Artificial Neural Networks for Repairing Language

Volker Weber Stefan Wermter
University of Hamburg, Computer Science Department
Vogt-Kölln-Straße 30, D-22527 Hamburg, Germany
{weber,wermter}@informatik.uni-hamburg.de
http://www.informatik.uni-hamburg.de/Arbeitsbereiche/NATS/staff/{weber,wermter}.html

Abstract. Spontaneous language contains many discontinuities caused by unusual order, false starts, repairs, repetitions, pauses, etc. Since data-driven artificial neural networks possess an inherent fault tolerance we use this property for dealing with such forms of "sequential noise". We describe an approach for a flat syntactic and semantic interpretation of ill-formed utterances in spontaneous dialogs using symbolic methods for communication and simple known mappings as well as connectionist methods for unknown mappings. As an example for fault-tolerant flat analysis we describe the use of our syntactic and semantic representation for recovering from repairs in our hybrid approach using real–world spontaneous dialog utterances.

1 Introduction

Previous connectionist approaches often concentrated on the analysis of relatively well-formed sentences from written text. In contrast to text processing, spontaneous language often contains ill-formed utterances or at least parts of utterances are ill-formed. Examples are unusual word order, false starts, repairs, repetitions or pauses. There are some approaches for dealing with such phenomena which make use of connectionist networks (artificial neural networks), for instance the connectionist architectures of Pollack for representing symbolic properties like compositionality [11], Elman's network for sequential representation [3], Dolan and Dyer's role binding system [2], Fanty's learned structural analysis [4], and St. John's semantic case role analysis [14]. In our previous work we used hybrid connectionist techniques for learning a flat robust scanning understanding of text language [17, 19, 18]. This work has shown the ability of connectionist networks for learning and generalizing flat representations. In addition, for analyzing spontaneously *spoken language*, we have to stress the importance of *fault-tolerant* processing with a learned sequential classification for hard real-world utterances. Our work for processing spontaneous language is based on three principles:

- *fault-tolerant processing* for instance for repair recovery
- *learning* syntax and semantics for dealing with irregular and regular utterances
- *screening approach* to cope with erroneous, messy spontaneous language

These principles were the guideline for building the hybrid-connectionist based flat screening system SCREEN[1]. The first corpus we chose contained spontaneous spoken and transcribed utterances from a railway counter. This RTC[2] corpus contains a number of spontaneous utterances with different syntactic and semantic structures as well as many irregularities, for instance repairs. However, SCREEN does not depend on the domain of traveling inquiries and we also transferred the knowledge incorporated in the system to a new meeting arrangements corpus (BMC[3] corpus) [21]. SCREEN's architecture consists of many symbolic or connectionist modules working in parallel for syntactic and semantic processing, speech analysis, and repair recovery. The communication and integration of these modules is performed by an incremental parallel interaction, similar to message passing. In this paper we will present the results for repair recovery and error correction based on the system's flat representation.

2 Flat Language Analysis

The example utterances we use in this paper are taken from the RTC corpus which contains transcriptions of spoken dialogs from a railway counter. These spontaneous real utterances incorporate various forms of "noise", for instance, interjections (eh, ...), hesitations (mm, ...), simple repairs as word repetitions (I I need ...) or phrase repairs (at morning

[1] SCREEN stands for Symbolic Connectionist Robust EnterprisE for Natural language.
[2] Corpus compiled at the University of Regensburg (FRG) containing travel inquiries.
[3] Corpus compiled at the University of Karlsruhe (FRG) containing meeting arrangements (also called Blaubeuren dialogs).

category name	basic syntactic category
N	noun
V	verb
A	adverb
R	preposition
C	conjunction
J	adjective
U	pronoun
D	determiner
M	numeral
I	interjection
P	past participle
O	other
/	pause
	abstract syntactic category
NG	noun group
VG	verb group
AG	adverbial group
PG	prepositional group
CG	conjunction group
MG	modus group (interrogative pronouns and confirmation words)
IG	interjection group (interjections, phonetic material, pauses)
SG	special group (words “without” a meaning (e.g. politeness))

