MODELING UTTERANCE-DRIVEN VISUAL ATTENTION DURING SITUATED COMPREHENSION

Ján Švantner, Igor Farkaš, Matthew Crocker†*

Abstract: Evidence from behavioral studies demonstrates that spoken language guides attention in a related visual scene and that attended scene information can influence the comprehension process. Here we model sentence comprehension within visual contexts. A recurrent neural network is trained to associate the linguistic input with the visual scene and to produce the interpretation of the described event which is part of the visual scene. A feedback mechanism is investigated, which enables explicit utterance-mediated attention shifts to the relevant part of the scene. We compare four models – a simple recurrent network (SRN) and three models with specific types of additional feedback – in order to explore the role of the attention mechanism in the comprehension process. The results show that all networks learn not only successfully to produce the interpretation at the sentence end, but also demonstrate predictive behavior reflected by the ability to anticipate upcoming constituents. The SRN performs expectedly very well, but demonstrates that adding an explicit attentional mechanism does not lead to loss of performance, and even results in a slight improvement in one of the models.

Key words: *Simple recurrent network, sentence comprehension, attention mechanism, visual scene*

Received: February 11, 2011
Revised and accepted: March 29, 2012

1. Introduction

Recurrent neural networks have proven to be a successful modeling tool for natural language processing. Elman [1] introduced a simple recurrent network (SRN) in the next word prediction task that has become a frequent training paradigm in processing the sequential data with recurrent networks. Since its invention SRNs

*Ján Švantner, Igor Farkaš
Department of Applied Informatics, Comenius University, Mlynská dolina, 824 48 Bratislava, Slovakia, {svantner,farkas}@fmph.uniba.sk
†Matthew W. Crocker
Department of Computational Linguistics and Phonetics, Saarland University, 66123 Saarbrücken, Germany, crocker@coli.uni-sb.de

have often been used for processing language (e.g. [2, 3]). Elman demonstrated that SRNs are able to learn an underlying grammar, when trained on (simplified) English sentences, creating meaningful context-dependent representations of words at the hidden layer. These results were repeated by Tong et al. [4] who compared SRNs with a recent popular model, echo-state networks (ESN), introduced by Jaeger [5]. It has been shown that ESN performance is similar to common statistical methods (variable-length Markov models) while a well-trained SRN can demonstrate superior prediction abilities [6].[1] What all these modeling approaches share is their focus on natural language processing as an independent domain.

However, human language typically does not occur in isolation, and the visual context provides a very frequent setting in which language is used. During the last decade, research into human language comprehension using the visual world paradigm (see the recent review in [7]) has revealed that spoken language can guide attention in a related visual scene and that scene information can immediately influence comprehension processes [8]. Findings have revealed the rapid and incremental influence of visual referential context [9, 8] and depicted events [10] on ambiguity resolution in online situated utterance processing. Further research has demonstrated that listeners even anticipate likely upcoming role fillers in the scene based on their linguistic and general knowledge (e.g. [11]). Knoeferle and Crocker [12] identified several cognitive characteristics based on the above mentioned findings, claiming that situated language comprehension is incremental, anticipatory, integrative, adaptive, and coordinated, which led to the proposal of the coordinated interplay account (Cia).

The recent CiaNet model [13], which serves as a motivation for the present work, instantiates the Cia's proposal and accounts for a range of observed empirical findings. CiaNet is a recurrent sigma-pi neural network that models the rapid use of scene information, exploiting an utterance-mediated attentional mechanism. The model was shown to achieve high levels of performance (both with and without scene contexts), while also exhibiting hallmark behaviors of situated comprehension, such as incremental processing, anticipation of appropriate role fillers, as well as the immediate use and priority of depicted event information through the coordinated use of utterance-mediated attention to the scene.

There exist several other models that link language with the visual world [14, 15, 16], including those mentioned in the recent review [7]. These models emphasize situated lexical learning and processing, however, and there remain very few attempts to model the compositional and incremental nature of visually situated sentence comprehension.

Inspired by the above mentioned CiaNet, we investigate more general network architectures that also learn to adapt the attention mechanism which helps the network focus on (and predict upcoming) relevant constituents and in principle allows generalization to more complex scenes (the attention mechanism in CiaNet is restricted to favor one of the two concurrent events). Specifically, our models differ from the former in that the inhibition operates at both the object and event levels (and not only at event level as in CiaNet). We have explored

[1] Although this evidence comes from the linguistic domain, the results may hold in other domains as well, where the sequential data is discrete (symbolic) in nature. All that matters for prediction is the structural dependencies among symbols, regardless of the domain.

multiple implementations of the attentional mechanism describing their strengths and weaknesses in situated comprehension tasks. An earlier, less detailed version of this work appeared in [17].

2. The Model

The network architecture, shown in Fig. 1, is based on a simple recurrent network (SRN, [1]). The network reconciles an incrementally presented utterance with a representation of the current visual context to incrementally and predictively recover the target event representation. The scene representations stand for encodings of the objects and events in the visual world, the linguistic representations are presented as short sentences. In each trial, the scene representation is presented at the input and the associated sentence is presented at the linguistic input, word by word. The network task is to produce a (partial) scene representation at the output. This process is mediated by the hidden layer that combines scene-related representations with symbolic language. The target is fixed, available at the output during processing of the entire sentence. The explicit feedback (from the output) is added to the network using a sigma-pi mechanism (explained below) to model the cognitively-motivated process of focusing attention on relevant constituents shown in the visual scene and mentioned in the associated sentence.

