HOW TO
WRITE A PAPER

STRUCTURE IS EVERYTHING

Technical Report Structure
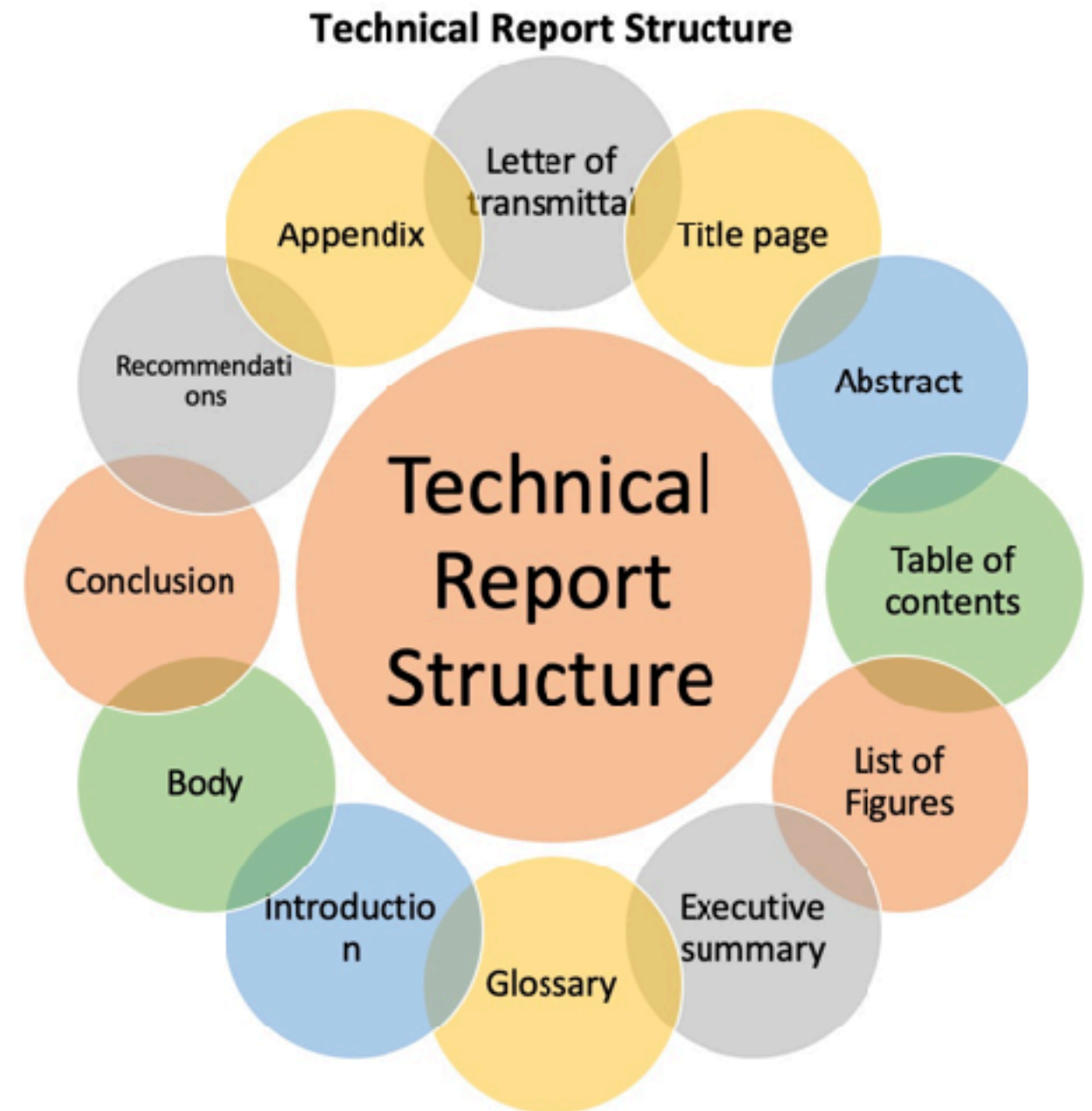
GENERIC PAPER STRUCTURE

Title

Abstract

Keywords

Table of Contents

Introduction

State of the Art

Methods

Results

And

Discussion

Conclusions

Acknowledgements

References

Supplementary Material

GENERIC PAPER STRUCTURE

Title

Abstract

Keywords

Table of Contents

Usually you write this in the end, We will get back to this later

Introduction

Methods

Results

And

Discussion

Conclusions

Acknowledgements

References

Supplementary Material

GENERIC PAPER STRUCTURE

Title

Abstract

Keywords

Table of Contents

Introduction

Methods

Results

And

Discussion

Conclusions

Acknowledgements

References

Supplementary Material

this is the body
of your paper and this may
require small adaptation

GENERIC PAPER STRUCTURE

Title
Abstract
Keywords
Table of Contents

Introduction
Methods
Results
And
Discussion

Conclusions
Acknowledgements
References
Supplementary Material

You write the conclusions
and introduction at the same time

The rest results naturally
from the work

BODY OF THE PAPER

Introduction:

What did you/others do? Why did you do it?

Methods:

How did you do it?

Results:

What did you find?

And

Discussion:

What does it all mean?

ADAPTING THE IMRAD BODY FOR A PRACTICAL ASSIGNMENT

Introduction

Implementation

Experimental Setup

Experimental Results and Analysis

Discussion

THE BEGINNING

Title, Abstract and Keywords are written in the end!!

This is your chance to:

    Convince people to read the paper

    Allow the reader to find the paper!

Make this easy for indexing and searching

Make it informative, attractive, effective

Use the proper keywords!

Title

Abstract

Keywords

Table of Contents

THE END

Conclusions

Acknowledgements

References

Supplementary Material

The conclusions summarise what has been said before and indicate future work.

Always end with a positive tone!

The references should be spotless!

Supplementary Material should not be necessary to fully understand the paper!

THE PROCESS

THE PROCESS

1. Write down the sections titles

2. Prepare all figures, tables, algorithms, charts, diagrams, etc. and place them on the document

If you don't have the final version, just include empty tables, a sketches of diagrams and charts, etc

Do not include screenshots of code or of parameters, it's lazy!

3. Describe your framework/implementation

focus on the relevant aspects, if more detail is required start with an overview of the framework and then talk about each of the modules

if this is for an assignment:

describe the implementation instead of showing it!!!

avoid showing screenshots of code

avoid unnecessary stuff (we have the code)

4. Write the state of the Art / Related work

This doesn't make sense for an assignment, (ignore)

THE PROCESS

5. Write the methodology / Experimental Setup

All the details pertaining the experimental setup should be clearly explained.

Usually I use a table to summarise them (this table is already in your document so it should be easy)

6. Results/Experimental Results

You already have all the tables and charts on the document, now you describe the main findings that are portrayed in those tables and charts.

You are helping the reader to understand the data, giving context, highlighting key findings, making comparisons, etc.

7. Discussion

Now you must extract knowledge from all that data. E.g. Is is algorithm A better than B? What are the best parameters? Is the solution robust? Does it apply to other instances and problems? What are the implications?

THE PROCESS

8. Conclusions and Introduction

Usually you want to use "army writing": in the introduction you state what you are going to say, then throughout the document you say it, in the conclusions you summarise what has been said.

Write a clear conclusion the conclusion should summarise the work and highlight its main contributions and experimental findings.

The introduction should state the context, motivation, scope , the relevance and the objectives of the work. It should summarise the key contributions and, when necessary, overview the structure of the document.

Pro Tip: do not copy the handout!

THE PROCESS

8. Title

Concise, descriptive, appealing

Pro Tip: A title is not an abstract! Keep it small

9. Abstract

The Problem: What you are trying to solve (e.g coronavirus)

Its Relevance: Why should others care (people are dying)

The Approach: How did you try to solve it (telling everybody to stay at home)

The Contributions: What are the main findings (while some people respect the instructions others do not which indicates that additional measures are required)

THE PROCESS

10. Keywords

Be smart, do not invent you own keywords!

