A CONNECTIONIST MODEL FOR IMPLICIT LEARNING

Michal Wierzchon

Consciousness-Lab, Institute of Psychology, Jagiellonian University, Krakow

CNC group seminar, Bratislava

Oct, 26th, 2011

overview

- implicit learning and artificial grammar learning
- theories on knowledge representation in artificial grammar learning
- connectionist models of artificial grammar learning
- our model
- conclusions and further directions

implicit learning

an <u>automatic</u> and <u>unconscious</u> process that leads to <u>abstract</u> knowledge that is hardly available to introspection

(cf. Reber, 1992; Higham et al., 2000)

probably stands behind an everyday life phenomena like:
- natural language acquisition
- second language learning
- expert knowledge
- procedural knowledge (e.g. sport)

the most common research paradigms are:
- sequence learning task
- control of dynamic system task
- **artificial grammar learning task**

INSTRUCTION

this is a memory experiment

please try to memorize sequences of letters that will be presented

MXR

VMRVXVR

VXVRMXT

INSTRUCTION

sequences were not random. They followed some rules that describe the relations between the elements of the sequences.

Your next task is to determine whether new sequences follow the same set of rules, or not.

MVR

irregular **regular**

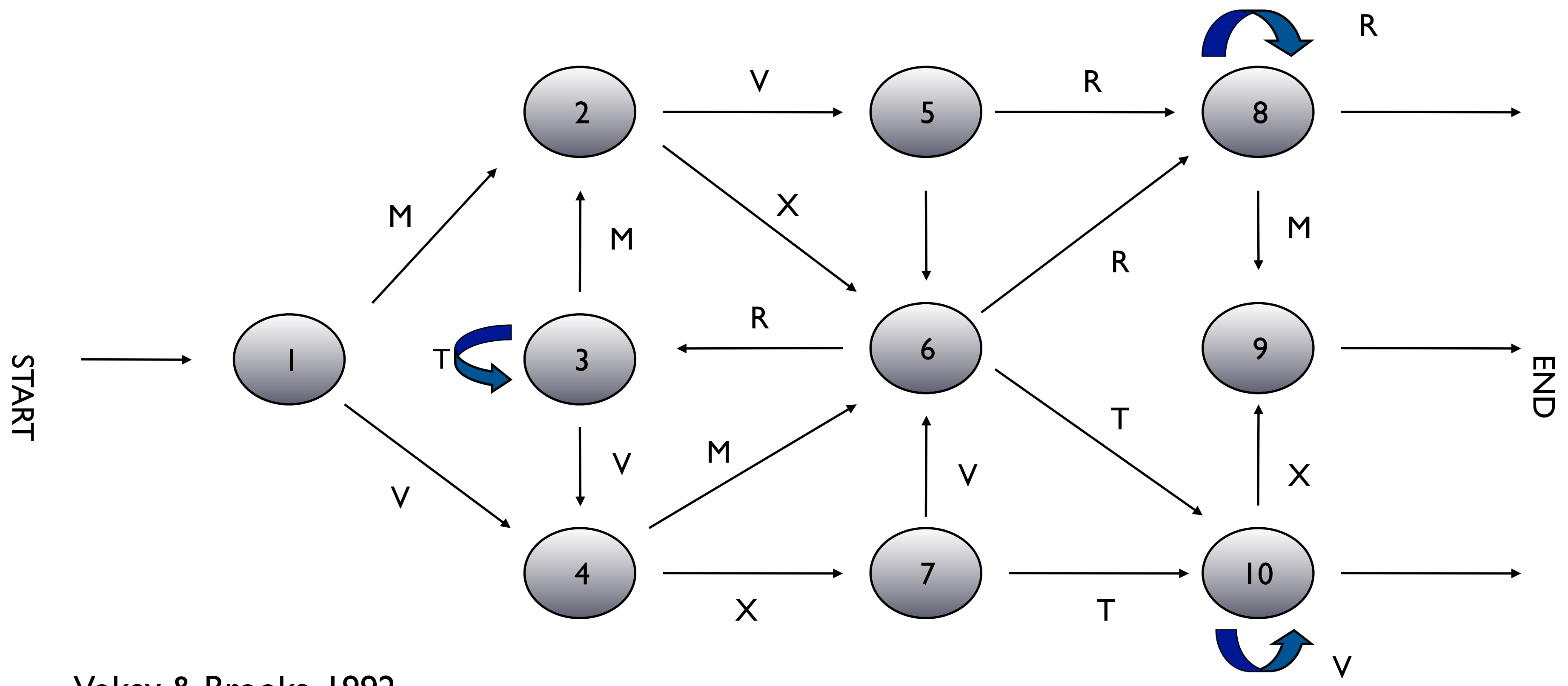
MVRVVVM

irregular ⟵ ⟶ regular

VMRMVXX

irregular **regular**

artificial grammar learning



Vokey & Brooks, 1992

Grammatical:

VXTVVX

MVRRM

Ungrammatical:

VXTTVX

RVRRM

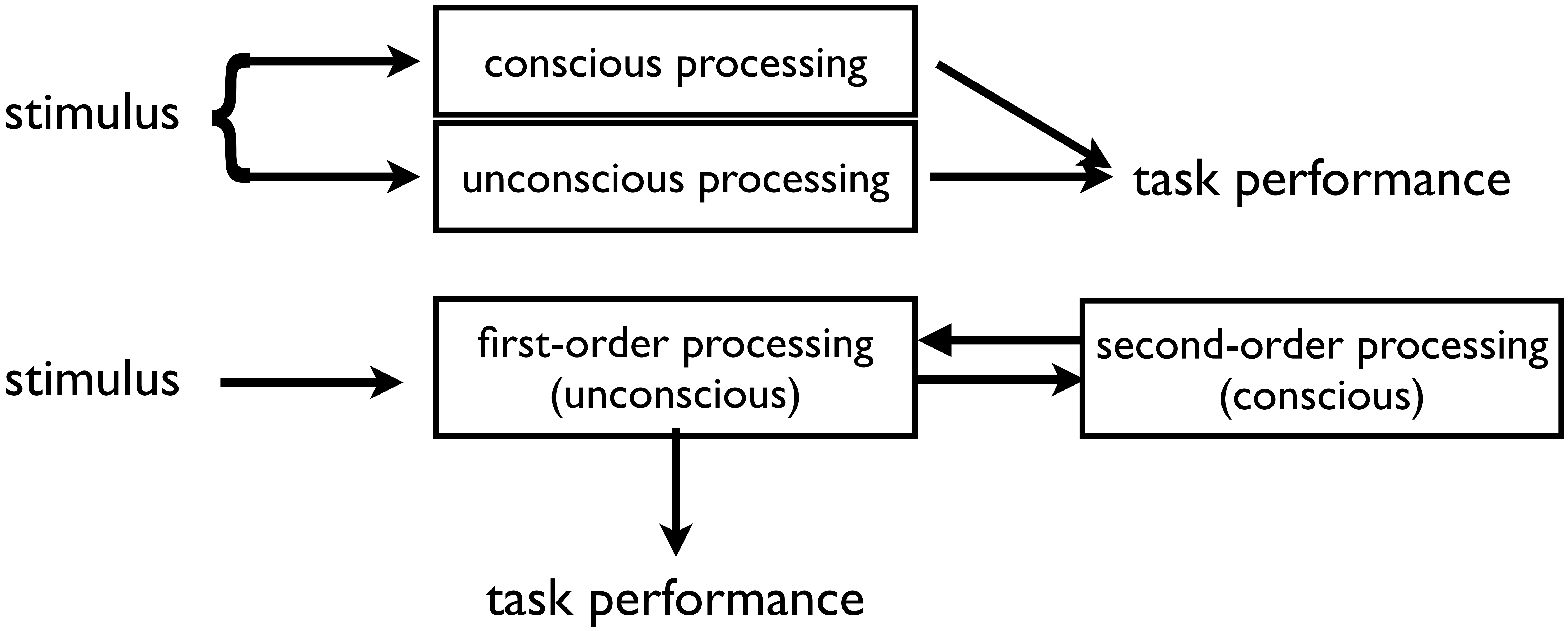
The task:

- learning phase
- information about the rules
- test phase

Result:

better than chance classification, although almost no verbal knowledge about the structure of the grammar

knowledge representation



knowledge representation

- knowledge in artificial grammar learning could be represented by:
  - abstract rules (e.g. Reber, 1989)
  - memorized exemplars (e.g. Brooks & Vokey, 1991) PTPWPT
  - memorized fragments (bigrams/trigrams - e.g. Perruchet & Pacteau, 1990) PTPWPT
  - statistic regularities of the material (e.g. Perruchet, 2008; Pothos, 2007) PTPWPT

p p p p p

knowledge representation

- you could control for that:
  - computing correlation between conscious knowledge for regular exemplars/ fragments and classification accuracy (e.g. Dulany, Carlson & Dewey, 1984)
  - controlling frequencies of regular fragments in the classification material (Higham, 1997)

