of each combination inside the sets of papers belonging to each conference. For readability, in Table 14 we show the most frequent patterns having percentage greater than 5% over ASE, FSE and ICSE.

In ASE the top 3 most frequent patterns have been: (*Analysis method*, *Technological*, *Evaluation*), (*Development method*, *Technological*, *Evaluation*), (*Analysis method*, *Theoretical*, *Evaluation*) with a percentage of ∼18%, ∼16%, and ∼8% respectively. In other words, among 100 possible patterns, the above three have been followed in ∼42% of the papers.

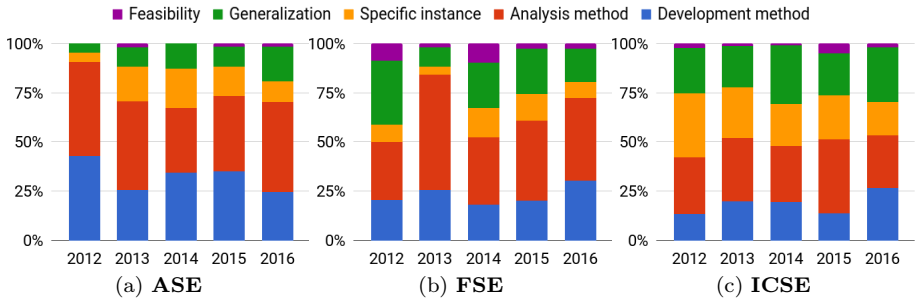
In FSE the top 3 most frequent patterns found are the same of ASE, i.e.: (*Analysis method*, *Technological*, *Evaluation*), (*Development method*, *Technological*, *Evaluation*), (*Analysis method*, *Theoretical*, *Evaluation*) with a percentage of ∼18%, ∼11%, and ∼8% respectively. Altogether the above three patterns cover ∼37% of the papers.

In ICSE the top 3 most frequent patterns are: (*Analysis method*, *Theoretical*, *Evaluation*), (*Specific instance*, *Technological*, *Evaluation*), (*Analysis method*, *Technological*, *Evaluation*) with a percentage of ∼12%, ∼10%, and ∼9% respectively. The three patterns cover ∼30% of the papers.

The above results seem to indicate that conference proceedings in the SE community are de facto converging towards a few paper genres, among the hundred patterns that would be possible in principle. However it is important to notice that the results in Table 14 are only limited to 5 editions from 3 conferences and larger studies would be needed to understand whether the results we observed would be generalizable. Limited to ASE, FSE and ICSE we found that a very frequent genre across the three conferences is a paper that presents a technological solution to a problem related with software analysis/evaluation/quality/correctness and performs an evaluation of the solution validity. Also very frequent is the twin type of paper (technological solution validated by evaluation) that tackles a problem relative to software development rather than analysis. As a last example, a high proportion of papers propose a theoretical result for analysis and evaluate it. We can't say whether this high concentration of paper types on a few patterns happens because these are the majority of papers that authors submit, or because these patterns are those that

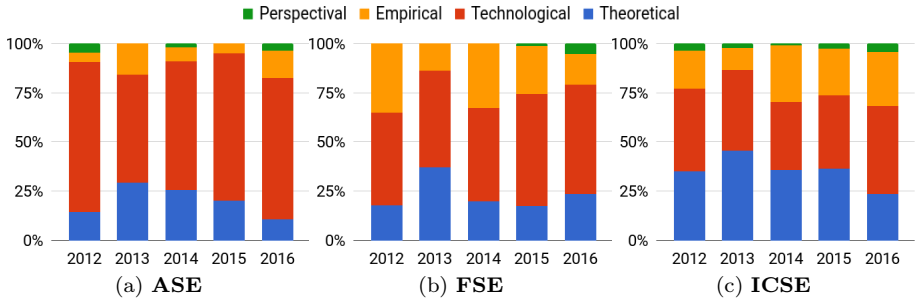
are more frequently accepted, because we could not look at submitted papers. Or, perhaps, we should get a bird's-eye view and try to reason on the respective meaning of the 100 combinations of dimensions, to understand whether they would all make sense in SE research.

