Writing for Computer Science

Javier Larrosa

UPC Barcelona Tech

.

Writing

Good writing is one of the most **important** and most **difficult** aspects of good research

Researchers spend a lot of time doing technical writing, besides writing papers:

- drafts
- research notes
- reports
- reviews
- e-mails to colleagues
- ...

In what follows we will assume the writing of a **paper**.

Writing a Paper

Writing a paper is **codifying** your thoughts

Includes structuring knowledge, determining **concepts**, establishing **notation**,...

- **Concepts** (definitions) determine what you are going to **reason** about
- **Notation** determines your **technical vocabulary**
  - Leibniz $\int_a^b ..dx$
  - $O()$

Writing a Paper

Writing a paper is **simplifying**

In Scientific writing simplicity is beautiful

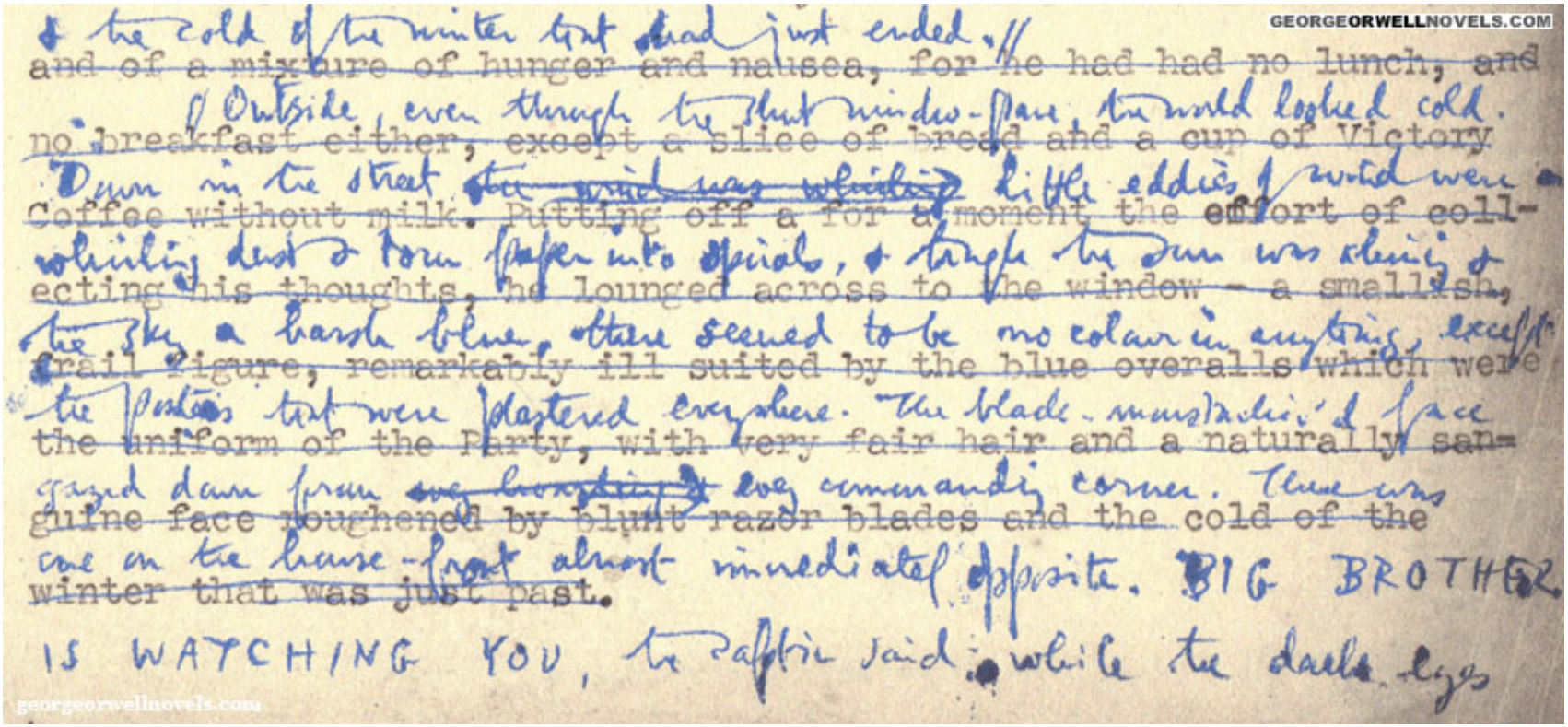
- Fitting months of creative thinking in a few lines (i.e, definitions, algorithms, theorems) is a final success.