Table 1: Syntactic categories

category name	Examples for basic semantic
NEED	need, would like
MOVE	go, ride
STATE	know, exist
AUX	can, could
SAY	say, ask
QUESTion	which, when (question words)
PHYSical	train, wagon (physical objects)
ANIMate	I, you (animate objects)
ABStract	connection, class (abstract objects)
HERE	on, in (time/location state words)
SouRCe	from, (time/location source words)
DESTination	to (time/location dest. words)
LOCation	Frankfurt, Hamburg
TIME	tomorrow, 3 o' clock
HOW	with, without
NEGation	no
NILL	a, the
	abstract semantic category
ACTion	action for full verb events
AUX-action	auxiliary action for aux. events
AGENT	agent of an action
OBJect	object of an action
RECIPient	recipient of an action
INSTRument	instrument for an action
MANNER	how to achieve an action
TiMe-AT	at what time
TiMe-FRoM	start time
TiMe-TO	end time
LoCation-AT	at which location
LoCation-FRoM	start location
LoCation-TO	end location
QUESTion	question phrases
MISC	miscellaneous words

Table 2: Semantic categories

/ at Monday morning ...), etc. For our experiments we used 172 of these utterances. As an example for illustrations in this paper we choose the sentence below. The RTC corpus is a German corpus but for easier illustration we have translated this sentence to English.

> I nee[d] · I need the best connection · from Etterzhausen to Hindelang · at Saturday

This utterance contains a number of pauses (·), a break within a word (nee[d]) and a self repair (I nee[d] · I need the best connection). Since traditional symbolic syntax parsers and semantic analyzers have difficulties dealing with such problems we decided to design a hybrid fault-tolerant system with a particular correction part for repairs. Because we do not emphasize the use of symbolic rules for analysis they are not violated as easily as in many symbolic systems. Furthermore, ill-formed structures are detected and removed from the utterance. This is necessary because spoken language often contains ill-formed utterances and since we are also using hypotheses from a speech-recognizer we also have to cope with the problem of potentially wrong word-hypotheses. However, for reasons of simplicity, in this paper we will focus on the analysis of one sentence hypothesis, the transcription of the utterance.

We are using a rather flat representation in SCREEN which consists of four basic parts: a syntactic and a semantic word description and a syntactic and semantic phrase description.

In previous work we discussed in detail the ability of learning and using a flat representation for a syntactic analysis which uses the categories of table 1 (ref. [20]), respectively for the semantic analysis which uses the categories of table 2 (ref. [16]). In this paper we focus on the correction part and the networks for repairs. Therefore we will give here only a brief description of the syntactic and semantic networks. For instance the syntactic and semantic analysis of a part of our example sentence results in the following output:

Utterance		I	need	the	best	connection
Syntax	basic	U	V	D	J	N
	abstract	NG	VG	NG		
Semantic	basic	ANIM	NEED	NILL	HOW	ABS
	abstract	AGENT	ACT	OBJ		

A basic category is assigned to each word of the utterance (syntax: I ← pronoun, need ← verb, the ← determiner, ...; semantics: I ← animate, need ← need event, the ← no special semantics) and an abstract category to each phrase (syntax: the best connection ← noun group; semantics: the best connection ← object).

Table 1 shows the used syntactic categories as well as their abbreviations. Table 2 illustrates the semantic categories which were developed with an emphasis on the travel domain and the general background described by Fillmore [5]. For the basic semantic categories we give a number of examples. One word may have different basic categories, for instance "best" could be *adverb*, *adjective*, or *noun*, and "train" could be an abstract object (if used in the sense of a train *connection*) or physical object. Abstract categories are more general and are used for phrases rather than for single words.

3 Principles of Repair–Recovery

Repairs may be described by a scheme of their elements (figure 1). They consist of three parts: the *original utterance*, the *editing phrase* and the *repair* itself. An editing phrase which marks a point of interruption is often introduced by an *editing term* (an interjection or hesitation) but might also be incomplete or completely missing. There is a *structural link* between the original utterance and the repair which is often indicated by a number of lexically identical words or syntactically and semantically similar words. If we look at figure 1 and restrict ourselves only on lexical identity we find "at ... morning" as a structural link but it can be argued that "the" respectively "Monday" also belong to the structural link because it fulfills a similar function. The structural link does not have to be lexically equal as in: "Down there is a *red* eh *pink* node". So some repairs are driven by *word-identity* and some by *category-identity* [8]. The part changed in the original utterance is the *reparandum* which is replaced by the *alteration*. Also the reparandum or alteration might be missing.