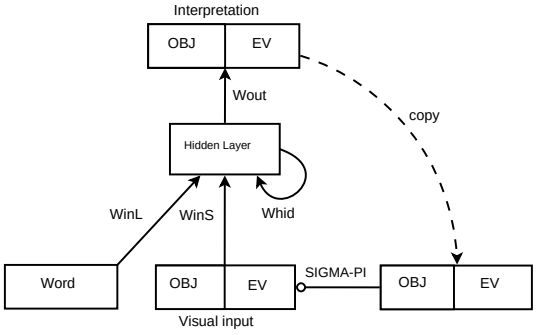

Fig. 1 *Architecture of A-SRN with a language-mediated, top-down attention mechanism (for description see the text).*

2.1 Scene representations

The scene representations are postulated to exist at two levels – the object level (OBJ) and the event level (EV). The objects are typically the constituents of the events, so there exists natural relationship between the two levels. Objects refer to physical agents/patients that can be focused on, whereas the event level refers to specific actions in the concrete context (with given semantic roles, i.e. the known agent and patient). The combination of both levels of representation is hence assumed to constitute a *semantic representation* of an event. The scene is assumed

to consist of multiple events that may, or may not, share a constituent, plus a random number of distractors (see Fig. 3). For instance, an agent of one event is a patient of the other event, or events share common patients.

2.1.1 Objects

Objects include human agents (e.g. TODDLER, WOMAN), animate agents (e.g. DOG, DONKEY) and one artificial agent (ROBOT) that can be involved in various meaningful activities, with or without a patient. Agents can operate on machines (FORKLIFT, BULLDOZER)[2], on objects (e.g. BARREL, HOUSE) or food items (e.g. APPLE, JUICE). The actions include moving (e.g. WALKS, SITS), physical manipulation (e.g. LIFTS, HOLDS), socially oriented activities (e.g. GREETS, LOOKS-AT) and sustenance (EATS, DRINKS). Agents and patients are manually assigned binary features that encode various physical and functional properties and form 40-dimensional vectors $\mathbf{c}_\mathrm{A}$ and $\mathbf{c}_\mathrm{P}$, respectively. These features include: size (one of 7 categories with a localist encoding), animacy (yes, no), category (human, animal, artifact, food), means of mobility (2-legged, 4-legged, wheeled, winged), agency, instrument, and several others, permitting differentiation of the entities from one another by at least one feature.

Each object can take a role of a distractor, denoted as $\mathbf{c}_\mathrm{D}$, using the same representation vector. Analogically, actions are described by 8 binary semantic features, namely animacy (of agent), contact (with object), motion, transitivity, effortfulness, temporality, egocentricity and flow. In fact, we found it useful to duplicate these features, resulting in 16-dimensional vectors $\mathbf{c}_\mathrm{V}$, in order to increase the differentiation of compressed event representations, performed by EV module.

Before training the recurrent model(s), we used the standard self-organizing map (SOM; [18]) to learn the localized representations of objects. Hence, the SOM is constructed in advance using only agents $\mathbf{c}_\mathrm{A}$, patients $\mathbf{c}_\mathrm{P}$ and distractors $\mathbf{c}_\mathrm{D}$ as inputs, one at a time. Actions are excluded from SOM training, they are included only in event-level representation. The SOM is trained to provide a topographically organized map of objects according to their semantic features. Each object is represented in the SOM by three most active units, focused around the winner (best matching unit, bmu), all other units are set to zero. The activity of unit i is calculated as $y_i = \exp(-\|\mathbf{x} - \mathbf{w}_i\|)$, where $\mathbf{w}_i$ is the unit's i weight vector and $\mathbf{x} \in \{\mathbf{c}_\mathrm{A}, \mathbf{c}_\mathrm{P}, \mathbf{c}_\mathrm{D}\}$. The activity of the three most active units is rescaled such that $y_\mathrm{bmu} = 1$. The resulting map for different objects can be seen in Fig. 2. Since these object representations are mostly localist, they do not interfere with one another in the map. The SOM size was chosen to have 64 units to allow unambiguous learning of each object (by assigning it a separate winner).

The purpose of using three most active units (instead of just a winner) is to allow the activation overlap between similar objects with neighboring winners (this actually helped the model to generalize better). The scene representation at the object level contains the superimposed representations (in SOM) of all objects in the current scene (i.e. all being simultaneously present) plus a few distractors resulting in SOM activation vector

[2]In fact, machines can serve as agents of some actions, too.

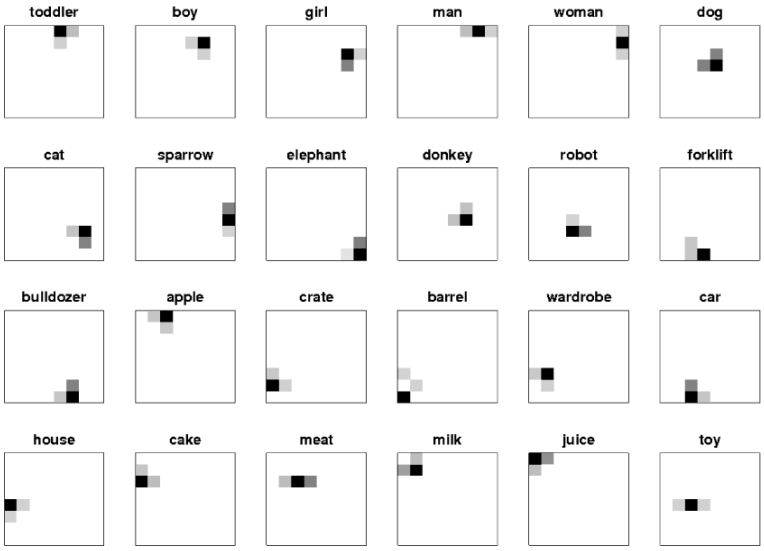

Fig. 2 *SOM representations of all objects used in the simulations. Topographic order according to semantic similarities is evident. Localist nature of representations allows their combinatorial use without causing interference.*

$$\mathbf{c}_{\text{in}}^{\text{all}} = \mathbf{c}_{\text{in}}^{(1)} \oplus \ldots \oplus \mathbf{c}_{\text{in}}^{(m)} \oplus \mathbf{c}_{\text{D}}^{(1)} \ldots \oplus \mathbf{c}_{\text{D}}^{(n)},$$

where the index in $\in \{\text{A}, \text{P}\}$, and m and n denote the number of different objects and distractors in the scene, respectively.

