The Art of Reading

How do I get the most out of this research paper?

Aaron Stump
Computer Science
The University of Iowa

Coffee and Conversations, Computer Science Dept., Nov. 14, 2018

Is Reading Important?

Is Reading Important?

"I like books. Have you read *A Game of Thrones*?"

Is Reading Important?

"I like books. Have you read *A Game of Thrones*?"

"I am a coder. Does reading code count?"

Is Reading Important?

"I like books. Have you read *A Game of Thrones*?"

"I am a coder. Does reading code count?"

"I have tried reading papers, but it is pretty awful."

Is Reading Important?

"I like books. Have you read *A Game of Thrones*?"

"I am a coder. Does reading code count?"

"I have tried reading papers, but it is pretty awful."

"I am too busy writing papers to read them."

Is Reading Important?

"I like books. Have you read *A Game of Thrones*?"

"I am a coder. Does reading code count?"

"I have tried reading papers, but it is pretty awful."

"I am too busy writing papers to read them."

"I hope I can finish all these papers so I can start research!"

Is Reading Important?

"I like books. Have you read *A Game of Thrones*?"

"I am a coder. Does reading code count?"

"I have tried reading papers, but it is pretty awful."

"I am too busy writing papers to read them."

"I hope I can finish all these papers so I can start research!"

Add your own!

Yes! (Reading is important)

- As educated people, read on many topics (history, culture, etc.)

- As researchers, we need to read **papers**
  - Learn!
  - Advanced material rarely in textbooks
  - Be able to cite previous related work
  - Reading code can also be good – but papers key

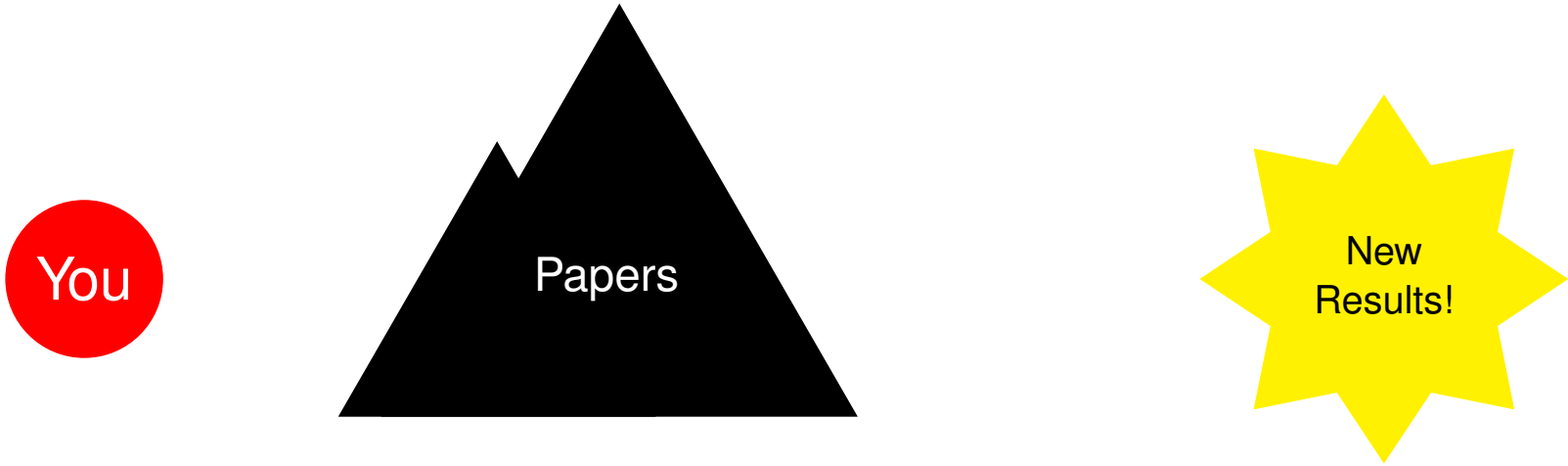
- But we cannot know everything...