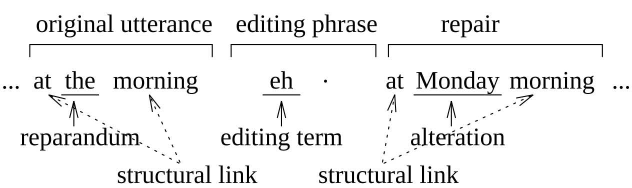

Figure 1: Structure of a repair (ref. [7])

Our approach makes use of this structure of repairs: First we try to reduce the editing phrase so that the original utterance and the repair are neighbor words or phrases, second we try to find a structural link by lexical, syntactic, or semantic equalities of words or phrases in direct neighborhood, third we decide depending on these equalities whether two subsequent words or phrases induce a repair or not.

4 Overview of the architecture

Based on principles of fault-tolerant processing, learning, and screening understanding we designed a parallel incremental hybrid architecture. The input can be a stream of word hypotheses from an underlying speech recognizer but here we illustrate the system behavior only with a single sentence hypothesis, the transcription of an utterance. The systems output is the repaired utterance together with its syntactic and semantic interpretation.

Parts of the architecture The six basic parts of the architecture of SCREEN are shown in figure 2: the speech sequence construction part, the speech evaluation part, the category part, the correction part, the subclause part, and the case frame part. Most modules are realized by connectionist feed-forward and recurrent networks, but there are also symbolic modules for simple mappings, like the detection of pauses. The communication between modules is done by symbols with confidence values which reflect activation values in case of modules with connectionist networks.

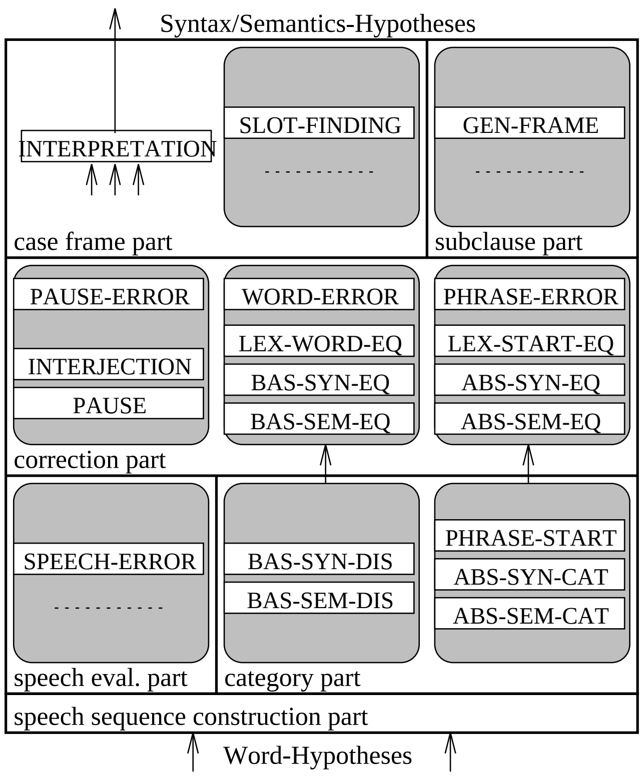

Figure 2: SCREEN: Modules of the six parts

Word-hypotheses from a speech recognizer are received by the *speech sequence construction part* and sequence hypotheses are constructed. Then word-hypotheses as part of sequence hypotheses are passed incrementally through the *speech evaluation part*. This part provides a syntactic and semantic plausibility of the output from a speech recognizer. The *category part* receives a word hypothesis in its respective context and provides a syntactic and semantic categorization for words. First, the syntactic and semantic categories of the words looked up in a lexicon are disambiguated depending on their current context (BAS-SYN-DIS, BAS-SEM-DIS, see network in figure 3). Then, the categorization of abstract syntactic and semantic categories (ABS-SYN-CAT, ABS-SEM-CAT, see net in figure 4) and the identification of phrase starts (PHRASE-START) are performed. The *correction part* checks and corrects pause errors as they occur in the editing phrase (pauses, interjections, word breaks), word repairs, and phrase repairs which might occur within sentences. Modules exist for hesitation detection (INTERJECTION, PAUSE), for the detection of lexical, syntactic, and semantic equality of two subsequent words (LEX-WORD-EQ, LEX-START-EQ, BAS-SYN-EQ, ABS-SYN-EQ, BAS-SEM-EQ, ABS-SEM-EQ, see network in figure 5), and for the detection of word or phrase repairs due to structural similarities of two subsequent words or phrases (WORD-ERROR, PHRASE-ERROR; see network in figure 6). The *subclause part* contains triggers for identifying individual subclauses within sentences and causes the system to generate new frames for subclauses (GEN-FRAME). Finally, the *case frame part* provides syntactic and semantic hypotheses about the parts of the sentence hypothesis by filling slots with words (SLOT-FINDING) and checking for constraints attached to the slots.