2.1.2 Events

To obtain representations $\mathbf{e}_{\text{in}}$ of events, an auto-associative network (AAN), modeled by a two-layer perceptron (i.e. with one hidden layer) is pretrained off-line using the vectors $[\mathbf{c}_{\text{A}}\ \mathbf{c}_{\text{V}}\ \mathbf{c}_{\text{P}}]$ to form the compressed distributed representations at the hidden layer with 48 units. Patient $\mathbf{c}_{\text{P}}$ is optional, so its components are set to zero in case of patient's absence. The size of the input vector for training the AAN was 40+16+40=96 dimensions. The functionality of the trained AAN was checked via accuracy of compressed representations using the encoding and decoding of novel agent-action-patient triplets. The accuracy reached almost 100% for testing data.

Once the AAN is trained, the event-level representation corresponding to a scene is taken as a superposition of all events in the situation, resulting in the vector

$$\mathbf{e}_{\text{in}}^{\text{all}} = \mathbf{e}_{\text{in}}^{(1)} \oplus \ldots \oplus \mathbf{e}_{\text{in}}^{(k)}.$$

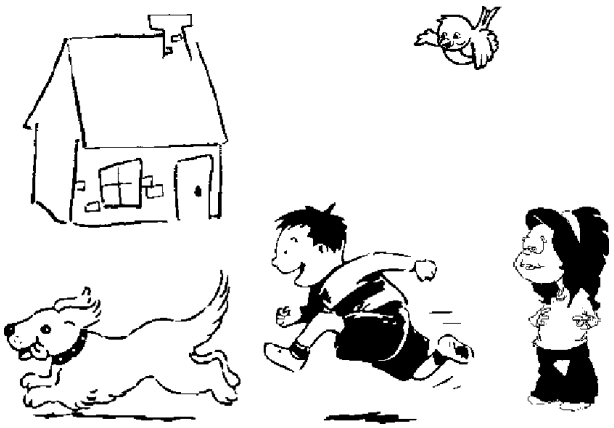

Fig. 3 *Example of a scene consisting of two events* (BOY CHASES DOG) *and* (GIRL LOOKS-AT BOY) *and two distractors* (HOUSE, SPARROW). *Both events share the constituent* (BOY).

The vector components are constrained in the interval $\langle 0, 1\rangle$.[3] Using the superposition is analogous to that used in CIANET – it encodes simultaneous information provided to the subject as the visual input. However, in CIANET the representational media for two events are separated whereas in our models the medium is shared. Unlike localist representations for objects, the superposition of distributed event representations leads to an overlap between the two codes, which expectedly makes the decompression task more difficult.[4]

2.2 Linguistic inputs

The lexicon consists of 40 words, with one-to-one mapping to the objects and actions. Words are treated as symbols and are assigned one-hot codes with 40-dimensions creating an input $\mathbf{l}_{\text{in}}$. The sentences have a SV(O) form, such as *toddler looks-at crate* or *woman walks*.

2.3 Models

All models use two output slots: $\mathbf{c}_{\text{out}}$ is the object-level output that tries to activate the target objects, taking part in the described event. $\mathbf{e}_{\text{out}}$ predicts the representation of the target event. Together, the network output (predicted scene interpretation) is given as $\mathbf{a}_{\text{out}} = [\mathbf{e}_{\text{out}}\ \mathbf{c}_{\text{out}}]$. The models have no linguistic output.

In total, we simulated four models. Beside the standard SRN, we have explored the behaviour of A-SRN shown in Fig. 1, to appreciate the role of the feedback mechanism in the sentence comprehension task. The third model, A-SRN$^+$, was

[3] Some components of the event vector could become larger than one after superposition (i.e. if both events had the same unit highly active), therefore all components were divided by the value of the most active component.

[4] We experimented with decreasing this overlap by manipulating the profile of the sigmoidal activation function (its gain and shift) of the hidden units of AAN in order to get sparser compressed codes, but this had no significant effect.

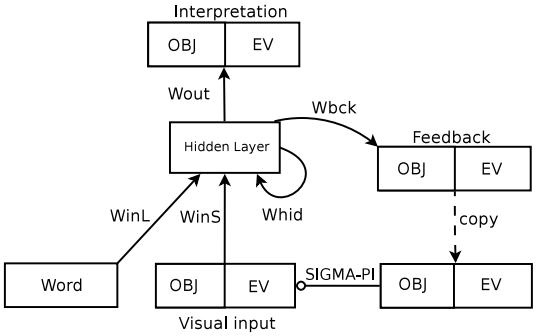

Fig. 4 *Architecture of the A-SRN$_{bck}$ model with an internal explicit language-mediated attention mechanism (for description see the text).*

motivated by our initial observations about the effects of feedback mechanism and was designed to help the network avoid undesirable object inhibition. It guarantees that input representation remains preserved to a certain degree (we used $\gamma = 0.3$ in Eq. 1) which is desirable in cases when output inhibition incorrectly inhibits valid inputs, hence hindering the correct output of the network. In terms of architecture, A-SRN$^+$ falls between A-SRN and SRN.

The last model, shown in Fig. 4, uses an alternative, internal attentional mechanism that is driven by direct connections from the hidden layer. It modulates the input similarly to A-SRN but allows a different flow of error during training by using an extra set of weights to separate the output representation (the scene interpretation) from the attentional information.