Think like this: how is someone using google going to find my paper? Who is my audience?

11. Acknowledgements

Thanks everybody that helped you and never forget the institution paying you!!!

12. References

Use Bibtex

Download the bibitem files from credible sources (Publishers, DBLP, DOI)

Assume that authors do not know how to cite their own work

AN EXAMPLE

Towards the Evolution of Multi-Layered Neural Networks: A Dynamic Structured Grammatical Evolution Approach

Filipe Assunção, Nuno Lourenço, Penousal Machado, Bernardete Ribeiro
CISUC, Department of Informatics Engineering,
University of Coimbra, Portugal
{fga,naml,machado,bribeiro}@dei.uc.pt

ABSTRACT

Current grammar-based NeuroEvolution approaches have several shortcomings. On the one hand, they do not allow the generation of Artificial Neural Networks (ANNs) composed of more than one hidden-layer. On the other, there is no way to evolve networks with more than one output neuron. To properly evolve ANNs with more than one hidden-layer and multiple output nodes there is the need to know the number of neurons available in previous layers. In this paper we introduce Dynamic Structured Grammatical Evolution (DSGE): a new genotypic representation that overcomes the aforementioned limitations. By enabling the creation of dynamic rules that specify the connection possibilities of each neuron, the methodology enables the evolution of multi-layered ANNs with more than one output neuron. Results in different classification problems show that DSGE evolves effective single and multi-layered ANNs, with a varying number of output neurons.

CCS CONCEPTS

•**Computing methodologies → Neural networks; Genetic programming;** *Supervised learning by classification;* •**Theory of computation** → Genetic programming;

KEYWORDS

NeuroEvolution, Artificial Neural Networks, Classification, Grammar-based Genetic Programming

1 INTRODUCTION

Machine Learning (ML) approaches, such as Artificial Neural Networks (ANNs), are often used to learn how to distinguish between multiple classes of a given problem. However, to reach near-optimal classifiers a laborious process of trial-and-error is needed to hand-craft and tune the parameters of ML methodologies. In the specific case of ANNs there are at least two manual steps that need to be

considered: (i) the definition of the topology of the network, i.e., number of hidden-layers, number of neurons of each hidden-layer, and how should the layers be connected between each other; and (ii) the choice and parameterisation of the learning algorithm that is used to tune the weights and bias of the network connections (e.g., initial weights distribution and learning rate).

Evolutionary Artificial Neural Networks (EANNs) or NeuroEvolution refers to methodologies that aim at the automatic search and optimisation of the ANNs' parameters using Evolutionary Computation (EC). With the popularisation of Deep Learning (DL) and the need for ANNs with a larger number of hidden-layers, NeuroEvolution has been vastly used in recent works [19].

The goal of the current work is the proposal of a novel Grammar-based Genetic Programming (GGP) methodology for evolving the topologies, weights and bias of ANNs. We rely on a GGP methodology because in this way we allow the direct encoding of different topologies for solving distinct problems in a plug-and-play fashion, requiring the user to set a grammar capable of describing the parameters of the ANN to be optimised. One of the shortcomings of previous GGP methodologies applied to NeuroEvolution is the fact that they are limited to the evolution of network topologies with one hidden-layer. The proposed approach, called Dynamic Structured Grammatical Evolution (DSGE) solves this constraint.

The remainder of the document is organised as follows: In Section 2 we detail Structured Grammatical Evolution (SGE) and survey the state of the art on NeuroEvolution; Then, in Sections 3 and 4, DSGE and its adaption to the evolution of multi-layered ANNs are presented, respectively; The results and comparison of DSGE with other GGP approaches is conducted in Section 5; Finally, in Section 6, conclusions are drawn and future work is addressed.

2 BACKGROUND AND STATE OF THE ART

In the following sub-sections we present SGE, which serves as base for the development of DSGE. Then, we survey EC approaches for the evolution of ANNs.

2.1 Structured Grammatical Evolution

Structured Grammatical Evolution (SGE) was proposed by Lourenço et al. [11] as a new genotypic representation for Grammatical Evolution (GE) [13]. The new representation aims at solving the redundancy and locality issues in GE, and consists of a list of genes, one for each non-terminal symbol. Furthermore, a gene is a list of integers of the size of the maximum number of possible expansions for the non-terminal it encodes; each integer is a value in the interval $[0, non\_terminal\_possibilities-1]$, where $non\_terminal\_possibilities$ is the number of possibilities for the expansion of the considered non-terminal symbol. Consequently, there is the need to pre-process

SOME ADDITIONAL REFERENCES

REFERENCES

For papers:

https://www.elsevier.com/connect/11-steps-to-structuring-a-science-paper-editors-will-take-seriously

https://spie.org/samples/9781510619142.pdf

For technical reports:

http://www.sussex.ac.uk/ei/internal/forstudents/engineeringdesign/studyguides/techreportwriting

https://www.wikihow.com/Write-a-Technical-Report

For survey papers:

http://ueberfachliche-kompetenzen.ethz.ch/dopraedi/pdfs/Mayer/guidelines_review_article.pdf

FORMATING

IT'S THAT SIMPLE

It's a formal document
Use latex (I suggest overleaf)
Use a template

Use a spell checker
Use grammarly

If your English is not good enough, enrol in a course

get cape.   wear cape.   fly.

GENERAL GUIDELINES FOR WRITING

READ THIS

REALLY, READ IT!

NEW EDITION

WILLIAM **STRUNK** JR.
AND
E.B. **WHITE**

"...still a little book, small enough and important enough to carry in your pocket, as I carry mine."
— *Charles Osgood*

