Title
A Connectionist Treatment of Grammar for Generation

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 12(0)

Author
Ward, Nigel

Publication Date
1990

Peer reviewed

A Connectionist Treatment of Grammar for Generation

Nigel Ward

Computer Science Division
University of California at Berkeley

Abstract

Connectionist language generation promises better interaction between syntactic and lexical considerations and thus improved output quality. To realize this requires a connectionist treatment of grammar. This paper explains one way to do so. The basic idea is that constructions and their constituents are nodes in the same network that encodes world knowledge and lexical knowledge. The principal novelty is reliance on emergent properties. This makes it unnecessary to make explicit syntactic choice or to build up representations of sentence structure. The scheme includes novel ways of handling constituency, word order and optional constituents; and a simple way to avoid the problems of instantiation and binding. Despite the novel approach, the syntactic knowledge used is expressed in a form similar to that often used in linguistics; this representation straightforwardly defines parts of the knowlege network. These ideas have been implemented in FIG, a 'flexible incremental generator.'

1 Introduction

A generator is faced with a great number of interdependent options: lexical, syntactic, and conceptual. To produce a good utterance a generator must arrive at a consistent set of choices. The best way to do this seems to be with explicit parallel consideration of all possible options and parallel computation of their interdependencies [Ward 89c]. This ensures that the relevant information is all available, something which is difficult for generators which make choices in sequence [Ward 89b].

For syntax, this implies considering many possible constructions at once. This technique appears to be used by human speakers – analysis of speech errors suggests that even normal speech is the result of competing 'plans' [Baars 80]. [Stemberger 85] realized that, in particular, human speakers can be modeled as having many 'phrase structure units' being 'partially activated' simultaneously, and stressed the importance of emergents.

I have written an intrinsically parallel generator, 'FIG,' motivated by considerations of cognitive modeling [Ward 89c]. FIG is a structured (aka localist) connectionist system. Structured connectionism allows parallelism while making it relatively easy to build, explain, and debug the system. (A distributed connectionist system would have other advantages but would be harder to develop.)

This paper presents a new approach to grammar for language generation. The key innovation is reliance on emergent properties, instead of making explicit choices and doing explicit structure-building. Despite the novel approach to processing, there is a declarative representation for linguistic knowledge.

To see that this approach works for syntactically non-trivial examples, consider that FIG's outputs include: "once upon a time there lived an old man and an old woman," "one day the old man went into the hills to gather wood," "a big peach bobbed down towards an old woman from upstream," "an old woman gave a peach to an old man," and "Mary was killed;" and corresponding Japanese sentences: "mukashi mukashi aru tokoro ni ojiisan to obaasan ga sunde imashita," "aru hi ojiisan wa yama e shibakari ni ikimashita," "kawakami kara ookii momo ga donburiko donburako to obaasan e nagarete kimashita," "ojiisan wa meeri ni momo o agemashita," and "meeri o koroshimashita."

Section 2 overviews contrasting and related treatments of syntax in generation, Section 3 summarizes the FIG approach to generation, Section 4 presents a representation for grammatical knowledge, Section 5 describes how this knowledge is used, Section 6 discusses the implementation, and Section 7 suggests fu-

[0] Thanks to Daniel Jurafsky, Robert Wilensky, and Dekai Wu. This research was sponsored by the Defense Advanced Research Projects Agency (DoD), monitored by the Space and Naval Warfare Systems Command under N00039-88-C-0292, and the Office of Naval Research under contract N00014-89-J-3205.

ture research directions.

2 Previous Research

Problems of grammar for generation have received a fair amount of attention in AI. Of the work which is concerned with the details of language, almost all is based on syntactic mechanisms adopted from linguistic theories. Most linguistic grammars come complete with mechanisms: transformations, parse-tree traversals, unification, and so on. Yet these mechanisms are not intended to be computational models. They are typically inspired by the goal of explaining sentence structure, a goal probably originally due to linguistics' focus on grammaticality.

Independent of linguistic theories, the most common metaphor for generation is as a sequence of choices among alternatives. For example, a generator may chose among words for a concept, among ways to syntactically realize a constituent, and among concepts to bind to a slot. These decisions are generally made in a fairly fixed (and generally top-down) order, thus most generators are not easily parallelizable.