In this respect, while in Table 14 we showed the most common paper genres, the categorization may be also informative to identify which patterns were never found among the 976 paper we studied. Perhaps we should acknowledge that some patterns that we did not find would indeed be unlikely. For example, it is not expected that one paper that belongs to *Feasibility study or exploration* could include a validation based on *Experience*. On the other hand there are several well plausible patterns that we did not find. Strangely enough, we did not identify any paper that provides an *Empirical* contribution in addressing (*Development method*), whereas we have several empirical papers concerned with (*Analysis method*). Why do SE researchers publish empirical studies concerned with testing, analysis, measuring, but not with design, specification, modeling? On the opposite site, we did not find any paper across the three conferences that proposes *Perspectival* contribution to solve problems related with *Analysis method*: again we don't know if this is because SE researchers in analysis and testing do not try novel perspectives or if such papers do not pass the selection process.



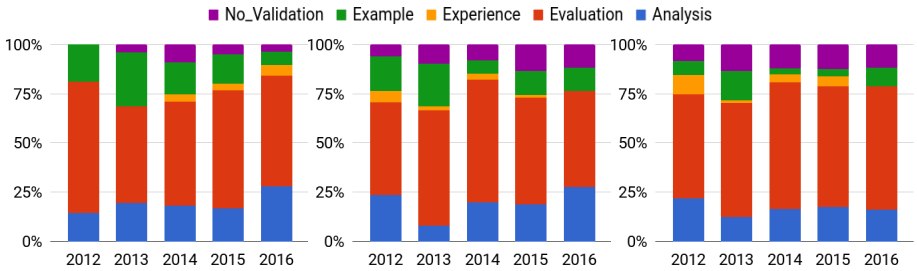
(a) **ASE** (b) **FSE** (c) **ICSE**

Figure 4: Problem per year



(a) **ASE** (b) **FSE** (c) **ICSE**

Figure 5: Contribution per year



(a) **ASE** (b) **FSE** (c) **ICSE**

Figure 6: Validation per year

Table 11: Matching Contribution to Problem

	Problem	Contribution			
		Empirical	*Perspectival*	*Technological*	*Theoretical*
ASE	Development method	-	1.32%	86.84%	11.84%
	Analysis method	3.00%	-	67.00%	30.00%
	Specific instance	5.56%	-	72.22%	22.22%
	Generalization	65.52%	10.34%	17.24%	6.90%
	Feasibility	-	-	66.67%	33.33%
FSE	Development method	-	-	75.36%	24.64%
	Analysis method	3.25%	-	62.60%	34.15%
	Specific instance	16.67%	3.33%	66.67%	13.33%
	Generalization	86.67%	5.00%	5.00%	3.33%
	Feasibility	57.14%	7.14%	14.29%	21.43%
ICSE	Development method	-	-	57.32%	42.68%
	Analysis method	3.76%	-	40.60%	55.64%
	Specific instance	1.94%	0.97%	63.11%	33.98%
	Generalization	80.56%	7.41%	6.48%	5.56%
	Feasibility	40.00%	30.00%	10.00%	20.00%

Table 12: Matching Validation to Problem

	Problem	Validation				
		Analysis	*Evaluation*	*Experience*	*Example*	*No_ Validation*
ASE	Development method	21.05%	56.58%	2.63%	15.79%	3.95%
	Analysis method	16.00%	64.00%	2.00%	16.00%	2.00%
	Specific instance	16.67%	50.00%	-	25.00%	8.33%
	Generalization	34.48%	37.93%	6.90%	6.90%	13.79%
	Feasibility	33.33%	-	33.33%	33.33%	-
FSE	Development method	21.74%	56.52%	-	17.39%	4.35%
	Analysis method	20.33%	62.60%	-	13.01%	4.07%
	Specific instance	3.33%	56.67%	3.33%	23.33%	13.33%
	Generalization	25.00%	36.67%	6.67%	5.00%	26.67%
	Feasibility	21.43%	42.86%	7.14%	7.14%	21.43%
ICSE	Development method	8.54%	67.07%	3.66%	13.41%	7.32%
	Analysis method	17.29%	69.92%	1.50%	9.02%	2.26%
	Specific instance	15.53%	66.02%	5.83%	7.77%	4.85%
	Generalization	24.07%	39.81%	5.56%	1.85%	28.70%
	Feasibility	10.00%	30.00%	-	-	60.00%