The computation of the scene input vector is model-dependent, so

$$\mathbf{s}'_{\text{in}}(t) = \begin{cases} \mathbf{s}_{\text{in}}(t) & \text{for SRN} \\ \mathbf{s}_{\text{in}}(t) .* \mathbf{a}_{\text{out}}(t-1) & \text{for A-SRN} \\ \gamma\ \mathbf{s}_{\text{in}}(t) + (1-\gamma)\ \mathbf{s}_{\text{in}}(t)\ .* \mathbf{a}_{\text{out}}(t-1) & \text{for A-SRN}^+ \\ \mathbf{s}_{\text{in}}(t) .* \sigma(\mathbf{W}_{\text{bck}}.\mathbf{a}_{\text{hid}}(t-1)) & \text{for A-SRN}_{\text{bck}} \end{cases} \tag{1}$$

In the above equation, the scene representation $\mathbf{s}_{\text{in}} = [\mathbf{c}_{\text{in}}^{\text{all}}\ \mathbf{e}_{\text{in}}^{\text{all}}]$ and the symbol '.*' denotes the component-wise multiplication of two vectors (hence implementing sigma-pi connection). To avoid propagation of the misleading activation from the previous sentence, sigma-pi activation is omitted at the beginning of each sentence, leaving only $\mathbf{s}_{\text{in}}(t)$ as the scene input.

The activation of the hidden layer in all models at time t is computed as

$$\mathbf{a}_{\text{hid}}(t) = \sigma(\mathbf{W}_{\text{inL}}.\mathbf{l}_{\text{in}}(t) + \mathbf{W}_{\text{inS}}.\mathbf{s}'_{\text{in}}(t) + \mathbf{W}_{\text{hid}}.\mathbf{a}_{\text{hid}}(t-1)) \tag{2}$$

and the network output

$$\mathbf{a}_{\text{out}}(t) = [\mathbf{c}_{\text{out}}(t), \mathbf{e}_{\text{out}}(t)] = \sigma(\mathbf{W}_{\text{out}}.\mathbf{a}_{\text{hid}}(t)), \tag{3}$$

where σ is the standard logistic function $\sigma(x) = 1/(1+\exp(-x))$.

2.4 Network training

We systematically searched for optimal model parameters which were then used in testing the models and performing the comparisons described below. The hidden layer in all networks was chosen to have 150 units. Networks were trained with the back-propagation-through-time (BPTT) algorithm by propagating the error after each word [19], using the learning rate 0.01.

We generated 10,000 scenes, each of which was associated with two events. Model's attention was driven by linguistic input to the single, major event of each situation. All generated events were consistent with the world, obeying semantic constraints. With each scene representation a number of distractors (ranging from 0 to 3) was added to the input, taken from the pool of remaining agents/patients. Randomly chosen 70% of situations were used for training and the remaining 30% for testing. Data sets were distinguished by major events used in the scenes.

3. Performance Evaluation

In order to evaluate the output accuracy, we need to interpret the model output. Since this consists of two different components (OBJ and EV), we need to evaluate both. For testing the accuracy of $\mathbf{e}_{\text{out}}$ we decode the corresponding output part (using the hidden-output weights of AAN) and count the percentage of correct decodings in the test set. Regarding $\mathbf{c}_{\text{out}}$, we compare this output with all possible combinations of OBJ representations (in SOM), i.e. $\mathbf{c}_{\text{tgt}}$. Analogically, we count the percentage of matches (for both agents and patients). The standard cosine measure is also applied to both EV and OBJ outputs. All measures can be evaluated after each word is presented to capture the progress during sentence processing. We looked at the output accuracy at the end of sentences and also on network's anticipatory behavior. Anticipation means the prediction of upcoming constituents during sentence processing (i.e. predicting an action when reading a subject word, and predicting a patient when reading a subject or action words).

The illustration of a trained A-SRN during processing at the sentence *boy chases dog* is shown in Fig. 5, and corresponds to the scene in Fig. 3. OBJ-related graphs contain 8×8 units, EV-related graphs contain 48-dimensional vectors, reshaped to an 8×6 matrix for convenience. On the right, OBJ input is the composition of various objects (including distractors), EV input is the superposition of two events. Both inputs are presented to the network at the sentence beginning. On the left, both targets comprise only information about the target event (and the pertaining objects). At the bottom, both inputs become overridden by the feedback attentional mechanism that filters out irrelevant objects and non-target event information. Visual inspection of the network outputs (in the middle) reveals that they match well with both corresponding targets.

3.1 Quantitative measures used

All measures used for model evaluations are listed in Tab. I, where they are verbally explained. The symbol 'x' refers to the processing step in a sentence (if x = 1, the first word is at the input). All measures share the property that the closer the