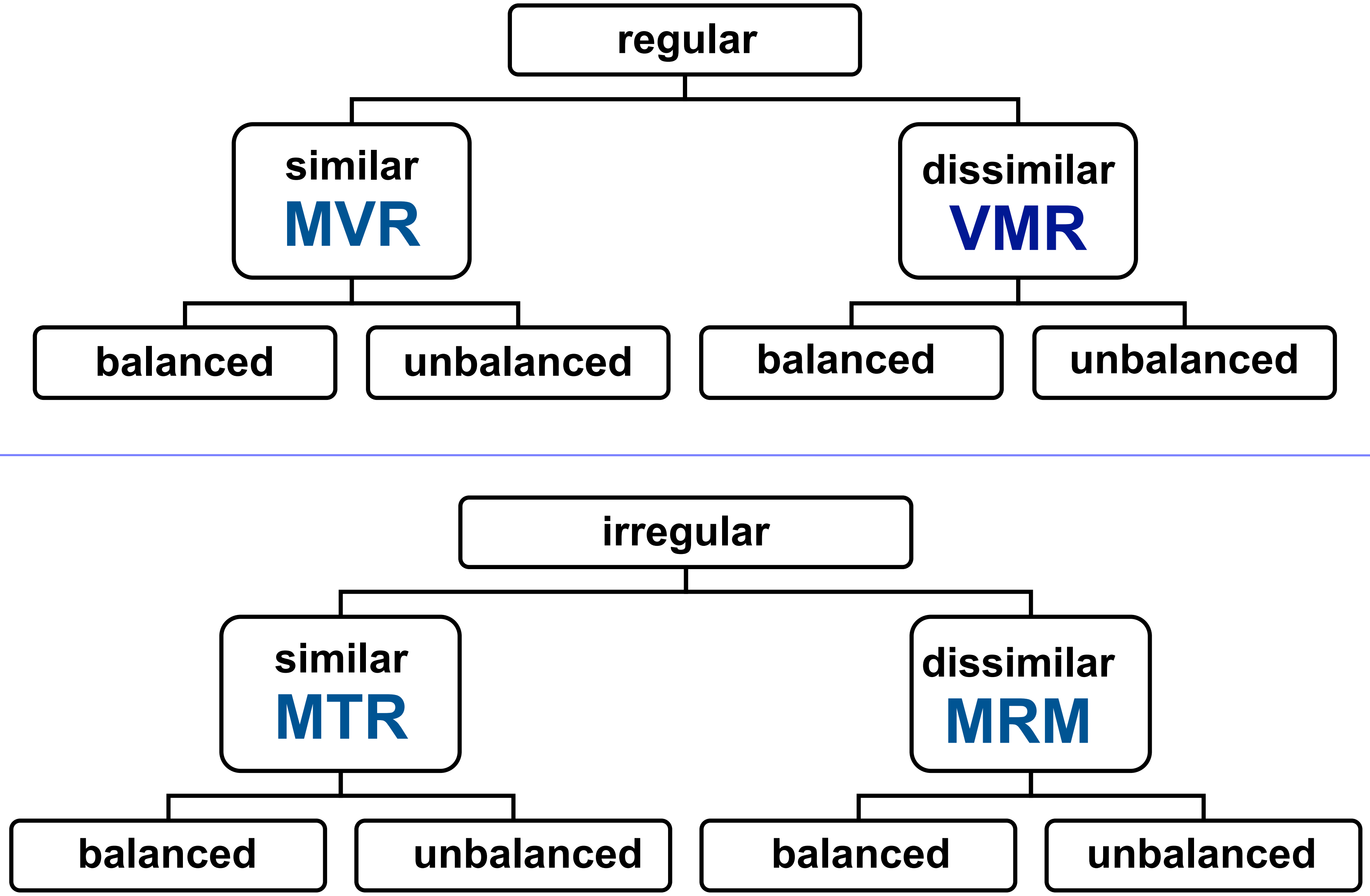
knowledge representation

learning phase

MXR

MXRVXT
MXTRRR
VTMXVR

classification phase



knowledge representation

- we have at least four theories of representations acquired in AGL
- we do not not whether a process is better described by dual model or unified model
- we could control the similarity effects, but one may always say that it may influence the results (e.g. we could also accidentally increase rule difficulty)
- an alternative measure that will allow us to determine which elements are most important for successive classification may be very useful

connectionist modeling of AGL

- auto-associative network is capable to simulate artificial grammar learning (Dienes, 1992)
- but
- opposite conclusions in Timmermans i Cleeremans (2001)

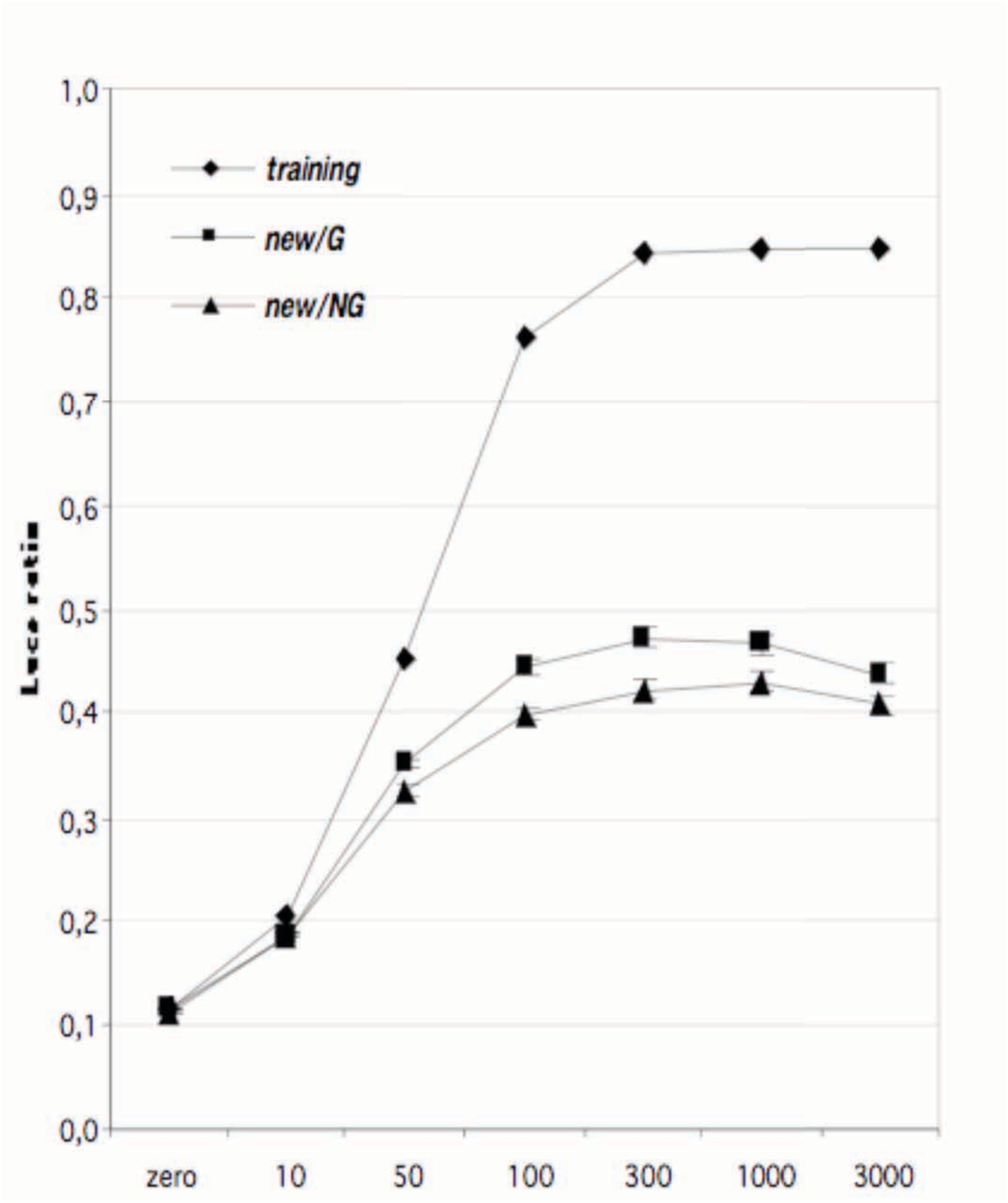
- since then
- most of models assumed that AGL requires **some knowledge about the context**
- thus **memory of the previously acquired informations should be implemented** in the architecture

connectionist modeling of AGL

- Tunney and Shanks (2003)
  - artificial grammar learning simulated by simple recurrent network
  - Higham et al (2000) version of the procedure was used to proof that both implicit and explicit learning effects could be simulated with SRN

- Timmermans and Cleeremans (2000; 2001)
  - artificial grammar learning simulated by simple recurrent network
  - authors criticized ‘abstractionists’ view - SRN bases its performance on similarity
  - Shanks et al (1997) biconditional grammars were used to rule out the interpretation claimed that SRN is able to reproduce sequential regularities only

- Onnis, Destrebecqz, Christiansen, Chater i Cleeremans (unpublished)
  - artificial grammar learning simulated by SRN, AARN, Jordan’s network, Buffer’s network & AAN
  - authors concluded that SRN works better than the other architectures

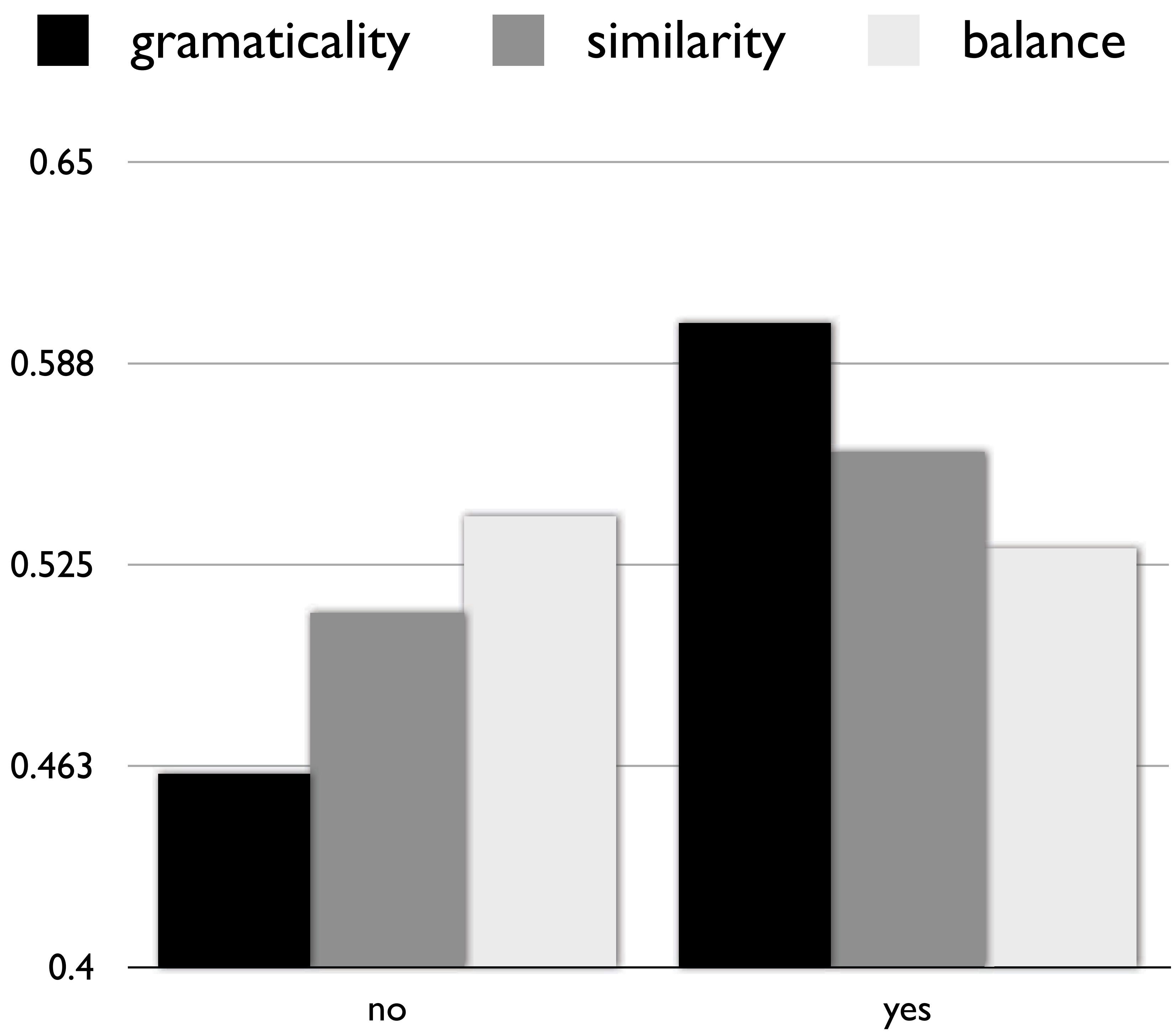


Timmermans and Cleeremans (2000) results

some surprising details

- large differences in Luce ration values for learning and testing material
- the effect was always analyzed for G/NG differences only - **similarity differences were never controlled!**

