How to write a technical paper or a research paper

by [Michael Ernst](#)

April, 2005
Last updated: July 1, 2024

Contents:

This document describes several simple, concrete ways to improve your writing, by avoiding some common mistakes. The [end of this document](#) contains more resources for improving your writing.

Some people believe that writing papers, [giving talks](#), and similar "marketing" activities are not part of research, but an adjunct to it or even an undesirable distraction. This view is inaccurate. The purpose of research is to increase the store of human knowledge, and so even the very best work is useless if you cannot effectively communicate it to the rest of the world. If a paper is poorly written, then readers might conclude you spent as little effort on the research that it describes.

Equally importantly, writing papers and giving talks will clarify your thinking and thereby improve your research. You may be surprised how difficult it is to clearly communicate your ideas and contributions; doing so will force you to understand them more deeply and enable you to improve them.

Know your message, and stay on message

The goal of writing a paper is to change people's behavior: for instance, to change the way they think about a research problem or to convince them to use a new approach. Determine your goal (also known as your thesis), and focus the paper around that goal.

As a general rule, your paper needs to convince the audience of three key points. If any of these is missing or unclear, the paper will not be compelling.

- The problem is **important**. The problem has a significant impact and consequences. You can buttress your argument by showing that others consider the problem important.
- The problem is **hard**. Explain that obvious techniques and existing approaches do not suffice. Showing what others have tried can be effective here.

- You have **solved** the problem. This is often demonstrated via experiments. Keep in mind how you expect the behavior of readers to change once they appreciate your contributions.
  You'll also need to convince readers that your contributions are novel. When expressing this, it is helpful to explain why no one else thought of your approach before (or why, if they thought of it, they would have rejected the approach) , and whether similar insights apply to other problems.

Before you write your paper, you need to understand your audience. Who will read your paper? What are their backgrounds, motivations, interests, and beliefs? What are the key points you want a reader person to take away from your paper? Once you know the thesis and audience, you can determine what points your document should make to achieve its purpose.

For each point in your paper, you need to explain both *what* and *why*. Start with what, but don't omit why. For example, it is not enough to state how an algorithm works; you should explain why it works in that way, or why another way of solving the problem would be different. Similarly, it is not sufficient to present a figure or facts. You must also ensure that reader understands the significance or implications of the figure and what parts of it are most important.

Which details to include

Your purpose is to communicate specific ideas, and everything about your paper should contribute to this goal. If any part of the paper does not support your main point, then delete or change that part. You must be ruthless in cutting every irrelevant detail, however true it may be. Everything in your paper that does not support your main point distracts from it.

Write for the readers, rather than writing for yourself. In particular, think about what matters to the intended audience, and focus on that. It is not necessarily what you personally find most intriguing.

A common mistake is to focus on what you spent the most time on. Do not write your paper as a chronological narrative of all the things that you tried, and do not devote space in the paper proportionately to the amount of time you spent on each task. Most work that you do will never show up in any paper; the purpose of infrastructure-building and exploration of blind alleys is to enable you to do the small amount of work that is worth writing about. Another way of stating this is that the purpose of the paper is not to describe what you have done, but to inform readers of the successful outcome or significant results, and to convince readers of the validity of those conclusions.

Likewise, do not dwell on details of the implementation or the experiments except insofar as they contribute to your main point. This is a particularly important piece of advice for software documentation, where you need to focus on the software's benefits to the user, and how to use it, rather than how you implemented it. However, it holds for technical papers as well — and remember that readers expect different things from the two types of writing!

The audience is interested in what worked, and why, so start with that. If you discuss approaches that were not successful, do so briefly, and typically only after you have discussed the successful approach. Furthermore, the discussion should focus on differences from the successful technique, and if at all possible should provide general rules or lessons learned that will yield insight and help others to avoid such blind alleys in the future.

Whenever you introduce a strawman or an inferior approach, say so upfront. A reader will (and should) assume that whatever you write in a paper is something you believe or advocate, unless very clearly marked otherwise. A paper should never first detail a technique, then (without forewarning) indicate that the technique is flawed and proceed to discuss another technique. Such surprises confuse and irritate readers. This mistake is often called "leading the reader down the garden path".

When there are multiple possible approaches to a problem, it is preferable to give the best or successful one first. Oftentimes it is not even necessary to discuss the alternatives. If you do, they should generally come after, not before, the successful one. Your paper should give the most important details first, and the less important ones afterward. Its main line of argument should flow coherently rather than being interrupted. It can be acceptable to state an imperfect solution first (with a clear indication that it is imperfect) if it is a

simpler version of the full solution, and the full solution is a direct modification of the simpler one. Less commonly, it can be acceptable to state an imperfect solution first if it is an obvious solution that every reader will assume is adequate; but use care with this rationalization, since you are usually wrong that every reader will jump to the given conclusion.