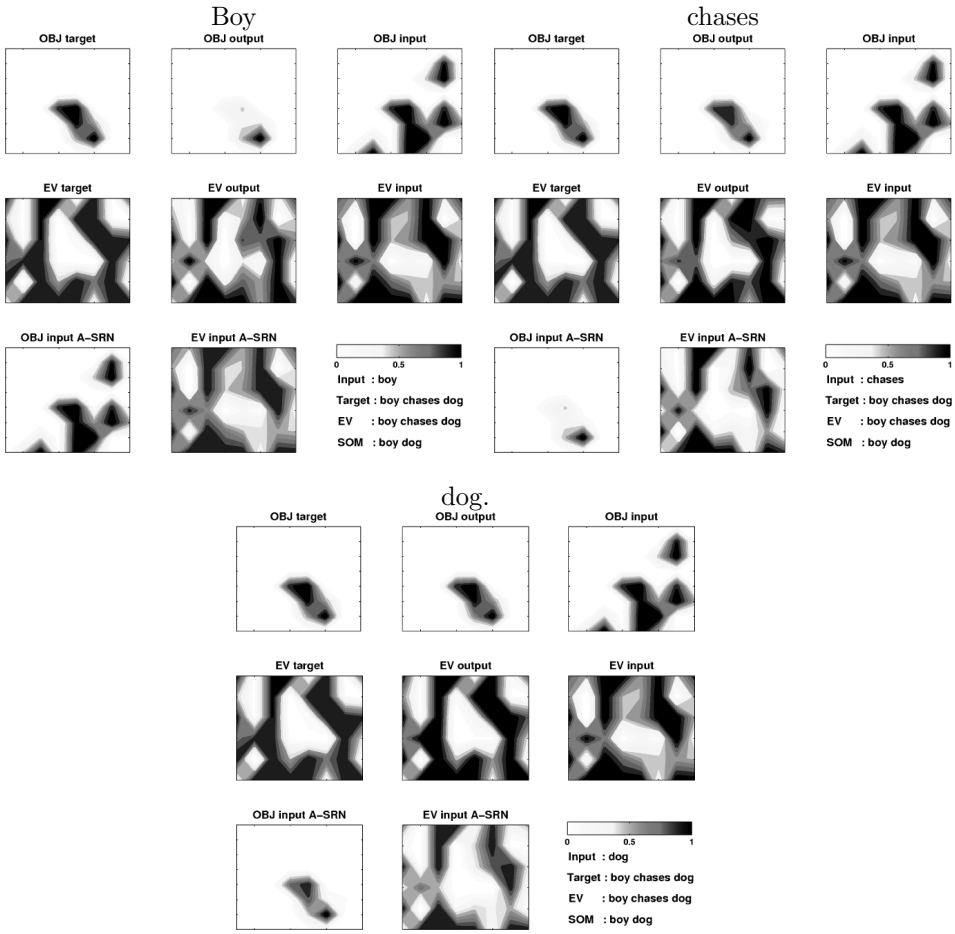

Fig. 5 *Example of A-SRN activation during sentence processing for sentence "Boy chases dog". We can notice improvement of output activation compared to targets for both EV and OBJ output parts.*

value to 1 (from below), the more accurate the value. Measures starting with EV- are related to the event level while measures starting with OBJ- are related to the object level. The symbol x $\in \{1, 2\}$. The measures subsequently appear in Tabs. II–VII.

4. Results

Results in all tables refer to the testing data. We focused on three factors when evaluating model performance: (1) We compared the accuracy of four models at the end of the sentence; (2) We manipulated the availability of the scene information during training and investigated its effect on model behavior; (3) We examined the

Acronym	Description
cos	cosine between the target situation vector and the network output (both OBJ and EV parts concatenated)
EV	quantifies the accuracy of network output by decoding it at sentence end; successful if both objects and action match the targets
EVa1	prediction of action when reading a subject; important measure since action cannot be retrieved from OBJ (unlike objects)
EVpx	prediction of the patient before the sentence end
EVa1_W	predictions of the possible actions from output after the first word; action is correctly decoded when it is consistent with the word
EVpx_W	predictions of possible patients; successful if consistent with the word (i.e. it exists in the corpus in the given context)
OBJx	prediction of agent/patient pairs; successful if both objects match the target

Tab. I *Quantitative measures used for evaluating the model performance.*

predictive properties of the model, i.e. the anticipation of upcoming consituents before the sentence end. All results shown in tables are averages of 5 simulations, all with standard deviations below 0.02. Standard deviation was larger only for results with fully omitted visual scene inputs, with highest values reaching 0.1.

4.1 End-of-sentence performance

The model's ability to yield the correct interpretation of the event, mediated by linguistic utterance, can be evaluated only at the end of the sentence. Tab. II shows that all models have learned to generate correct output with high accuracy for both parts of its representation. SRN was observed to perform sufficiently, which suggests that the feedback mechanism used in A-SRN models is not essential for this relatively simple task. However, the feedback mechanism used in A-SRN_bck improved the accuracy to nearly 100%.

Model	cos	EV	OBJ
SRN	0.995	0.985	0.986
A-SRN	0.981	0.899	0.949
A-SRN^+	0.986	0.949	0.976
A-SRN_bck	0.995	0.996	0.992

Tab. II *Model performance with respect to the target event, evaluated at the end of sentence.*

In the case of trained A-SRN, examination of its behavior revealed that it might be the suboptimality of its strict attention mechanism that sometimes inhibits (via sigma-pi connection) the target objects/actions at the input, hence reducing the

output accuracy towards the end of the sentence. To test this hypothesis, we introduced the model A-SRN$^+$, and its performance was observed to be expectedly better than A-SRN.

4.2 Restriction of the scene input

We restricted the availability of the visual input during training, either completely, or by randomly choosing 50% of the sentences (in each training epoch). The purpose of this manipulation was twofold: to simulate the lack of visual input (for example, to simulate mere listening about the given event) but also to force the network to rely more on the linguistic pathway in predicting the output. The models were then tested on two types of novel inputs – those with and without available visual inputs. Corresponding results are shown in Tab. III.

The simulations reveal that partial omission of scene inputs during training positively affects model accuracy, especially that of A-SRN. Interestingly, A-SRN also yields better performance on testing data patterns with corresponding scene inputs, compared to the training mode with 100% availability of the scene information (Tab. II).

The complete removal of the scene input during training led to excessive bonding between visual contexts and spoken language resulting in good performance for data without visual scene input (see EVe and OBJe in Tab. III). However, when testing the network with available visual inputs, the results deteriorated (for both EV and OBJ measures) showing that the network does not exhibit the ability to correctly comprehend the described event within the visual world. Because of the top-down attentional mechanism in A-SRN-based models, these models were able to handle this type of testing much better. One possibility is that these models took advantage of the initial output representation evoked by the (sole) linguistic input and feeding back as the scene input that eventually contributed to the higher accuracy at the sentence end.

Model	%	cos	EV	OBJ	EVe	OBJe
SRN	50	0.995	0.995	0.989	0.995	0.992
A-SRN	50	0.991	0.989	0.988	0.991	0.990
A-SRN$^+$	50	0.993	0.992	0.990	0.995	0.994
A-SRN$_{\text{bck}}$	50	0.997	0.998	0.994	1.000	1.000
SRN	0	0.929	0.504	0.627	0.999	0.997
A-SRN	0	0.963	0.769	0.823	0.998	0.994
A-SRN$^+$	0	0.947	0.671	0.688	1.000	0.994
A-SRN$_{\text{bck}}$	0	0.970	0.863	0.822	0.999	0.999

Tab. III *Model performance with respect to the target event for 50% and 100% empty situation input, evaluated at the end of the sentence. Performance was computed for test data with full (EV, OBJ) and empty (EVe, OBJe) scene input.*

4.3 Anticipation of upcoming constituents

Tabs. IV–VI refer to the prediction accuracy (constituent anticipation) during sentence processing. All four models predict the target action (EVa1) with ∼50% accuracy (Tab. IV). However, these predictions are almost always consistent with the world knowledge (∼97%, Tab. V).