behavioral data



G vs. NG:
$F[1,71] = 111.01, p < .001, \eta^2 = .61$

S vs. NS:
$F[1,71] = 10.97, p < .01, \eta^2 = .13$

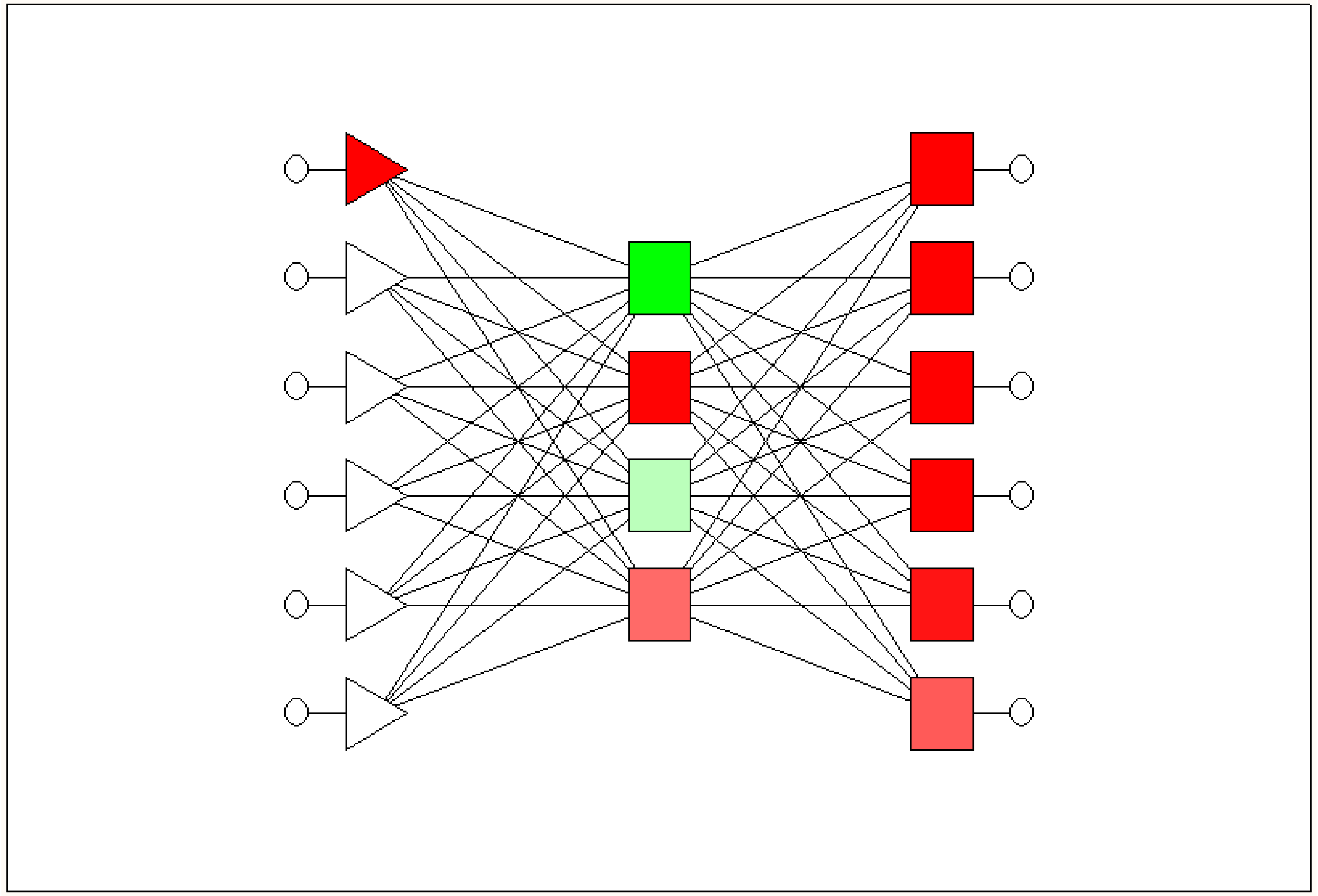
B vs. UB:
$F[1,71] = .50$, ns

our model

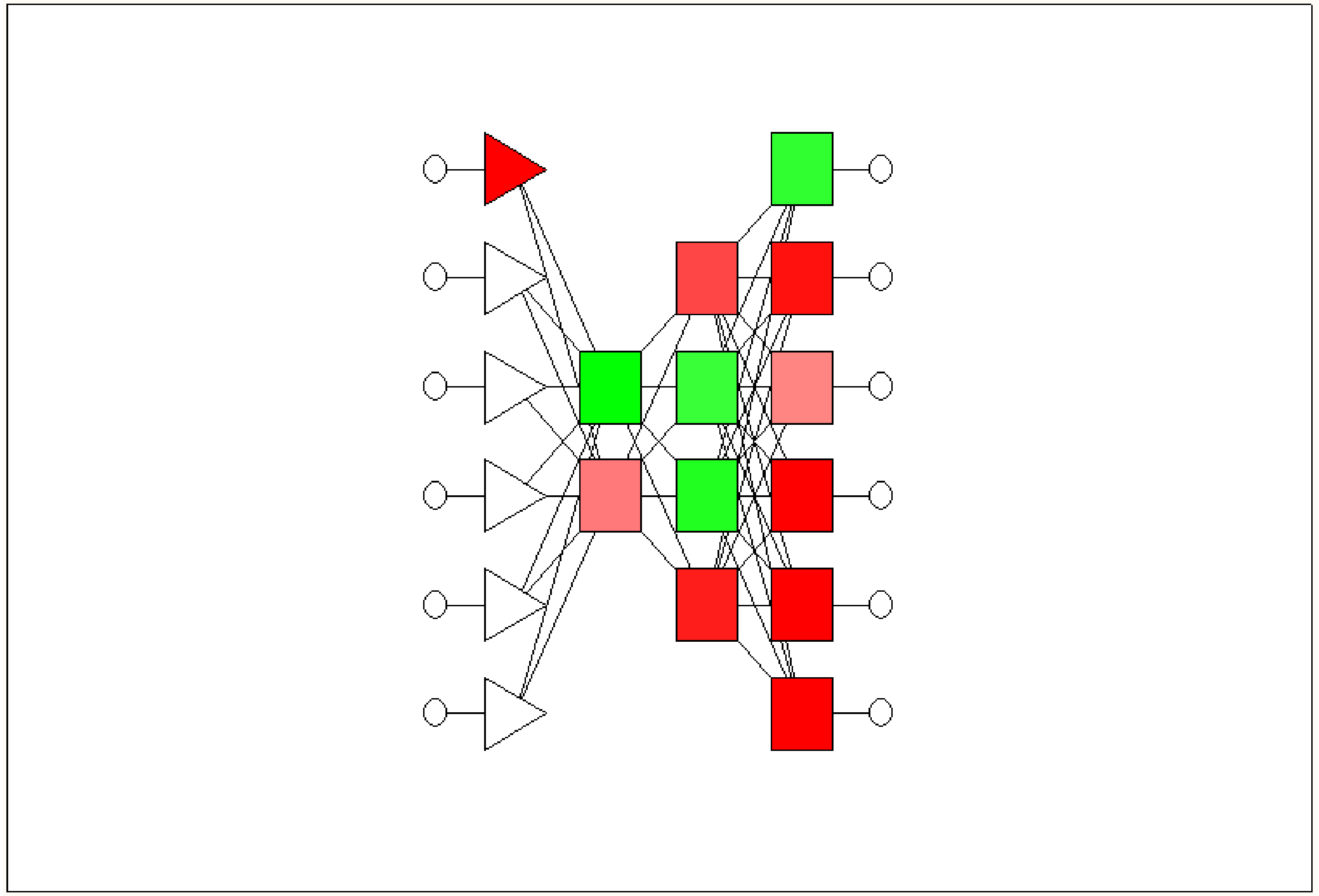
- is it possible to simulate AGL results with simpler architecture than SRN?
- how could we determine network parameters (e.g. number of hidden units)?
- is it possible to control the similarity effects (whether our model is not driven by similarity to grater extent than humans)?

