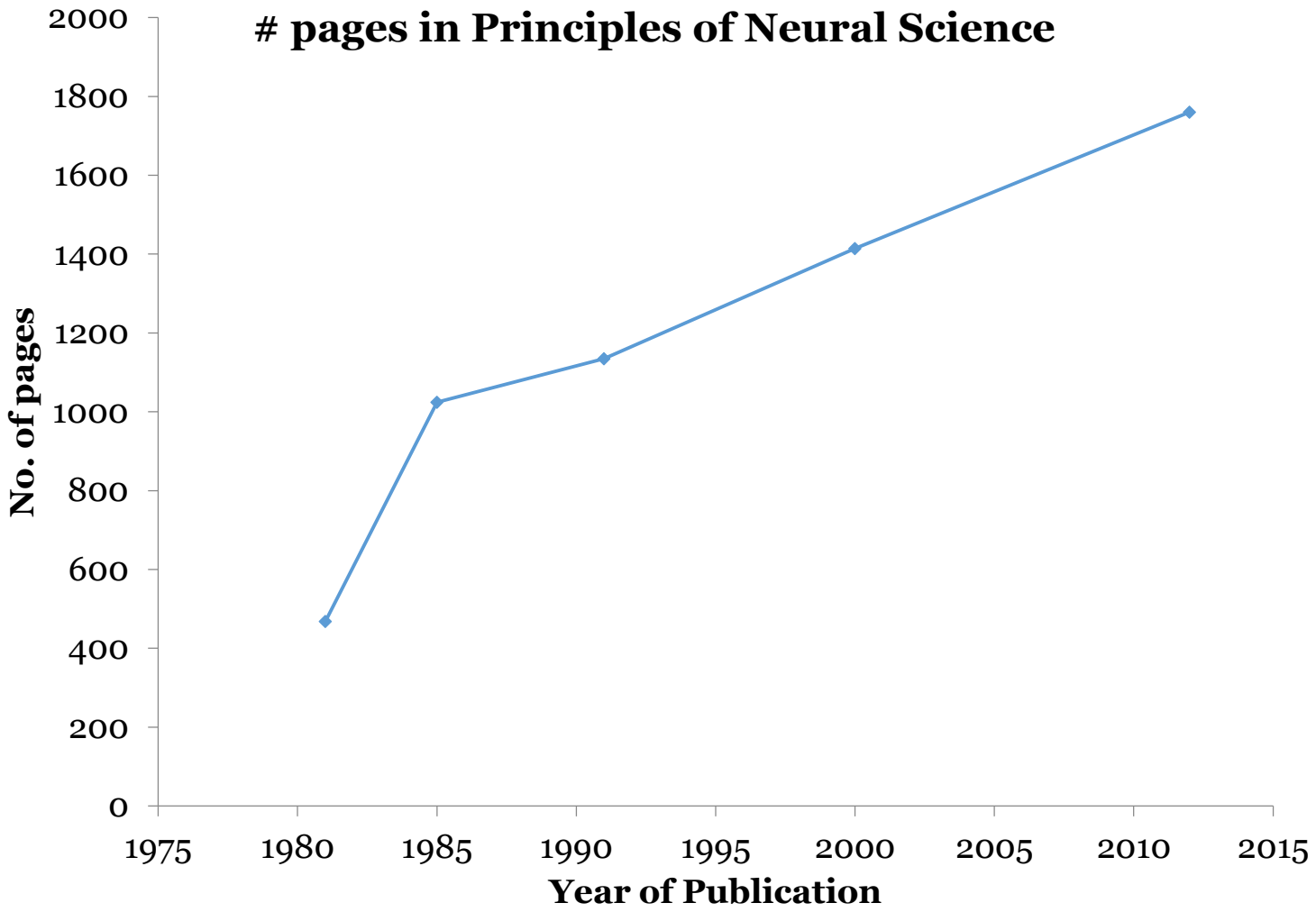
An introduction to computational modelling of visual word recognition

Colin J. Davis & James S. Adelman

easyNet

The Leverhulme Trust

Models are needed to simplify (make sense of) large amounts of data



(thanks to Daniel Wolpert)

Charles Darwin on data/theories

- "About thirty years ago there was much talk that geologists ought only to observe and not theorize; and I well remember some one saying that at this rate a man might as well go into a gravel pit and count the pebbles and describe all the colours. How odd it is that anyone should not see that **all observation must be for or against some view if it is to be of any service!**"