Make the organization and results clear

A paper should communicate the main ideas of your research (such as the techniques and results) early and clearly. Then, the body of the paper can expand on these points; a reader who understands the structure and big ideas can better appreciate the details. Another way of saying this is that you should give away the punchline. A technical paper is not a joke or a mystery novel. The reader should not encounter any surprises, only deeper explanations of ideas that have already been introduced. It's particularly irritating when an abstract or introduction states, "We evaluated the relationship between baldness and beekeeping", with the key results buried pages later. A better abstract would say, "Male beekeepers are 25% more likely to be bald (p=.04), but there is no statistically significant correlation for female beekeepers."

The same advice applies at the level of sections and paragraphs. It is a bad approach to start with a mass of details and only at the end tell the reader what the main point was or how the details related to one another. Instead, state the point first and then support it. The reader is more likely to appreciate which evidence is important and why, and is less likely to become confused or frustrated.

For each section of the paper, consider writing a mini-introduction that says what its organization is, what is in each subpart, and how the parts relate to one another. For the whole paper, this is probably a paragraph. For a section or sub-section, it can be as short as a sentence. This may feel redundant to you (the author), but readers haven't spent as much time with the paper's structure as you have, so they will truly appreciate these signposts that orient them within your text.

Some people like to write the abstract, and often also the introduction, last. Doing so makes them easier to write, because the rest of the paper is already complete and can just be described. However, I prefer to write these sections early in the process (and then revise them as needed), because they frame the paper. If you know the paper's organization and outlook, then writing the front matter will take little effort. If you don't, then it is an excellent use of your time to determine that information by writing the front matter. To write the body of the paper without knowing its broad outlines will take more time in the long run. Another way of putting this is that writing the paper first will make writing the abstract faster, and writing the abstract first will make writing the paper faster. There is a lot more paper than abstract, so it makes sense to start with that and to clarify the point of the paper early on.

It is a very common error to dive into the technical approach or the implementation details without first appropriately framing the problem and providing motivation and background. Readers need to understand what the task is before they are convinced that they should pay attention to what you are saying about it. You should first say what the problem or goal is, and — even when presenting an algorithm — first state what the output is and probably the key idea, before discussing steps. Avoid providing information that isn't useful to readers/users. It just distracts from the important content.

Getting started: overcoming writer's block and procrastination

Some writers are overwhelmed by the emptiness of a blank page or editor buffer, and they have trouble getting started with their writing. Don't worry! Here are some tricks to help you get started. Once you have begun, you will find it relatively easier to revise your notes or first draft. The key idea is to write *something*, and you can improve it later.

Start verbally. Explain what the paper needs to say to another person. After the conversation is over, write down what you just said, focusing on the main points rather than every word you spoke. Many people find it easier to speak than to write. Furthermore, getting feedback and giving clarifications will help you discover problems with your argument, explanation, or word choice.

Outline. You may not be ready to write full English paragraphs, but you can decide which sections your paper will have and give them descriptive titles. Once you have decided on the section structure, you can write a little outline of each section, which indicates the subsection titles. Now, expand that into a topic sentence for each paragraph. At this point, since you know the exact topic of each paragraph, you will find the paragraph easy to write.

Stream-of-consciousness notes. Write down everything that you know, in no particular order and with no particular formatting. Afterward, organize what you wrote thematically, bringing related points together. Eventually, convert it into an outline and proceed as above. While writing notes, use phrases/keywords, *not* complete sentences. The phrases are quicker to write and less likely to derail your brainstorming; they are easier to organize; and you will feel less attached to them and more willing to delete them.

Divide and conquer. Rather than trying to write your entire document, choose some specific part, and write just that part. Then, move on to another part.

Re-use. Find other text that you have written on the topic and start from that. An excellent source is your [progress reports](#) — you are writing them, aren't you? This can remind you what was hard or interesting, or of points that you might otherwise forget to make. You will rarely want to re-use text verbatim, both because you can probably convey the point better now, and also because writing for different audiences or in different contexts requires a different argument or phrasing. For example, a technical paper and a [technical talk](#) have similar aims but rather different forms.

You must be willing to delete and/or rewrite your notes and early drafts. If you wrote something once, you can write it again (probably better!). Early on, the point is to organize your ideas, not to create finished sentences.

Brevity

Be brief. Make every word count. If a word does not support your point, cut it out, because excess verbiage and fluff only make it harder for the reader to appreciate your message. Use shorter and more direct phrases wherever possible.

Make your writing crisp and to the point. Eliminate any text that does not support your point. Here is one way you might go about this; it is time-consuming but extremely effective. First, examine each section of the paper in turn and ask what role it serves and whether it contributes to the paper's main point. If not, delete it. Next, within each section, examine each paragraph. Ask whether that paragraph has a single point. If not, rewrite the paragraph. Also ask whether that point contributes to the goals of the section. If not, then delete the paragraph. Next, within each paragraph, examine each sentence. If it does not make a single, clear point that strengthens the paragraph, delete or rewrite it. Finally, within each sentence, examine each word, and delete or replace those that do not strengthen their point. You will need to repeat this entire process multiple times, keeping a fresh perspective on the paper.