our model
multilayer perceptron network

mlp 6-4-6

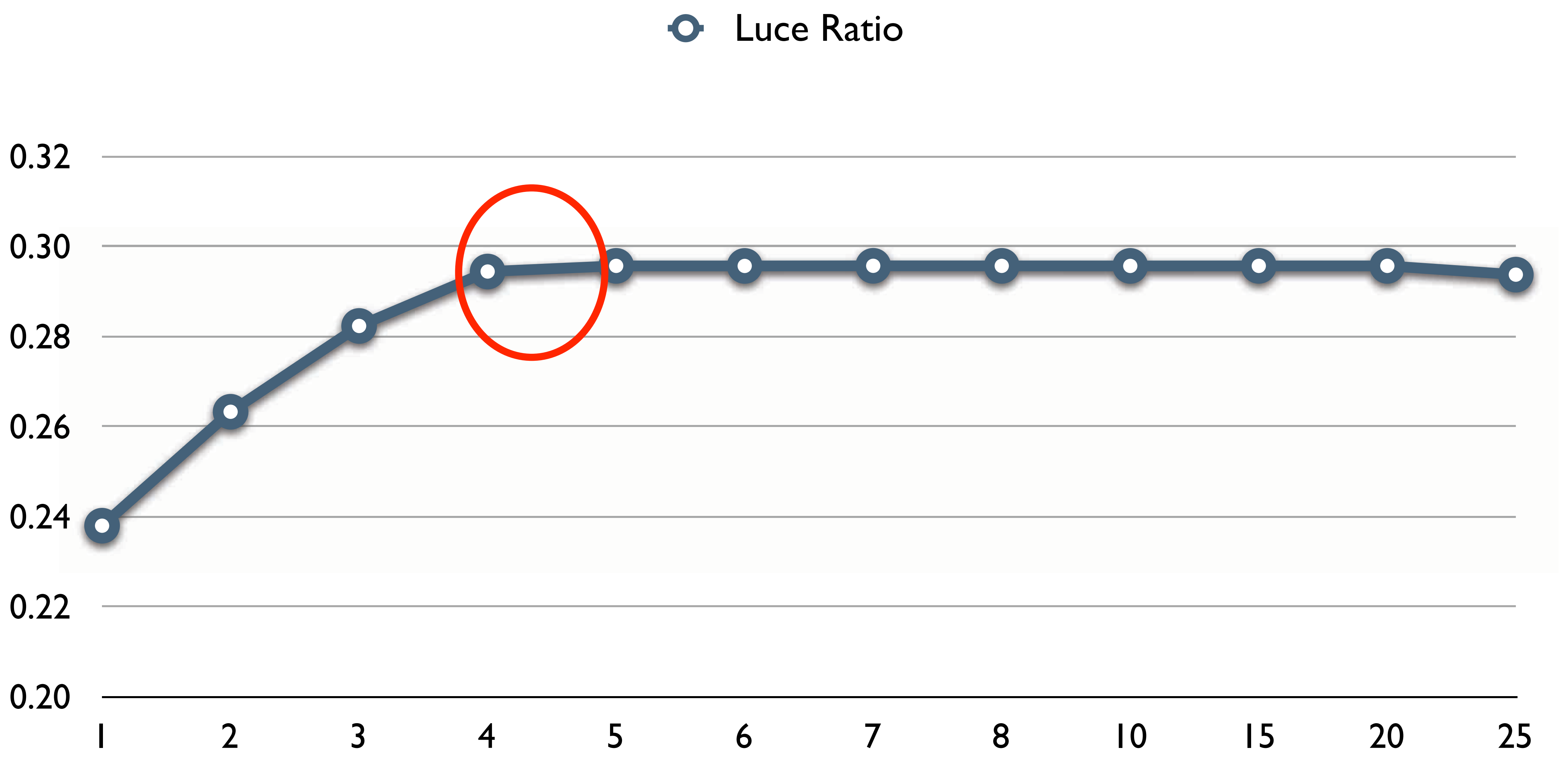


mlp 6-2-4-6



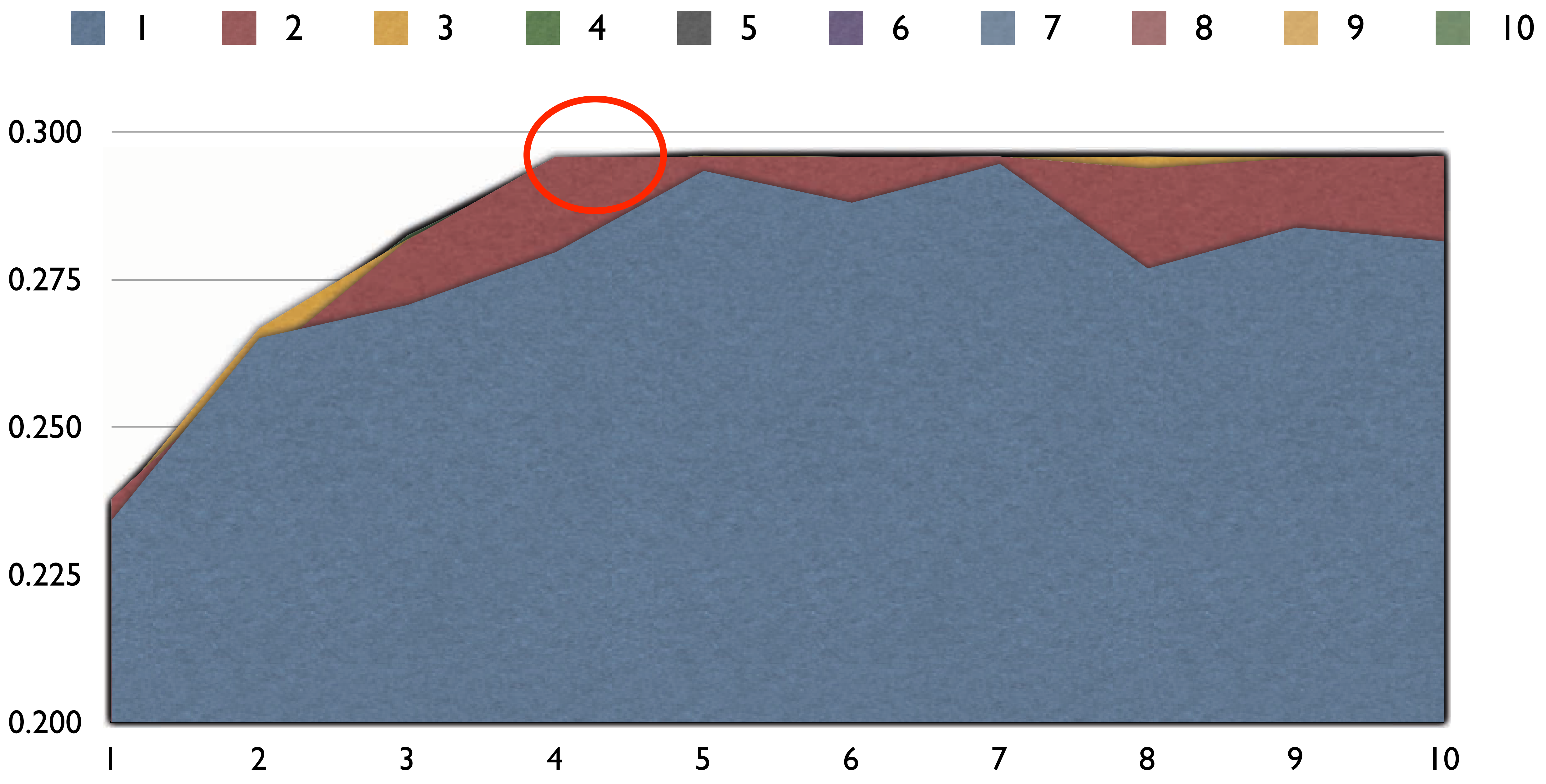
our model

how the architecture was chosen?

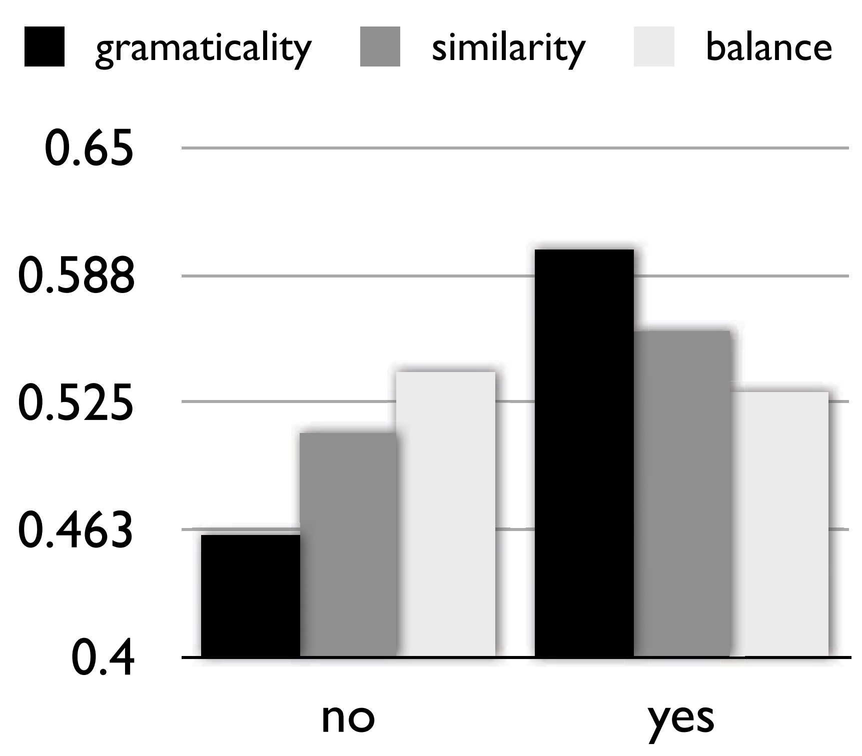


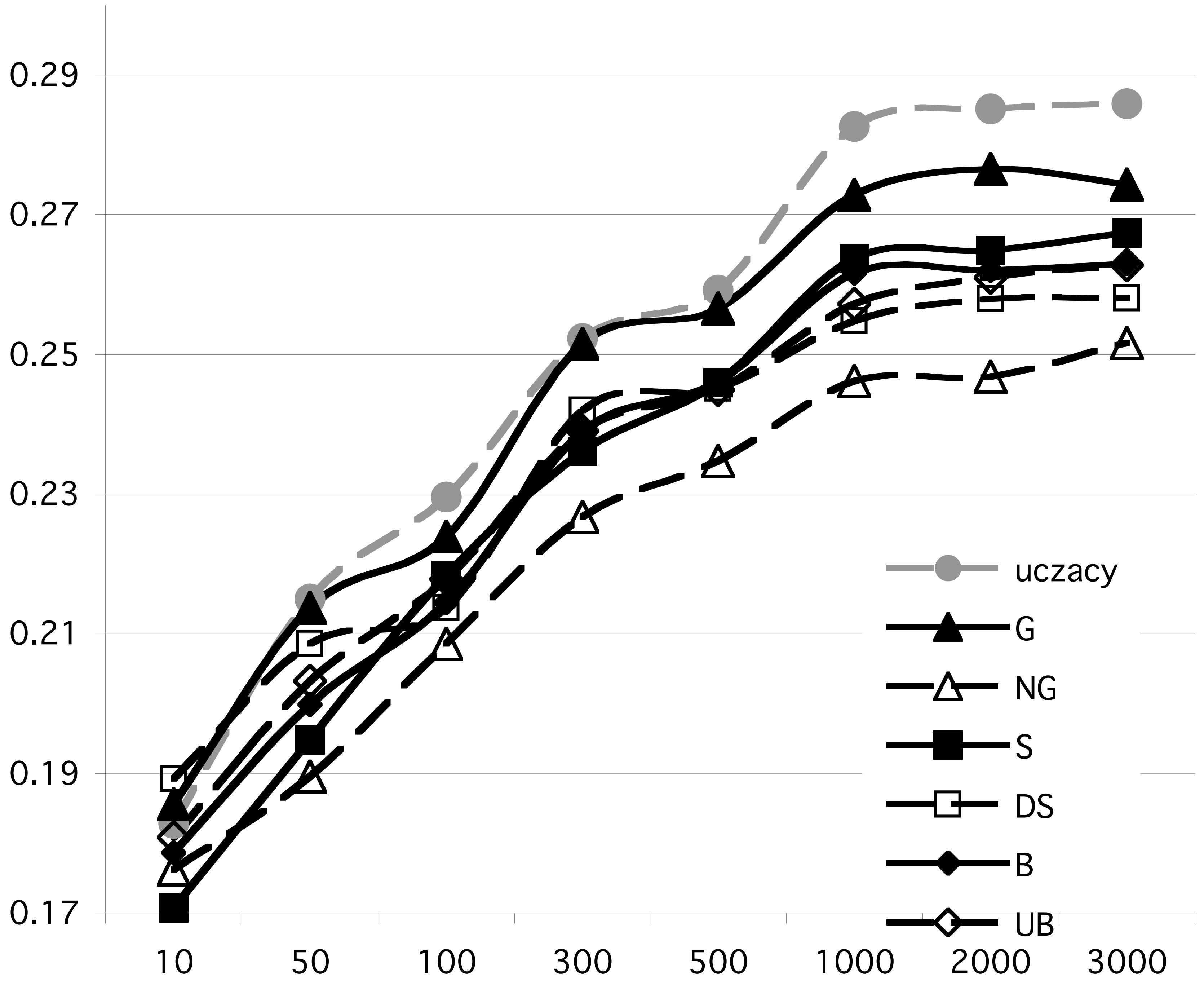
our model

how the architecture was chosen?