Yes! (Reading is important)

- As educated people, read on many topics (history, culture, etc.)

- As researchers, we need to read **papers**
  - Learn!
  - Advanced material rarely in textbooks
  - Be able to cite previous related work
  - Reading code can also be good – but papers key

- But we cannot know everything...

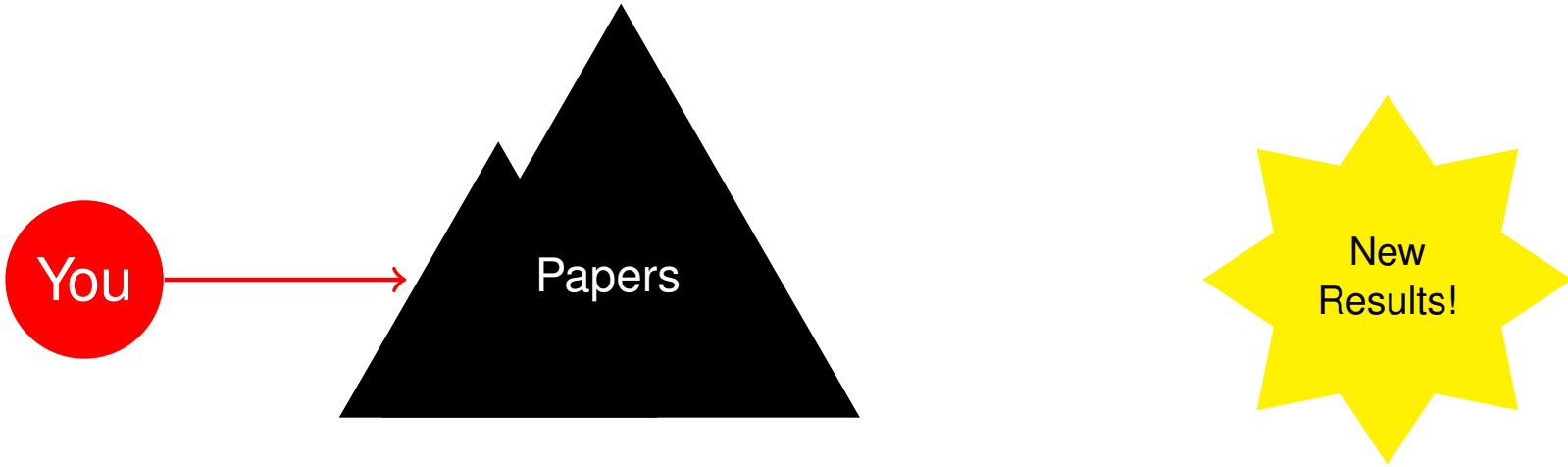


Yes! (Reading is important)

- As educated people, read on many topics (history, culture, etc.)

- As researchers, we need to read **papers**
  - Learn!
  - Advanced material rarely in textbooks
  - Be able to cite previous related work
  - Reading code can also be good – but papers key

- But we cannot know everything...