In FIG grammar is important but grammaticality is not. Grammar is a tool used in the process of expressing meaning, not a goal in itself. Structure building, structure mapping, and explicit syntactic choice are dispensed with. In FIG the structure of the output is emergent, and choices are also largely emergent. One advantage of relying on emergents is that there is no need to order choices [Ward 89b], and thus the process is naturally parallelizable.

Previous connectionist research has in general not strayed far from traditional approaches to grammar. [Stolcke 89] directly implemented unification grammar. [Kalita and Shastri 87] implemented a standard symbolic generator [McDonald 1983]. Perhaps the most original connectionist generator is Gasser's CHIE [Gasser 88]. Yet even in CHIE there are choices (represented by neuron firings) and these happen sequentially, in order. The exact timing of firings seems crucial. CHIE freely uses winner-take-all subnetworks, which also cuts down on the amount of effective parallelism.

3 The FIG Approach to Generation

The task is generation of natural language from thoughts. (Most generators, presume that the input has been pre-processed by a 'what-to-say' component, and thus only need to take the 'message' and do some language-specific processing. I think this division of the task is unnecessary and unwise [Ward 88b].) Reduced to bare essentials, a generator's task is to get from concepts (what the speaker wants to express) to words (what he can say). From this point of view, the key problem in generation is computing the relevance (pertinence) of a particular word, given the concepts to express. Syntactic and other knowledge mediates this computation of relevance. Therefore FIG is based on word choice – every other consideration is analyzed in terms of how it affects word choice.

Processing in FIG is done with spreading activation in a network. The basic FIG algorithm is:

1. each node of the input conceptualization is a source of activation
2. activation flows through the network
3. when the network settles, the most highly activated word is selected and emitted
4. activation levels are updated to represent the new current state
5. steps 2 through 4 repeat until all of the input has been conveyed

An utterance is simply the result of successive word choices, thus FIG is incremental in a strong sense.

Activation represents relevance; flow of activation represents implications of relevance; and updating the activation of a node represents computing its relevance. The general direction of activation flow is from concepts to words, but nodes representing syntax 'redistribute' activation also, and feedback is pervasive.

The network must be designed so that, when it settles, the node which is most highly activated corresponds to the best next word. This paper discusses the design of the network structures which encode syntactic knowledge.

4 Knowledge of Syntax

FIG's treatment of syntax is based on Construction Grammar [Fillmore et al 89]. This approach describes the grammar of a language 'directly, in terms of a collection of grammatical constructions' [Fillmore 88]. The key characteristics here are that Construction Grammar is declarative and that the units of syntactic knowledge, namely constructions, are densely related to words, meaning, and each other.

The encodings of syntactic knowledge shown below are taken directly from FIG. These are for illustrative purposes only. I do not claim that these represent the facts of English, nor the best way to describe them in a grammar. The examples are intended simply to illustrate the representational tools and computational mechanisms available in FIG.

Figure 1 shows FIG's definition of **noun-phr**, representing a simplification of the English noun-phrase construction. This construction has three constituents:

```
(defp noun-phr
  (constituents (np-1  obl  article  ((article .8) (a-w .2) ))
                (np-2  opt  adjective((adjective .6)))
                (np-3  obl  noun      ((noun .9))) ))
```

Figure 1: The English Noun-Phrase Construction

```
(defp go-p
  (constituents (gp-1  obl go-w   ((go-w .1)))
                (gp-2  opt epart  ((vparticle .6) (directionr .2)))
                (gp-3  opt noun   ((prep-phr .6) (destr .2)))
                (gp-4  opt verb   ((purpose-clause .6) (purposer .2))) ))
```

Figure 2: Representation of the Valence of "Go"

```
(defp ex-there
  (inhibit subj-pred passive)
  (constituents (et-1 obl therew ((therew .9)))
                (et-2 obl verb   ((verb .9)))
                (et-3 obl noun   ((noun .5))) ))
```

Figure 3: Representation of the Existential "There" Construction

```
(defw peachw    (cat noun)   (expresses momoc)  (grapheme "peach")
     (nebors (initial-phoner consnt-initial .5)) )

(defs noun   (maximals (noun-phr .4)))

(defw go-w   (cat verb)  (expresses ikuc)  (valence (go-p .2))
     (grapheme (inf "go") (past "went") (pastp "gone") (presp "going")) )

(defc introductoryc  (english (ex-there .1) (a-w .1)) )

(defr purposer (english (to2w .4) (purpose-clause .1))
     (japanese (ni-w .6)))
```

Figure 4: Some Knowledge Related to Constructions

np-1, **np-2**, and **np-3**. **np-1** and **np-3** are obligatory, **np-2** is optional. Glossing over the details for the moment, the list at the end of the definitions specifies how to realize the constituent. For example, **np-1** should be realized as an **article**, with the default being **a-w** (representing the word "a"), and so on.