Prediction of the patient can be assessed at two steps. At reading a subject (EVp1), the predictability of the patient is around 50% w.r.t. the target but it grows to over 80% in terms of consistency with world knowledge. Prediction of a patient one step later (EVp2) grows to about 65% w.r.t. target, and to about 95% w.r.t. world knowledge.

Prediction at the level of agents and patients (OBJx) is slightly less accurate. Upon processing the first word, the accuracy of predicting both objects remains at around 50% (having the agent accurate but the patient inaccurate), and only grows to ∼60% when processing the verb.

Model	EVa1	EVp1	EVp2	OBJ1	OBJ2
SRN	0.522	0.575	0.706	0.501	0.625
A-SRN	0.503	0.510	0.645	0.452	0.588
A-SRN^{+}	0.491	0.517	0.667	0.484	0.608
A-SRN$_{\mathrm{bck}}$	0.498	0.545	0.697	0.479	0.597

Tab. IV *Network anticipation of upcoming constituents with respect to target.*

Model	EVa1$_{\mathrm{W}}$	EVp1$_{\mathrm{W}}$	EVp2$_{\mathrm{W}}$
SRN	0.975	0.872	0.964
A-SRN	0.971	0.836	0.939
A-SRN^{+}	0.969	0.843	0.952
A-SRN$_{\mathrm{bck}}$	0.971	0.857	0.963

Tab. V *Network anticipation accuracy with respect to world knowledge.*

The models with omitted scene-related inputs (Tab. VI) exhibit decreased prediction ability because of missing visual scene information. When no situation inputs are presented during training, no model can rely on this type of information, thus ignoring it also for the test set with visual information available. In addition, the prediction in the dataset without the visual input was not achieved by any model.

4.4 Hidden-layer activations

If the network is able to correctly predict the output, this ability should imply that some organization of the network's internal representations at the hidden-layer has taken place. We performed an analysis of the hidden representations, using the traditional technique (hierarchical clustering), first presented in [1]. That is, the training data were again presented to the trained network in a single sweep, the

Model	%	EVa1	EVp1	EVp2	OBJ1	OBJ2
SRN	50	0.509	0.521	0.681	0.449	0.549
A-SRN	50	0.475	0.456	0.639	0.423	0.506
A-SRN$^+$	50	0.492	0.477	0.664	0.454	0.569
A-SRN$_{\text{bck}}$	50	0.489	0.556	0.675	0.492	0.561
SRN	0	0.201	0.039	0.074	0.015	0.031
A-SRN	0	0.203	0.033	0.085	0.006	0.022
A-SRN$^+$	0	0.183	0.039	0.090	0.005	0.016
A-SRN$_{\text{bck}}$	0	0.201	0.042	0.080	0.015	0.020

Tab. VI *Network anticipation of upcoming constituents with respect to target for models with omitted scene inputs.*

hidden-layer activation vectors were recorded, shuffled with respect to the current input word, and averaged over contexts. Like Elman, we could observe some degree of internal organization between words, albeit to a lesser degree. However, there were two important differences. First, our network was not trained on a next-word prediction task but rather mapping to a static target. Second, our linguistic input is modulated (and noised) by situational inputs. The example of a hierarchical cluster diagram of A-SRN$_{\text{bck}}$ model is shown in Fig. 6. The structure of word hidden representations in the other three models was somewhat less evident.

4.5 Visual scenes with multiple events

To simulate a more realistic world environment, we created a data set with multiple events per visual scene ($k \geq 2$). Each scene contained two or three events (with a 50:50 ratio) and a random number of distractors. Similarly as in the previous text, the network task was to select the target event mediated by the utterance.

Model	cos	EV	OBJ	EVa1	EVp1	EVp2
SRN	0.995	0.989	0.989	0.463	0.447	0.586
A-SRN	0.986	0.957	0.970	0.426	0.347	0.508
A-SRN$^+$	0.989	0.981	0.984	0.450	0.363	0.523
A-SRN$_{\text{bck}}$	0.995	0.997	0.993	0.447	0.430	0.558

Tab. VII *Model performance for multiple events per visual scene.*

Tab. VII shows that all models were able to process situations with multiple events, having suprisingly better performance at the end of the sentences. The complexity of the visual scene has probably forced the models to rely on linguistic inputs, resulting in behavior similar to what we observed in the case of restricted scene input (Section 4.2). On the other hand, prediction accuracy suffers from multiple object and event possibilities, resulting in deteriorated performance.

In sum, the presented simulations reveal that all models achieve very high levels of accuracy with respect to meaning interpretation at the end of the sentence, with small differences between them. In addition, all models demonstrate a cer-