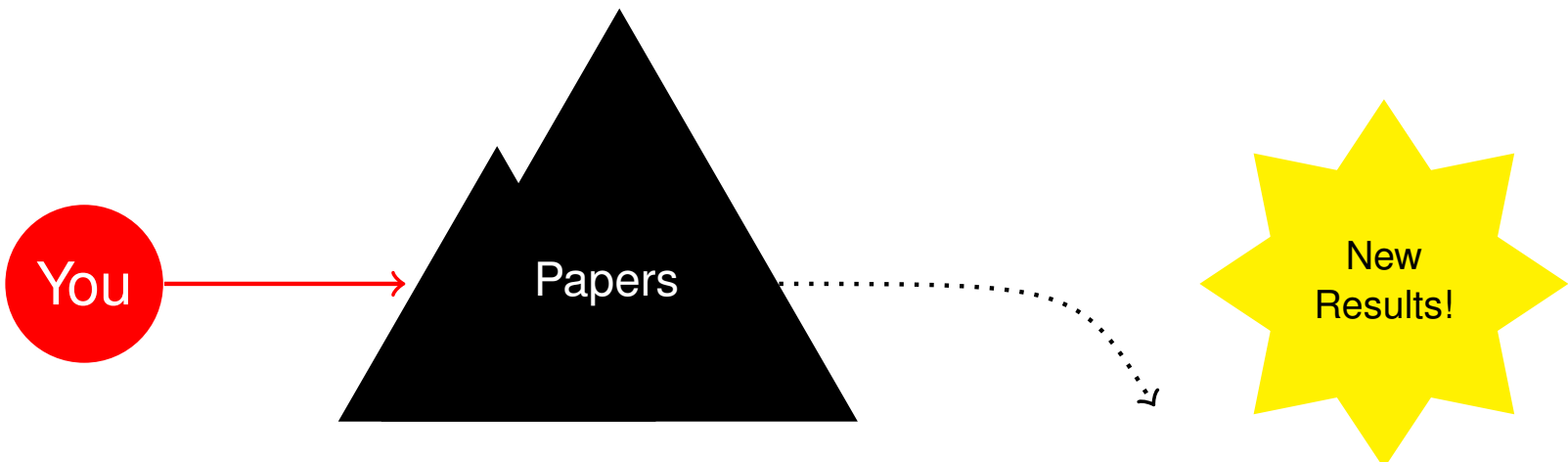


Yes! (Reading is important)

- As educated people, read on many topics (history, culture, etc.)

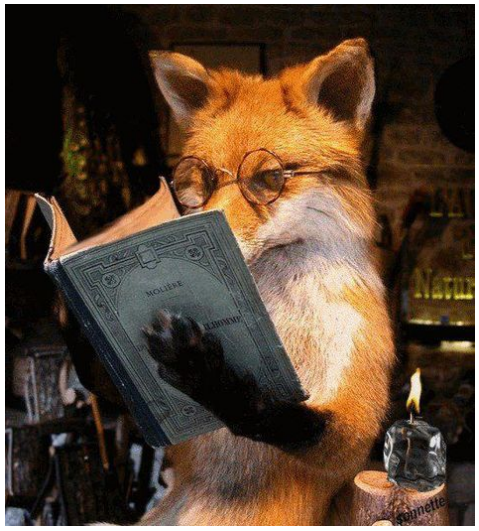
- As researchers, we need to read **papers**
  - Learn!
  - Advanced material rarely in textbooks
  - Be able to cite previous related work
  - Reading code can also be good – but papers key

- But we cannot know everything...

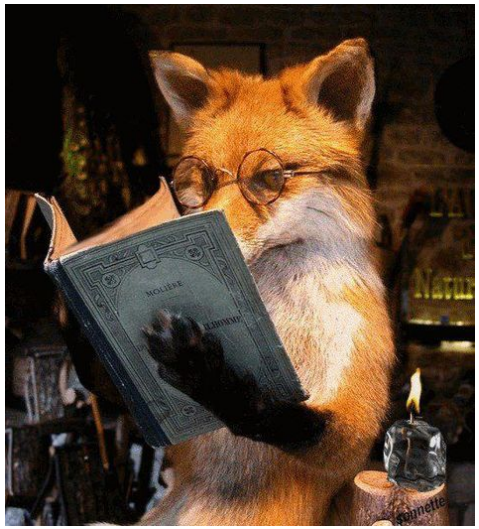


But how do I do it?

But how do I do it?



But how do I do it?



It all depends on what “it” is

- Completely digest every detail?

It all depends on what “it” is

- Completely digest every detail?
- Strengthen your core knowledge?

It all depends on what “it” is

- Completely digest every detail?
- Strengthen your core knowledge?
- Explore a new direction?

It all depends on what “it” is

- Completely digest every detail?
- Strengthen your core knowledge?
- Explore a new direction?
- Assess the contribution?

It all depends on what “it” is

- Completely digest every detail?
- Strengthen your core knowledge?
- Explore a new direction?
- Assess the contribution?
- Relate to your own work?