Table 13: Matching Validation to Contribution

	Contribution	Validation				
		Analysis	*Evaluation*	*Experience*	*Example*	*No_ Validation*
ASE	Theoretical	28.00%	46.00%	2.00%	20.00%	4.00%
	Technological	15.06%	62.05%	1.81%	17.47%	3.61%
	Empirical	33.33%	41.67%	4.17%	4.17%	16.67%
	Perspectival	50.00%	-	50.00%	-	-
FSE	Theoretical	23.53%	50.00%	-	19.12%	7.35%
	Technological	16.23%	64.94%	1.30%	14.29%	3.25%
	Empirical	24.64%	37.68%	5.80%	5.80%	26.09%
	Perspectival	20.00%	20.00%	-	-	60.00%
ICSE	Theoretical	17.11%	68.42%	1.32%	9.87%	3.29%
	Technological	12.64%	69.54%	4.60%	9.20%	4.02%
	Empirical	25.51%	37.76%	7.14%	1.02%	28.57%
	Perspectival	-	-	-	8.33%	91.67%

Table 14: Patterns percentage

	Problem	Contribution	Validation	Percentage
ASE	Analysis method	Technological	Evaluation	18,03%
	Development method	Technological	Evaluation	16,39%
	Analysis method	Theoretical	Evaluation	7,79%
	Specific instance	Technological	Evaluation	6,15%
	Analysis method	Technological	Example	5,33%
	Development method	Technological	Analysis	5,33%
FSE	Analysis method	Technological	Evaluation	17,57%
	Development method	Technological	Evaluation	11,15%
	Analysis method	Theoretical	Evaluation	8,11%
	Generalization	Empirical	Evaluation	6,42%
	Generalization	Empirical	No_Validation	5,07%
ICSE	Analysis method	Theoretical	Evaluation	11.93%
	Specific instance	Technological	Evaluation	9.63%
	Analysis method	Technological	Evaluation	8.72%
	Development method	Technological	Evaluation	8.03%
	Generalization	Empirical	Evaluation	7.57%
	Specific instance	Theoretical	Evaluation	5.73%
	Generalization	Empirical	No_Validation	5.50%
	Generalization	Empirical	Analysis	5.28%

6.6. Threats to validity

This section discusses threats to the internal, external and construct validity of the study presented in this paper. Concerning the internal validity, i.e., the amount of confidence on the reported results, different aspects can be considered:

Authors' expertise. Our own expertise may have influenced the individual coding results during the classification process. To reduce this risk, the classification process forces a random assignment of each paper to two of the authors and includes the participation of a third author as an independent arbiter to solve coding disagreements.

Lack of independence. All the authors contributed both to the definition of the categorization scheme and to the classification study. This is an intrinsic threat of this type of studies, and no mitigation has been performed. To prevent this threat an independent classification should be planned.

Scheme. The proposed paper categorization scheme is the result of merging, extending and revising the types and definitions provided in previous relevant works. The resulting scheme could not be the most appropriate for SE conferences. The threat has been mitigated by applying an iterative process as described in Section 4. However, future improvements and revisions of the proposed paper categorization scheme by the community are encouraged.

Source data. The proposed paper categorization scheme has been applied to the papers published in ASE, FSE and ICSE in five years. The analysis of further conferences or broader periods could have provided different results. However, because the 3 selected conferences are acknowledged as top-tier conferences in SE field, we consider the sample to be a good representative in terms of variety, structure and scope of the SE conference papers. On the other hand, also specific conferences focusing on sub-fields of SE research could be considered to get a more comprehensive and flexible scheme. We are planning as a future work the application of the paper categorization scheme to different conferences.

Trustworthiness. It could be that what is claimed in the abstract, title, and keywords is not true. This threat could rarely occur, as the papers have all undergone an extensive review process before being published. However, we did observe that the abstracts can be incomplete, as we discussed concerning papers classified as *No_ Validation*. To mitigate this threats a check against full text was performed limited to papers without validation.

Forcing classification. We forced the selection of one main group for *problem*, *contribution* and *validation*. This might have influenced the observed results, as in some cases a paper could be classified with different values for a dimension. For instance, it could be the case that a paper provides a contribution that could be classified both as technological and theoretical. To reduce this threat, during the classification process, a consensus among three authors (the two evaluators and the arbiter) was sought for each paper. However, we acknowledge that

the threat remains and a study allowing for example the choice of more values should be performed.

Informal definitions. The proposed categorization scheme relies on a narrative definition of terms that might allow for different interpretation from different users. In this study this risk was dealt with by several workshop meetings. In general the inclusion of examples aiming to clarify the values assigned to each dimension might mitigate the risk.