  (letter to friend, 1861)

Computers and models

- Computers as a metaphor …
- and as a tool for simulating mind
- Herbert Simon (1916-2001)
- “as of 2016 the most cited person in Artificial Intelligence and Cognitive Psychology on Google Scholar” (276,193 citations)

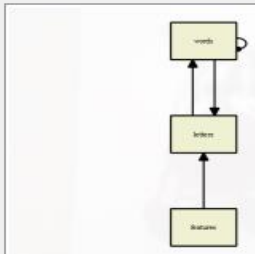
Simon and Newell's (1958) predictions

1. That within ten years, a digital computer will be the world's chess champion, unless the rules bar it from competition.
2. That within ten years a digital computer will discover and prove an important new mathematical theorem.
3. That within ten years a digital computer will write music that will be accepted by critics as possessing considerable aesthetic value.
4. That within ten years most theories in psychology will take the form of computer programs, or of qualitative statements about the characteristics of computer programs.

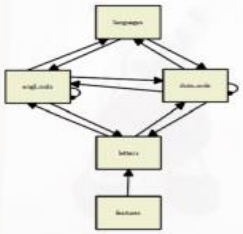
Why computational modelling?

- Requires modeller to be explicit about workings of model and all its assumptions
- Successful implementation is an existence proof
- Allows analysis & exploration of systems that are otherwise too complex to study
- Allows generation of novel predictions

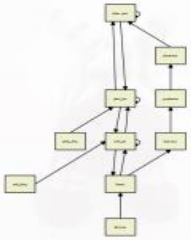
Verbal Models	**Computational Models**
Hard to communicate model (know what I mean?)	The program is the model
No specific tools for generating predictions	Predictions are derived from simulations
Predictions are often vague and are unquantified	Predictions are exact
Predictions may be contradictory	A single model can't produce contradictory predictions
Hard to falsify	Readily falsifiable



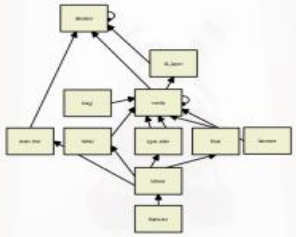
IA



Bilingual IA

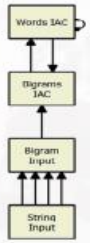


CDP+

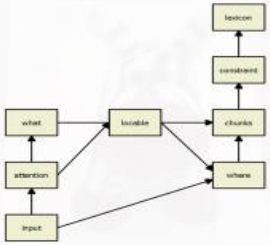


Spatial Coding

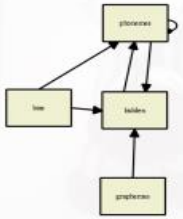
Relative Position



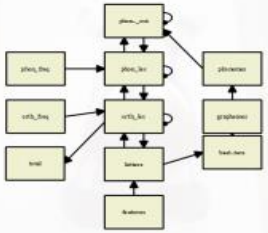
Overlap Open Bigram



LTRS



PMSP (recurrent)



DRC

The promise of modeling

WARWICK
THE UNIVERSITY OF WARWICK

Computational models — mathematical descriptions of theory complete enough to calculate predictions — promise key benefits.

- Open and objective
- Quantitative and precise
- Self-consistent

Should contribute to theoretical and practical falsifiability

WARWICK
THE UNIVERSITY OF WARWICK

Problems with modeling in practice

Models are frequently discussed, but much less frequently are simulations published.

Problems: Proprietariness

- Models are used and extended by their authors and sometimes their co-authors
- Programs aren’t always available
- Programs are often inflexible and idiosyncratic
- Source code often not viable for re-use in terms of compatibility or readability

Problems: Limited range of models

- Relatively few researchers have the skills (or alternatively, resources) to engage in modeling
- Other researchers do not have their ideas turned into models, which could they could then test

Problems: Poor understanding

- Complex model descriptions in articles are often incomplete or vague in places
- Source code is specific, but readability is usually lacking, even for experts
  - Code can have a poor structural relationship with the verbal description of the model, limiting understanding
- Neither can substitute for the experience of interacting with (breaking) the model