Some Modules The modules of the categorization part and of the correction part are most important for repair recovery. We will describe some of these modules in this paragraph in more detail. For the modules BAS-SYN-DIS (BAS-SEM-DIS), input is a sequence of words and output is a sequence of disambiguated basic syntactic (semantic) categories. For the module ABS-SYN-CAT (ABS-SEM-CAT) a sequence of words with their disambiguated basic syntactic (semantic) categories is mapped to a sequence of abstract syntactic (semantic) categories. Since both tasks are sequential learning tasks simple recurrent networks [3] have been used for training and generalization. For each training item the number of input and output units depends on the respective word representation. There are 13 (17) input and output units for BAS-SYN-DIS (BAS-SEM-DIS) for the 13 (17) basic syntactic (semantic) categories. The network in figure 3 shows the 13 input and output units labeled with their basic syntactic representation. The activation values in the input layer of this figure represent the syntactic entry of our lexicon for the word `best'. `Best' could be a noun,

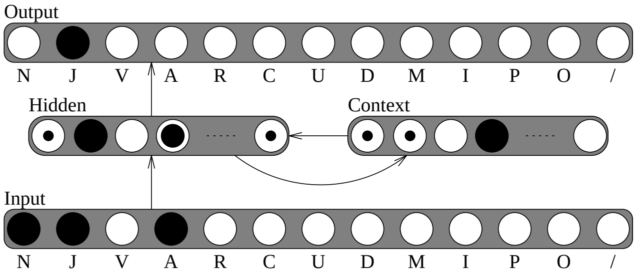

Figure 3: BAS-SYN-DIS: Recurrent network for basic syntactic analysis

adjective, or adverb. In the context of the example sentence, `adjective' has been chosen and therefore the output layer represents the disambiguated syntax J for the word `best'. For the module ABS-SYN-CAT there are 13 input units (which is the output of the disambiguation of BAS-SYN-DIS) for the basic syntactic categories, and 8 output units for the abstract syntactic categories. The network in figure 4 shows the units of the input and output layers la-

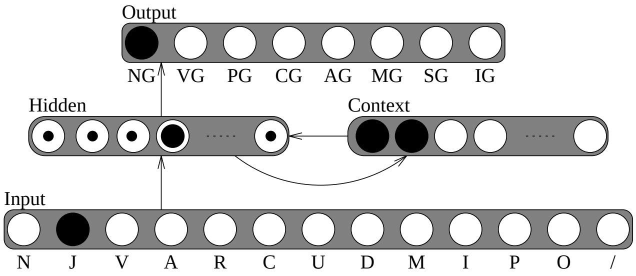

Figure 4: ABS-SYN-CAT: Recurrent network for abstract syntactic analysis

beled with the corresponding interpretations. The modules BAS-SEM-DIS and ABS-SEM-CAT are trained and used in the same way with their corresponding labels and sizes. Since context plays an important role for BAS-SYN-DIS, BAS-SEM-DIS, ABS-SYN-CAT and ABS-SEM-CAT we have used recurrent networks.

