Automatic and Controlled Sentence Production: A Computational Model

Eugene V. Buyakin (eugene.buyakin@gmail.com)
Department of Psychological Science, Birkbeck, University of London
Malet Street, London WC1E 7HX, UK

Richard P. Cooper (R.Cooper@bbk.ac.uk)
Department of Psychological Science, Birkbeck, University of London
Malet Street, London WC1E 7HX, UK

Abstract

We present a computational model of sentence production that emulates variation of the output of lexicalization and grammatical encoding of the abstract pre-lexical message, in terms of complexity and accuracy of the generated sentence as well as fluency and cognitive costs of the sentence production. The model integrates approaches from routine action selection models built on Dual Systems Theory (Norman & Shallice, 1986) with 'A Blueprint for the Speaker' developed by Levelt (1989). The paper describes and justifies the model architecture, explores factors affecting language variation in production, and applies the model for testing relationship between complexity, accuracy, and fluency (CAF) of language production as debated within Second Language Acquisition (SLA) research. A simulation that generated 78,750 sentences provides evidence of the trade-off relationship between CAF parameters as speakers have to sacrifice performance on one of the CAF factors in order to improve the remaining two.

Keywords: sentence production; attentional control; spreading activation model; language variation; complexity, accuracy, and fluency (CAF);

Introduction

One of the fascinating aspects of natural language is its capacity to express the same ideas in different ways. Speakers of the same language can employ different linguistic devices, including lexical choice, syntactic arrangements, prosodic features and so on to convey the same information. This paper presents a computational model that aims to emulate and quantify syntactic and lexical aspects of language variation and explore its cognitive determinants, in particular, the degree of attentional control invested by the speaker. The model is developed on the basis of the Dual Systems Theory (Norman & Shallice, 1986) and in line with models of routine action selection (Cooper & Shallice, 2000, Cooper et al., 2014) that combine routine, automatic selection of highly learned, repetitive daily activities and consciously controlled biasing of the selection mechanism by the supervisory attentional system when it is deemed necessary. The same principle can be applied to language production, with highly familiar linguistic forms being applied automatically unless higher order cognitive processes controlled by attention intervene and bias the language production process toward specific communicative goals and linguistic devices.

The topic of automaticity and control in language production has been investigated from a number of angles with the primary focus on identifying the components of the language production stack (usually higher order more complex components) that are assumed to be consciously controlled, and segregating the elements that are largely automatic (Meyer et al., 2007). The model presented in this paper attempts to take a different view and looks for a balance between controlled and automatic processing within the components of language production.

Discussions regarding the automaticity and control of language production are normally conducted within the framework developed in Levelt's tradition (Levelt, 1989) with the 'blueprint of a speaker' as the standard architecture of the language production. One of the fundamental aspects of Levelt's architecture is a division between abstract thinking as manipulation of concepts aimed to generate a unit of communication called the *message*, expressed in terms of some non-linguistic knowledge representation system, and linguistic encoding charged with transformation of the message into natural language. Message generation is performed by the 'conceptualizer' and results in a pre-lexical abstractly formulated proposition with a number of nested predicates establishing various types of relationships between the concepts. The 'formulator' or linguistic encoder performs, in its turn, two sequential transformations: from the message to surface structure (grammatical representation) and from the surface structure to inner speech (phonological representation). The focal point of language production research is at the boundary between grammatical and phonological encoding, whereas the boundary between the conceptualizer and the formulator historically has somewhat unfairly been considered less relevant. The model presented in this paper sits at the intersection of the conceptualizer and the formulator and is concerned with the transformation of the message into the surface structure.