Figure 2 shows the construction for the case frame of the word "go." First comes **go-w**, for the word "go," which is obligatory. Next come (optionally): a verb-particle representing direction (as in "go *away*" or "go *back* home" or "go *down* to the lake"), a prepositional phrase to express the destination, and a purpose clause.

Figure 3 shows the representation of the existential "there" construction, as in "there was a poor cobbler." The 'inhibit' field indicates that this construction is incompatible with the **passive** construction and also with **subj-pred**, the construction responsible for the basic SVO ordering of English.

Figure 4 shows knowledge about when and where constructions are relevant. Constructions are associated with words. For example **noun-phr** is the 'maximal' of **noun** (actually, of all nouns), and **go-p** is the 'valence' (case frame) of **go-w**. These links encode knowledge about constructions and their heads; other relations between words and constructions are discussed in Section 5.5. Constructions are also associated with the meanings they can express. For example, **ex-there** is listed under the concept **introductory**, representing that this construction is appropriate for introducing some character into the story, and **purpose-clause** is listed as a way to express the **purposer** relation. Constructions are also associated with other constructions. For example, the fourth constituent of **go-p** subcategorizes for **purpose-clause** (Figure 2); and there are associations among incompatible constructions, for example the 'inhibit' link between **ex-there** and **subj-pred** (Figure 3).

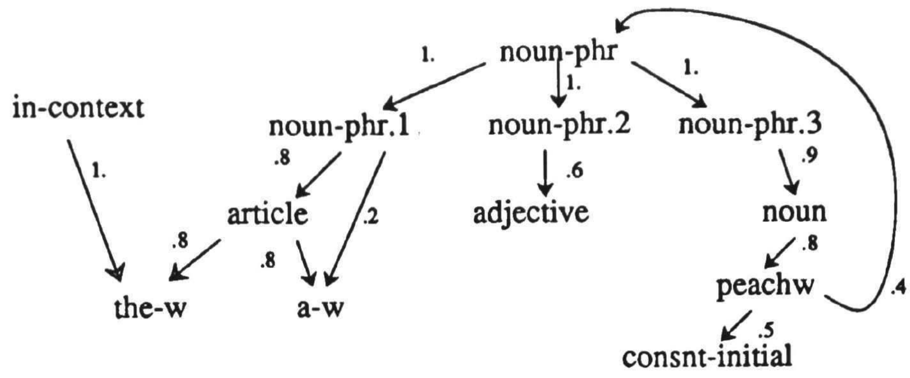

Figure 5: A Fragment of the Network

5 Using This Knowledge

While the above representations resemble those used by several modern linguistic theories, FIG uses them in a novel way.

5.1 Constructions in the Network

In FIG constructions and constituents are nodes in the knowledge network. Their activation levels represent their current relevance. Constructions receive activation from nodes linked to them, and transmit activation to other nodes. Figure 5 shows a fragment of FIG's network, where the numbers on the links are their weights. This is partially specified by the knowledge shown in the previous figures [1]. It is generally possible to give procedural interpretations to links. For example, the link from **peachw** to **noun-phr** 'advises' that: if you want to say a noun, you should 'consult' the knowledge in **noun-phr**.

Activation flow among the various nodes in the network provides, among other things, pervasive interaction among lexical choices and syntactic considerations. The rest of this section explains how this handles some basic functions of syntax.

5.2 Constituency

The links defined above suffice to handle constituency. Consider for example the fact that nouns need to be preceded by articles in this particular kind of noun phrase. Suppose that **peachw** is activated, perhaps because a **peachc** concept is in the input. Activation flows from **peachw** to **noun-phr**, from **noun-phr** to **article**, and from there to **a-w** and **the-w**. **a-w** also receives activation via a direct link from **noun-phr**.

In this way the relevance of a noun increases the relevance rating of articles. Provided that other activation levels are appropriate, this will cause some article to become the most highly activated word, and thus be selected and emitted. Note that FIG does not first choose to say a noun, then an article; both 'decisions' are considered and made together, as activation levels settle.

