

Example Rewrite of an Abstract for a Journal Submission

The original abstract has 114 words.

Both software testing and programming foundations are not easy subjects to be taught. In case of programming, experiences have suggested that the mastering of testing can contribute to enhance the students' skills of comprehension and analysis. So, automatic test-based assessment systems have been developed and adopted in programming courses. Aiming at exploring the adoption of assessment systems also in the teaching of software testing, this paper introduces `ProgramTester`—a tool for the submission and automatic evaluation of programming assignments based on testing activities. Results from two experiments involving the use of `ProgramTester` in testing courses are presented, providing evidences on the feasibility of its adoption to support the teaching of testing concepts and criteria.

Consider the first sentence:

Both software testing and programming foundations are not easy subjects to be taught.

Changes:

- It is unclear what “programming foundations” means that is different from “programming”. So drop “foundations”.
- “Testing and programming” reverses the order of those phases of software development. Put them in temporal order.
- The negative phrase “not easy” is more straightforward as positive phrase “difficult”. The latter is stronger and seems appropriate in the sentence.
- The phrase “subjects to be taught” is simpler as just “to teach”.

Suggested rewrite:

Programming and software testing are both difficult to teach.

Now consider the second sentence:

In case of programming, experiences have suggested that the mastering of testing can contribute to enhance the students' skills of comprehension and analysis.

Questions:

- Whose experiences? The authors, other observers, or the community at large?
- Who is doing the “mastering”? “
- “Comprehension and analysis” of what?
- By “experiences have suggested” do the authors mean “we observe” a fact or “our observations suggest” a hypothesis you plan to test? (But you do not seem to support this hypothesis in your work.) Or do you mean something different from this?

Changes:

- Turn the abstract nouns “mastering”, “comprehension”, and “analysis” into strong verbs.
- Change passive voice sentence to active voice.
- Comprehension and analysis seem to refer to programming, so add “programs” as the new verb’s objects.
- Add clause “But they are linked” to replace “in the case of programming”.
- Eliminate clutter “have suggested that” and “can contribute to enhance”.
- Add “than those who do not” to emphasize implied comparison.

Suggested rewrite:

But they are linked; students who master testing can more effectively analyze and comprehend programs than those who do not.

Consider the third and fourth sentences:

So, automatic test-based assessment systems have been developed and adopted in programming courses. Aiming at exploring the adoption of assessment systems also in the teaching of software testing, this paper introduces ProgramTester—a tool for the submission and automatic evaluation of programming assignments based on testing activities.

Changes:

- Remove redundancy and clutter to combine sentences into one.
- Use active voice.
- Tie back to “linkage” concept in previous sentence.

Suggested rewrite:

To exploit this linkage, we developed ProgramTester—a tool that evaluates student programs using software testing techniques—and developed methods to use it in both programming and software testing courses.

Consider sentence five:

Results from two experiments involving the use of ProgramTester in testing courses are presented, providing evidences on the feasibility of its adoption to support the teaching of testing concepts and criteria.

Questions:

- What does “testing courses” mean? Are they courses on testing (noun) or are we testing (verb) courses? Likely it is the former.
- What is meant by “concepts and criteria”. Are these “testing concepts and testing criteria” or some other kind of criteria? What are the significant differences between “testing concepts” and “testing criteria”?

Changes:

- Change to active voice using “we” as subject.
- Break into two sentences.
- Dropped explicit reference to testing since it is clear from previous sentence.

Suggested rewrite:

To determine the efficacy of this approach, we used ProgramTester in two different courses. This paper describes the use of ProgramTester and the results of these experiments.

The original abstract has 114 words.

Both software testing and programming foundations are not easy subjects to be taught. In case of programming, experiences have suggested that the mastering of testing can contribute to enhance the students' skills of comprehension and analysis. So, automatic test-based assessment systems have been developed and adopted in programming courses. Aiming at exploring the adoption of assessment systems also in the teaching of software testing, this paper introduces ProgramTester—a tool for the submission and automatic evaluation of programming assignments based on testing activities. Results from two experiments involving the use of ProgramTester in testing courses are presented, providing evidences on the feasibility of its adoption to support the teaching of testing concepts and criteria.

The rewritten abstract has 87 words. It is shorter than the original, but I think conveys the essence of the original in a clearer and more direct manner. Of course, it should be edited further, especially if I unintentionally distorted the meaning of the original.

Programming and software testing are both difficult to teach. But they are linked; students who master testing can more effectively analyze and comprehend programs than those who do not. To exploit this linkage, we developed ProgramTester—a tool that evaluates student programs using software testing techniques—and developed methods to use it in both programming and software testing courses. To determine the efficacy of this approach, we experimented with ProgramTester in two different courses. This paper describes the use of ProgramTester and the results of these experiments.

The remainder of the paper should be edited in a similar way.

Purpose of Paper

The overall work is interesting, but more thought needs to be put into how you present the results—for example, about what the purpose of the paper is. What you say in the Abstract, in the Introduction, and in the Conclusions do not seem consistent.

In the last paragraph of the Introduction, you state that the “main” goal is introduce your tool and illustrate its usage in teaching. You accomplish this goal, but it is not a interesting one from a scientific/engineering perspective. You want argue from the results of your study that there is novelty and significance in the tool and its use. Hopefully, there are results that transcend the specifics of the tool you develop and are applicable to other situations. (That is, there is a contribution to the “science”.)

In the next to last paragraph of the Introduction you mention other goals related to increased program comprehension and increased software quality as a result of using `ProgTest`. It is unclear which of these goals you address in your study.

In section 4, you give two “experiments” in use of the tool. You use the tool in two classes, gather data, and examine the data in detail. What are the hypotheses for your experiments? Does your data support or refute your hypotheses? Given that you do not mention control groups or data from other experiments using other tools and methods, concluding that the use of the tool results in an improvement seems problematic.