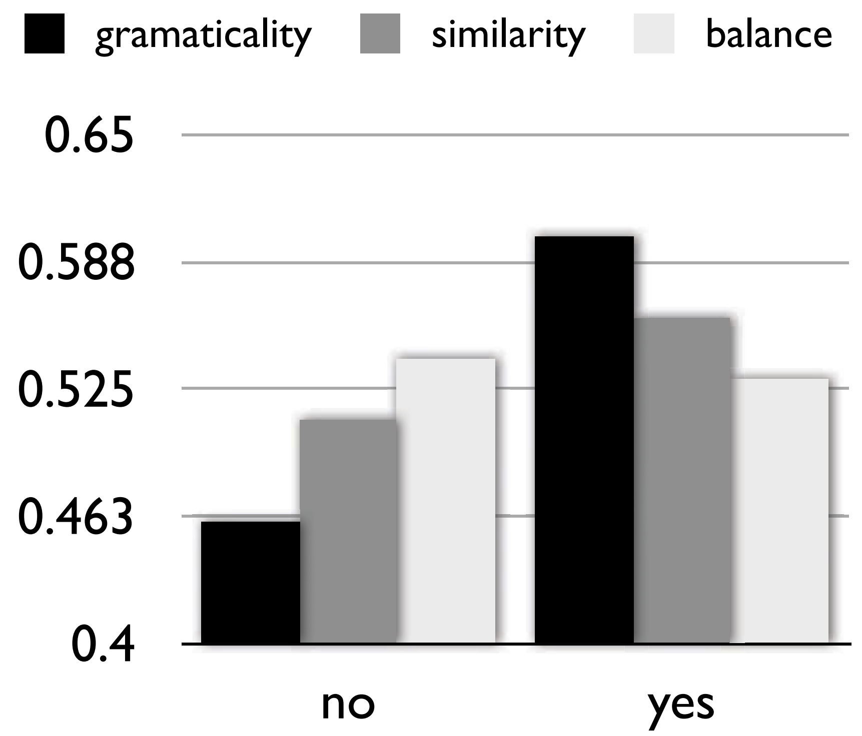
results for mlp 6-4-6

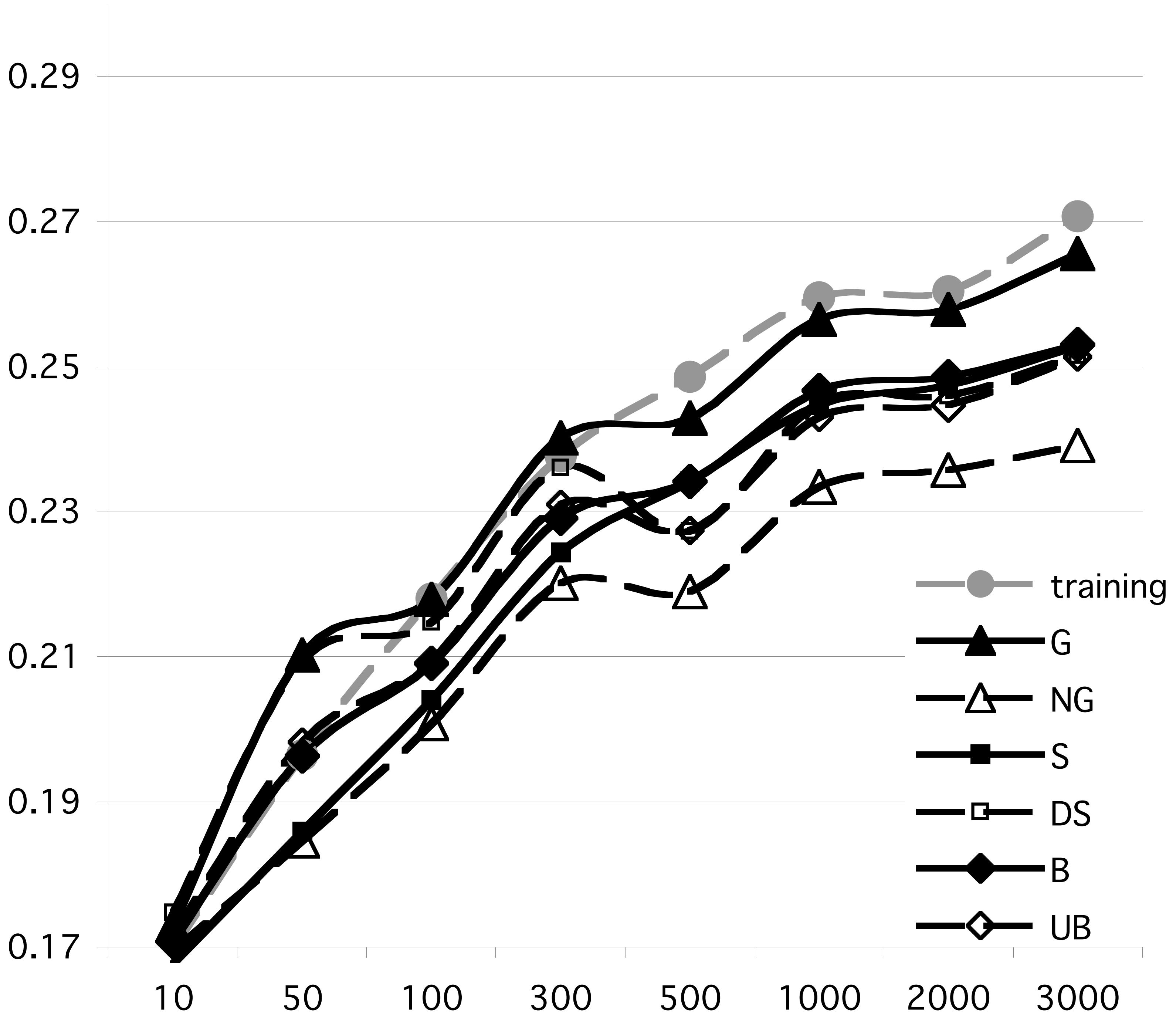




$R^2 = 0.98$

results for mlp 6-2-4-6



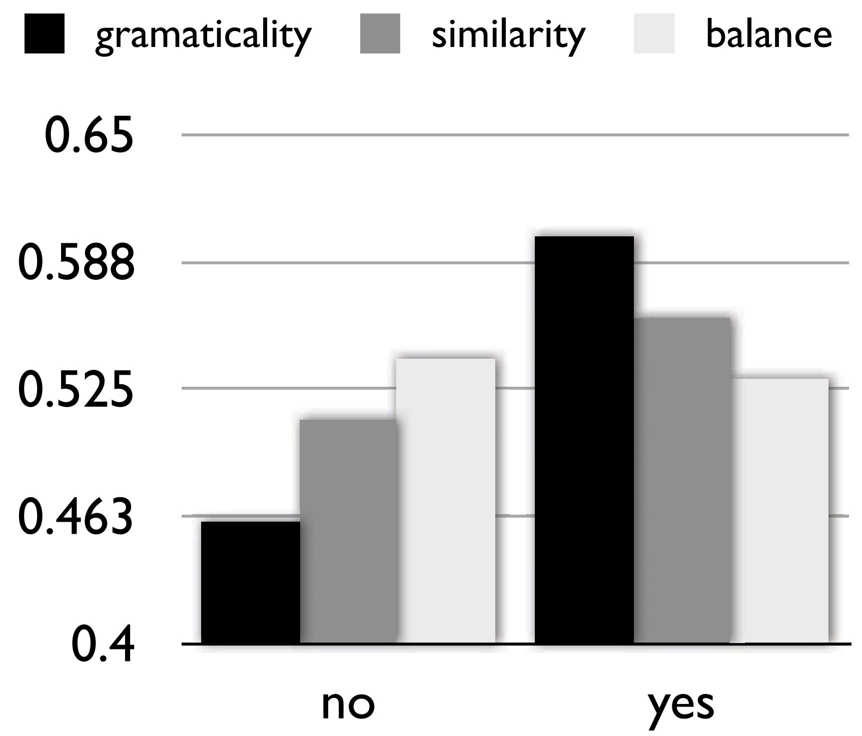


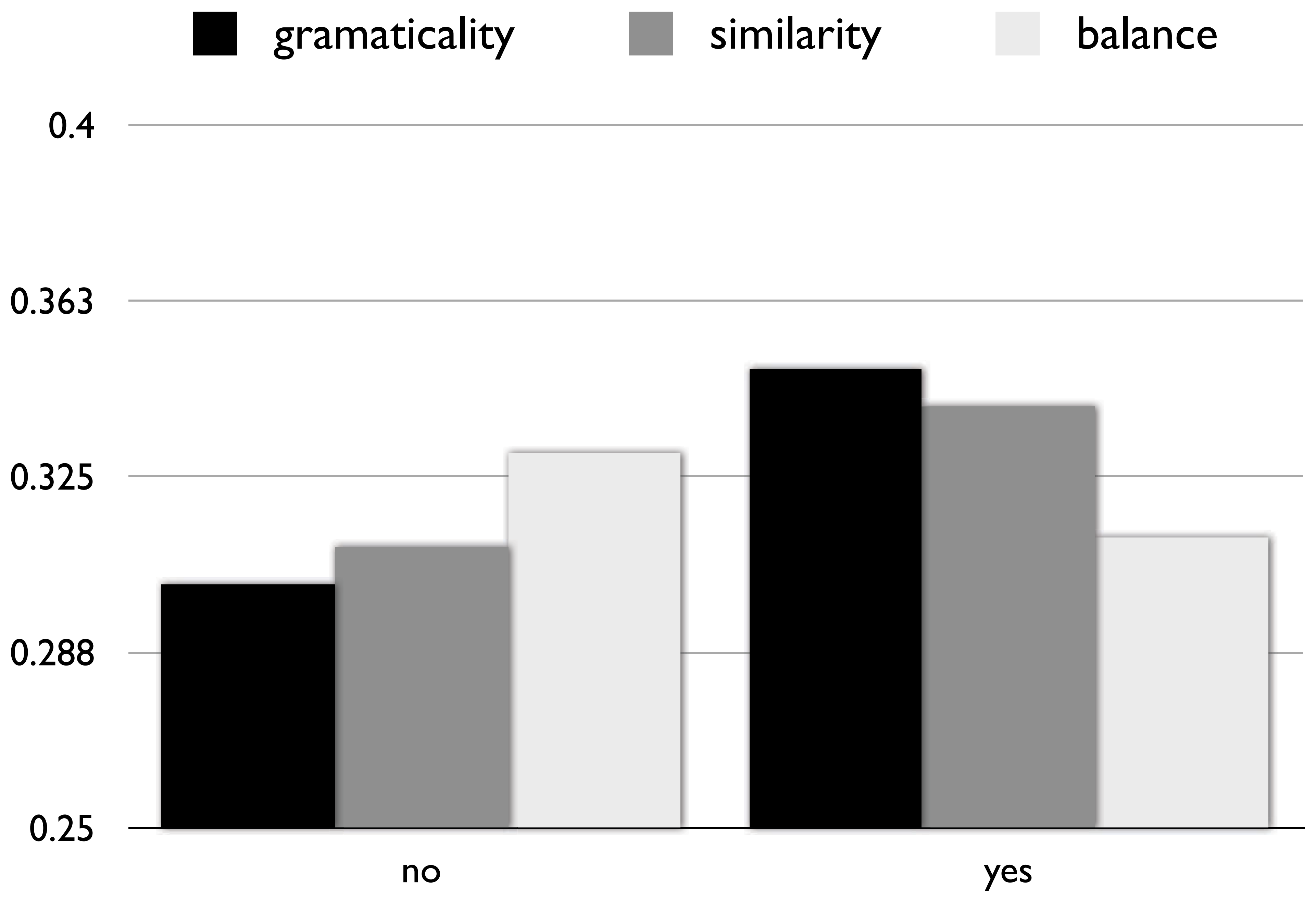
$R^2 = 0.89$

conclusions

- is it possible to simulate AGL results with simpler architecture than SRN?
  - yes, MLP seems to work quite well
- how could we determine network parameters (e.g. number of hidden units)?
  - by testing performance of a set of networks and choosing the simplest one that seems to work
- is it possible to control the similarity effects (whether our model is not driven by similarity to grater extent than humans)?
  - yes, and the results seems to be fitted better with MLP then other architectures.