External validity threats concern potential issues that may prevent the generalization of the results. In the presented study only papers published in three conferences along 5 years have be considered. Therefore, the results might not be a good representative of SE conferences. The risk is mitigated since 976 papers over 5 years could be considered a good representative of the current software engineering research trends in SE. However, the generalization based only on a subset of papers represents an issue concerning several empirical contributions, including ours.

With respect to construct validity, i.e., threats regarding the extent of the utilized approach may concern the source data and the summary data collected. Making paper categorization by reading only title, abstract and keywords, may have influenced the findings: considering the entire paper content might have produced different data and would have certainly provided more accurate results. However, we followed the procedure adopted in previous similar studies, e.g., [6], assuming that the abstract was a good representative of paper type. Mitigation of this threat has been done only for papers classified as *No_ Validation*, in which case two of the authors read the entire paper and re-assessed that validation value (see next section). Concerning the metrics we adopted to analyze the categorization results, we provide only simple percentage and ranking figures, which are the most commonly used in empirical analysis and do not represent per se a potential threat.

7. Discussion and Conclusion

This section puts on the table some interesting discussion points.

Investigated topics. It is outside the scope of the paper to draw inferences about trends observed in SE research topics over the years, even because we would need to enlarge our observation to be able to infer valid conclusions. However, observing such scores may be useful to conference organizers when the program committee members are selected and when submissions are invited. In addition to the CFP, authors will likely look at proceedings of previous years before deciding whether a conference is a good venue for their research topic. Therefore, if a relevant SE topic is not represented, organizers might consider to provide more explicit invitations to authors working in those topics, and to enhance the expertise in the program committee.

While we had only access to published papers, it would also be interesting to investigate the relation between the topics of submitted papers against those of accepted ones. Mapping such comparisons on the expertise of reviewers might for instance help understanding whether a conference edition provided fair treatment to all topics, or otherwise the process can be improved for better comprehensiveness.

Scheme improvement. As we said in the threats to validity, our categorization scheme is based on informal descriptions and we understand, as we ourselves noticed during the study, that it may not be always easy to choose between two different values. More specifically, the cases where we found potential overlaps most often included:

- Validation through Analysis or through Evaluation: perhaps a clearer distinction should be forced in the definition. However, again, the cause is often in the abstract wording.
- Specific instance vs. Development or Analysis, given that on the specific instance the authors apply a development or analysis. We wonder whether it is worth keeping the distinction (originally introduced in Shaw's scheme) or drop it.
- Theoretical vs. Technological: for the nature of SE papers, often a theoretical methodology is the solution implemented into a tool and hence a paper often addresses both type of problems.

The provided scheme is a best effort solution, but improvements are certainly possible.

Paper validation. In recent years awareness has grown within the SE community that before a paper can be published convincing evidence should be provided that the presented results can be trusted. Having said this, we have been somehow surprised to still find a percentage of papers in all the three conference supposedly having no validation. As we explained in Section 5.3, we classified papers based on abstract, and then we suspected that in some cases the abstract could not tell the whole story. Thus, to better understand this issue, we decided to read the full text for the 94 papers (specifically 12 in ASE + 31 in FSE + 51 in ICSE) classified as *No_ Validation*. From this follow up analysis, we discovered that indeed many of them really do no provide any validation (48%), while the remaining ones actually include a kind of validation but do not mention it in the abstract. The detailed numbers are reported in Table 15.

Concerning the 45 papers really including no validation, we notice that for all of them the contribution was classified either as *Perspectival* or as *Empirical*. In a sense, for these cases the lack of a validation could be in line with the nature of the papers: concerning perspectival papers, these are expected to provide breakthrough novel ideas with good arguments, but not necessarily sustained by data or analyses. Concerning papers whose main contribution is an empirical study, a reflection is needed. By their very nature the results shown in an empirical paper

are founded on objective data and as such, as far as the number manipulation is sound, they can be considered trustable. We found anyhow many papers whose contribution is empirical that corroborate further the data with careful statistical analyses (we classify these as *Analysis*) or with further validation, e.g. by asking practitioners if they agree with the reported data (we classify these as *Evaluation*). The remaining cases of papers classified as *No_ Validation* on the basis of abstract, but including in the full text an explicit validation, evidence an incompleteness or inaccuracy by the authors in writing the abstract itself. This observation also supports our following recommendation concerning abstract writing good practice.