The
ELEMENTS
of
STYLE

FOURTH EDITION

FOREWORD BY ROGER ANGELL
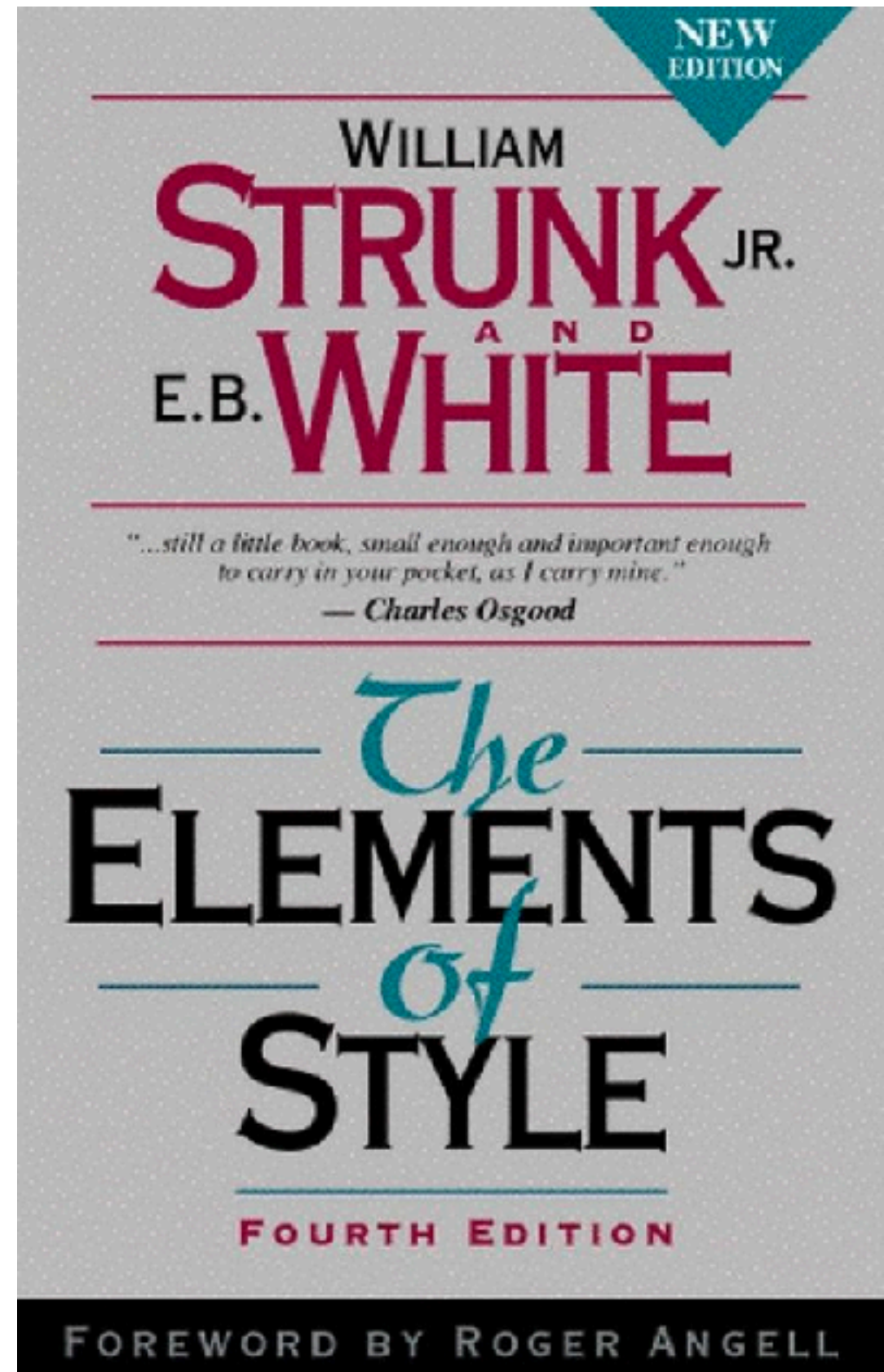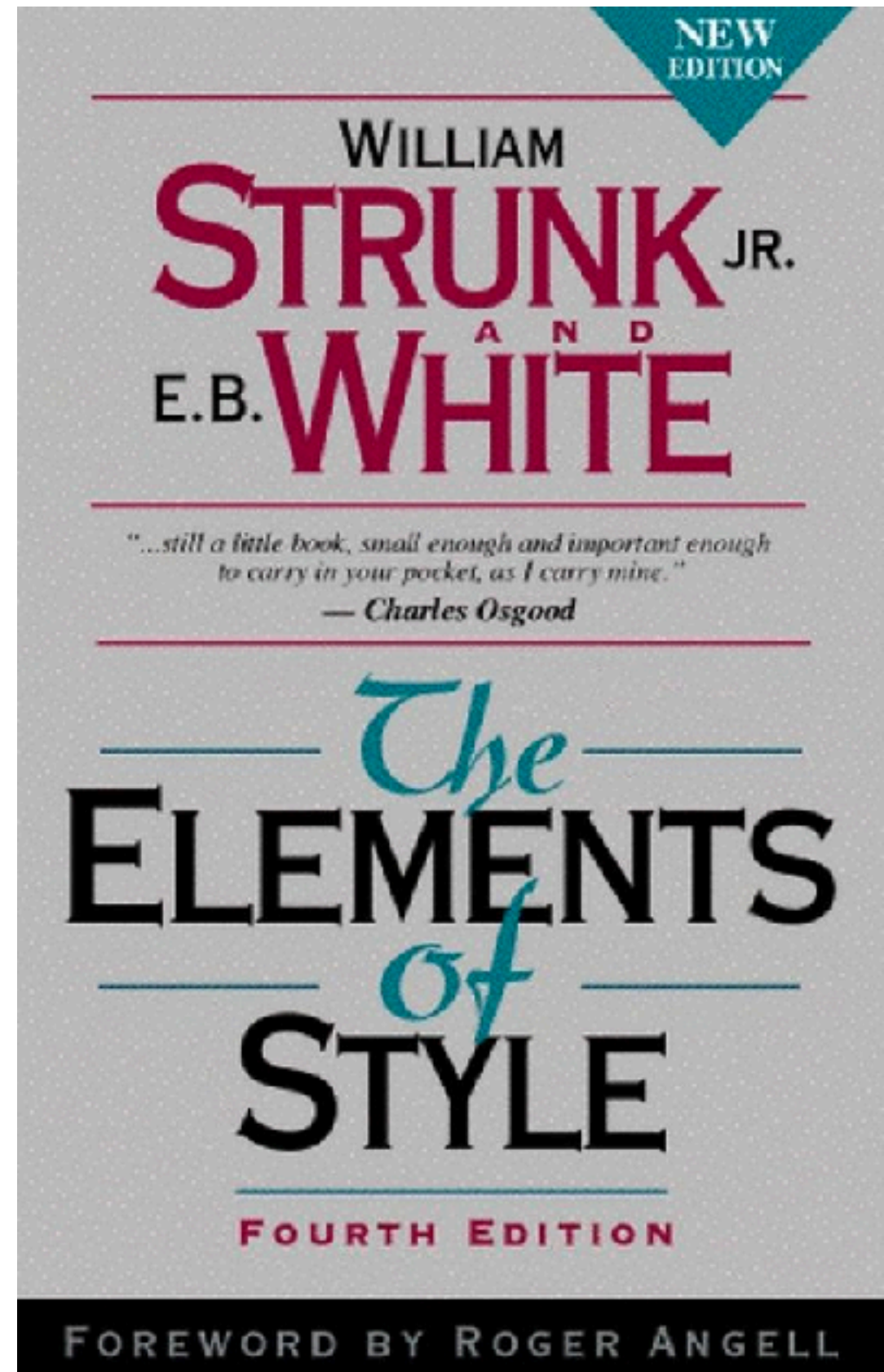
BE CLEAR!

Keep it simple
This is not a novel!

'Style guides are generally wine and water but this one ... is pure wine.' *Oldie*