If there are additional sources of activation (such as **in-context**) then **the-w** will receive more activation, and thus "the" will be output. Also, if the node **vowel-initial** is more highly activated than the node **consnt-initial**, "an" instead of "a" will be output, thanks to the inflection mechanism (not described further).

There is also the question of specifying where a given concept should appear and what syntactic form it should take. This problem, subcategorization, is handled in FIG by simultaneously activating a concept and the syntactic form it should take. For example, the third constituent of **go-p** specifies that 'the direction of the going' be expressed as a 'verbal particle.' Activation will thus flow to an appropriate word node, such as **downw**, both via the concept filling the **directionr** slot and via the syntactic category **vparticle**. Thanks to this sort of activation flow, FIG tends to select and emit an appropriate word in an appropriate form [Ward 88a].

Syntactic considerations manifest themselves only through their effects on the activation levels of words. Syntax is never 'in control' of word choice [2]; the syntactic structure of the result is emergent.

[1] The mapping from s-expressions to network structures is not always trivial. For example, the link from **noun** to **peachw** comes from the statement that **peachw** had 'cat' **noun**; and the link from **peachw** to **noun-phr** is inherited by **peachw** from the 'maximals' information on **noun**.

[2] Post hoc examination of FIG output might make one think, for example, 'this exhibits the choice of the existential-there construction.' In FIG there is indeed an inhibit link between the nodes **ex-there** and **subj-pred**, and so when generating the network tends to reach a state where only one of these nodes is highly activated. The most highly activated construction can have a strong effect on word choices, which is why the appearance of syntactic choice arises.

5.3 Word Order

FIG is an incremental generator, that is, it selects and emits words one by one in order. At each time the activation level of a word must represent its *current* relevance. To this end, one job of constructions is to activate things which are currently syntactically appropriate. In FIG the current syntactic state is represented in the state of each construction node, namely, their activation levels and 'cursors.' The cursor of a construction points to the currently appropriate constituent and ensures that it is relatively highly activated. To be specific, the cursor gives the location of a 'mask' specifying the weights of the links from the construction to constituents. The mask specifies a weight of 1.0 for the constituent under the cursor, and for subsequent constituents a weight proportional to their closeness to the cursor [3]. This is parallelism among constituents.

For example, if the cursor of **noun-phr** points to **np-1**, then articles will receive a large proportion of the activation of **noun-phr**. Thus, an article is likely to be the most highly activated word and therefore selected and emitted. After an article is emitted the cursor is advanced to **np-2**, and so on. Advancing cursors is described in Section 5.5.

In this way constructions 'shunt' activation to words which should appear early in the input. FIG has no central process which plans or manipulates word order. Each construction simply activates nodes which it 'thinks' currently are relevant. In this sense word order is emergent.

5.4 Optional Constituents

When building a noun-phrase a generator should emit an adjective if semantically appropriate, otherwise it should ignore that option and emit a noun next. FIG does this without additional mechanism.

To see this, suppose "the" has been emitted and the cursor of **noun-phr** is on its second constituent, **np-2**. As a result adjectives get activation, via **np-2**, and so to a lesser extent do nouns via **np-3**. There are two cases: If the input includes a concept linked (indirectly perhaps) to some adjective, that adjective will receive activation from it. In this case the adjective will receive more syntactic activation than any noun does, and hence have more total activation, so it will be selected next. If the input does not include any concept linked to an adjective, then a noun will have more activation than any adjective (since only the noun receives semantic activation also), and so a noun will be selected next.

Most generators use some syntax-driven procedure to inspect semantics and decide explicitly whether or not to realize an optional constituent. In FIG, the decision to include or to omit an optional constituent (or adjunct) is emergent — if an adjective becomes highly activated it will be chosen, in the usual fashion, otherwise some other word, most likely a noun, will be.

5.5 Updating Constructions

Recall that FIG, after selecting and emitting a word, updates activation levels to represent the new state. In particular, it must advance the cursors of constructions as their constituents are completed [4]. Why is a separate update mechanism necessary? Most generators simply choose a construction and then 'execute' it straightforwardly. However, in FIG no construction is ever 'in control.' For example, one construction may be strongly activating a verb, but activation from other constructions may 'interfere,' causing an adverbial, for example, to be emitted instead. Therefore, in FIG constructions need feedback on what words have been output.