Problems: Model testing

- The quality of experimental tests of models is more clearly demonstrable with simulations
  - Indeed, experiments could be designed with sight of such simulations
- Results inconsistent with models would be less easily dismissed if inconsistent simulations were presented

WARWICK
THE UNIVERSITY OF WARWICK

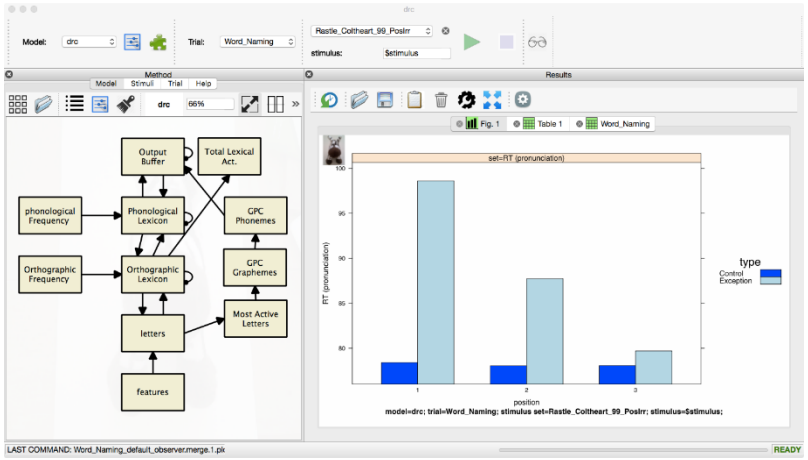
Ad-hoc software is partly at fault

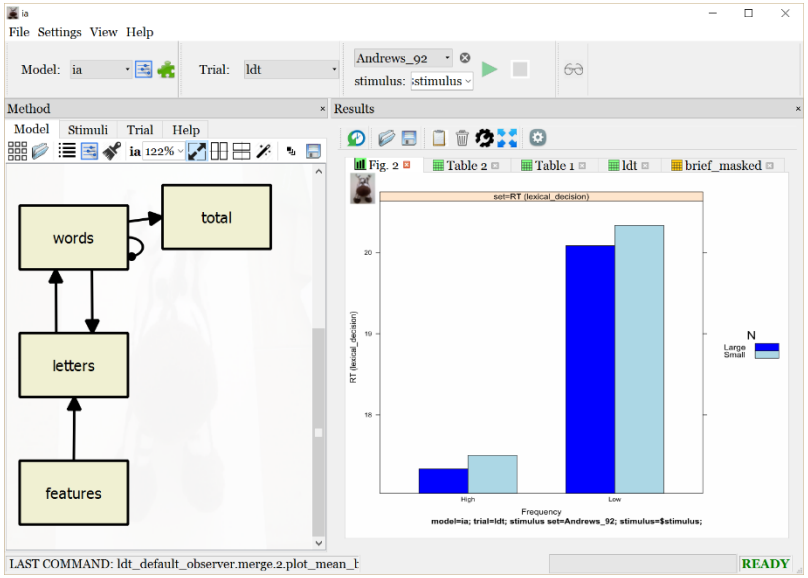
- *Model-specific software* has limited extensibility and high barrier-to-entry
- *Teaching-specific software* requires learning that does not generalize, and becomes outdated rapidly
- *Framework-specific toolkits* are rarely exhaustive, limit generalization and do not match scope of researcher interests

easyNet is our answer

easyNet

The Leverhulme Trust





easyNet: Accessible but serious

- Multiple types of users and uses: experienced and novice; model development and simple simulation
- Give a real path to learn about modeling and working with state-of-the-art models
- Remove from individual modelers the burden of providing accessible interfaces
- Provide efficient methods for manipulating and extending model concepts

easyNet: Models and more models

- Multi-model and multi-framework: For different models and types of model
- Reduce unnecessary burden on learners
- Open up new types of hybrid models
- Make fairer model comparisons more easy
- First, provide access to well-known models
- Do and encourage new modelling in *easyNet*

Session 1

WARWICK
THE UNIVERSITY OF WARWICK

- Introduction to the *easyNet* interface
- A classic example: interactive activation
- Familiarization worksheet
- Extending interactive activation: primed lexical decision
- Extending interactive activation: bilingual interactive activation

Session 2: Reading aloud, dual-route theory

Session 3: Form priming, the state-of-the-art