Connections and symbols II

AT1

Emmanuel Dupoux
summary of preceding session

• “computational reduction”:
  – reduction of unboundedly complex behavior to the combination of simple ones
    • simple set of primitive processes
    • finite set of data types
    • a finite set of operations that combine the primitive processes to make more complex ones
  – what computational mechanisms underly complex behaviors (like language, reasoning, etc)?
    • Symbolic IA: (sequential & deterministic) computations with symbols and rules
      – eg: Turing machines, rewrite rules, finite state automata
    • Connexionist IA: (parallel & stochastic) computation with (continuous valued) neurone-like units
      – eg: Multilevel Perceptrons, Boltzman machines

• The Fodor & Pylyshyn challenge:
  – (current) connectionist architectures fail to capture complex behaviors
  – (future) connectionist architectures are ‘mere’ implementation of symbolic architectures
The Fodor & Pylyshyn argument

• mental representations have a constituent structure
  – they are not atomic or hollistic but have parts with specific roles
    • eg: the red cow; cheese or desert, Vx R(x), A→B
  – Some constituents can be recursive
    • eg: P. thinks that « M. is nice » → J thinks that « P thinks that « M is nice » »

• mental processes are structure sensitive
  – eg: combinatorial semantics
    • semantics of « J. loves M. » derived from semantics of « J. », « loves » and « M. »
  – eg: logical inferences:
    • A→B, A entails B ; this does not depend on the meaning of A and B but on the structure of the representations

• as a result, mental computations are
  – systematic
    • all humans are mortal -> John is moral, Mary is mortal, etc.
    • « Paul likes fruits » grammatical -> « Paul likes fruits » also grammatical, etc
  – productive (achieve discrete infinity)
    • the list of thoughts/sentences is not finite (I can construct new thoughts with old ones)

• connectionist representations have none of these properties
Possible responses to the Fodor & Pylyshyn critique

– level confusion
  • F&P talk about a descriptive level not a computational one; the descriptive level is compatible with many architectures including connectionnist ones; indeed, none of the physical implementations of symbol structures would satisfy the F&P criteria (eg, a physical computer).

– process confusion
  • F&P talk about conscious deliberative explicit thought processes (which are symbolic), not intuitive ones (which could be subsymbolic)

– Implementation matter
  • constructing a neurally plausible implementations of symbol manipulation is non trivial and interesting, and could reveal unexplained phenomena (eg graceful degradation)

– artificial dichotomy
  • There are many systems intermediate between classical architectures and connectionist ones. It is an empirical issue which one is appropriate to modelling human cognition.

– F&P criticize some classes of connectionist architectures, they do not demonstrate their points for all possible architectures.
  • Potential counterexamples:
    – Recurrent Networks (Elmann)
    – Tensor products (Smolensky)
Elman

• structure of the paper
  – representing time
  – SRN architecture
  – xor through time
  – badiiguuu
  – word segmentation (15 words)
  – part of speech (13 categories, 29 words, 15 sentence templates)
Backprop applied
• structure of Smolensky
  – representing structures by fillers and roles
    • examples: trees, lists, etc
  – tensor products and filler/role binding (definition)
    • local, semilocal and distributed
  – unbinding (exact and selfaddressed)
  – capacity and graceful saturation
  – continuous and infinite structures
  – binding and unbinding networks
  – analogy between binding units and hebb weights
  – example of a stack
  – structured roles
example of tensor product representations

Paul loves Mary -> loves(Paul,Mary)
-> pred=loves, arg1=Paul, arg2=Mary
-> pred*loves+arg1*Paul+arg2*Mary

Local

Distributed
binding and unbinding

Fig. 8. A network using sigma-pi binding units to perform tensor product binding.

Fig. 9. A network using multiplicative junctions to perform tensor product binding.

Fig. 10. An extension of the network of Fig. 8 that can perform two variable bindings in parallel.
• extensions of Elman’s SRN
  – computational capacity of SRN

  – reservoir computing
    • http://reservoir-computing.org
• extensions:
  – implementation of a phonological theory (Optimality Theory) in a tensor product network with energy relaxation
    • see the Harmonic Mind (Smolensky & Legendre)
  – Escaping the explosion in nb of neurons: holographic reduced representations
    • define A * B as an operation that preserves the dimensions (eg xor, circular convolution)
Conclusions

• What about the F&P Challenge?
  – tensor products are an interesting implementation/alternative to symbolic systems
  – recurrent networks could also be an alternative, but much less understood

• The hidden debate
  – innate vs learner structures (to be continued...)

Conclusions

• empirical impact of the debate
  – past tense in English
    • rule: play->played, fax->faxed
    • exceptions: sing->sang, put->put
  • Pinker & Prince (1988)
  • procedural vs declarative memory (Ullman et al, 1997; Pinker & Ullman, 2002)

Conclusions

• empirical impact of the debate (cont)
  – statistical learning vs algebraic learning in infants
      Peña, M., Bonatti, L., Nespor, M., Mehler J. (2002). Signal-Driven Computations in Speech

  – exemplar-based versus abstract representations
    • object recognition (Biederman & Gerhardstein, 1993), face
      recognition, speech recognition (eg Goldinger, 1988; Johnson
      1997, Pierrehumbert 2001)
Extensions

• computational reduction: finding the right architecture

• other connectionist architectures
  – Kohonen’s maps (competitive learning) (Kohonen, 1982)
  – Adaptive Resonance Theory (Grossberg, 1976)
  – Reinforcement learning (Barto, Sutton, Anderson, 1983)

• other computational frameworks
  – Probabilistic/Bayesian frameworks
  – Predictive Coding/Free Energy