The difference between obl and opt constituents is whether or not the update mechanism can skip over them. (Since, for example, if there are no adjectives, the cursor of **noun-phr** should not remain stuck forever at the second constituent.) More than one construction may get updated after a word is output. For example, emitting a noun may cause updates to both the **prep-phr** construction and the **noun-phr** construction. Constructions which are 'guiding' the output should be scored as more relevant, so the update process adds activation to those constructions whose cursors have changed. It also sets a floor under their activation levels. After the last constituent of a construction has been completed, the cursor is reset and the floor is removed.

This type of bottom-up influence on constructions models an important factor affecting the syntactic form of utterances. [Bock and Warren 85] has shown that people can realize, after emitting some words, that in order to continue they must use a certain construction, for example, a passive. Similarly, FIG may output some words without any syntactic plan, then, based on the words output, the update mechanism will activate appropriate constructions, and those constructions will henceforth help guide production.

[3] For unordered construction the weight on all construction-constituent links is uniform and unchanging.

[4] Determining when a constituent is complete is not trivial. The current implementation uses a simple matching process, using the 'triggers' or each constituent (the third atoms in their descriptions).

5.6 No Instantiation or Binding

Most generators employ special mechanisms for instantiation and binding. One thing these are used for is handling multiple copies, for example, several noun phrases, or several instances of "a" in a single sentence. A connectionist system must also address this issue. An example of a problem that might occur otherwise is: in the case where several words of a category are highly activated, a node linked to all of them would receive more activation than when only one such word were active. For example **noun-phr** might receive activation from many words of category **noun**.

For this reason FIG uses a special rule for activation received across inherited links: the maximum (not the sum) of these amounts is used. For example, this rule applies to the 'maximal' links from nouns to **noun-phr**, which means that **noun-phr** effectively 'ignores' all but the most highly activated noun.

An earlier version of FIG handled this differently: by making copies of words and constructions. For example, it would make a copy of **noun-phrase** for each noun-expressible concept, and bind each copy to the appropriate concept, and to copies of **a-w** and **the-w**. Once I had started using instances and binding, it seemed natural to use those mechanisms for other problems (notably slots and cases). This approach worked, but it meshed so poorly with the basic activation-flow mechanism that I went back to a more pure spreading activation model. I conjecture that everything which seems to require 'instantiation' or 'binding' can be handled better by an appropriate refinement to the spreading activation mechanism.

5.7 Extended Example

This section describes how FIG produces "the old woman went to a stream to wash clothes." For this example the input is the set of nodes **ikucl**, **old-womancl**, **sentakucl**, **kawacl**, and **pastc**. These nodes are linked to each other as follows: **ikucl**'s **agentr** is **old-womancl**, its **purposer** is **sentakucl**, and its **destr** is **kawacl**; and **old-womancl**'s **prag-roler** is **topicc**. The concepts here have Japanese names because the input is the result of parsing the Japanese sentence "obaasan wa kawa e sentaku ni ikimashita," (old-woman TOPIC stream DEST wash-clothes PURPOSE go-POLITE-PAST).

Initially, each node of the input has 12 units of activation. Figure 6 shows the activation levels of selected nodes after activation flows, before any word is output. At this point the most highly activated word node is **the-w**, thus it will be selected and emitted first. The major source of activation for **the-w** is the first constituent of **noun-phr**, **np-1** (shown in capitals to indicate that it is the constituent currently under the cursor.) **np-1** receives energy from **noun-phr**, **noun-phr** receives most of its activation from **old-womanw**, which receives activation from **old-womancl** and from **noun**. **noun** is activated, among other reasons, by the first constituent of **subj-pred**.

One construction not mentioned previously is **backforep**, the construction responsible for putting adverbials of time and place at the beginning of a sentence. This construction has no effect on this sentence, since there are no concepts present expressible in this way.

After "the" is emitted **noun-phr** becomes even more highly activated and its cursor is moved to **np-2**. The most highly activated word becomes **old-womanw**, largely due to activation from **np-3**.