SRN 6-4-4-6



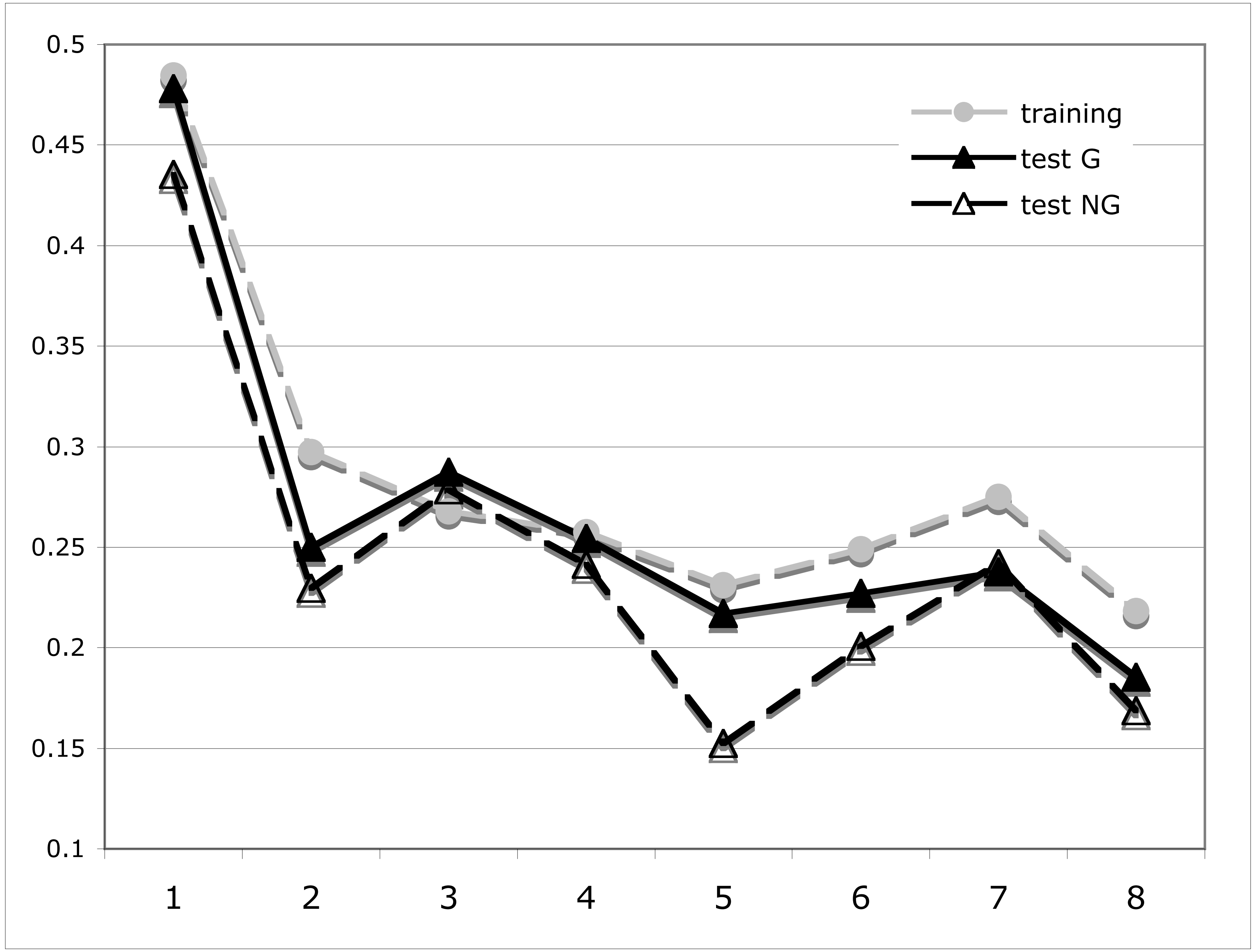


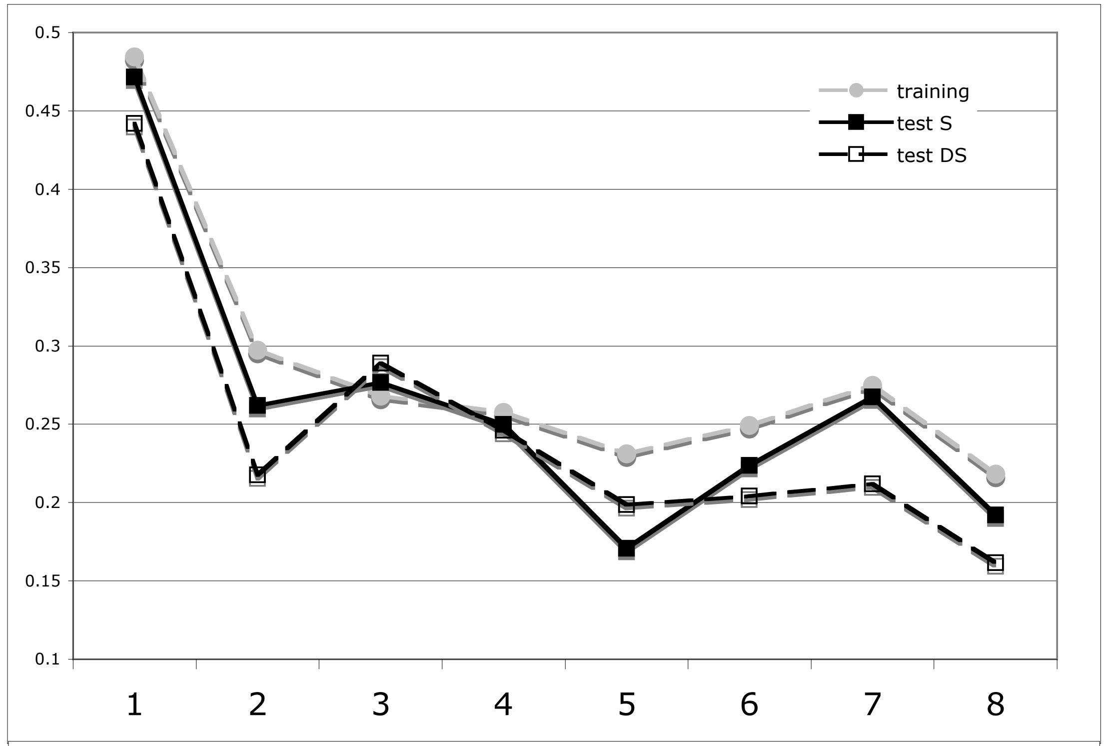
$R^2 = 0.79$

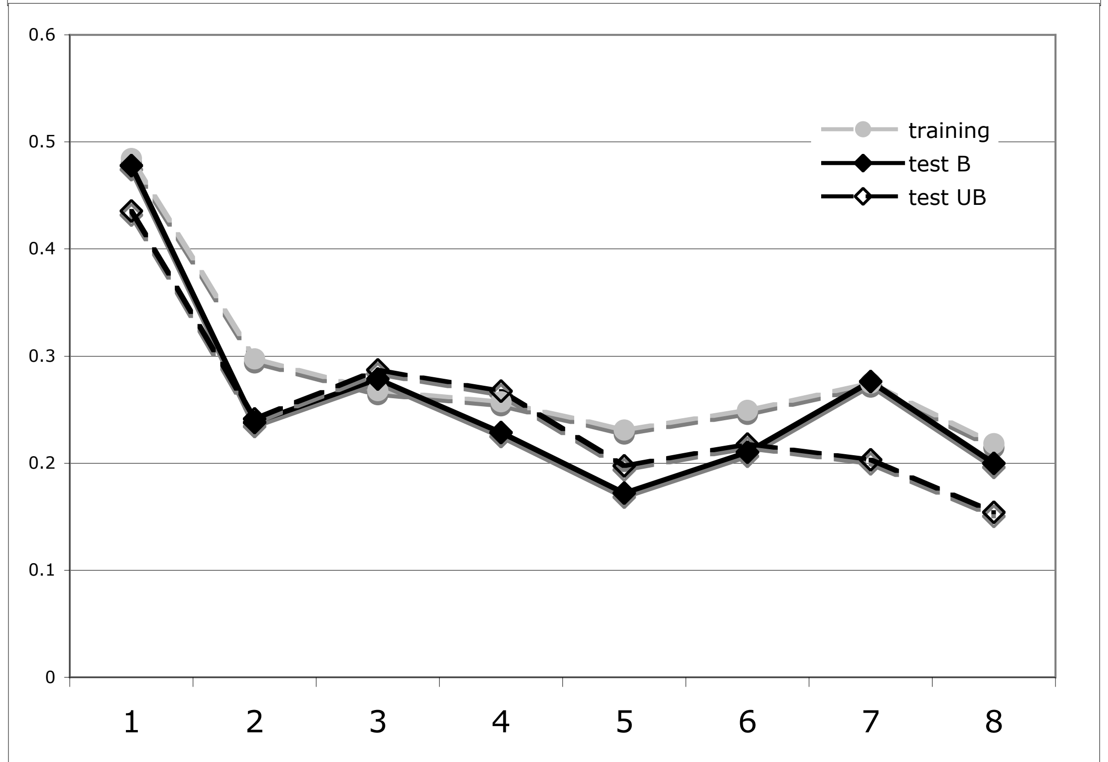
conclusions

- It seems that SRN context layer is not needed for AGL simulation
  - information about previous steps of learning could be stored in the connection weights
  - as only SRN should be able to simulate finite state grammars (Omlin & Gilles, 2000) it seems that the classification follows some other rules