Some people find it easier to follow this approach bottom-up, first cutting/rewriting words, then sentences, etc.

Writing style

Passive voice has no place in technical writing. It obscures who the actor was, what caused it, and when it happened. Use active voice and simple, clear, direct phrasing.

First person is rarely appropriate in technical writing.

- First person *is* appropriate when describing something that the author of the paper did manually. Recall that your paper should not be couched as a narrative.
- Do *not* use "we" to mean "the author and the reader" or "the paper". For example, do not write "In this section, we ...".

- Do *not* use "we" to describe the operation of a program or system. "We compute a graph" makes it sound like the authors did it by hand. As a related point, do not anthropomorphize computers: they *hate* it. Anthropomorphism, such as "the program thinks that ...", is unclear and vague.

Avoid puffery, self-congratulation, superlatives, and subjective or value judgments: give the objective facts and let the reader judge. Avoid vague terms like "sizable" and "significant" (which are also subjective). Don't overuse the word "novel". When I see a paper that is full of these, my rule of thumb is that the paper is trying too hard to cover up for scanty evidence.

Do not use words like "clearly", "easily", "obviously", and "trivially", as in "Obviously, this Taylor series sums to π." If the point is really obvious, then you are just wasting words by pointing it out. And if the point is not obvious to readers who are not intimately familiar with the subject matter the way you are, then you are offending readers by insulting their intelligence, and you are demonstrating your own inability to communicate the intuition.

Prefer singular to plural number. In "sequences induce graphs", it is not clear whether the two collections are in one-to-one correspondence, or the set of sequences collectively induces a set of graphs; "each sequence induces a graph" avoids this confusion. Likewise, in "graphs might contain paths", it is unclear whether a given graph might contain multiple paths, or might contain at most one path.

When describing an experiment or some other event or action that occurred in the past, use *past tense*. For example, the methodology section might say "We ran the program". It would be ungrammatical and confusing to use present tense, as in "We run the program". Present tense is for ongoing events ("I write this letter to inform you...") or regular events ("I brush my teeth each day"), but not past events ("Yesterday, I eat dinner with my family"). It is also correct to say "Our methodology was to run the program", where you use past tense "was" and the infinitive "to run".

When describing the paper itself, use *present tense*. "This paper shows that ...". The reason for this is that the reader is experiencing the paper in real time.

Avoid gratuitous use of the future tense "will ...", as in, "switching the red and green wires will cause the bomb to explode". It is unclear when the action will occur. If it is an immediate effect, use the shorter and more direct "switching the red and green wires causes the bomb to explode".

Use "previous work" instead of "existing work". Your work exists, so "existing work" would refer to it as well.

In a list with 3 or more elements list, put a serial comma between each of the items (including the last two). As a simple example of why, consider this 3-element grocery list written without the clarifying last comma: "milk, macaroni and cheese and crackers". It's not clear whether that means { milk, macaroni and cheese, crackers } or { milk, macaroni, cheese and crackers }. As another example, "I would like to thank my parents, Rene Descartes and Ayn Rand," suggests rather unusual parentage, whereas "I would like to thank my parents, Rene Descartes, and Ayn Rand," shows a debt to four people. I've seen real examples that were even more confusing than these.

In English, compound adjectives are hyphenated (except those whose first words end with "ly", in some style guides) but compound nouns are not. Consider "the semantics provide name protection" versus "the name-protection semantics".

Prefer unambiguous words to ambiguous ones. Do not use "as" or "since" to mean "because". Do not use "if" to mean "whether".

Use quotations sparingly. A clear paraphrase of the points that are relevant to your own work (along with a proper citation) is usually better than a long quotation from a previous publication.

Avoid third-person pronouns when you can. The old standard was "he", which is masculine chauvinist. The new standard is "he or she", which can be viewed as heteronormative and which some people find clumsy. An emerging standard is "they" as a first-person *singular* pronoun, which is inclusive but grammatically incorrect and [confusing](#) (see comments above about singular vs. plural number).

Some of the suggestions in this document are about good writing, and that might seem secondary to the research. But writing more clearly will help you think more clearly and often reveals flaws (or ideas!) that had previously been invisible even to you. Furthermore, if your writing is not good, then either readers will not be able to comprehend your good ideas, or readers will be (rightly) suspicious of your technical work. If you do not (or cannot) write well, why should readers believe you were any more careful in the research itself? The writing reflects on you, so make it reflect well.

Figures

Use figures! Different people learn in different ways, so you should complement a textual or mathematical presentation with a graphical one. Even for people whose primary learning modality is textual, another presentation of the ideas can clarify, fill gaps, or enable the reader to verify his or her understanding. Figures can also help to illustrate concepts, draw a skimming reader into the text (or at least communicate a key idea to that reader). Figures make the paper more visually appealing.