How to be Clear, Correct and Persuasive on the Page

WRITE TO THE POINT
SAM LEITH
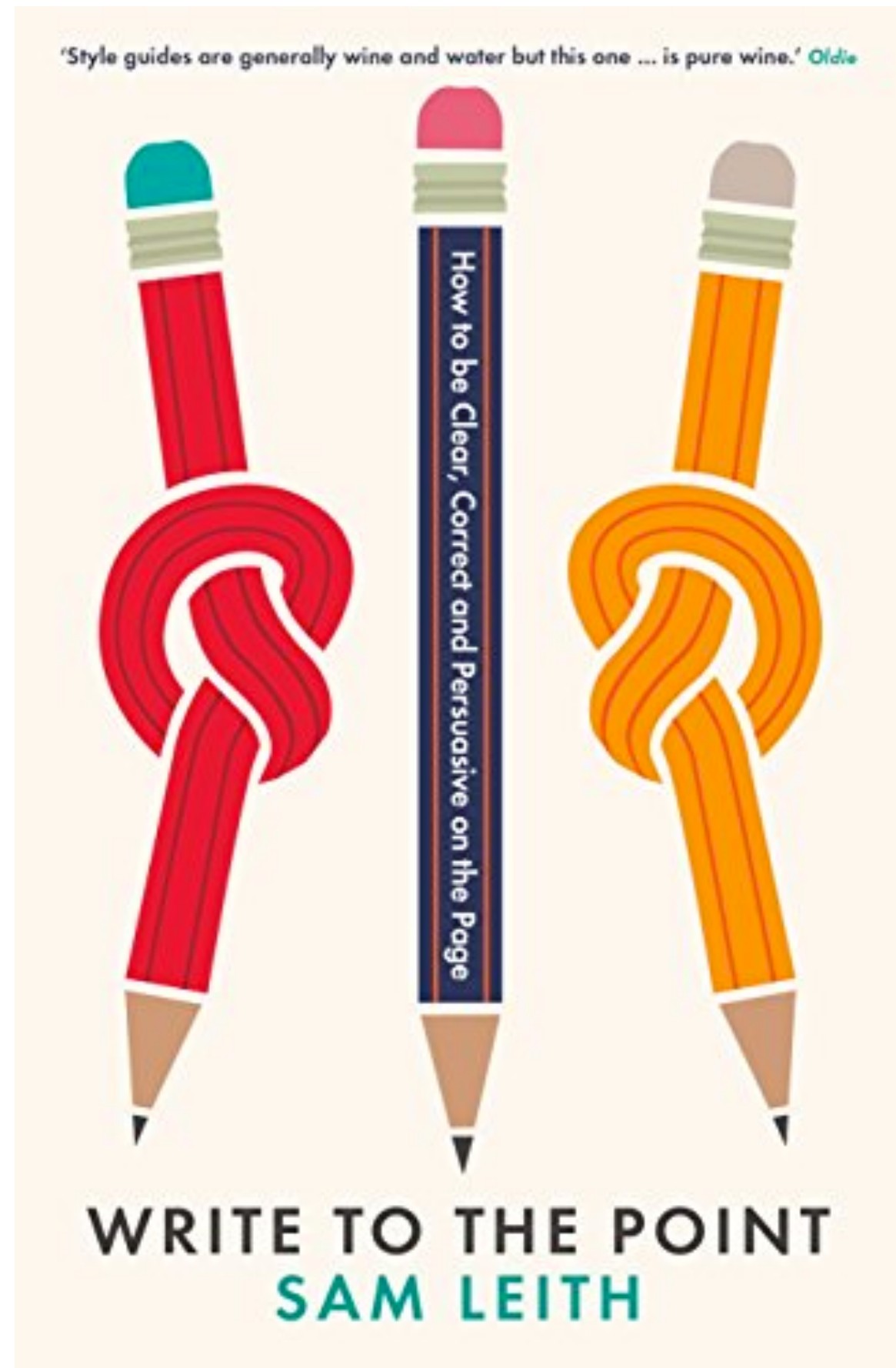
WRITE TO EXPRESS, NOT TO IMPRESS

Again, be clear!
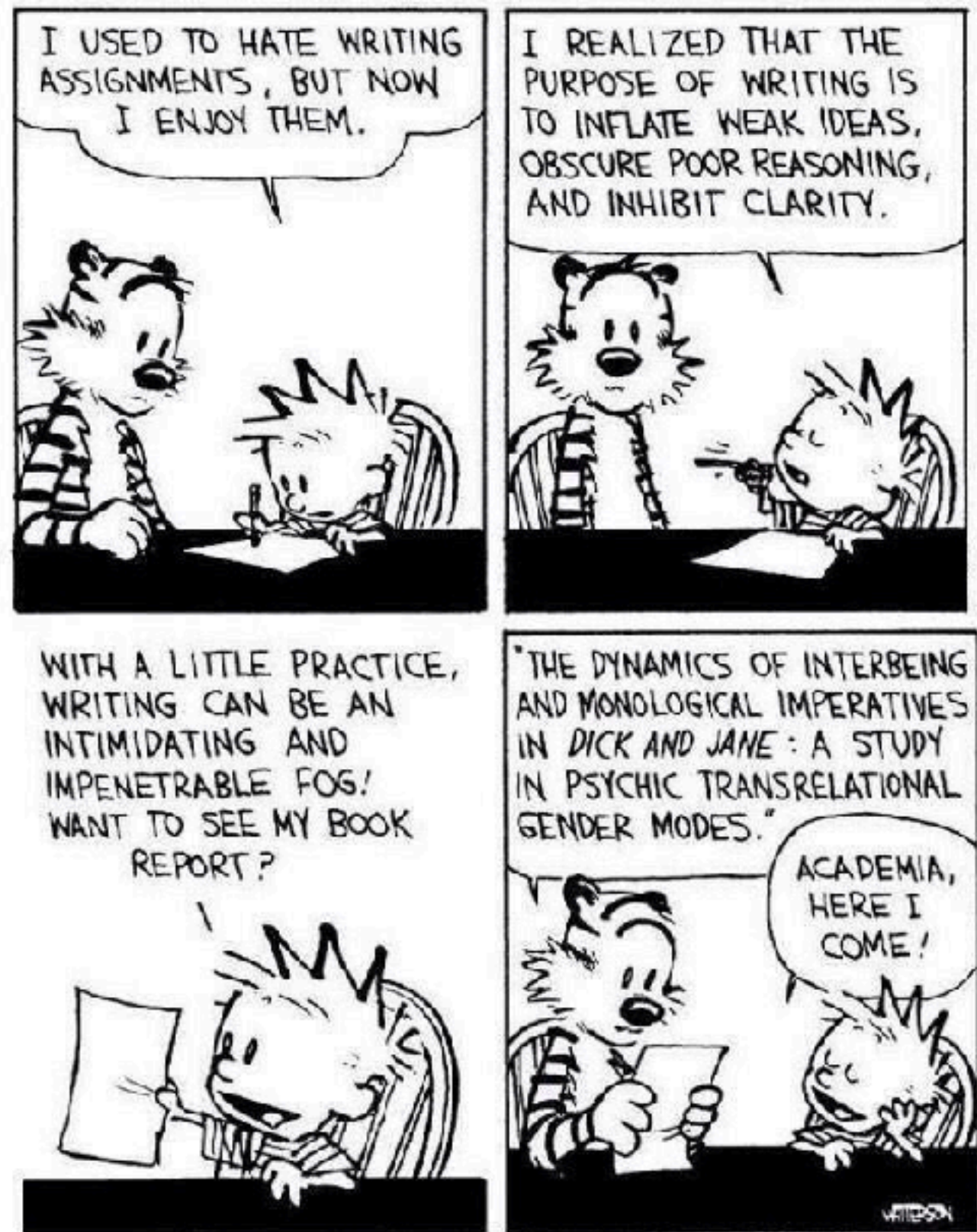
IT'S ALL ABOUT THE READER

Simple concise sentences

Be assertive, be precise, project confidence, avoid the passive

Avoid adverbs (e.g. very, highly, etc.)

Write a great opening sentence:
"All happy families are alike; each unhappy family is unhappy in its own way." Anna Karenina, by Leo Tolstoy