Table 15: Actual validation in the 94 papers originally classified as No_Validation

Validation	ASE	FSE	ICSE
Analysis	3	7	7
Evaluation	5	4	9
Experience	-	-	3
Example	2	5	4
No_Validation	2	15	28

Abstract. The increasing use of abstract readings for secondary studies evidences the importance of very clearly summarizing the paper content. This is in line with what Shaw already claimed, that "...people judge papers by their abstracts and read the abstract in order to decide whether to read the whole paper. *It's important for the abstract to tell the story.*" [6]. In this direction, our study also addresses the research question in Section 5.3: *Do abstracts of ASE, FSE and ICSE papers carry sufficient information for understanding the paper type?* In our experience, many abstracts clearly allowed us to identify the type of problem, contribution, validation as well as the addressed research topics. However, some abstracts did not carry a complete view of the paper content, e.g., they do not provide either a clear distinction between *Technological* and *Theoretical* for the main contribution, or the adopted type of validation. In [26], Budgen and coauthors show that the use of structured abstracts can provide improved readability and make it easier to understand the paper content. The proposed structure for abstracts in [26] includes: *Background*, *Aiming*, *Method*, *Results* and *Conclusions* and provides a description of each of them (Table 3 of [26]). Our proposed categorization scheme can further contribute to improve structured abstracts. In particular, the types of problems, contributions and validation defined in Section 4 integrate the descriptions of *Aiming*, *Method* and *Results* respectively. Specifically, for the *Results* part, it is important to include an explicit reference to the type of validation adopted, which, as evidenced by our study, is missing in several papers. This recommendation is again in line with what reported in [6]: "*...Acceptance rates were highest for papers whose abstracts indicate that analysis or experience provides evidence in support of the work*".

The abstract of the present paper has been conceived as an example of the integration of our categorization scheme with structured abstract.

In conclusion, we hope that our proposed categorization scheme can provide insights for both authors and reviewers of SE conferences in their respective roles to shape the future of the conferences with increasing quality and impact. We believe that a meta-analysis of how we report our research results (are we doing well, how can we do better?) is a fundamental activity for the discipline, as other research fields have long recognized. On our side, we plan to extend the present study by considering other venues. One open question is whether the scheme that has been originally conceived for conference papers could also be applied to journal articles. As journal submissions generally allow for more relaxed space and more relaxed evaluation, we do not know if this might have an impact on paper structure and require a different scheme, or instead our same scheme could still fit. Finally, we described in this paper in detail the process and reasoning we followed in the hope that other colleagues will improve our proposal, as only a scheme shared within the community may have validity.

Acknowledgements

Breno Miranda wishes to thank the postdoctoral fellowship jointly sponsored by CAPES (Coordination for the Improvement of Higher Education Personnel) and FACEPE (Foundation for Science and Technology Development of the State of Pernambuco) (APQ-0826-1.03/16; BCT-0204-1.03/17).

References

[1] K. N. Nwogu, The medical research paper: Structure and functions, English for Specific Purposes 16 (2) (1997) 119 – 138. doi:https://doi.org/10.1016/S0889-4906(97)85388-4.
URL http://www.sciencedirect.com/science/article/pii/S0889490697853884

[2] Y. Ruiying, D. Allison, Research articles in applied linguistics: structures from a functional perspective, English for Specific Purposes 23 (3) (2004) 264 – 279. doi:https://doi.org/10.1016/S0889-4906(03)00005-X.
URL http://www.sciencedirect.com/science/article/pii/S088949060300005X

[3] K.-J. Stol, B. Fitzgerald, A holistic overview of software engineering research strategies, in: Proceedings of the Third International Workshop on Conducting Empirical Studies in Industry, CESI '15, IEEE Press, Piscataway, NJ, USA, 2015, pp. 47–54.
URL http://dl.acm.org/citation.cfm?id=2819303.2819319

[4] M. Montesi, P. Lago, Software engineering article types: An analysis of the literature, J. Syst. Softw. 81 (10) (2008) 1694–1714. doi:10.1016/j.jss.

2007.11.723.
URL http://dx.doi.org/10.1016/j.jss.2007.11.723

[5] R. Wieringa, N. Maiden, N. Mead, C. Rolland, Requirements engineering paper classification and evaluation criteria: A proposal and a discussion, Requir. Eng. 11 (1) (2005) 102–107.