It is extremely helpful to give an example to clarify your ideas: this can make concrete in the reader's mind what your technique does (and why it is hard or interesting). A running example used throughout the paper is also helpful in illustrating how your algorithm works, and a single example permits you to amortize the time and space spent explaining the example (and the reader's time in appreciating it). It's harder to find or create a single example that you re-use throughout the paper, but it is worth it.

A figure should stand on its own, containing all the information that is necessary to understand it. Good captions contain multiple sentences; the caption provides context and explanation. For examples of good, informative captions, see the print editions of magazines such as *Scientific American* and *American Scientist*. The caption should state what the figure illustrates or what conclusion a reader should draw from it. Don't write an obvious description of what the figure is, such as "Code example". Never write a caption like "The Foobar technique"; the caption should also say what the Foobar technique is, what it is good for, or how it works. The caption may also need to explain the meaning of columns in a table or of symbols in a figure. However, it's even better to put that information in the figure proper; for example, use labels or a legend. When the body of your paper contains information that belongs in a caption, there are several negative effects. The reader is forced to hunt all over the paper in order to understand the figure. The flow of the writing is interrupted with details that are relevant only when one is looking at the figure. The figures become ineffective at drawing in a reader who is scanning the paper — an important constituency that you should cater to!

As with naming, use pictorial elements consistently. Only use two different types of arrows (or boxes, shading, etc.) when they denote distinct concepts; do not introduce inconsistency just because it pleases your personal aesthetic sense. Almost any diagram with multiple types of elements requires a legend (either explicitly in the diagram, or in the caption) to explain what each one means; and so do many diagrams with just one type of element, to explain what it means.

Some writers label all the types of figures differently — some as "figure", others as "table" or "graph" or "picture". This differentiation has no benefits, but it does have a drawback: it is very hard for a reader to find "table 3", which might appear after "figure 7" but before "freehand drawing 1". You should simply call them all figures and number them sequentially. The body of each figure might be a table, a graph, a diagram, a screenshot, or any other content.

Put figures at the top of the page, not in the middle or bottom. If a numbered, captioned figure appears in the middle or at the bottom of a page, it is harder for readers to find the next paragraph of text while reading, and harder to find the figure from a reference to it.

Avoid bitmaps, which are hard to read. Export figures from your drawing program in a vector graphics format. If you must use a bitmap (which is only appropriate for screenshots of a tool), then produce them at very high resolution. Use the biggest-resolution screen you can, and magnify the portion you will capture.

Don't waste text in the paper (and tax the reader's patience) regurgitating information that is expressed more precisely and concisely in a figure. For example, the text should not repeat the numbers from a table or graph.

Text in the paper should add insight or explanations, or summarize the conclusions to be drawn from the data in the figure.

Computer program source code

Your code examples should either be real code, or should be close to real code. Never use synthetic examples such as procedures or variables named `foo` or `bar`. Made-up examples are much harder for readers to understand and to build intuition regarding. Furthermore, they give the reader the impression that your technique is not applicable in practice — you couldn't find any real examples to illustrate it, so you had to make something up.

Any boldface or other highlighting should be used to indicate the most important parts of a text. In code snippets, it should never be used to highlight syntactic elements such as "public" or "int", because that is not the part to which you want to draw the reader's eye. (Even if your IDE happens to do that, it isn't appropriate for a paper.) For example, it would be acceptable to use boldface to indicate the names of procedures (helping the reader find them), but not their return types.

Naming

Give each concept in your paper a descriptive name to make it more memorable to readers. Never use terms like "approach 1", "approach 2", or "our approach", and avoid acronyms when possible. If you can't think of a good name, then quite likely you don't really understand the concept. Think harder about it to determine its most important or salient features.

It is better to name a technique (or a paper section, etc.) based on *what* it does rather than *how* it does it.

Use terms consistently and precisely. Avoid "elegant variation", which uses different terms for the same concept to avoid boredom on the part of the reader or to emphasize different aspects of the concept. While elegant variation may be appropriate in poems, novels, and some essays, it is not acceptable in technical writing, where you should clearly define terms when they are first introduced, then use them consistently. If you switch wording gratuitously, you will confuse the reader and muddle your point. A reader of a technical paper expects that use of a different term flags a different meaning, and will wonder what subtle difference you are trying to highlight. Thus, don't confuse the reader by substituting "program", "library", "component", "system", and "artifact", nor by conflating "technique", "idea", "method" and "approach", nor by switching among "program", "code", and "source". Choose the best word for the concept, and stick with it.

Do not use a single term to refer to multiple concepts. If you use the term "technique" for every last idea that you introduce in your paper, then readers will become confused. This is a place that use of synonyms to distinguish concepts that are unrelated (from the point of view of your paper) is acceptable. For instance, you might always use "phase" when describing an algorithm but "step" when describing how a user uses a tool.

