Some Advice on Writing a Technical Report

Alan T. Sherman
DRAFT: April 27, 1996

The *Technical Report (TR)* is a common written form through which computer scientist communicate their findings. Each TR should have a focused topic that is developed logically along some clearly identified perspective. The major components of a TR are title, author information, date, keywords, informative abstract, body, acknowledgments, references, and appendices. Typically, the body is organized into four sections: motivation, methods, results, and discussion. This document offers advice and specifications for writing TRs.

Overview

Communicating results is a crucial aspect of doing research. Through such communication other people can learn about and benefit from the findings. Often such communication includes a written document known as a *Technical Report (TR)*. The successful researcher must master this important written form.

A TR should explain what you did, why you did it, what you discovered, and what is significant of your findings. The report should identify clearly what is novel about your work, and how it relates to prior knowledge. There should be a focused topic, and an attitude about this topic. The topic should be developed according to the attitude in a thorough, logical, and orderly fashion. Throughout, the author should be helpful to the reader.

The report should include the following components: descriptive title, author name and affiliation, date, informative abstract, list of keywords, body, acknowledgments, and list of references. Additional separate appendices, where appropriate, may also be included. The standard four-part outline for the body of a technical report is motivation, methods, results, and discussion.

There is no minimum or maximum length requirement--the length should be appropriate for what you have to say. Many TRs are about 10--20 pages long, but it is not uncommon for TRs to be significantly longer. Regardless of length, it is usually an effective strategy to explain in successive ``layers." For example, lengthy TRs often begin with a relatively short overview section for readers who wish an executive summary. Quality and conciseness, not quantity, will be rewarded.

This document aims to help students learn the basics of computer science technical reports. Although my advice is highly subjective, I hope the reader will benefit from issues raised, even if she disagrees with my particular point of view. Although technical writing is a crucial part of writing TRs, this document is not a tutorial in effective technical writing. Several sources of information about technical writing are listed in the references. The rest of this document describes the following important aspects of TRs: thesis, components, organization, delivery formats, special advice for experimental projects, common mistakes to avoid, additional advice, and other important communication forms.

The Thesis

Every TR should have a *thesis*--a topic together with an attitude about the topic. The attitude helps focus the subject and provide a framework along which the topic subject can be explored. For example, the topic might be partitioning algorithms for the geometric Steiner tree problem, and the attitude might be that Steele's theory of Euclidean functionals provides a powerful tool for analyzing the performance of such approximation algorithms. The introduction of each TR should clearly identify its thesis and an organizational plan for developing the thesis.

Many researchers find it useful to think in terms of questions and answers. I recommend that you carry out and communicate your research by raising and answering focused questions. For example, you might ask ``What are the performance limits of polynomial-time approximation algorithms?'' or ``What does the theory of probabilistic proof checking say about such performance limits?''

The Components

A technical report should include each of the following items:

1. A logical, accurate, descriptive, and grammatically correct title. Please note: the title ``CMSC 441 Course Project'' is not descriptive.

   Titles should be as short as possible, while still satisfying the foregoing criteria. Avoid cute titles that violate these criteria. I often like two-part titles because they provide short and long forms (e.g. ``Statistical Techniques for Cryptanalysis: An Experimental Study using Real and Simulated English''). I try to avoid titles that exceed 17 words.

2. Author name and affiliation, and date. For example, your affiliation might be ``Department of Computer Science and Electrical Engineering, University of Maryland Baltimore County.'' You might also like to include the city and state of your affiliation, your email address, and a URL to your home page. UMBC students: Please note that there is no punctuation in your University name.

3. An informative abstract of approximately 200 words. Make sure that your abstract is informative---your abstract should serve as a substitute for your paper. Briefly summarize your main findings. Concretely summarize; do not introduce. Immediately get to the point in the first sentence. Do not cite any references in the abstract, and do not begin the abstract with the weak, hackneyed, and boring phrase ``This paper ...''. The abstract should be informative yet understandable to most researchers in your general field. I like the abstract to fit on one title page, including the title, author name and affiliation, date, and list of keywords.

4. A list of appropriate keywords. These keywords should identify the field of your report and its major topics. Choose keywords to be helpful to researchers in locating your work in document-retrieval systems. What words and phrases should someone use to find your report? Be specific, and use only standard phrases. Browse some computer science journal (e.g. Journal of Algorithms) to get a feel for what is an appropriate keyword and what is not. *Computing Reviews* publishes an annual classification system including keywords which many journals follow.

   Many journals use three levels of keywords: general terms (e.g. cryptology), subject descriptors (e.g. differential cryptanalysis) recognizable to most researchers, and implicit terms--specific words or phrases that act as proper names (e.g. RSA Cryptosystem) which might not be recognizable to all readers.

5. Body of technical report. Write a clear, informative, and thoughtful description and critique of what you did. Where appropriate, include carefully drawn graphs and diagrams. Be sure to motivate, present, and interpret your findings.

   Focus on the scientific content of the project--your questions and answers. Identify and explain interesting and important phenomena. Emphasize what is new about your project. In addition, briefly comment on the engineering aspects of your work: what problems did you face, what decisions did you make, and what are the consequences of these decisions? Although it is crucial to explain your experimental procedures, be concise and do not bore your reader with lengthy descriptions of routine implementation concerns.

   Pay attention to important transitional sentences, especially the first and last sentences of the report. There are three standard ways to begin the introduction: startling statement, dramatic incident, and quotation. I like to end each report with a powerful sentence that concisely summarizes the significance of the entire project.

6. Acknowledgments. Acknowledge any help you received, including any use of computer equipment. Be specific.

7. Complete and accurate list of references cited in the technical report. There are three reasons for citing works: to give credit where credit is due, to be helpful to the reader to identify useful related work, and to identify he context and background of your work. Adopt a bibliographic style used by some major refereed computer science journal (e.g. use the style for the \it Journal of the ACM\/).