After "old woman" is emitted **noun-phr** is reset – that is, the cursor is set back to **np-1** and it thereby becomes ready to guide production of another noun phrase. Also, now the cursor on **subj-pred** advances to **sp-2**. As a result verbs, in particular **go-w**, become highly activated.

go-w is selected. Because **pastc** has more activation than **presentc** etc., **go-w** is inflected and emitted as "went." After this **subj-pred**'s cursor advances to its third constituent. At the same time, **go-p**'s cursor advances to its second constituent, thus it activates directional particles, although it happens that there is no semantic input to any such word. The most highly activated words are prepositions, due to activation from the first constituent of **prep-phr**, which in turn receives its energy from **sp-3** and to a lesser extent from **gp-3**. Of the various prepositions, **to1w** receives the most activation from **directionr**, which also receives its activation from the third constituent of **go-p**.

After "to" is emitted, the cursor of **prep-phr** is advanced. The key path of activation flow is now from the second constituent of **prep-phr** to **noun** to **streamw** to **noun-phr** to **article** to **a-w**. Thus "a" is emitted.

Then the cursor of **noun-phr** advances and "stream" is emitted. It is "stream" rather than some other noun because **streamw** is linked to **kawac**, **kawac** fills a **destinationr** relation, and **destinationr** is listed in **gp-3**.

At this point the cursor of **go-p** is on **gp-4**. From this constituent activation flows to **purpose-clause**, and in due course "to" and "wash clothes" are emitted.

In this way FIG has produced: "the old woman went to a stream to wash clothes." All the nodes of the input having been expressed, FIG ends.

6 Implementation

```
---PATTERNS---                  ----WORDS-----              ---CONCEPTS---
 16.0 BACK-FOREP                21.3 THE-W                  23.6 IKUC1
      BF-1 bf-2 bf-3            21.0 A-W                    19.7 OLD-WOMANC1
  4.9 NOUN-PHR                  12.6 OLD-WOMANW             16.7 KAWAC1
      NP-1 np-2 np-3            10.7 STREAMW                15.5 SENTAKUC1
  3.1 SUBJ-PRED                 10.6 GO-W                   12.0 PASTC
      SP-1 sp-2 sp-3             8.6 RIVERW                  9.5 CONSNT-INITIAL
  2.1 GO-P                       5.5 WASH-CLOTHESW           6.1 VOWEL-INITIAL
      GP-1 gp-2 gp-3 gp-4        3.4 TO2W                    5.9 TOPICC
  1.8 PURPOSE-CLAUSE                 ---                    ----OTHER-----
      PC-1 pc-2 pc-3                 ---                    16.0 TIMES
  0.3 PREP-PHR                       ---                     3.8 ARTICLE
      PP-1 pp-2                      ---                     2.6 NOUN
      ---                            ---                     1.8 AGENTR
```

Figure 6: Activation Levels of Selected Nodes After Activation Flow

FIG currently has six types of nodes: concepts, relations, words, constructions, constituents, and instances of concepts. They are distinguished for clarity and efficiency but not for activation flow. The cursor update process and semantic update process (not described further) do, however, examine node types; and of course only words are selected and emitted.

Although links have been differentiated above in terms of their intended meaning, activation flows across all links in the same way, except: 1. there are **english** and **japanese** links; no activation flows across the ones for the language not in use, 2. the weights of links from constructions to constituents are modified by the mask according to the cursor, and 3. inhibit links transmit negative activation.

In accordance with the intuition that a word is not truly appropriate unless it is both syntactically and semantically appropriate, the activation level for words is given by the product (not the sum) of incoming syntactic and semantic activation. 'Syntactic activation' is activation received from constituents and syntactic categories.

Currently FIG has 284 nodes and about 600 links. Before each word choice, activation flows until the network settles down, with cutoff after 8 cycles. This takes about .15 seconds on average (simulating parallel activation flow on a Symbolics 3670), thus FIG outputs words faster than a human speaker.

The correct operation of FIG depends on having correct link weights. This is not a major problem. Many of the link weights are uniform. For example, all links from syntactic categories to their members have weight .8, all 'inhibit' links have weight .7, and so on. Many of the others have a rationale: for example, the link from **np-1** to articles has relatively high weight because articles get very little activation from other sources. No single weight is meaningful; the way it functions in context is. For example, the exact weight of the link from the first constituent of **subj-pred** to **noun** is not crucial, as long as the product of it and the weight on the **agentr** relation is appropriate.

Also crucial in generation is the flow of activation through the network structures encoding world-knowledge. World knowledge is also used when monitoring the output and updating activation levels. For details see [Ward 88a].