End with something memorable that ties everything together
"He loved Big Brother." 1984 by George Orwell
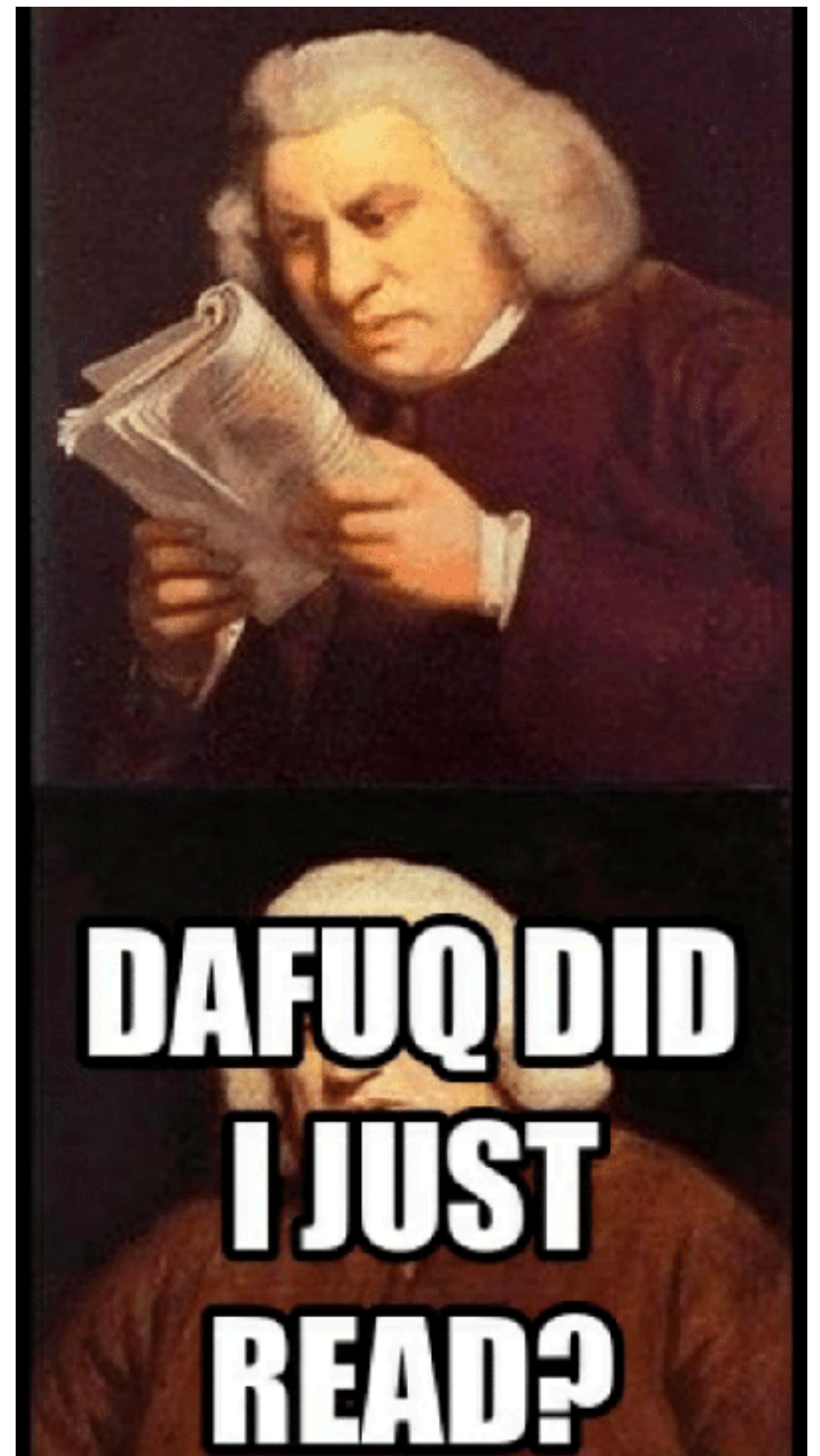
IF THE READER IS LOST, YOU HAVE FAILED!

Technical documents can be difficult to follow! This is not a novel, no plot twists!

Give a roadmap to the reader!
- E.g. At the end of the introduction: "The paper is structured as follows: In chapter 1 we..."

Tie chapters together and place the reader on the map! e.g.:
- By the end of an chapter: "in this chapter we talked about this and that, in the next one we will see this"
- In the beginning of a chapter: "In chapter X and Y we introduced whatever, now we are going to put these ideas into practice.

Remind the readers, e.g.:
- "as we mentioned before (see section 3) ..."

YOU ARE NOT A WRITER AND THAT'S OK!

If you follow the guidelines all you have to do is "fill in the blanks"
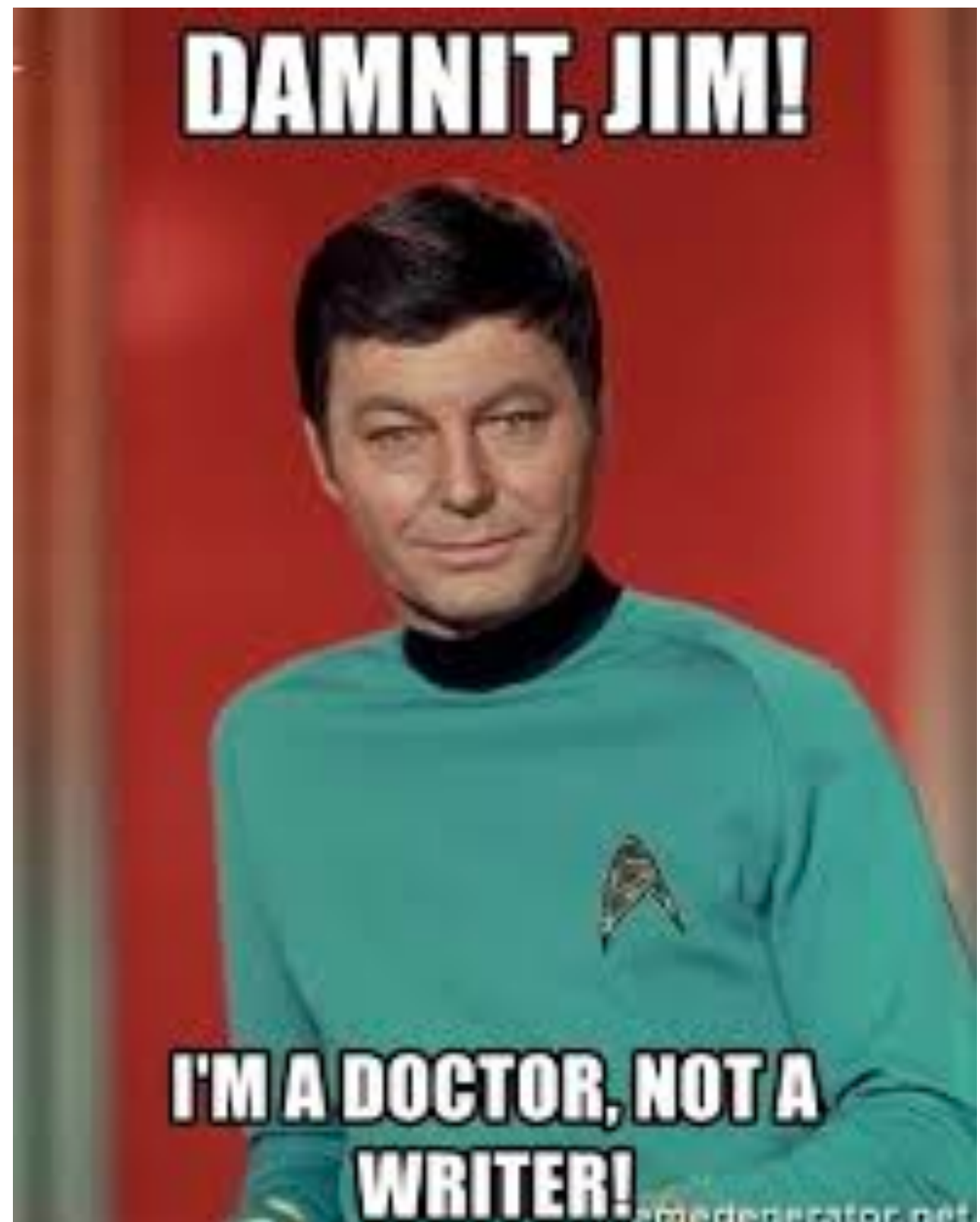
I DON'T LIKE TO WRITE!