Writing a Paper

Revision

Most of the writing time is spent in revisions

Fragment from first page of 1984's draft by George Orwell



Deletion

An important part of writing is **deleting**

- Be ready to dislike anything you have previously written

100 years Dow Jones chart:

Inform vs Entertain

The primary objective of Scientific Writing is to **inform**, not entertain

Quote from a paper on concurrent systems:

We have already seen, in our consideration of **what is**, *that the usual simplified assumptions* **lead inexorably** *to a representation that is* **desirable, because a solution is always desirable**; *but* **repugnant, because it is false**. *And we have presented* **what should be**, *assumptions whose nature is not susceptible to easy analysis but are the only tenable alternative to* **ignorance** *(absence of solution). Our choice is then Hobson's choice, to make do with what material we have - viable assumptions - and to discover whether the intractable can be* **teased** *into a useful form.*

The reader's fixed energy allocation principle

Assumption

The reader has a fixed amount of resources (time, attention, energy,...) devoted to read a paper

Corollary

Effort used to understand the form is effort not used to understand the content

The reader's profile

Prototypical reader

- Intelligent
- Experts in the field, but not on your particular research
- Busy and most probably will only skim your paper

Also write to yourself

Write to yourself

Writing is not the end of the research process, it is **integral** to it

- Vague concepts become concrete
- Complex notions become simpler
- New concepts become useful
- Written material facilitates discussion

Write to yourself

Spending time in writing your past research is a good **investment** for your future research.

Organization of a Scientific Article

Standard Structure

Allows to do **selective reading**: find quickly what you are looking for

- Title
- Abstract (short summary)
- Introduction (long summary)
- Preliminaries (technical context)
- Contribution (hypothesis, ideas)
- Contribution (demonstration, evidence)
- Related Work (high level context)
- Conclusions and Future work (learned lessons)
- References

Standard Structure

.

Break the standard structure if there is a good reason for it:

- A section is too short or too long w.r.t. others
- Some material is too far from where it is needed
- ...

.

Being original is not a good reason

Selective Reading

Number of readers of a paper in a good conference (educated guess):

- Title and Authors (300)
- Abstract (100)
- Introduction (50)
- Preliminaries (10)
- Contribution, ideas (10)
- Contribution, results (20)
- Related Work (10-20)
- Conclusions and Future work (50)

Title and Authors

- **Title**:
  - Try to be **sexy**, **concise** and **informative**
    - *Limited Discrepancy AND/OR Search and Its Application to Optimization Tasks in Graphical Models.*
    - *Of Mousetraps and Men: A Cautionary Tale*
    - *Boosting Search with Variable Elimination*
- **Authors**:
  - Use the same name on all your papers
    - better short and simple but catchy
  - In CS the order (relatively) matters

Limited Discrepancy AND/OR Search and Its Application to Optimization Tasks in Graphical Models

Javier Larrosa, Emma Rollon
UPC Barcelona Tech
Barcelona, Spain

Rina Dechter
University of California, Irvine
Irvine, California, USA

Abstract

If your paper was a movie, the abstract would be the **official trailer**

.

The purpose of the abstract is to allow readers to judge whether or not the article is of relevance to them. It is a **snapshot** of the paper.