   I like to list and number references by alphabetical order of author name. When citing references in the body of the report, always explain why the reference is being cited. For example, do not cite previous work without critically explaining how it relates to your work. I like to mention the author name in the textual citation, followed by the corresponding reference number (e.g. ``In 1976, Diffie and Helman [14] proposed the concept of public-key cryptography.").

8. Appendices for supplemental information and for information that is too detailed or voluminous to fit into body the of the technical report. For example, if your project involves any computer programming, you should include a nicely documented and formatted listing of all source code you wrote.

Organization

Although you are free to organize your report in any way you see fit, I highly recommend that you organize the body of your report along the following standard outline for scientific papers:

- motivation
- methods
- results
- discussion.

In thinking about organization, I find it helpful to separate logical organization from explicit numbered sectioning. For logical organization, I think in terms of hierarchies. Part of the organizational task is to embed the logical organization into numbered sections. For example, the logical introduction might include one or more numbered sections, depending on what needs to be said. A short report might begin with one section: 1. Introduction. A longer report might begin with a more elaborate logical introduction consisting of four numbered sections: 1. Introduction 2. Overview 3. Background 4. Previous work. As the report evolves you may wish to modify the organization.

In describing the purpose of your project, restrict yourself to scientific and engineering reasons; do not discuss reasons that are related only to school. Do not repeat sentences from the abstract ver batim.

In the conclusion, you should explain what it all means to you. If you discuss philosophy, do so in the discussion section.

Special Notes on Experimental Work

Be sure to explain your procedures, to present your results, and to interpret your results. Summarize your findings in meaningful ways, visualizing important data (e.g. in graphs) whenever possible.

If you are experimentally measuring the running time of a computer program, test your program on many randomly chosen inputs of a variety of sizes, including large inputs. Since the behavior of your program might vary significantly among inputs of the same size, for each input size, try several inputs of that size and report the sample mean and standard deviation for that size; do not simply try one input per size.

Be sure to explain your procedures in sufficient detail so that other researchers can verify and replicate your findings.

Delivery Forms

I prefer to receive TRs on standard 8.5 x 11 inch paper, with exactly one staple in the upper-left corner, without any cover. Please hand in any source code or other appendices separately, not attached in any way to the main report. I prefer not to receive technical reports in binders, which simply add bulk.

Print any source code on 8.5 x 11 inch paper with carefully inserted pagebreaks. If your listing comes out of the printer with the pages attached to each other, then burst the listing (i.e. separate the pages and arrange them like pages in a book) before handing in the listing.

In addition, I recommend that you make your report available on the WWW using your favorite hypertext language (e.g. html).

Evaluation

I evaluate each TR on the basis of its scientific merit, effective presentation, and appropriateness for assignment. I reward thorough analysis, originality, and insightfulness. Scientific merit includes correctness, significance, novelty, nontriviality, and completeness.

I evaluate source code on the basis of its correctness, completeness, design, modularity, documentation, coding, user interface, and testing.

Common Mistakes to Avoid

Adhere carefully to the following guidelines:

1. In the introduction of your report, clearly identify a focused well-defined question. Answer this question in the rest of your report.

2. Analyze and interpret your data, and discuss the significance and limitations of your findings. Do not simply report your data.

3. Be sure that your technical report is complete in the sense that it has each of the following components: descriptive title, author name and affiliation, date, informative abstract, list of keywords, body, acknowledgments, and references.

4. In your abstract, specifically and concretely state your findings; do not vaguely describe what you set out to do. Your abstract should summarize, not introduce. Do not begin your abstract with the hackneyed phrase ``This paper."

Additional Advice

Start early and do not wait until the last moment. Expect system downtime and personal illness, especially the week before the project is due.

I strongly recommend that you prepare your written report using a document preparation system. Such systems enable you to edit your document the numerous times required to create excellent prose. In addition, they will enable you to produce high-quality printed output. If you do not already know such a system, now is a good time to learn. I recommend the Latex system because it and its relative TEX produce high-quality results for mathematical typesetting, and Latex provides high-level document support (e.g. indexing and cross referencing). As for text editors, I recommend Gnu Emacs because it is powerful, extensible, customizable, and self-documenting.

For drawing graphs and doing statistical analysis of your data, I recommend the program xmgr, which runs on the SGI workstations.

Whenever working on a large project, you should save your notes and preliminary drafts. Many people find this material useful, and it will be helpful to you if you are ever challenged to show that the work is your own. In addition, you should keep a copy of the final report in case the original is lost.

Other Important Communication Forms

Other written forms of technical communication important to researchers are: grant proposal, white paper, extended abstract, journal article, research monograph, conference proceedings, cover letter, letter of recommendation, and job application letter. Important oral communication forms include: conference presentation, technical lecture, dog-and-pony show, telephone conversation, and job interview. These topics are beyond the scope of this document.

References

1. Miller, Casey, and Kate Swift, *The Handbook of Nonsexist Writing*, Harper and Row.
2. Sherman, Alan T., ``How to solve and write up homework problems'' (January 29, 1991).
3. Higham, Nicholas J., *Handbook of Writing for the Mathematical Sciences*, SIAM Press (1993).
4. Dornan, Edward A.; and CHarles W. Dawe, *The Brief English Handbook*, Little, Brown and Company (1984).
5. *Letitia Baldrige's New Complete Guide to Executive Manners*, Rawson Associates (NY, 1993).
6. Strunk, William Jr.; and E. B. White, *The Elements of Style*, Macmillan (New York, 1972).

Alan T. Sherman, sherman@cs.umbc.edu
Last modified: April 27, 1995