BAS-SYN-EQ (ABS-SYN-EQ, BAS-SEM-EQ, ABS-SEM-EQ) tests whether the basic syntactic (abstract syntactic, basic semantic, abstract semantic) categories of two subsequent words are equal. For BAS-SYN-EQ, BAS-SEM-EQ, ABS-SYN-EQ and ABS-SEM-EQ we use feedforward networks (figure 5), since their input (the output of BAS-SYN-DIS, BAS-SEM-DIS, ABS-SYN-CAT and ABS-SEM-CAT) is a collection of analog values. The input to BAS-SYN-EQ (BAS-SEM-EQ) is the disambiguation result of BAS-SYN-DIS (BAS-SEM-DIS) for a word and its

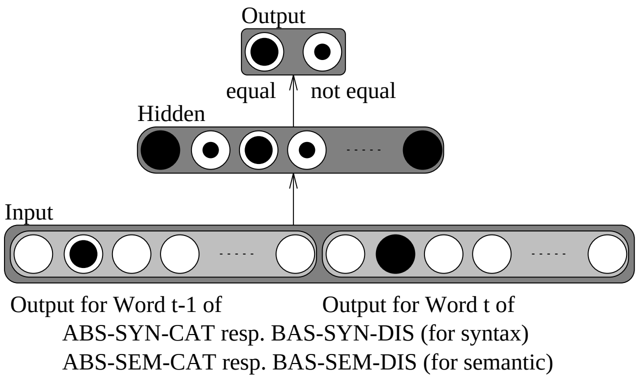

Figure 5: EQ-nets: Feedforward networks for testing equality

predecessor. For ABS-SYN-EQ (ABS-SEM-EQ) the input is the categorization result of ABS-SYN-CAT (ABS-SEM-CAT) for a word and the final categorization result for the previous phrase. Therefore we use 13 basic syntactic (17 basic semantic, 8 abstract syntactic, 15 abstract semantic) categories, that is 26 (34, 16, 30) input units for two words. The output consists of two units: equality and its negation (non-equality) to provide the possibility for faster training. A single output is computed from these two activations by (unit0 * (unit1 - 1.0)).

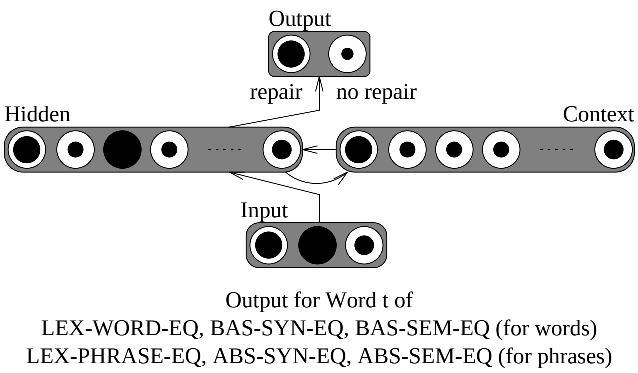

Figure 6: WORD-/PHRASE-ERROR: Recurrent networks for repairs

WORD-ERROR (PHRASE-ERROR) detects repairs for words (phrases). These modules are simple recurrent networks (figure 6) with three inputs corresponding to the outputs of the EQ-modules, hidden units and the output with the two units for repair and no repair. If the networks detect a repair the corresponding alteration replaces the reparandum.

5 An Example

The sentence mentioned in section 2 is shown in figure 7 with a number of snapshots of the running system. This sentence has an ill-formed syntax and contains some ungrammatical phenomena like pauses, a word break and a repair. Each word possesses five labels with a Hinton-diagram for the activation values. The `CONF`idence is intentionally left blank because it is only needed for interaction with the speech recognizer and speech sequence construction and evaluation parts. The upper two labels are for syntactic analysis, the lower ones for semantic analysis. In both cases, the basic decision is made, before the abstract decision. The long rectangle marks a phrase start if it is filled. In SCREEN parsing is incremental and parallel. So syntax, semantics and repair recovery modules work at the same time but on different words since some of them depend on others. Therefore the last word is always depicted without any label and its predecessor has only the basic labels.

In figure 7 `I' is disambiguated to be a pronoun and an animate. It starts a new phrase which is supposed to be a noun phrase and an agent. At first, there is also a basic categorization for the broken word `nee[d]' and a pause is just passed to the system. In the next step the broken word and pause have been deleted by the PAUSE-ERROR module. The second `I' has entered the system and is labeled in the same way as the first. The word `need' is just categorized as verb and need event. The equal structure of the two words and phrases `I' causes all equality modules to assign them as equal. As a result, the WORD-ERROR detects the repair and the altering `I' replaces the reparandum `I'. Incrementally more words are passed to the system and categorized in their particular context. Some additional pauses are detected and deleted. Finally we reach the end of the utterance and the repaired sentence found is: "I need the best connection from Etterzhausen to Hindelang on Saturday". We interpret the final abstract syntactic category at the start of a phrase (found to be effective in [19]) and the final abstract semantic category at the end of a phrase (found to be effective in [17]). So `on Saturday' is assigned to be a prepositional phrase and a time and not a location as supposed for `on'.