Kinds of reading

- Reading for evaluation/assessment
  - Reviewing a paper
  - Reading competing work
  - Literature review (your comps!)
- Reading for core knowledge
  - Work directly relevant for your research
  - Background material for your area (e.g., some field of math)
- Exploratory reading
  - What was that crazy paper at Top Conference about?
  - Is there anything to this cryptocurrency stuff?
  - Subfield X seems to use related techniques to mine.

More?

Three Reading Adventures

A Framework for Defining Logics

ROBERT HARPER

Carnegie-Mellon University, Pittsburgh, Pennsylvania

FURIO HONSELL

Università di Udine, Udine, Italy

AND

GORDON PLOTKIN

Edinburgh University, Edinburgh, United Kingdom

Abstract. The Edinburgh Logical Framework (LF) provides a means to define (or present) logics. It is based on a general treatment of syntax, rules, and proofs by means of a typed λ-calculus with dependent types. Syntax is treated in a style similar to, but more general than, Martin-Löf's system of arities. The treatment of rules and proofs focuses on his notion of a *judgment*. Logics are represented in LF via a new principle, the *judgments as types* principle, whereby each judgment is identified with the type of its proofs. This allows for a smooth treatment of discharge and variable occurrence conditions and leads to a uniform treatment of rules and proofs whereby rules are viewed as proofs of higher-order judgments and proof checking is reduced to type checking. The practical benefit of our treatment of formal systems is that logic-independent tools, such as proof editors and proof checkers, can be constructed.

Categories and Subject Descriptors: F.3.1 [**Logics and Meanings of Programs**]: Specifying and Verifying and Reasoning about Programs; F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic

General Terms: Algorithms, Theory, Verification

Additional Key Words and Phrases: Formal systems, interactive theorem proving, proof checking, typed lambda calculus

A Framework for Defining Logics

ROBERT HARPER

Carnegie-Mellon University, Pittsburgh, Pennsylvania

FURIO HONSELL

Università di Udine, Udine, Italy

AND

GORDON PLOTKIN

Edinburgh University, Edinburgh, United Kingdom

Abstract. The Edinburgh Logical Framework (LF) provi
It is based on a general treatment of syntax, rules, and pr
dependent types. Syntax is treated in a style similar t
system of arities. The treatment of rules and proofs foc
are represented in LF via a new principle, the *judg*
judgment is identified with the type of its proofs. This al
and variable occurrence conditions and leads to a uniform treatment of rules and proofs whereby rules are viewed as proofs of higher-order judgments and proof checking is reduced to type checking. The practical benefit of our treatment of formal systems is that logic-independent tools, such as proof editors and proof checkers, can be constructed.

Categories and Subject Descriptors: F.3.1 [**Logics and Meanings of Programs**]: Specifying and Verifying and Reasoning about Programs; F.4.1 [**Mathematical Logic and Formal Languages**]: Mathematical Logic

General Terms: Algorithms, Theory, Verification

Additional Key Words and Phrases: Formal systems, interactive theorem proving, proof checking, typed lambda calculus

- JACM 1993 paper, 1700 citations
- 42 pages long
- Suffered through for about 1 year as an early doctoral student
  - Lacked knowledge of conventions
  - Paper also tough reading

Infinitary Logics

- I got curious about “logics” with infinite formulas
  - Infinite branching reasonably well understood
  - Infinitely deep much less studied

- Deep dive in the library, including Finnish dissertations

- Wrote this up on my blog after the dust settled:

  `https://queuea9.wordpress.com/2012/07/12/infinitary-logics/`

Writing a summary of what you read!

Treasured friends

However obscure the venue
However ignored or neglected
Some works seem to befriend you
Now enlightened, enriched, and corrected

- Henk Barendregt, “Lambda Calculi with Types”, Handbook of Logic in CS, 1993
- Jean-Yves Girard, “Proofs and Types”, 1989

To conclude

- Reading is an art
  - What should I read?
  - How? (evaluation, core knowledge, exploratory)
- You will improve with practice
- Your research will benefit

> Take your place in the company of educated men and women throughout the ages!