When you present a list, be consistent in how you introduce each element, and either use special formatting to make them stand out or else state the size of the list. Don't use, "There are several reasons I am smart. I am intelligent. Second, I am bright. Also, I am clever. Finally, I am brilliant." Instead, use "There are four reasons I am smart. First, I am intelligent. Second, I am bright. Third, I am clever. Fourth, I am brilliant." Especially when the points are longer, this makes the argument much easier to follow. Some people worry that such consistency and repetition is pedantic or stilted, or it makes the writing hard to follow. There is no need for such concerns: none of these is the case. It's more important to make your argument clear than to achieve "elegant variation" at the expense of clarity.

Choose good names not only for the concepts that you present in your paper, but for the document source file. Don't name the file after the conference to which you are submitting (the paper might be rejected) or the year. Even if the paper is accepted, such a name won't tell you what the paper is about when you look over your files in later years. Instead, give the paper or its folder/directory a name that reflects its content. Another benefit is that this will also lead you to think about the paper in terms of its content and contributions.

Here is a piece of advice that is specific to computing: do not use the vague, nontechnical term "bug". Instead, use one of the [standard terms](#) fault, error, or failure. A fault is an underlying defect in a system, introduced by a human. A failure is a user-visible manifestation of the fault or defect. In other circumstances, "bug report" may be more appropriate than "bug".

Numbers and measurements

Digits of precision:

- Don't report more digits of precision than the measurement process reliably and reproducibly produces. The 3rd or 4th digit of precision is rarely accurate and generalizable; if you don't have confidence that it is both repeatable and generalizable to new experiments, omit it. Another way to say this is that if you are not confident that a different set of experiments would produce all the same digits, then don't report so much precision.
- Don't report more digits of precision than needed to convey your message. If the difference between 4.13 and 4 will not make a difference in convincing readers, then don't report the extra digits. Reporting extra digits can distract readers from the larger trends and the big picture. Including an inappropriate number of digits of precision can cast suspicion on all of your results, by giving readers the impression that you are statistically naive.
- Use a consistent number of digits of precision. If the measured data are 1.23, 45.67, and 891.23, for example, you might report them as 1.23, 45.7, and 891, or as 1.2, 46, and 890, or as 1, 50, and 900. (An exception is when data are known to sum to a particular value; I would report 93% and 7% rather than either 93% and 7.4% or 90% and 7%. Often it's appropriate to report percentages as whole numbers rather than using the same precision.)
- If you do any computations such as ratios, your computations should internally use the full precision of your actual measurements, even though your paper reports only a limited number of digits of precision.
- If a measurement is exact, such as a count of items, then it can be acceptable to give the entire number even if it has many digits; by contrast, timings and other inexact measurements should always be reported with a limited number of digits of precision.

Do not confuse relative and absolute measurements. For instance, suppose your medicine cures 30% of patients, and the placebo cures 25% of patients. You could report that your medicine's cure rate is .3, the placebo's cure rate is .25, and your medicine's cure rate is either .05 greater or 20% greater. (Other correct, but less good, ways to say the same thing are that it cures 20% more, 120% as many, or 1.2 times as many patients.) It would be inaccurate to state that your medicine cures 5% more patients or your medicine cures 120% more patients. Just as you need to correctly use "120% more" versus "120% as many", you need to correctly use "3 times faster than" versus "3 times as fast as". A related, also common, confusion is between "3 times faster than and 3 times as fast as". And, "2 times fewer" makes absolutely no sense. I would avoid these terms entirely. "Half as many" is a much better substitute for "2 times fewer".

Given the great ease of misunderstanding what a percentage means or what its denominator is, I try to avoid percentages and focus on fractions whenever possible, especially for base measurements. For comparisons between techniques, percentages can be acceptable. Avoid presenting two different measurements that are both percentages but have different denominators.

Processing data

Your paper probably includes tables, bibliographies, or other content that is generated from external data. Your paper may also be written in a text formatting language such as LaTeX. In each of these cases, it is necessary to run some external command to create some of the content or to create the final PDF.

All of the steps to create your final paper should be clearly documented — say, in comments or in a notes file that you maintain with the paper. Preferably, they should be automated so that you only have to run one command that collects all the data, creates the tables, and generates the final PDF.

If you document and automate these steps, then you can easily regenerate the paper when needed. This is useful if you re-run experiments or analysis, or if you need to defend your results against a criticism by other

researchers. If you leave some steps manual, then you or your colleagues are highly likely to make a mistake (leading to a scientific error) or to be unable to reproduce your results later.

One good way to automate these tasks is by writing a program or creating a script for a build system such as Ant, Gradle, Make, Maven, etc.