additional analyses

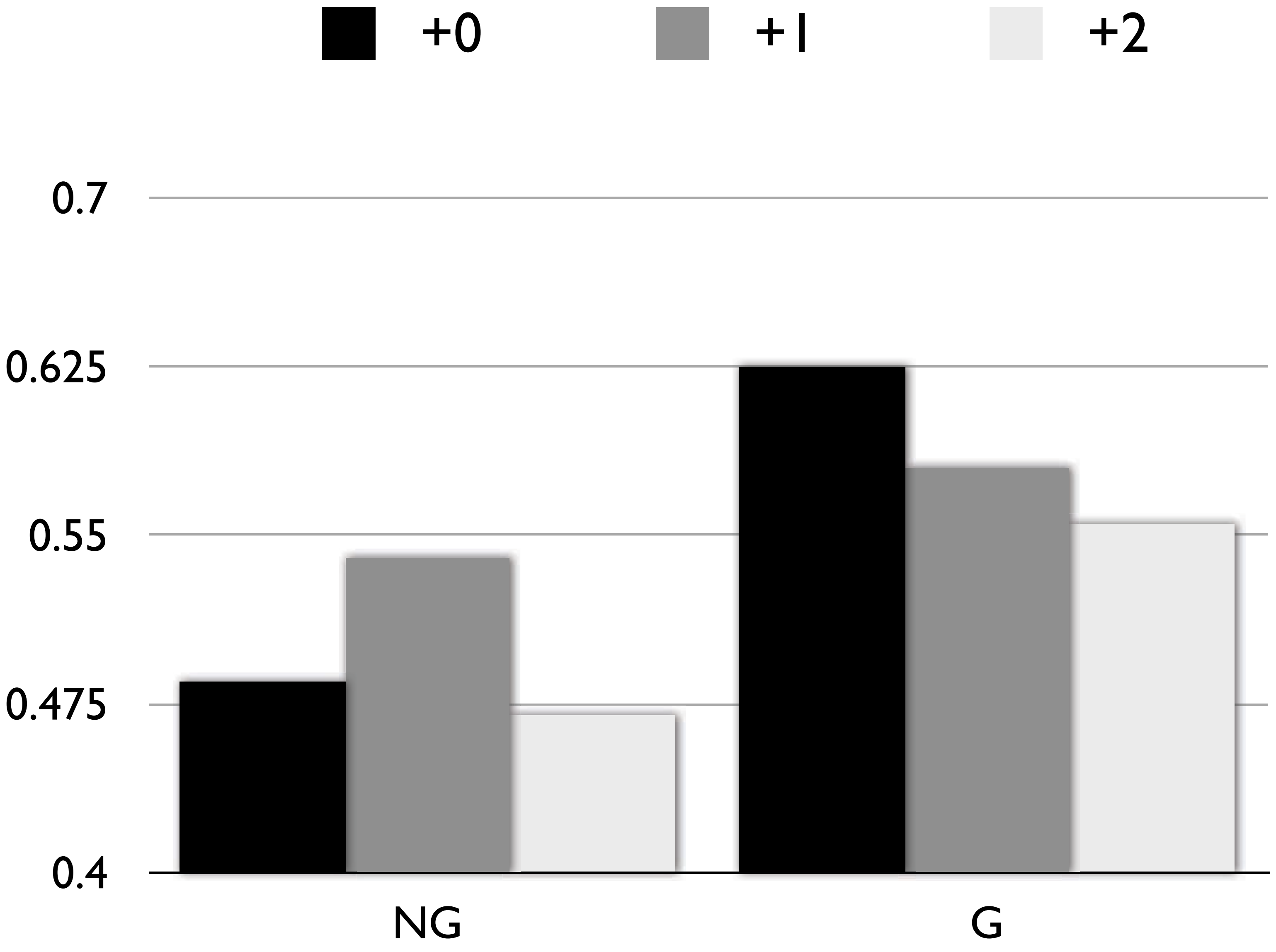




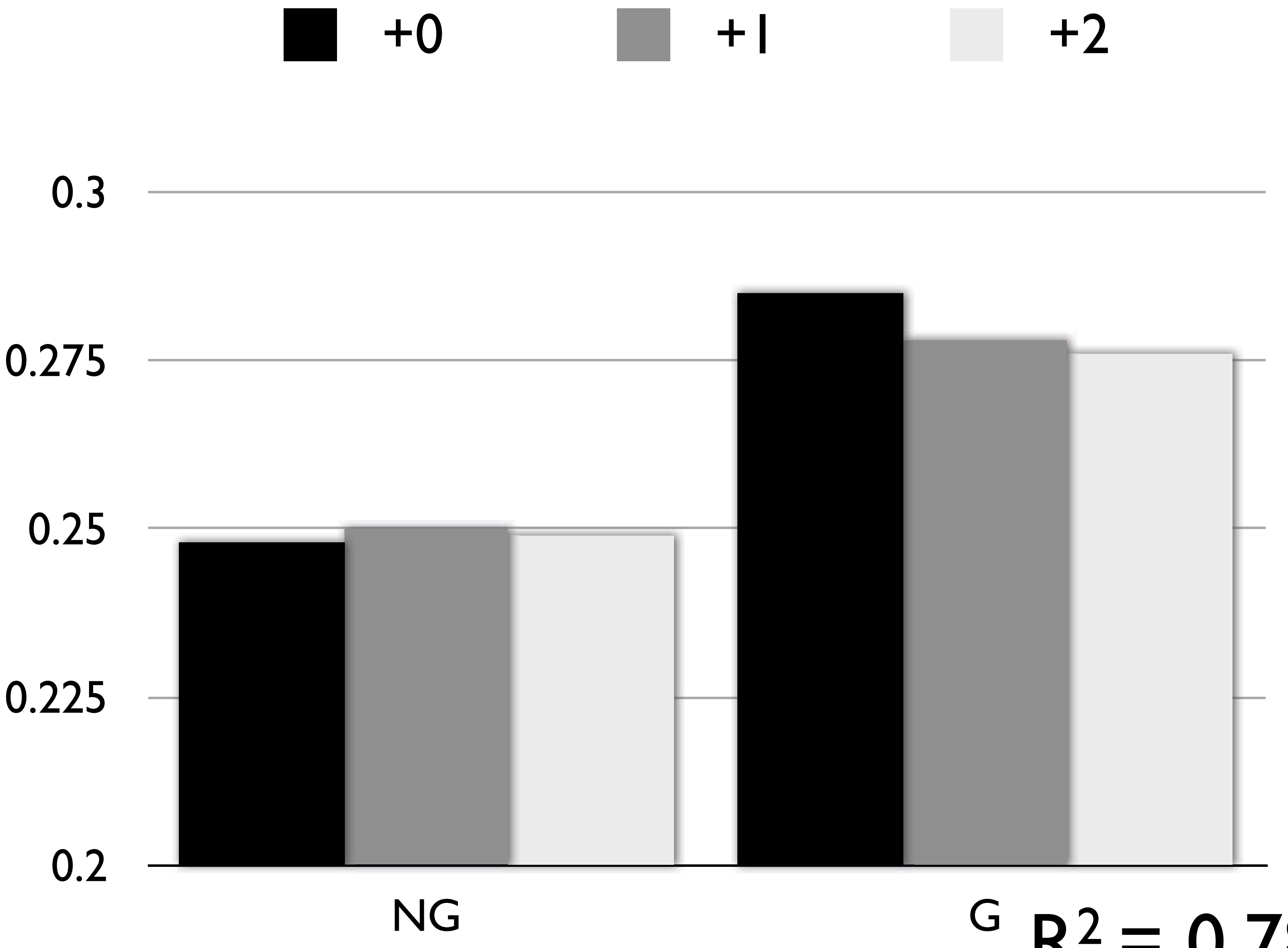


verification of the model

behavioral data



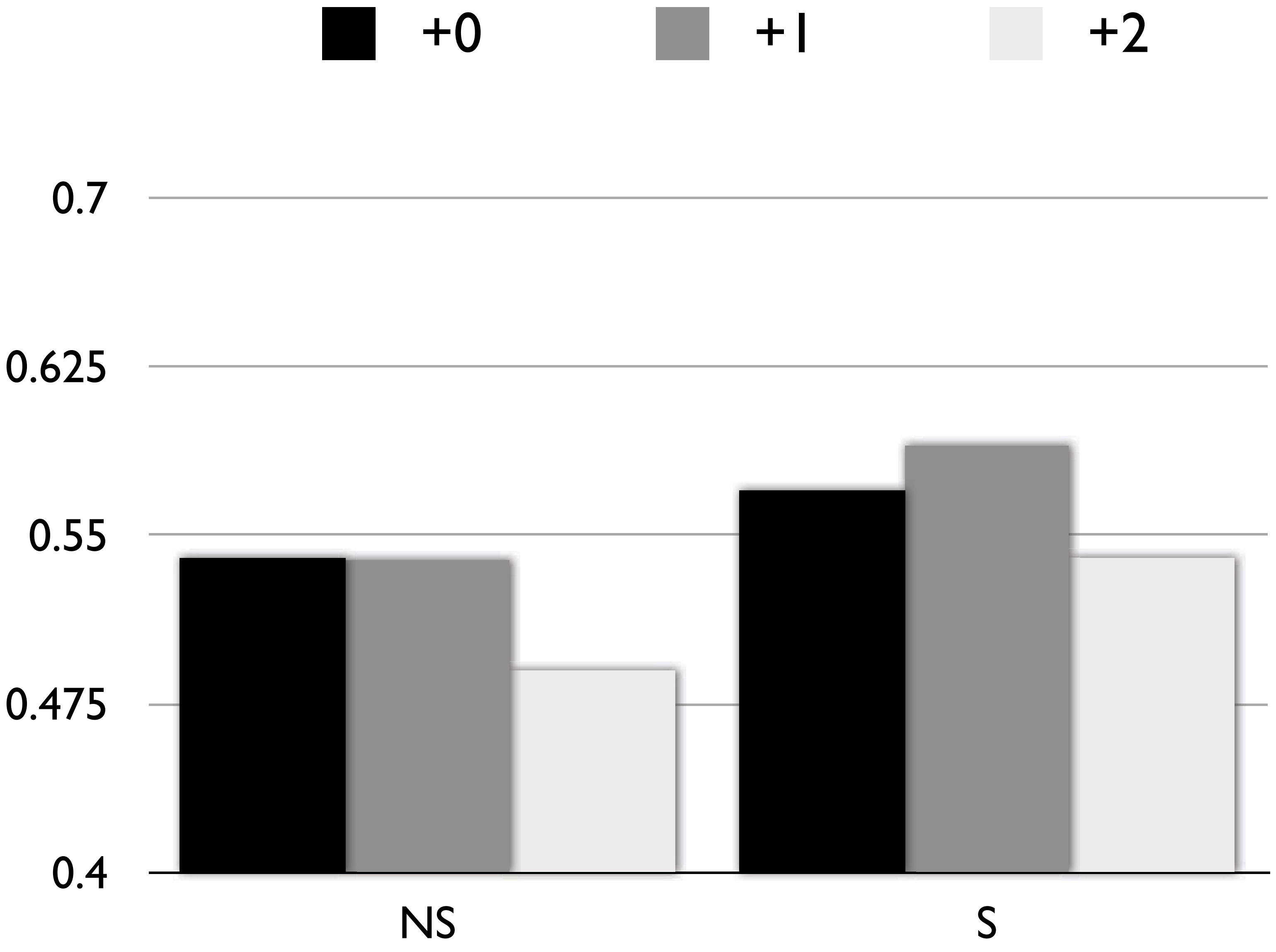
model



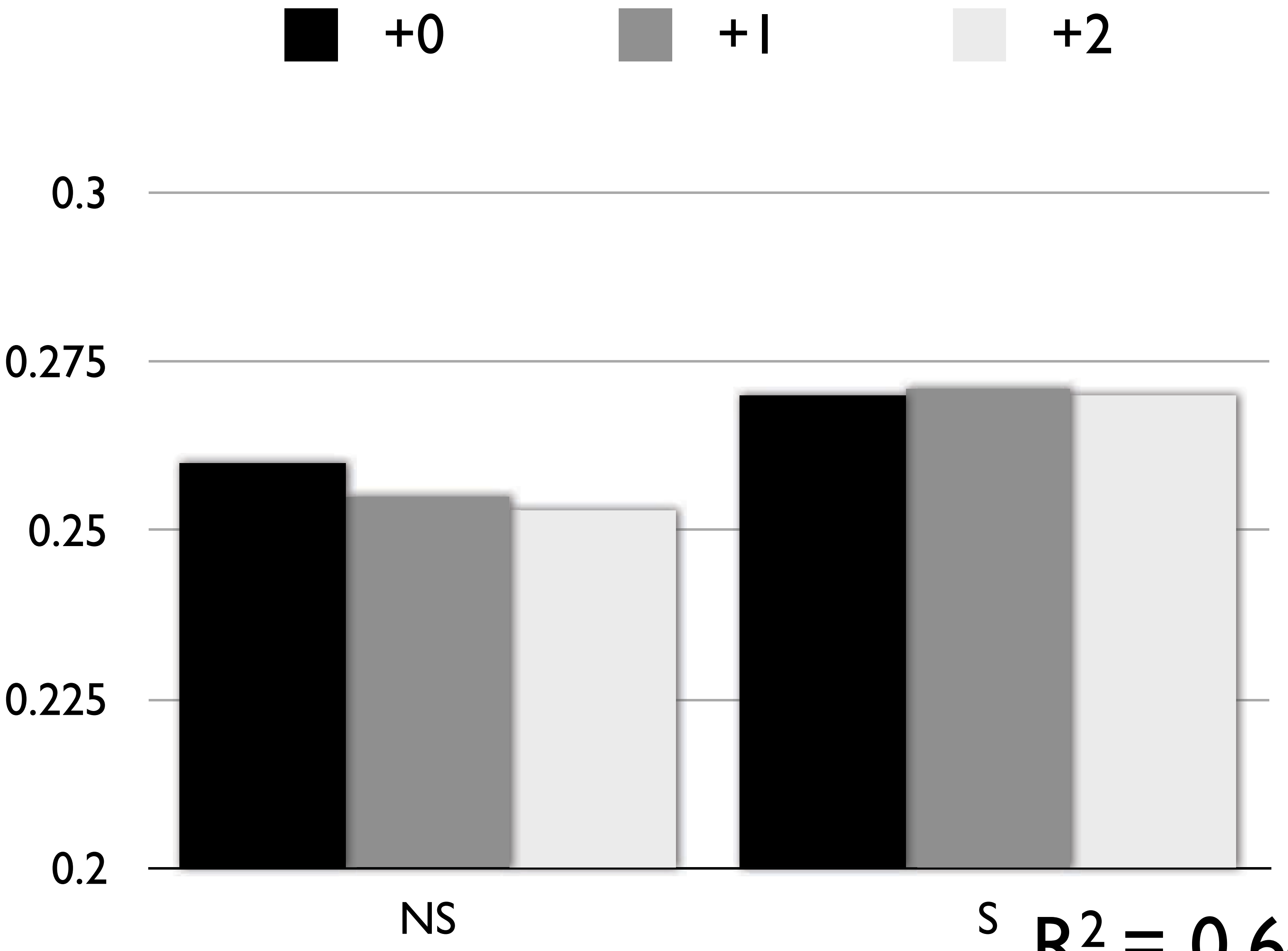
$R^2 = 0.78$