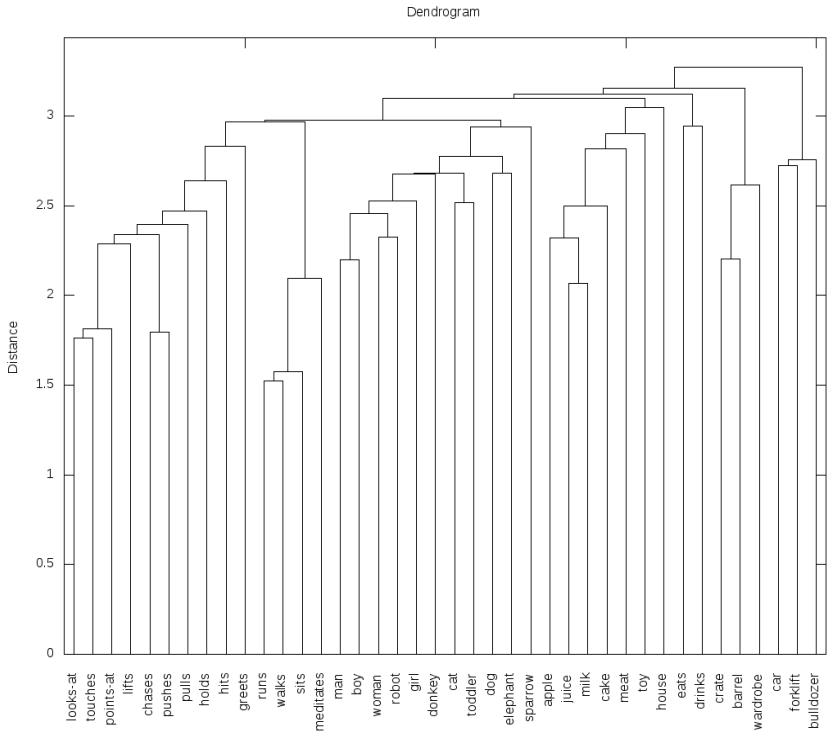

Fig. 6 *Hierarchical cluster diagram of the hidden-unit activation vectors of a trained A-SRN*$_{\mathrm{bck}}$ *model with available scene input.*

tain degree of anticipatory behavior, measured by predicting the representations of upcoming constituents before the sentence end. Only the A-SRN-based models, however, have the explicit attentional mechanisms necessary to account for cognitive behavior revealed in visual world experiments, and the model performance is indeed largely consistent with the findings of Knoeferle and colleagues [10].

5. Conclusion

This paper addresses the modeling of situated language processing as revealed by psycholinguistic experiments in the visual world paradigm. We introduced several recurrent neural network models with an explicit attentional mechanism and compared them with a standard SRN to better understand the role of the feedback in a visually-situated sentence comprehension task. All models can almost perfectly learn to generate the end-of-sentence representation that is interpreted as the sentence meaning in the visual context. Having read the sentence, each network correctly selects the target scene event and its corresponding constituents (agent/patient). All networks also demonstrate predictive behavior reflected by the ability to anticipate upcoming constituents. The SRN performs expectedly

very well, but we have shown that adding an explicit attentional mechanism (in A-SRN$_{\text{bck}}$) results in slight improvement of the performance. The availability of the attentional mechanism helps the A-SRN models to perform better on testing data with and without the scene information when trained on inputs with limited scene information. From a cognitive perspective, A-SRN's attentional mechanism helps the network focus on the relevant scene event and incorporates into the model the visual attention system on an abstract level. In addition, it reveals similar anticipatory shifts in visual attention that have been found using the visual world paradigm [10, 12].

We have shown that the models are also able to process linguistic utterances in the absence of a visual context, but adding the scene input helps the network to correctly identify the described event within the visual world and it enables to anticipate upcoming event constituents. A-SRN models differ crucially from CIANET [13] that served as our motivation, in their potential to deal with relatively complex visual scenes containing more than two events, and arbitrary numbers of objects. This property allows use of more realistic world scenes and the ability to deal with complex (possibly recursive) sentences with multiple relations between their constituents.

Regarding the world complexity, we expect that the benefits of the A-SRN model (i.e. anticipation of objects in the scene) over standard SRNs may in fact increase, as the knowledge of the network scales up. That is, when there is a larger difference between what the network learns during training, and what is actually depicted when processing a given sentence.

Acknowledgments

This work was supported by the Slovak Grant Agency for Science, #1/0439/11 (I. F., J. Š.), in part by the Humboldt foundation (I. F.), and the Cluster of Excellence "Multi-model Computing and Interaction" (M.W.C.) funded by the German Science Foundation (DFG).

References

[1] Elman J.: Finding structure in time. Cognitive Science, 14, 1990, pp. 179–211.

[2] Lawrence S., Giles C. L., Fong S.: Natural language grammatical inference with recurrent neural networks. IEEE Transactions on Knowledge and Data Engineering, **12**, 1, 2000, pp. 126–140.

[3] Christiansen M., Chater N.: Toward a connectionist model of recursion in human linguistic performance. Cognitive Science, **23**, 2, 1999, pp. 157–205.

[4] Tong M. H., Bickett A. D., Christiansen E. M., Cottrell G. W.: Learning grammatical structure with echo state networks. Neural Networks, **20**, 3, 2007, pp. 424–432.

[5] Jaeger H.: The "echo state" approach to analyzing and training recurrent neural networks. Technical report GMD 148, German National Research Center for Information Technology, 2001.

[6] Čerňanský M., Tiňo P.: Comparison of echo state networks with simple recurrent networks and variable-length Markov models on symbolic sequences. Proceedings of the 17th International Conference on Artificial Neural Networks, 2007, pp. 618–627.

[7] Huettig F., Rommers J., Meyer A.: Using the visual world paradigm to study language processing: A review and critical evaluation. Acta Psychologica, 137, 2011, pp. 151–171.

[8] Tanenhaus M., Spivey-Knowlton M., Eberhard K., Sedivy J.: Integration of visual and linguistic information in spoken language comprehension. Science, 268, 1995, pp. 1632–1634.

[9] Spivey M., Tanenhaus M., Eberhard K., Sedivy J.: Eye-movements and spoken language comprehension: Effects of visual context on syntactic ambiguity resolution. Cognitive Psychology, 45, 2002, pp. 447–481.

[10] Knoeferle P., Crocker M., Scheepers C., Pickering M.: The influence of the immediate visual context on incremental thematic role-assignment: Evidence from eye-movements in depicted events. Cognition, 95, 2005, pp. 95–127.

[11] Kamide Y., Altmann G., Haywood S.: Prediction in incremental sentence processing: Evidence from anticipatory eye movements. Journal of Memory and Language, 49, 2003, pp. 133–156.