Related work

A related work section should not only explain what research others have done, but in each case should compare and contrast that to your work and also to other related work. After reading your related work section, a reader should understand the key idea and contribution of each significant piece of related work, how they fit together (what are the common themes or approaches in the research community?), and how your work differs. Don't write a related work section that is just a list of other papers, with a sentence about each one that was lifted from its abstract, and without any critical analysis nor deep comparison to other work.

Unless your approach is a small variation on another technique, it is usually best to defer the related work to the end of the paper. When it comes first, it gives readers the impression that your work is rather derivative. (If this is true, it is your responsibility to convey that clearly; if it is not true, then it's misleading to intimate it.) You need to ensure that readers understand your technique in its entirety, and also understand its relationship to other work; different orders can work in different circumstances.

Just as you should generally explain your technique first, and later show relationships with other work, it is also usually more effective to defer a detailed discussion of limitations to a later section rather than the main description of your technique. You should be straightforward and honest about the limitations, of course (do mention them early on, even if you don't detail them then), but don't destroy the coherence of your narrative or sour the reader on your technique.

Feedback

Get **feedback**! Finish your paper well in advance, so that you can improve the writing. Even re-reading your own text after being away from it can show you things that you didn't notice. An outside reader can tell you even more.

When readers misunderstand the paper, that is always at least partly the author's fault! Even if you think the readers have missed the point, you will learn how your work can be misinterpreted, and eliminating those ambiguities will improve the paper.

Be considerate to your reviewers, who are spending their time to help you. Here are several ways to do that.

As with submission to conferences, don't waste anyone's time if there are major flaws. Only ask someone to read (a part of) your paper when you think you will learn something new, because you are not aware of serious problems. If only parts are ready, it is best to indicate this in the paper itself (e.g., a TODO comment that the reader will see or a hand-written annotation on a hardcopy) rather than verbally or in email that can get forgotten or separated from the paper.

It is most effective to get feedback sequentially rather than in parallel. Rather than asking 3 people to read the same version of your paper, ask one person to read the paper, then make corrections before asking the next person to read it, and so on. This prevents you from getting the same comments repeatedly — subsequent readers can give you new feedback rather than repeating what you already knew, and you'll get feedback on something that is closer to the final version. If you ask multiple reviewers at once, you are de-valuing their time — you are indicating that you don't mind if they waste their time saying something you already know. You might ask multiple reviewers if you are not confident of their judgment or if you are very confident the paper already is in good shape, in which case there are unlikely to be major issues that every reviewer stumbles over.

It usually best not to email the document, but to provide a location from which reviewers can obtain the latest version of the paper, such as a version control repository or a URL you will update. That way, you won't clutter inboxes with many revisions, and readers can always get the most recent copy.

Be generous with your time when colleagues need comments on their papers: you will help them, you will learn what to emulate or avoid, and they will be more willing to review your writing.

Some of your best feedback will be from yourself, especially as you get more thoughtful and introspective about your writing. To take advantage of this, start writing early. One good way to do this is to write a periodic progress report that describes your successes and failures. The progress report will give you practice writing about your work, oftentimes trying out new explanations.

Whereas you should start writing as early as possible, you don't need to put that writing in the form of a technical paper right away. In fact, it's usually best to outline the technical paper, and get feedback on that, before you start to fill in the sections with text. (You might think that you can copy existing text into the paper, but it usually works out better to write the information anew. With your knowledge of the overall structure, goals, and audience, you will be able to do a much better job that fits with the paper's narrative.) When outlining, I like to start with one sentence about the paper; then write one sentence for each section of the paper; then write one sentence for each subsection; then write one sentence for each paragraph (think of this as the topic sentence); and at that point, it's remarkably easy just to flesh out the paragraphs.

When to submit your paper for publication

You should not submit your paper too early, when it does not reflect well on you and a submission would waste the community's reviewing resources. You should not submit your paper too late, because then the community is deprived of your scientific insights. In general, you should err on the side of submitting too late rather than too early.

A rule of thumb is to submit only if you are proud for the world to associate your name with the work, *in its current form*. If you know of significant criticisms that reviewers might raise, then don't submit the paper.

Submitting your paper prematurely has many negative consequences.

- You will waste the time of hard-working reviewers, who will give you feedback that you could have obtained in other ways.
- You will get a reputation for shoddy work.
- You will make the paper less likely to be accepted in the future. Oftentimes the same reviewers may serve two different venues. Reviewing a paper again puts a reviewer in a negative state of mind. I have frequently heard reviewers say, "I read an earlier version of this paper, it was a bad paper, and this version is similar." (This is unethical because reviewers are not supposed to talk about papers they have reviewed, but nonetheless it is very common.) Now the paper will likely be rejected again, and the whole committee gets a bad impression of you. A reviewer who has read a previous version of the paper may read the resubmission less carefully or make assumptions based on a previous version. To sum up: it's harder to get a given paper accepted on its second submission, than it would have been to get the identical paper accepted on its first submission.