verification of the model

behavioral data



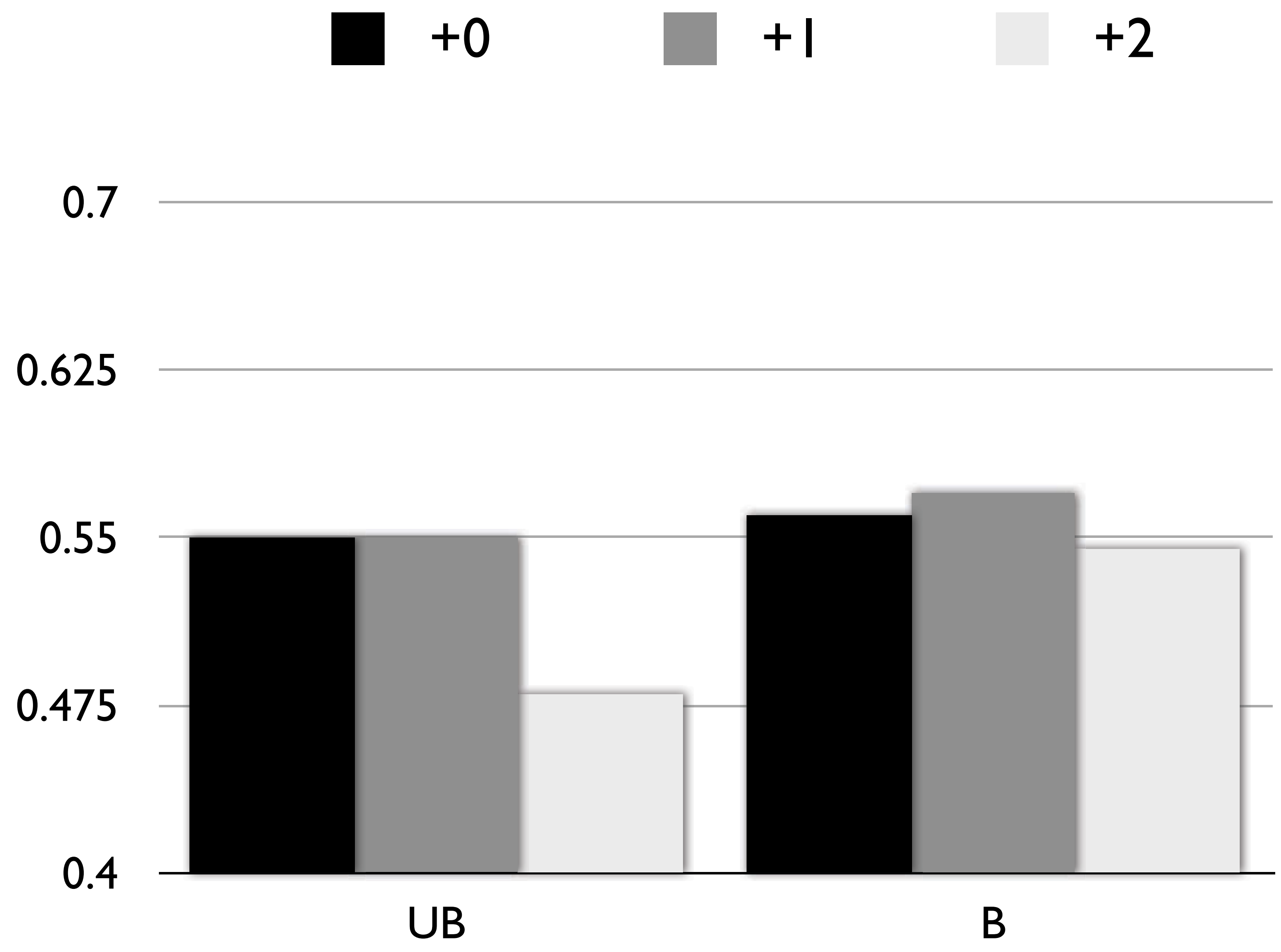
model



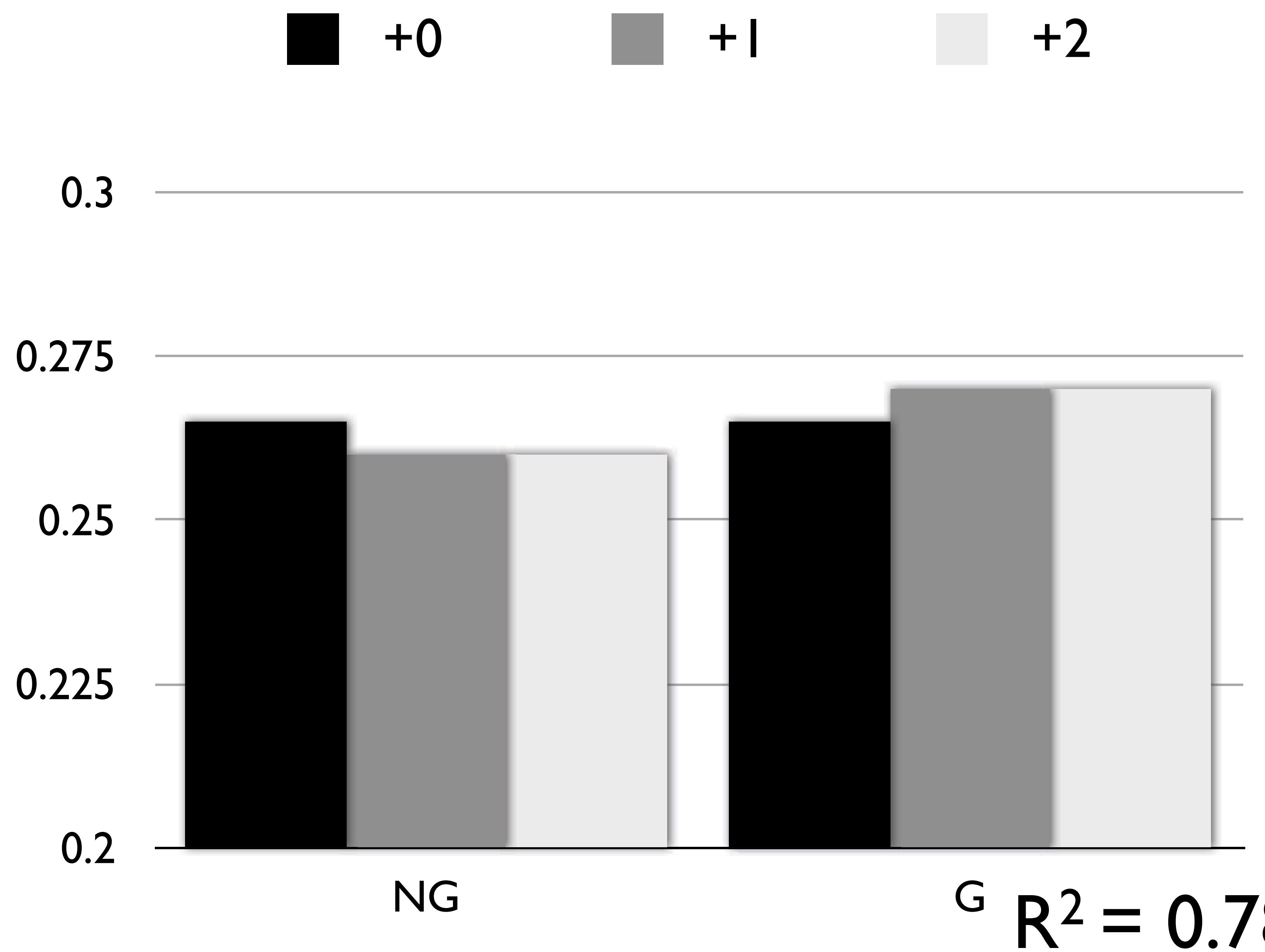
$R^2 = 0.64$

verification of the model

behavioral data



model



$R^2 = 0.78$

Thank you for your attention!