- Single paragraph of 100-200 words
- Summary of scope, achievement and conclusions
- Self contained
- Written for as broad audience as possible
- Should not contain acronyms, abbreviations, mathematics
- Specific
  - *space requirements can be significantly reduced*
  - *space requirements can be reduced by a 60%*
- It may determine the reviewers

Example

MiniMaxSat: An Efficient Weighted Max-SAT Solver

Federico Heras FHERAS@LSI.UPC.EDU
Javier Larrosa LARROSA@LSI.UPC.EDU
Albert Oliveras OLIVERAS@LSI.UPC.EDU
Technical University of Catalonia, LSI Department
Jordi Girona 1-3, 08034, Barcelona, Spain.

Abstract

In this paper we introduce MiniMaxSat, a new Max-SAT solver that is built on top of MiniSat+. It incorporates the best current SAT and Max-SAT techniques. It can handle hard clauses (clauses of mandatory satisfaction as in SAT), soft clauses (clauses whose falsification is penalized by a cost as in Max-SAT) as well as pseudo-boolean objective functions and constraints. Its main features are: learning and backjumping on hard clauses; resolution-based and substraction-based lower bounding; and lazy propagation with the two-watched literal scheme. Our empirical evaluation comparing a wide set of solving alternatives on a broad set of optimization benchmarks indicates that the performance of MiniMaxSat is usually close to the best specialized alternative and, in some cases, even better.

Example

On the Practical use of Variable Elimination in Constraint Optimization Problems: 'Still-life' as a Case Study

Javier Larrosa LARROSA@LSI.UPC.EDU
Enric Morancho ENRICM@AC.UPC.EDU
David Niso NISO57@CASAL.UPC.EDU
Universitat Politecnica de Catalunya
Jordi Girona 1-3, 08034 Barcelona, Spain

Abstract

Variable elimination is a general technique for *constraint processing*. It is often discarded because of its high space complexity. However, it can be extremely useful when combined with other techniques. In this paper we study the applicability of variable elimination to the challenging problem of finding *still-lifes*. We illustrate several alternatives: variable elimination as a stand-alone algorithm, interleaved with search, and as a source of good quality lower bounds. We show that these techniques are the best known option both theoretically and empirically. In our experiments we have been able to solve the $n = 20$ instance, which is far beyond reach with alternative approaches.

Introduction

If your paper was a movie, the introduction would be the **plot**

.

The Introduction is like a expanded version of the abstract. It should tell (and high-level explain) your achievements.

- The article's **topic**
- The **problem** being studied and why it is important.
- The **approach** to the solution
  - Consider giving an **example** if the main idea can be extracted out of it
  - Why this is **original**
  - Why it is **useful**
- **Scope** and limitations of the solution
- Relevant literature should be cited as needed
- Do not include complex definitions or mathematics

Preliminaries

.

The Preliminaries are used to provide **background knowledge** (definitions, ideas from previous works) that is used in the paper.

- Always **cite the sources**, but you can change notation for your convenience.
- **Be concise** (assume the reader knows it, else he can go to the sources)
- Consider adding **examples** if they can be re-used later on (e.g. to illustrate your contributions)

Preliminaries

fact

Sometimes I have fully understood a paper years later, when reading the preliminaries of a posterior paper mentioning it here.

Algorithm elim-mpe
Input: A belief network $BN = \{P_1, ..., P_n\}$; an ordering of the variables, d; observations e.
Output: The most probable assignment.
1. **Initialize:** Generate an ordered partition of the conditional probability matrices, $bucket_1, \ldots, bucket_n$, where $bucket_i$ contains all matrices whose highest variable is X_i. Put each observed variable in its bucket. Let $S_1, ..., S_j$ be the subset of variables in the processed bucket on which matrices (new or old) are defined.
2. **Backward:** For $p \leftarrow n$ downto 1, do
for all the matrices $h_1, h_2, ..., h_j$ in $bucket_p$, do
• (bucket with observed variable) **if** $bucket_p$ contains $X_p = x_p$, assign $X_p = x_p$ to each h_i and put each in appropriate bucket.
• **else**, $U_p \leftarrow \bigcup_{i=1}^{j} S_i - \{X_p\}$. Generate functions $h_p = \max_{X_p} \Pi_{i=1}^{j} h_i$ and $x_p^o = argmax_{X_p} h_p$. Add h_p to bucket of largest-index variable in U_p.
3. **Forward:** Assign values in the ordering d using the recorded functions x^o in each bucket.