Here are some *bad* reasons to submit a paper.

- You need feedback on the paper.

  It's true that the feedback from reviewers is extraordinarily valuable to you and will help you improve the paper. However, you should get feedback from other scientists (your friends and colleagues) before submitting for publication.

- Submitting the paper in its current form means more papers on your c.v. and more opportunities for others to learn about your work.

Those are true facts, and some people do "salami-slice" their research into as many papers as possible — such papers are called a "least publishable unit". However, doing so leads to less impact than publishing fewer papers, each one with more content. If a paper contains few contributions, it is less likely to make a big impression, because it is less exciting. In addition, readers won't enjoy reading many pages to learn just a few facts.

Note: This point refers to taking a single research idea or theme and splitting it into multiple publications. When there are multiple distinct research contributions, it can be appropriate to describe them in different papers.

- To roll the dice for acceptance on a paper that is better than some other published papers.

The reviewing process can be frustrating, because it contains a great deal of randomness: the same paper would be rejected by some reviewers and accepted by others. However, all great papers are accepted and all bad papers are rejected. For mediocre papers, luck plays a role. Your goal should not be to write great papers, not mediocre ones. Find a way to improve your paper. Recognize the great value of reviews: they provide a valuable perspective on your work and how to improve it, even if you feel that the reviewer should have done a better job.

- You are tired of the paper and need a break from it.

If you aren't excited about the paper, it is unlikely that other people will be. Furthermore, the period after submitting the paper is not a time to take a break, but an opportunity to further improve it.

After you submit a paper, don't stop working on it! You can always improve the research. For instance, you might expand the experiments, improve the implementation, or make other changes. Even if your paper is accepted, you want the accepted version to be as impressive as possible. And if the paper is rejected, you need to have a better paper to submit to the next venue.

Responding to conference reviews

(This section is most relevant to fields like computer science where conferences are the premier publication venue. Responding to journal reviews is different.)

Many conferences provide an author response period: the authors are shown the reviews and are given limited space (say, 500 words) to respond to the reviews, such as by clarifying misunderstandings or answering questions. The author response is sometimes called a "rebuttal", but I don't like that term because it sets an adversarial tone.

Your paper will only be accepted if there is a champion for the paper: someone who is excited about it and will try to convince the rest of the committee to accept the paper. Your response needs to give information to your champion to overcome objections. If there isn't a champion, then the main goal of your response is to create that champion. Your response should also give information to detractors to soften their opposition.

After reading the reviews, you may be disappointed or angry. Take a break to overcome this, so that you can think clearly.

For every point in the reviews, write a brief response. Do this in email-response style, to ensure that you did not miss any points. You will want to save this for later, so it can be better to do this in the paper's version control repository, rather than in a WYSIWYG editor such as Google Docs. (This assumes you have a version control repository for the paper, which you should!) Much of this text won't go in your response, but it is essential for formulating the response.

Summarize (in 5 or so bullet points, however many make sense) the key concerns of the reviewers. Your review needs to focus on the most important and substantive critiques. The authors of the paper should agree on this structure before you start to write the actual response.

Your response to each point will be one paragraph in your response. Start the paragraph with a brief heading or title about the point. Do not assume that the reviewers remember everything that was written by every

reviewer, nor that they will re-read their reviews before reading your response. A little context will help them determine what you are talking about and will make the review stand on its own. This also lets you frame the issues in your own words, which may be clearer or address a more relevant point than the reviews did.

Organize your responses thematically. Group the paragraphs into sections, and have a small heading/title for each section. If a given section has just one paragraph, then you can use the paragraph heading as the section heading. Order the sections from most to least important.

This is better than organizing your response by reviewer, first addressing the comments of reviewer 1, then reviewer 2, and so forth. Downsides of by-reviewer organization include:

- It can encourage you not to give sufficient context.
- It does not encourage putting related information together nor important information first.
- You want to encourage all reviewers to read the entire response, rather than encouraging them to just look at one part.
- When multiple reviewers raised the same issue, then no matter where you address it, it's possible for a reviewer to overlook it and think you failed to address it.
- You don't want to make glaringly obvious which issues in a review you had to ignore (for reasons of space or other reasons).
- You don't want to make glaringly obvious that you spent much more time and space on one reviewer than another.

In general, it's best not to mention reviewer names/numbers in your response at all. Make the response be about the science, not about the people.

In your responses, admit your errors forthrightly. Don't ignore or avoid key issues, especially ones that multiple reviewers brought up.

Finally, be civil and thankful the reviewers. They have spent considerable time and energy to give you feedback (even if it doesn't seem to you that they have!), and you should be grateful and courteous in return.

Rejection

If you submit technical papers, you will experience rejection. In some cases, rejection indicates that you should move on and begin a different line of research. In most cases, the reviews offer an opportunity to improve the work, and so you should be very grateful for a rejection! It is much better for your career if a good paper appears at a later date, rather than a poor paper earlier or a sequence of weak papers.