IT TAKES TIME

Tolstoy took 6 years to write "War and Piece"
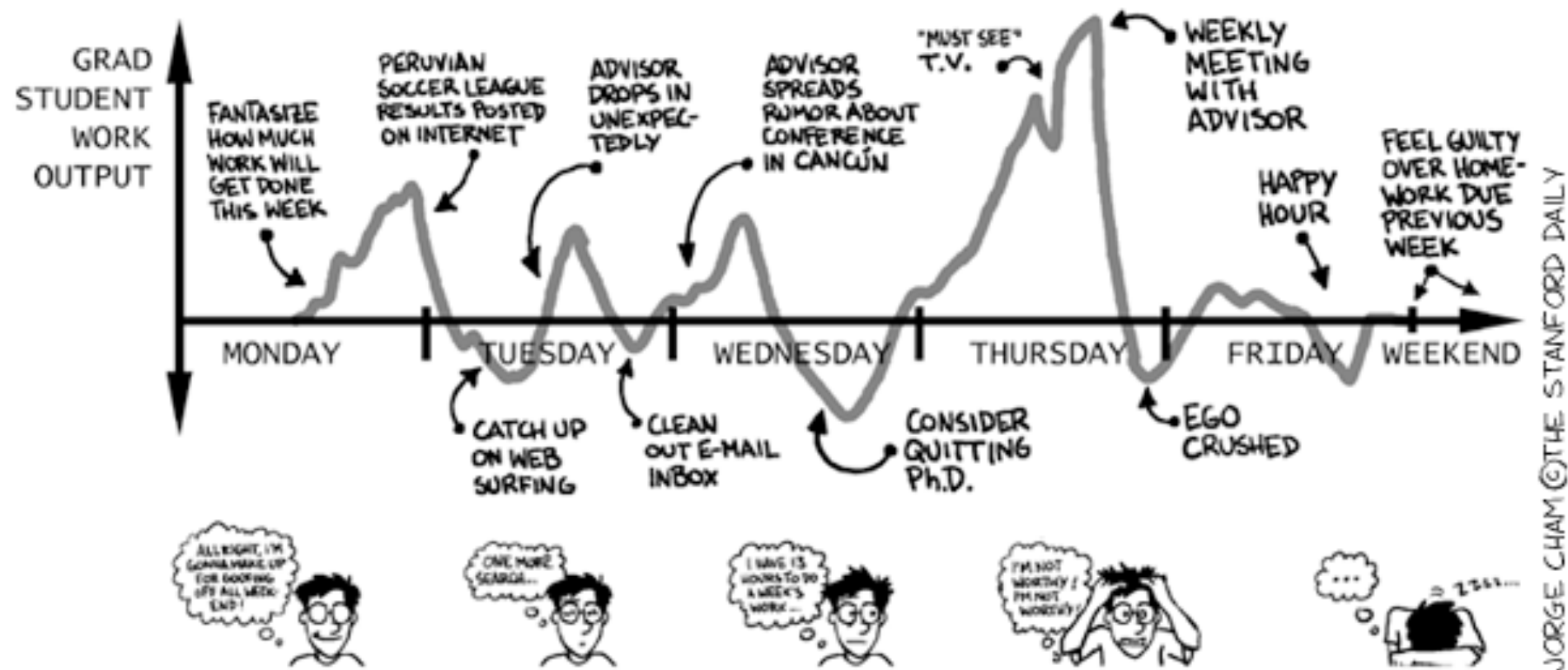
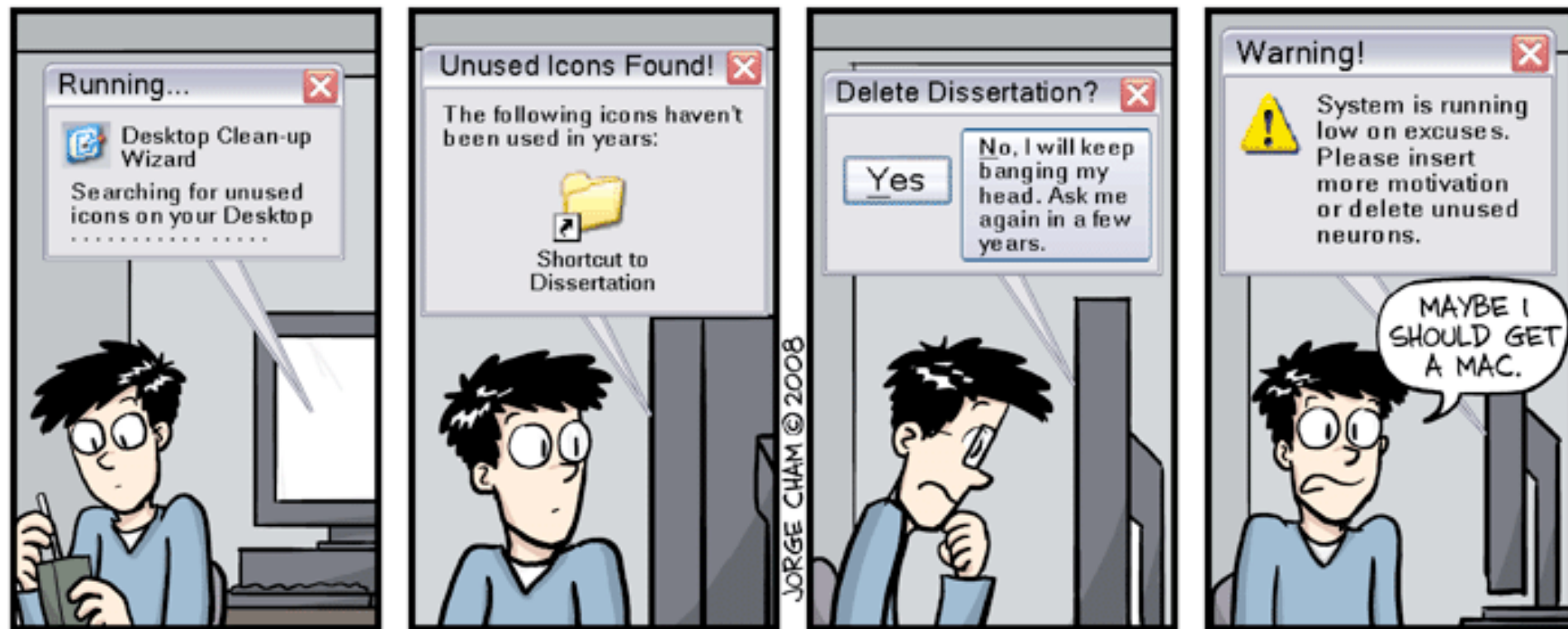Good news: your paper is shorter