6 Results

In this section we illustrate the overall performance of repair-recovery using connectionist modules. The overall performance for training and test sets is illustrated in table 3. The results for the modules are based on a set of 172 utterances with 3154 patterns. We used about 1/3 for training and 2/3 for test. A training or test pattern is assigned to be correct if the output unit with the maximal activation is equal to the desired category. For the EQ-modules and repair-modules the output is counted as correct if its output is above 0.5 and is desired to show equality respectively repair and vice versa. The results for categorization are only given for completeness and because they build the basis for the EQ- and repair modules. The EQ-networks show a very high performance (100% for syntax on training and close to 100% in all other cases). The results for repair-networks (WORD-ERROR and PHRASE-ERROR)

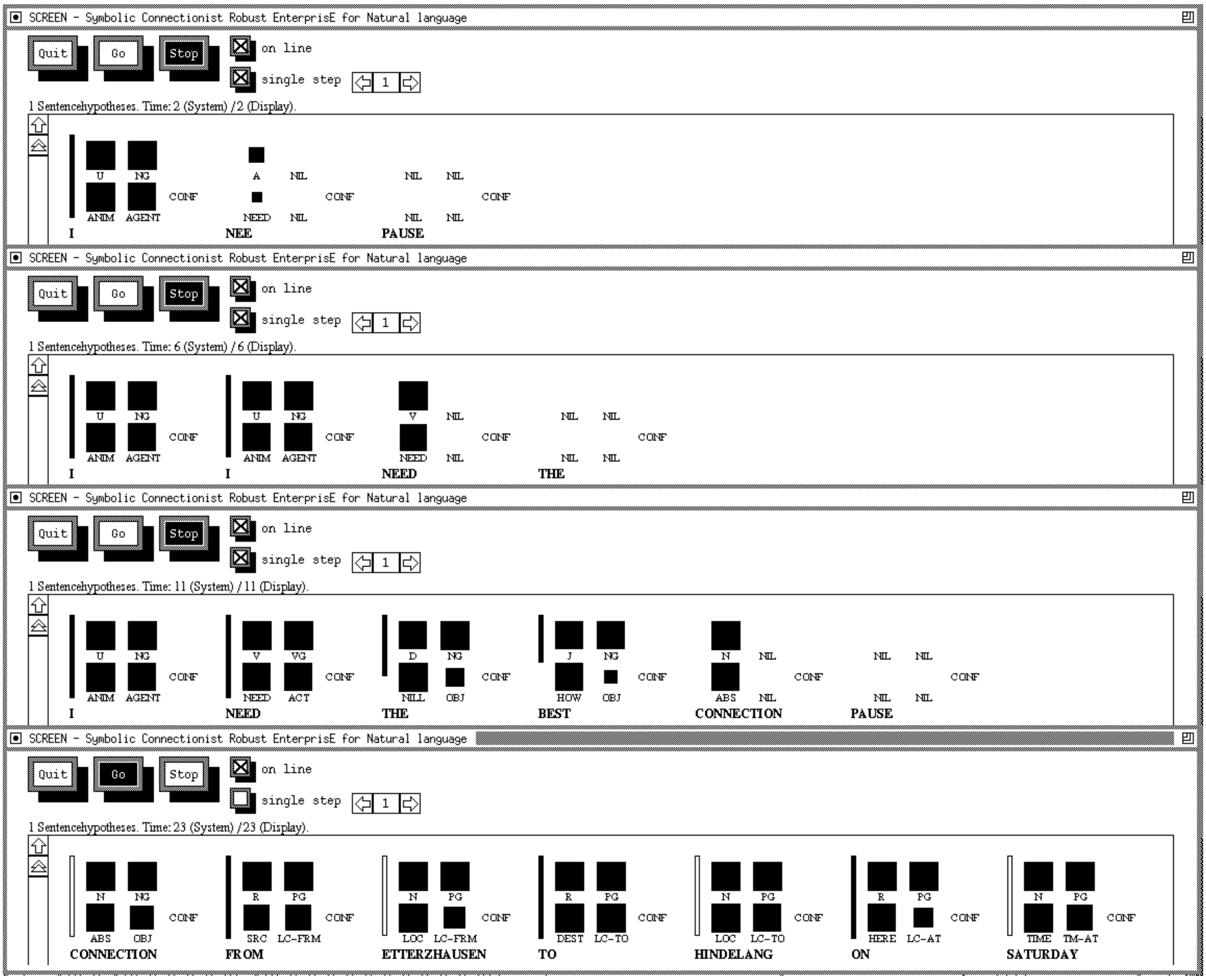

Figure 7: Four system snapshots of the category part. The abbreviations of the categories are explained in table 1 and 2.

are also above 94% in all cases. The small number of errors which have been made in these cases rely on incomplete reduction of an editing phrase and some errors have been made on category-identity driven repairs.

Module	No. of units			correct assignments	
	I	H	O	train	test
BAS-SYN-DIS	13	14	13	99%	93%
BAS-SYN-EQ	26	5	2	100%	98%
ABS-SYN-CAT	13	14	8	91%	85%
ABS-SYN-EQ	16	4	2	100%	99%
BAS-SEM-DIS	17	14	17	96%	84%
BAS-SEM-EQ	34	8	2	99%	98%
ABS-SEM-CAT	17	14	15	81%	77%
ABS-SEM-EQ	30	7	2	98%	95%
PHRASE-START	13	7	1	93%	89%
WORD-ERROR	3	4	2	95%	94%
PHRASE-ERROR	3	1	2	99%	98%

Table 3: Training and generalization performance

7 Discussion and Conclusion

We have described a new approach for fault-tolerant analysis of spontaneous utterances using a flat syntactic and semantic interpretation as well as several artificial neural networks. The networks as well as the whole architecture are designed for incremental and massively parallel processing. We chose artificial neural networks because of their inherent fault tolerance and their ability of learning. Our screening approach is able to detect structural references between parts of utterances which are used for repair recovery and correction.