[6] M. Shaw, Writing good software engineering research papers: Minitutorial, in: Proceedings of the 25th International Conference on Software Engineering, ICSE '03, IEEE Computer Society, Washington, DC, USA, 2003, pp. 726–736.
URL http://dl.acm.org/citation.cfm?id=776816.776925

[7] R. Glass, I. Vessey, V. Ramesh, Research in software engineering: an analysis of the literature, Information and Software Technology 44 (8) (2002) 491 – 506. doi:http://dx.doi.org/10.1016/S0950-5849(02)00049-6.
URL http://www.sciencedirect.com/science/article/pii/S0950584902000496

[8] M. V. Zelkowitz, D. Wallace, Experimental validation in software engineering, Information and Software Technology 39 (1997) 735–743.

[9] C. Theisen, M. Dunaiski, L. Williams, W. Visser, Writing good software engineering research papers: Revisited, in: 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C), 2017, pp. 402–402. doi:10.1109/ICSE-C.2017.51.

[10] J. P. Ioannidis, D. Fanelli, D. D. Dunne, S. N. Goodman, Meta-research: evaluation and improvement of research methods and practices, PLoS biology 13 (10) (2015) e1002264.

[11] A. Bertolino, A. Calabro, F. Lonetti, E. Marchetti, B. Miranda, What paper types are accepted at the International Conference on Software Engineering?, in: 2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C), 2017, pp. 399–401. doi:10.1109/ICSE-C.2017.50.

[12] P. J. Runkel, J. E. McGrath, Research on Human Behavior: A Systematic Guide to Method, Holt, Rinehart and Winston, Inc., 1972.

[13] D. I. K. Sjoeberg, J. E. Hannay, O. Hansen, V. B. Kampenes, A. Karahasanovic, N. K. Liborg, A. C. Rekdal, A survey of controlled experiments in software engineering, IEEE Transactions on Software Engineering 31 (9) (2005) 733–753. doi:10.1109/TSE.2005.97.

[14] C. Zannier, G. Melnik, F. Maurer, On the success of empirical studies in the international conference on software engineering, in: Proceedings of the 28th International Conference on Software Engineering, ICSE '06, ACM, New York, NY, USA, 2006, pp. 341–350. doi:10.1145/1134285.1134333.
URL http://doi.acm.org/10.1145/1134285.1134333

[15] C. Ghezzi, Reflections on 40+ years of software engineering research and beyond: an insider's view, [Online; accessed 24-August-2016] (2009).
URL http://www.slideshare.net/carloghezzi18/icse-2009-keynote-15919951

[16] M. Montesi, J. M. Owen, From conference to journal publication: How conference papers in software engineering are extended for publication in journals, Journal of the American Society for Information Science and Technology 59 (5) (2008) 816–829. doi:10.1002/asi.20805.
URL http://dx.doi.org/10.1002/asi.20805

[17] T. Systä, M. Harsu, K. Koskimies, Inbreeding in software engineering conferences (2012).
URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.361.7040&rep=rep1&type=pdf

[18] B. Vasilescu, A. Serebrenik, T. Mens, M. G. van den Brand, E. Pek, How healthy are software engineering conferences?, Science of Computer Programming 89, Part C (2014) 251 – 272. doi:http://dx.doi.org/10.1016/j.scico.2014.01.016.
URL http://www.sciencedirect.com/science/article/pii/S0167642314000318

[19] P. Bourne, Ten simple rules for getting published, PLOS Computational Biology 1 (5).
URL http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.0010057

[20] J. Swales, Genre Analysis: English in Academic and Research Settings (Cambridge Applied Linguistics), Cambridge University Press, 1990.

[21] R. Holmes, Genre analysis, and the social sciences: An investigation of the structure of research article discussion sections in three disciplines, English for Specific Purposes 16 (4) (1997) 321 – 337. doi:http://dx.doi.org/10.1016/S0889-4906(96)00038-5.
URL http://www.sciencedirect.com/science/article/pii/S0889490696000385

[22] C. R. Miller, Genre as social action, Quarterly Journal of Speech 70 (1984) 151–76.

[23] Systems and software engineering – vocabulary, ISO/IEC/IEEE 24765:2010(E) (2010) 1–418doi:10.1109/IEEESTD.2010.5733835.

[24] J. R. Landis, G. G. Koch, The measurement of observer agreement for categorical data, biometrics (1977) 159–174.

[25] M. F. Porter, An algorithm for suffix stripping, Program 14 (3) (1980) 130–137.

[26] D. Budgen, A. J. Burn, B. Kitchenham, Reporting computing projects through structured abstracts: a quasi-experiment, Empirical Software Engineering 16 (2) (2011) 244–277. doi:10.1007/s10664-010-9139-3.
URL http://dx.doi.org/10.1007/s10664-010-9139-3