Traditionally, the challenge of grammatical encoding is seen in conversion of a non-linear propositional formula (i.e., message) expressed in some strict logical code of abstract reasoning into a temporally organized, linear sequence of words adhering to the grammatical rules (not following any formal logic) of a specific natural language. Linearization (sequencing of words) and rule-based assignment of grammatical roles (such as subject or main verb) are the intricacies of the language production calling for explanation. For this purpose, the mapping between the conceptually encoded message and linguistically encoded sentence (surface structure) is largely assumed to be one-to-one. Some models explicitly rely on a precise mapping of concepts (items in the message) into words (items in the sentence) (e.g., WEAVER++, Roelofs, 1997) and lexical competition affects only the timing of the word choice, but not the not the particular choice of a word. Other models (e.g., Dell, 1986, Oppenheim et al., 2010) assume some competition between alternative words with the possibility of different lexicalization, but still settle on the one lemma (one word) for one concept principle, which means that, at least, the predicate structure of the message is translated into the sentence literally. These approaches lead to the understanding that the syntactic structure of the generated sentence is largely predetermined by the structure of the message. Thus, the Dual-path model (Chang, et. al., 2002, 2006) posits that, besides the set of concepts (and the concepts are assumed to be in one-to-one relationship to words), the message contains some additional information about thematic roles played by the concepts. On top of the roles the message is supplemented by the ‘event semantics’ (the indicator of relative importance, salience or priority of a certain role) as well as by extra information units flagging properties like definiteness or number. In fact, that combination provides non-ambiguous binding instructions for the construction of the sentence. This, classical, approach can be interpreted either as an assumption that the conceptualizer possesses the knowledge of the language (and is therefore capable of giving viable non-contentious instructions regarding the message construction) or as the assumption that the formulator is powerful enough to accurately encode any product of abstract thinking, incorporating not only the logic of the message, but all the nuances demanded by thematic role assignment and event semantics.

The first assumption is at odds with the division of labour between the two stages of language production postulated in the blueprint of a speaker, which can be traced to fundamental debates in cognitive science on modularity of mind (Fodor, 1983) and Whorfian linguistic relativity. The second assumption conflicts with the idea of limits of language both in philosophical terms as well as in practical terms, studied, for example, in linguistic expression of sensory experiences (smell, taste, space, etc.).

Existing computational models of language production such as those cited above are concentrated on accounts of three empirical linguistic phenomena: speech errors (e.g., Dell, 1986), the time course of language production (e.g., WEAVER++; Roelofs, 1997) and language learning (e.g., Chang et al., 2002, 2006, Oppenheim et al., 2010). Language variation, which is a cornerstone in applied linguistics in general and in second language acquisition (SLA) in particular and studied in many different angles, from learning to social aspects, and at many different levels (e.g., Bates & MacWhinney, 1987, Geeraerts et al., 2010), has received much less attention in the language production research.

The model presented in this paper draws on one of the approaches to language variation based on notions of complexity, accuracy, and fluency (CAF). These concepts are widely used to measure second language proficiency (Housen et al., 2012). The measurement of CAF is particularly convenient as it allows one to abstract away from the content of the language produced and focus on quantitative characteristics of the linguistic output and hence, as will be shown in the model description, to experiment with artificial toy languages. An important theme of CAF research is the relation between the CAF variables, with two competing theories suggesting either the existence of a trade-off between the three characteristics explained by a single pool of cognitive resources allocated between the three tasks (Foster & Skehan, 1996) or the existence of a direct correlation between the complexity of the encoding task and the quality of the language produced (Cognition hypothesis, Robinson & Gilabert, 2007, Salimi & Dadashpour, 2012). As a demonstration of the model utility it is applied to emulating and testing both hypotheses for an artificially constructed language.

Architecture and Processes of the Model

The model consists of two independent parts: the sentence generation engine and the language model that can be plugged into the engine provided it satisfies the architectural constraints. For compatibility with the engine, the language shall contain the concepts inventory (i.e., the semantic memory), and the lexicon (set of words, dictionary), as well as projections of the concepts into the lexicon (referred to as the semantic association matrix or simply semantics), and finally a bigram language model (referred to as the syntactic association matrix or collocations). The semantics connects every concept to a sub-set of words (rather than just a single word). The strength of association may be interpreted as the degree of meaning overlap or the probability distribution for a given concept to be expressed (represented) by a given word in different contexts.