[12] Knoeferle P., Crocker M.: The coordinated interplay of scene, utterance, and world knowledge: Evidence from eye-tracking. Cognitive Science, 30, 2006, pp. 481–529.

[13] Mayberry M., Crocker M., Knoeferle P.: Learning to attend: A connectionist model of situated language comprehension. Cognitive Science, 33, 2009, pp. 449–496.

[14] Roy D., Mukherjee N.: Towards situated speech understanding: Visual context priming of language models. Computer Speech and Language, **19**, 2, 2005, pp. 227–248.

[15] Yu C., Ballard D., Aslin R.: The role of embodied intention in early lexical acquisition. Cognitive Science, 29, 2005, pp. 961–1005.

[16] Gold K., Scassellati B.: A robot that uses existing vocabulary to infer non-visual word meanings from observation. Proceedings of the 22nd Conference on Artificial Intelligence (AAAI-07), Vancouver, Canada, 2007.

[17] Švantner J., Farkaš I., Crocker M.: Modeling utterance-mediated attention in situated language comprehension. Proceedings of the 33rd Annual Conference of the Cognitive Science Society, Boston, MA, 2011, pp. 2235–2240.

[18] Kohonen T.: The self-organizing map. Proceedings of the IEEE, **78**, 9, 1990, pp. 1464–1480.

[19] Rumelhart D., Hinton G., Williams R.: Learning internal representations by error propagation, MIT Press, Cambridge, MA, **1**, 1986, pp. 318–362.

Appendix A: Training method

The training method for BPTT at time t with window of size $T = 3$ uses the target $\mathbf{tgt}(t)$ and errors $\mathbf{f}_1, \mathbf{f}_2, \mathbf{f}_3$ for two different error propagation paths.[5] Steps 2 and 3 are repeated T-times (with index $s \in \{0, 1, \ldots, T-1\}$) before proceeding to step 4. The symbol $\top$ denotes transpose operation.

———	(0) *init.*	———
$\Delta\mathbf{W}_{\mathrm{X}}$	:=	$\mathbf{0}$, for $\mathrm{X} \in \{\mathrm{inL}, \mathrm{inS}, \mathrm{hid}, \mathrm{bck}, \mathrm{out}\}$
———	(1) *out. weights*	———
$\mathbf{f}_1$	:=	$\mathbf{tgt}(t) - \mathbf{a}_{\mathrm{out}}(t)$
$\mathbf{f}_1$	:=	$\mathbf{f}_1 . * (\mathbf{a}_{\mathrm{out}}(t))'$
$\Delta\mathbf{W}_{\mathrm{out}}$	:=	$\Delta\mathbf{W}_{\mathrm{out}} + \mathbf{f}_1 . \mathbf{a}_{\mathrm{hid}}(t)^\top$
$\mathbf{f}_1$	:=	$\mathbf{W}_{\mathrm{out}}^\top . \mathbf{f}_1$
$\mathbf{f}_2$	:=	$\mathbf{0}$
======	(2) *hid. weights*	======
$\mathbf{f}_2$	:=	$\mathbf{f}_1 + \mathbf{f}_2$
$\mathbf{f}_2$	:=	$\mathbf{f}_2 . * (\mathbf{a}_{\mathrm{hid}}(t-s))'$
$\Delta\mathbf{W}_{\mathrm{inL}}$	:=	$\Delta\mathbf{W}_{\mathrm{inL}} + \mathbf{f}_2 . \mathbf{l}_{\mathrm{in}}(t-s)^\top$
$\Delta\mathbf{W}_{\mathrm{inS}}$	:=	$\Delta\mathbf{W}_{\mathrm{inS}} + \mathbf{f}_2 . \mathbf{s}_{\mathrm{in}}^{'}(t-s)^\top$
$\Delta\mathbf{W}_{\mathrm{hid}}$	:=	$\Delta\mathbf{W}_{\mathrm{hid}} + \mathbf{f}_2 . \mathbf{a}_{\mathrm{hid}}(t-s-1)^\top$
$\mathbf{f}_3$	:=	$\mathbf{f}_2$
$\mathbf{f}_2$	:=	$\mathbf{W}_{\mathrm{hid}}^\top . \mathbf{f}_2$
$\mathbf{f}_1$	:=	$\mathbf{0}$
———	(3) *feedback*	———
$\mathbf{f}_1$	:=	$\mathbf{W}_{\mathrm{inS}}^\top . \mathbf{f}_3$
$\mathbf{f}_1$	:=	$\mathbf{f}_1 . * (\mathbf{a}_{\mathrm{X}}(t-s-1))' . * \mathbf{s}_{\mathrm{in}}(t-s)$
$\Delta\mathbf{W}_{\mathrm{X}}$	:=	$\Delta\mathbf{W}_{\mathrm{X}} + \mathbf{f}_1 . \mathbf{a}_{\mathrm{hid}}(t-s-1)^\top$
$\mathbf{f}_1$	:=	$\mathbf{W}_{\mathrm{X}}^\top . \mathbf{f}_1$
======	(4) Δ*weights*	======
$\mathbf{W}_{\mathrm{X}}$	:=	$\mathbf{W}_{\mathrm{X}} + \alpha\Delta\mathbf{W}_{\mathrm{X}}$, for $\mathrm{X} \in \{\mathrm{inL}, \mathrm{inS}, \mathrm{hid}, \mathrm{out}\}$
———	(5) *end*	———

In the above equations, $\mathbf{s}_{\mathrm{in}}^{'}(t-s)$ follows the definition from eq. 1 and the subscript X in step 3 is a notation for the output or backward layer, based on a type of the model (A-SRN or A-SRN$_{bck}$ respectively). In case of SRN, step 3 is not executed. Model A-SRN$^+$ is trained in the same way as A-SRN.

[5] In case of A-SRN models; in case of SRN, only one error propagation path is used.