Bad news: You are not Tolstoy
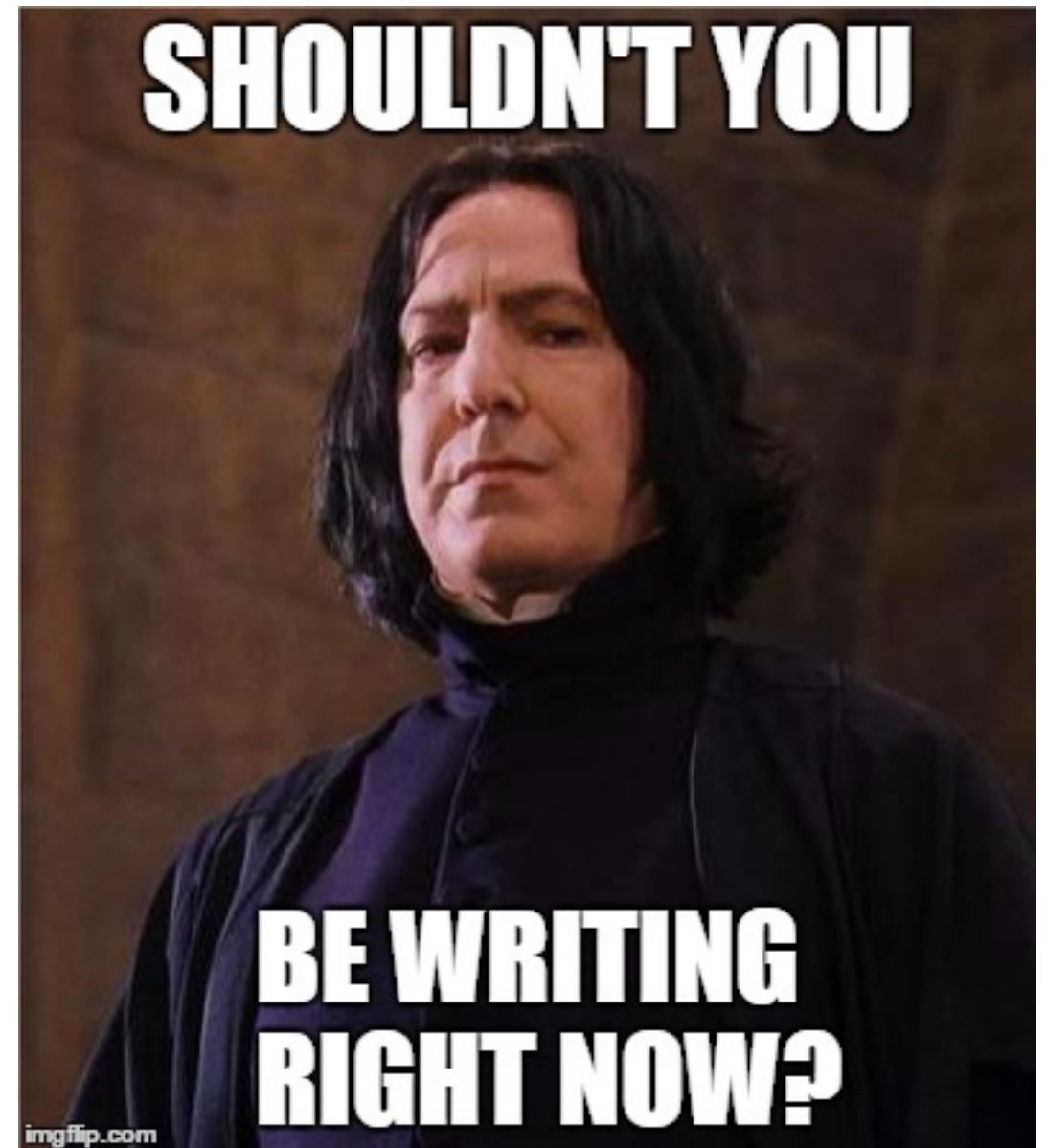
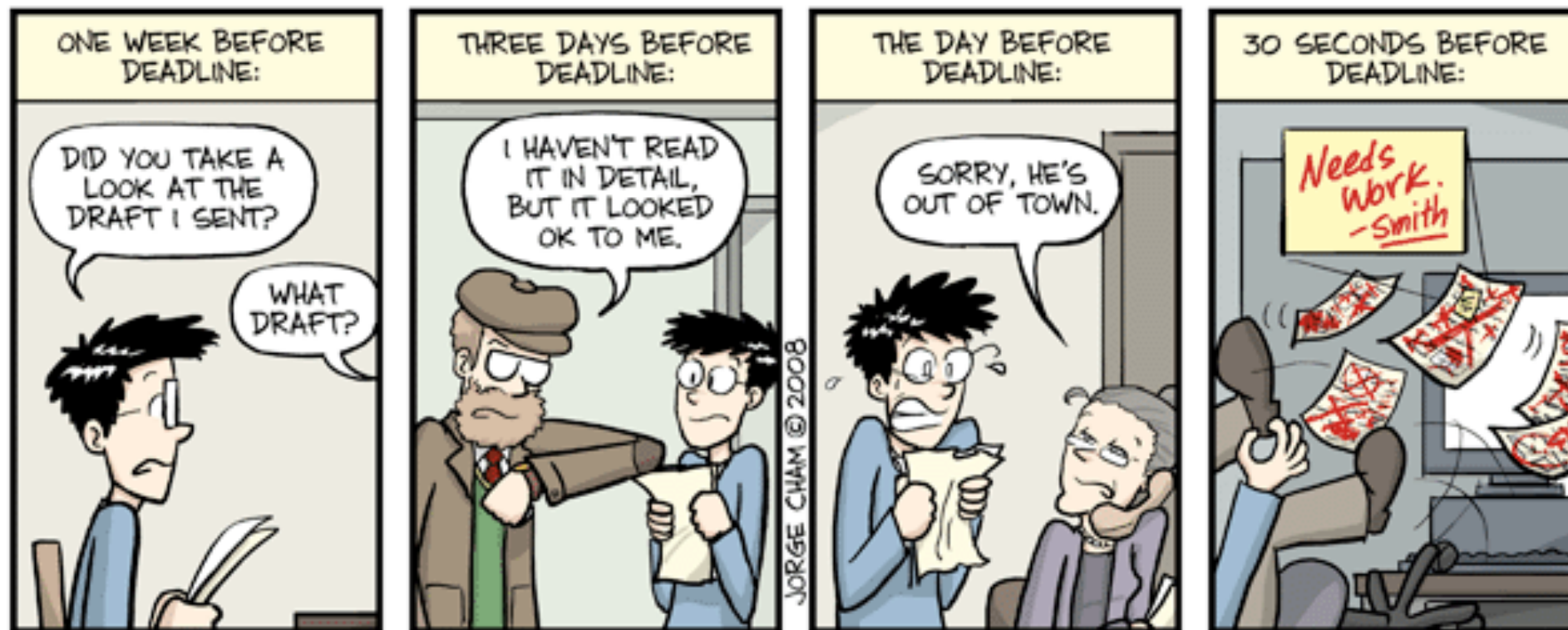# IF YOU DON'T KNOW THE STATE OF THE ART YOU SHOULDN'T BE WRITING
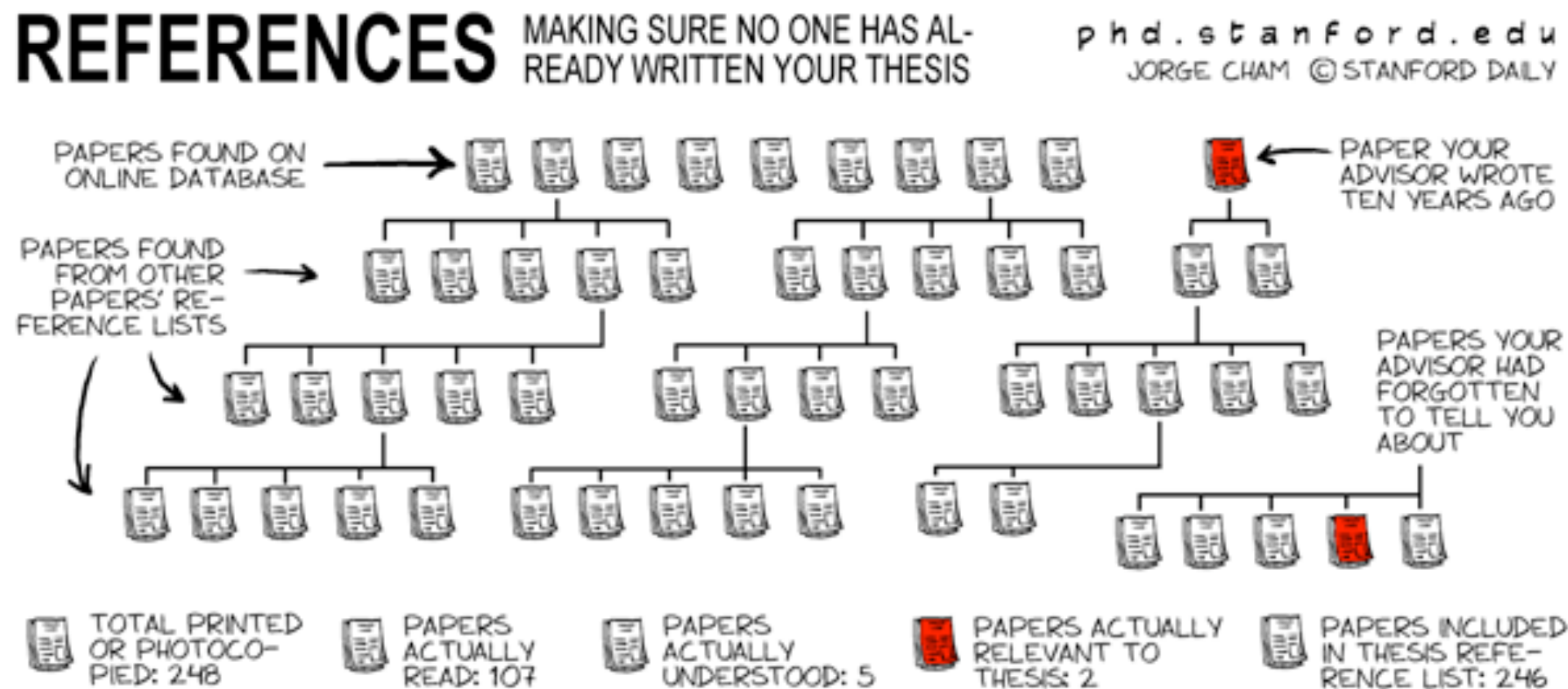