Even small flaws or omissions in an otherwise good paper may lead to rejection. This is particularly at the elite venues with small acceptance rates, where you should aim your work. Referees are generally people of good will, but different referees at a conference may have different standards, so the luck of the draw in referees is a factor in acceptance.

The wrong lesson to learn from rejection is discouragement or a sense of personal failure. Many papers — even papers that later win awards — are rejected at least once. The feedback you receive, and the opportunity to return to your work, will invariably improve your results.

Don't be put off by a negative tone in the reviews. The referees are trying to help you, and the bast way to do that is to point out how your work can be improved. I often write a much longer review, with more suggestions for improvement, for papers that I like; if the paper is terrible, I may not be able to make as many concrete suggestions, or my high-level comments may make detailed comments moot.

If a reviewer didn't understand something, then the main fault almost always lies with your writing. If you blame a lazy or dumb reviewer, you are missing the opportunity to improve. Reviewers are not perfect, but they work hard to give you helpful suggestions, so you should give them the benefit of the doubt. Remember that just as it is hard to convey technical ideas in your paper (and if you are getting a rejection, that is evidence that you did not succeed!), it is hard to convey them in a review, and the review is written in a few hours rather than the weeks you spent on the paper (not to mention months or years of understanding the

concepts). You should closely attend to both the explicit comments, and to underlying issues that may have led to those comments — it isn't always easy to capture every possible comment in a coherent manner. Think about how to improve your research and your writing, even beyond the explicit suggestions in the review — the prime responsibility for your research and writing belongs with you.

Norman Ramsey's advice

Norman Ramsey's nice [Teach Technical Writing in Two Hours per Week](#) espouses a similar approach to mine: by focusing on clarity in your writing, you will inevitably gain clarity in your thinking.

Don't bother to read both the student and instructor manuals — the student one is a subset of the instructor one. You can get much of the benefit from just one part, his excellent "principles and practices of successful writers":

Principles

1. Correctness. Write correct English, but know that you have more latitude than your high-school English teachers may have given you.
2. Consistent names. Refer to each significant character (algorithm, concept, language) using the same word everywhere. Give a significant new character a proper name.
3. Singular. To distinguish one-to-one relationships from n-to-m relationships, refer to each item in the singular, not the plural.
4. Subjects and verbs. Put your important characters in subjects, and join each subject to a verb that expresses a significant action.
5. Information flow. In each sentence, move your reader from familiar information to new information.
6. Emphasis. For material you want to carry weight or be remembered, use the end of a sentence.
7. Coherence. In a coherent passage, choose subjects that refer to a consistent set of related concepts.
8. Parallel structure. Order your text so your reader can easily see how related concepts are different and how they are similar.
9. Abstract. In an abstract, don't enumerate a list of topics covered; instead, convey the essential information found in your paper.

Practices

1. Write in brief daily sessions. Ignore the common myth that successful writing requires large, uninterrupted blocks of time — instead, practice writing in brief, daily sessions.
2. Focus on the process, not the product. Don't worry about the size or quality of your output; instead, reward yourself for the consistency and regularity of your input.
3. Prewrite. Don't be afraid to think before you write, or even jot down notes, diagrams, and so on.
4. Use index cards. Use them to plan a draft or to organize or reorganize a large unit like a section or chapter.
5. Write a Shitty First Draft™. Value a first draft not because it's great but because it's there.
6. Don't worry about page limits. Write the paper you want, then cut it down to size.
7. Cut. Plan a revision session in which your only goal is to cut.

Other resources

- [Norman Ramsey's advice](#), excerpted [immediately above](#).
- ["Hints on writing an M.Eng. thesis"](#), by Jeremy Nimmer
- my notes on [reviewing a technical paper](#), which indicate how to recognize — and thus produce — quality work
- my notes on [choosing a venue](#) for publication
- my notes on [giving a technical talk](#): a talk has the same goal as a paper, namely to convey technical ideas
- my notes on [making a technical poster](#)
- Ronald B. Standler's [advice on technical writing](#)
- [Dave Patterson's Writing Advice](#)

- [Advice on SIGPLAN conference submissions (at bottom of page)](#)
- *The Elements of Style*, William Strunk Jr. and E. B. White, is classic book on improving your writing. It focuses at a low level, on English usage.
- *Style: Toward Clarity and Grace*, by Joseph M. Williams, is another general-purpose writing guide, with a somewhat higher-level focus than that of Strunk & White.
- *The Sense of Style: The Thinking Person's Guide to Writing in the 21st Century*, by Steven Pinker, is an excellent guide to writing. It gives reasons (from psychology and other scientific fields) for its advice, making it more authoritative than someone's opinion.

Back to [Advice compiled by Michael Ernst](#).

[Michael Ernst](#)