J. Larrosa and R. Dechter

function BE(F, CC)
1. **if** $F = \emptyset$ **then** $\{CC$ contains a constant function $f\}$
2. $\quad BestCost \leftarrow f$
3. **else**
4. $\quad x_i \leftarrow$ SelectVariable(F);
5. $\quad B_i \leftarrow \{f \in C | x_i \in var(f)\}$
6. $\quad f_i \leftarrow \text{elim}_i(\sum_{f \in B_i} f)$;
7. $\quad CC \leftarrow CC \cup \{f_i\} - B_i$;
8. $\quad$ BE$(F - \{x_i\}, CC)$;
9. $\quad x_i \leftarrow$ best extension of the assignment to $(x_1, \ldots, x_{i-1})$ relative to $\sum_{f \in B_i} f$

Related Work

.

The Related Work is used to compare the new results to similar results in the literature

- Research is usually an extension of or correction to previous work
- It is not always easy to relate different papers
- Puts yours and related work in common terminology
- Also helps readers to understand the work
- Points to standard references
- May appear after the Introduction (1), before the Conclusions (2) or diluted along the paper (3)
  1. Helps to understand the technical contents of your contribution
  2. Helps identify the originality of your work

Contribution

- A small portion of the paper (i.e, 20-40% in a conference paper, 60% in a journal paper)
- The structure should be evident in the (sub-)section headings
- Narrative **flow** and **clear logical structure** is essential
- **It must include**:
  - Detailed hypothesis and major original concepts (already presented informally in the Introduction)
  - Chain of reasoning that leads to the conclusions
  - Details of contributions (central proofs, algorithms, definitions)
  - Summarize experimental setup
  - Important results (summarized in a table?)
  - Details that support the conclusions
- **You may omit** (or report briefly):
  - Minor results
  - Details of proofs of lemmas or minor theorems

Usual Structures of the Contribution

- **Chain**
  - When the motivation of one results comes from the previous one
- **By specificity**
  - First give outline and then fill in details
  - **Example:** Introduce an algorithm in terms of generic data structures and later on show the best choices (e.g. a particular type of hash table)
- **By example**
  - First apply ideas to some typical case and then explain more formally
  - **Example:** Introduce a search algorithm in the context of chess and then generalize
- **By complexity**
  - First explain over a simple case and then generalize to the most general case
  - **Example:** Introduce an algorithm in the Integers domain, and then generalize it to Rings (algebraic abstraction)

Conclusions, Bibliography, Appendices

- **Conclusion**: closing summary
  - Do not repeat the abstract
  - Can **look beyond** the current paper and be **speculative** about variations, extensions, consequences,...
- **Bibliography**: list of cited references, links and (rarely) personal communications
  - Most people do not read the papers they cite
  - Typos are propagated
  - Some people self-cite too much
- **Appendices**: holds bulky material that would interfere with the narrative flow, or material that most readers do not care about (unaggregated results, code, lengthly proofs)
  - Usually not necessary (consider giving a link instead)

Composition

Fact

An article is not a novel

- Do not keep secrets until the end
- *Spoilers* should be all over the place

Fact

An article is not a diary

- Do not get influenced on **how did you get there**
- Do not get influenced on **how much effort you put** on the different parts

Composition

1. Think of **what do you want to say**: choose what to emphasize and what to discard
2. A conference paper reports a sharp result (one idea)
3. A journal paper may group several related ideas (possibly from conference papers)

Composition

1. Write a **template**: work out a logical sequence of sections and subsections (writing the titles)
2. **Sketch each section** in 20-200 words
3. Write a first draft of the **Introduction**
4. Write a first draft of the sections containing the **contributions**
5. **Iterate** over the different sections **editing**, **deleting** and **adding**
   - During the first iterations do not worry much about style.
   - Along iterations move from the **what to say**, to the **how to say** it
   - Expect to dislike what you wrote the day before
6. Leave the **Abstract** and the **Conclusion** for the end.