Artificial neural networks have been applied to the analysis of well-formed texts [10, 14], monitoring of speech-production [1, 9, 12], and speech-based analysis [6]. However, these approaches often lack the ability to analyze *real-world* utterances either from speech or transcriptions. Since traditional connectionist models often rely on artificial corpora [10], repair and monitoring approaches are used for cognitively plausible modeling of speech production. In the context of the speech translation system JANUS [15] the connectionist parser PARSEC [6] is designed for fault-tolerant analysis but does not make use of a specialized correction part for dealing with often appearing ill-formed structures like repairs. We used the fault-tolerance of the flat connectionist analysis and showed that this analysis could be used for a discovery and correction of ill-formed structures like repairs. We claim that artificial neural networks are able to cope with ill-formed spontaneous language in a robust fault-tolerant manner even under conditions of real-world input.

Acknowledgments

This research was supported in part by the German Research Community (DFG) under contract DFG Ha 1026/6-2 and by the Federal Secretary for Research and Technology (BMBF) under contract #01IV101A0. We would like to thank S. Haack, M. Löchel, M. Meurer, U. Peters, M. Schrattenholzer and U. Sauerland for their assistance with corpus labeling and network training.

References

[1] Gary S. Dell, `A spreading-activation theory of retrieval in sentence production', *Psychological Review*, **93**, 283–321, (1986).

[2] Charles Patrick Dolan and Michael G. Dyer, `Symbolic schemata, role binding and the evolution of structure in connectionist memories', in *Proceedings of the 1st International Conference on Neural Networks*, volume II, pp. 287–298. (San Diego, CA), (1987).

[3] Jeffrey L. Elman, `Finding structure in time', *Cognitive Science*, **14**(2), 179–211, (1990).

[4] Mark A. Fanty, `Context-free parsing in connectionist networks', Technical Report 174, University of Rochester, Rochester, NY, (November 1985).

[5] Charles J. Fillmore, `The case for case', in *Universals of Linguistic Theory*, eds., E. Bach and R. Harms, 1–90, (1979).