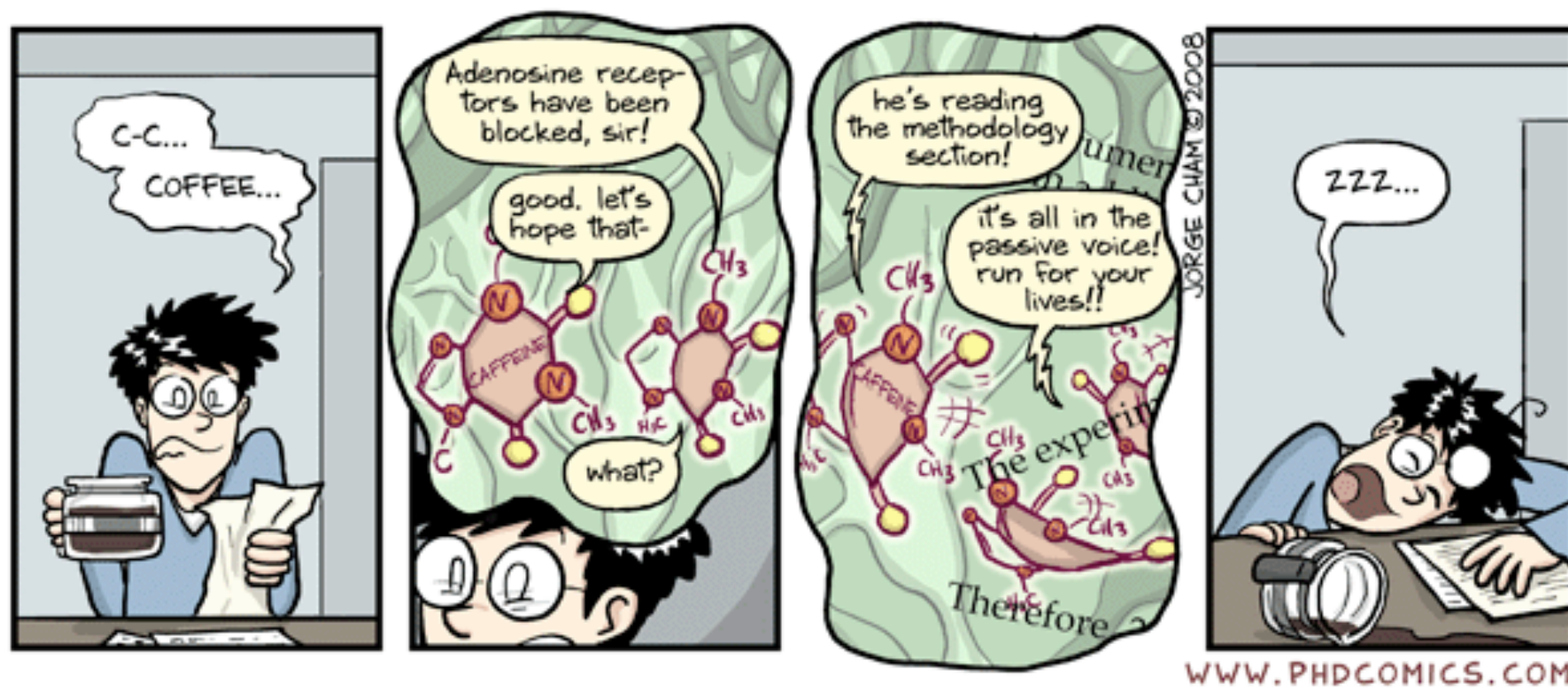
KNOW YOUR WORK CYCLE
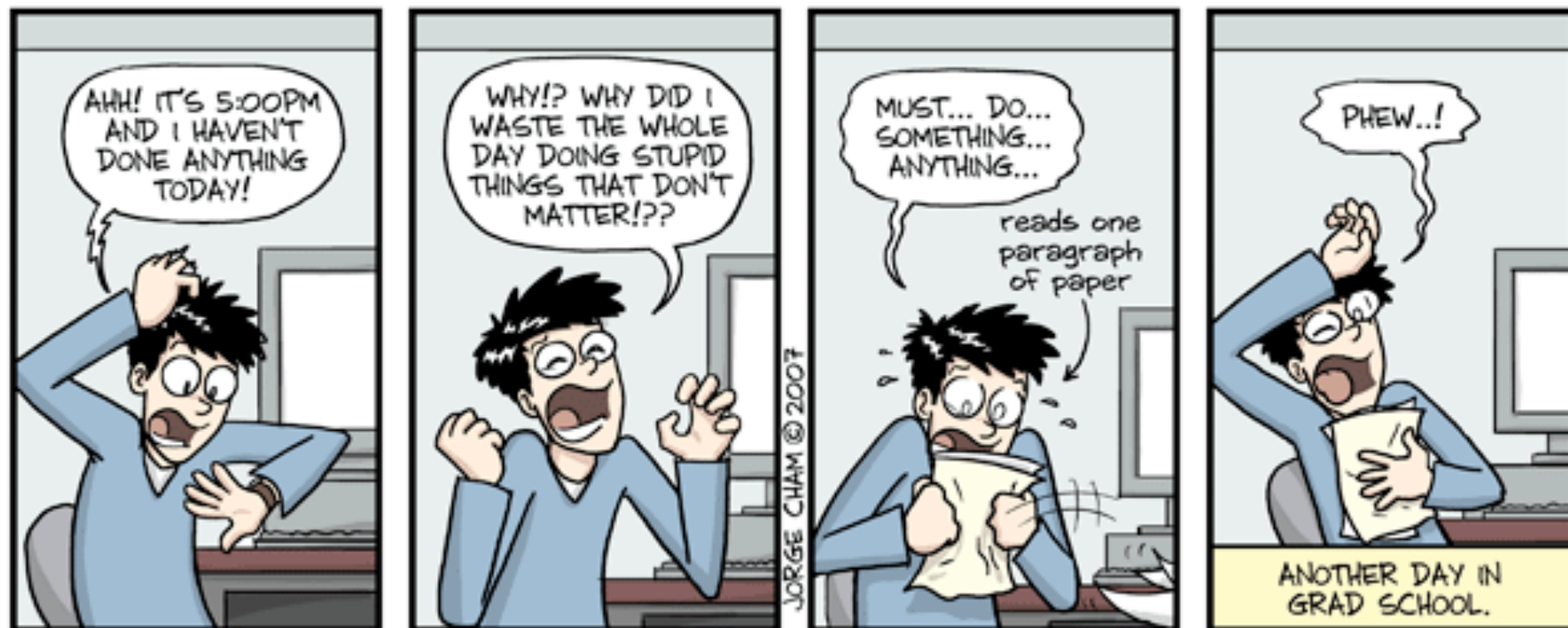
DO A BIT EVERYDAY

DON'T OVER-RELY ON YOUR SUPERVISOR

USE SIMPLE LANGUAGE

SET GOALS AND DEADLINES

AVOID PROCRASTINATION