FIG is, of course, extensible. Adding new concepts, words or constructions is generally straightforward; they can be encoded by analogy to similar nodes, and usually the same link weights suffice. Occasionally new nodes and links interact in unforeseen ways with other knowledge in the system, causing other nodes to get too much or too little activation. In these cases it is necessary to debug the network. Sometimes trial-and-error experiments are required, but often the acceptable range of weights can be determined by examination. This is a kind of back-propagation by hand; it could doubtless be automated.

Besides just increasing the amount of knowledge in FIG's network, I would like to make it model human speech errors and to use its grammatical knowledge structures for parsing also.

7 Summary

FIG's treatment of syntax is connectionist and novel. It relies heavily on parallelism and emergents. It does not build up any syntactic structures, nor even make explicit syntactic choices. The only explicit choices needed are the successive choices of words. It is also novel in that the network representations of linguistic knowledge affect word choice and order directly, rather than just affecting a parse tree. Thus the implementation corresponds clearly and directly to the knowledge-level theory.

This treatment of syntax works well with the rest of FIG — all types of knowledge are well integrated and interact freely at run time. I have explained elsewhere how this is important for: accurate and flexible word choice [Ward 88a], producing natural-sounding output for machine translation [Ward 89a], and modeling the key aspects of the human language production process [Ward 89c].

This work is not traditional linguistics, artificial intelligence, or connectionism, but uses techniques from all three fields. I have presented a syntactic mechanism which is compatible with intuitions about syntactic knowledge and also with connectionist processing. I hope this will stimulate further work in empirical computational linguistics, modeling human language production, and building and useful parallel generation systems.

References

[Baars 80] Baars, Bernard K,. The Competing Plans Hypothesis: an heuristic viewpoint on the causes of errors in speech, in *Temporal Variables in Speech*, Hans W. Dechert and Manfred Raupach (eds.), Mouton, 1980.

[Bock and Warren 85] Bock, J. K., and Richard Warren, Conceptual Accessibility and Syntactic Structure in Sentence Formulation, *Cognition* 21, 1985.

[Dell 86] Dell, Gary S., A Spreading Activation Theory of Retrieval in Sentence Production, *Psychological Review* 93, 3, 1986.

[Fillmore 88] Fillmore, Charles, *On Grammatical Constructions*, course notes, UC Berkeley Linguistics Department, 1988.

[Fillmore et al 89] Fillmore, Charles, Paul Kay, and M. C. O'Connor, Regularity and Idiomaticity in Grammatical Constructions: The Case of Let Alone, *Language* 64, 3, pp 501-538, 1988.

[Gasser 88] Gasser, Michael, A Connectionist Model of Sentence Generation in a First and Second Language, PhD Thesis, also Technical Report UCLA-AI-88-13, Los Angeles, 1988.

[Kalita and Shastri 87] Kalita, Jugal, and Lokendra Shastri, Generation of Simple Sentences in English Using the Connectionist Model of Computation, *9th Cognitive Science Conference*, Erlbaum, 1987

[McDonald 1983] McDonald, David, Description Directed Control: Its Implications for Natural Language Generation, *Computers and Mathematics with Applications*, 9, 1 1983.

[Stemberger 85] Stemberger, J. P., An Interactive Activation Model of Language Production, in *Progress in the Psychology of Language, Volume 1*, Andrew W. Ellis, ed., Erlbaum, 1985.

[Stolcke 89] Stolcke, Andreas, Processing Unification-based Grammars in a Connectionist Framework, *11th Cognitive Science Conference*, Erlbaum, 1989.

[Ward 88a] Ward, Nigel, Issues in Word Choice, *COLING-88*, Budapest, August 1988.

[Ward 88b] Ward, Nigel, An Open Design for Generation, *Proceedings of the AAAI Workshop on Text Planning and Realization*, St. Paul, MN, August 1988.

[Ward 89a] Ward, Nigel, Towards Natural Machine Translation, Proceedings of the EIC Workshop on Artificial Intelligence, Published as Technical Research Report AI89-29~ 37, Institute of Electronics, Information, and Communication Engineers, Tokyo, June 1989.

[Ward 89b] Ward, Nigel, On the Ordering of Decisions in Machine Translation, *Proceedings of the National Conference of the Japanese Society for Artificial Intelligence*, Tokyo, July 1989.

[Ward 89c] Ward, Nigel, Capturing Intuitions about Human Language Production, *Proceedings, Cognitive Science Conference*, Ann Arbor, August 1989.