[6] Ajay N. Jain, `Generalization performance in PARSEC - a structured connectionist parsing architecture', in *Advances in Neural Information Processing Systems 4*, eds., John E. Moody, Steve J. Hanson, and Richard R. Lippmann, 209–216, Morgan Kaufmann, San Mateo, CA, (1992).

[7] Willem J. M. Levelt, `Monitoring and self-repair in speech', *Cognition*, **14**, 41–104, (1983).

[8] Willem J. M. Levelt, *Speaking: From Intention to Articulation*, chapter 12: Self-Monitoring and Self-Repair, 458–499, MIT Press, Bradford Book, Cambridge, MA, 1989.

[9] D. G. MacKay, *The Organisation of Perception and Action*, Springer-Verlag, New York, 1987.

[10] James L. McClelland and Alan H. Kawamoto, `Mechanisms of sentence processing: Assigning roles to constituents of sentences', in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, eds., James L. McClelland, David E. Rumelhart, and The PDP research group, volume 2., Psychological and Biological Models, chapter 19, 272–331, MIT Press, Cambridge, MA, (1986).

[11] Jordan B. Pollack, `Recursive distributed representations', *Artificial Intelligence*, **46**(1-2), 77–105, (1990).

[12] Ulrich Schade and Hans-Jürgen Eikmeyer, "`wahrscheinlich sind meine Befehle soo sprunghaft und und und eh ehm zu zu telegraph": Konnektionistische Modellierung von "Covert Repairs"', in *GWAI-91: Proceedings der 15. Fachtagung für Künstliche Intelligenz, Bonn, Deutschland*, ed., Thomas Christaller, 264–273, Springer, Berlin, Heidelberg, (1991).

[13] E. A. Schegloff, G. Jefferson, and H. Sacks, `The preference for self-correction in the organization of repair in conversation', *Language*, **53**, 361–382, (1977).

[14] Mark F. St. John, *The Story Gestalt – Text Comprehension by Cue–Based Constraint Satisfaction*, Ph.D. dissertation, Department of Psychology, Carnegie Mellon University, 1990.

[15] Alexander Waibel, Ajay N. Jain, A. McNair, J. Tebelskis, L. Osterholtz, H. Saito, O. Schmidbauer, T. Sloboda, and M. Woszczyna, `JANUS: Speech-to-speech translation using connectionist and non-connectionist techniques', in *Advances in Neural Information Processing Systems 4*, eds., John E. Moody, Steve J. Hanson, and Richard R. Lippmann, 183–190, Morgan Kaufmann, San Mateo, CA, (1992).

[16] Volker Weber and Stefan Wermter, `Towards learning semantics of spontaneous dialog utterances in a hybrid framework', in *Hybrid Problems, Hybrid Solutions — Proceedings of the 10th Biennial Conference on AI and Cognitive Science (AISB-95)*, ed., John Hallam, pp. 229–238. (Sheffield, UK), (1995).

[17] Stefan Wermter, `Learning natural language filtering under noisy conditions', in *Proceedings of the Tenth IEEE Conference on Artificial Intelligence for Applications*, pp. 941–944, (1994).

[18] Stefan Wermter, *Hybrid Connectionist Natural Language Processing*, Chapman and Hall, London, UK, 1995.

[19] Stefan Wermter and Matthias Löchel, `Connectionist learning of flat syntactic analysis for speech/language systems', in *Proceedings of the International Conference on Artificial Neural Networks (ICANN'94)*, eds., Maria Marinaro and Pietro G. Morasso, volume 2, pp. 941–944, London, (May 1994). (Sorrento, Italy), Springer-Verlag.

[20] Stefan Wermter and Volker Weber, `Learning fault-tolerant speech parsing with SCREEN', in *Proceedings of the 12th National Conference on Artificial Intelligence (AAAI-94)*, volume 1, pp. 670–675, Menlo Park, (July/August 1994). Seattle, Washington, AAAI Press/The MIT Press.

[21] Stefan Wermter and Volker Weber, `Artificial neural networks for automatic knowledge acquisition in multiple real–world language domains', in *Proceedings of the 8th International Conference on Neural Networks and their Applications (NEURAP'95)*. (Marseilles, FRA), (December 1995).