The model implements the sentence generation process as a combination of two types of searches, referred to as semantic and syntactic search (see Figure 1). Semantic search is a process of looking for a word expressing the concept (through spreading activation in the semantic matrix), it is assumed to be slow and cognitively costly (and can be thought of as a multi-layer convolution-based transformation of the concept into a word). Syntactic search is the one-step activation of the word typically following a given word in a speech (based on bigram statistics). It is assumed to be fast, automatic and require little cognitive effort.

Message. The sentence generation engine operates on messages, modelled as hierarchical trees of concepts, and produces sentences, modelled as hierarchical trees of words. It is assumed that the relationship between concepts and words is not necessarily one-to-one, but that concepts may be expressed, to different degrees of approximation, by a range of different words.

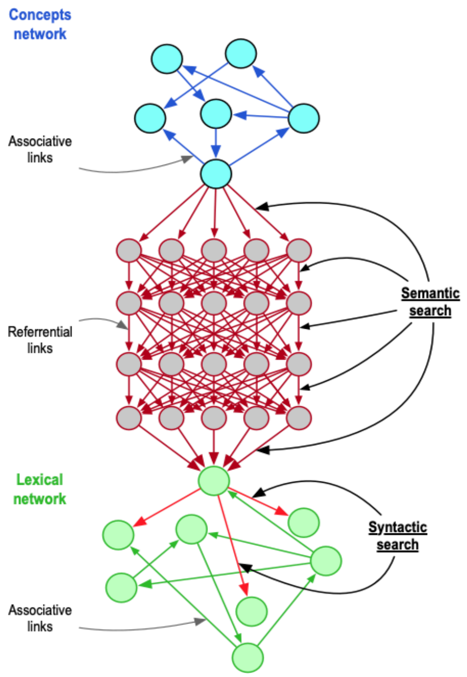

Figure 1: Semantic and syntactic search within the model

The hierarchy of concepts that constitutes the message can be thought of in a traditional fashion as a nested proposition with a number of predicates corresponding to various aspects of the thought to be articulated or to the picture to be described. For example, for a picture that one might describe as:

‘A dog with big teeth chases a scared to death kitten (1)
with blue eyes and soft fur.’

there may exist some underlying message that can look as follows:

```
x: CHASE
├── x0: DOG
│   └── x00: TEETH
│       └── x001: HUGE
└── x1: CAT
    ├── x10: EYES
    │   └── x100: BLUE
    ├── x11: FUR
    │   └── x110: SOFT
    └── x12: SCARED
        └── x120: DEATH
```
(2)

Here, ‘X’ denotes what is referred to as the top concept. The model assumes that the position of the concept in the hierarchy reflects its salience, i.e., the top concept is the topic or theme of the sentence, the concepts at the next level are the most important characteristics (attributes, properties) of the topic and the concepts deep down the tree provide some details that may be of less importance.

Sentence. The purpose of the model is to transform the message into a sentence. The sentence is understood as a syntactic tree with words as nodes. The model does not implement inflection or word ordering, it is concerned only with a functional encoding of the sentence (determined by the hierarchy of relationships between the words). In the example above the sentence corresponding to the message, besides the canonical (full, one-to-one) sentence (1), may include:

‘A dog chases a cat’ (4)
‘A dog with scary teeth chases a blue-eyed cat’ (5)
‘There is a dog with big teeth’ (6)
‘A dog chases a kitten with dark-blue eyes’ (7)
‘A canine runs after a feline’ (8)

Note, that surface variation corresponding to positional encoding of the surface structure (e.g., ‘A dog chases a cat’ can be alternatively expressed as ‘A cat is chased by the dog’) is not a subject of the model presented in this paper. Note also, that each of the sentence examples in (4) to (8) is just an approximation of the message. Actually, even the original canonical sentence (1) can be thought of as just an approximation of some richer message. One can imagine other branches, as looking at the scene a speaker perceives a lot more details than are encoded in the message tree (2). For a realistic simulation, the message should be assumed to consist of dozens of concepts, and it can be argued that the potentially large size of the message is a reason why approximation is necessary in the first place.

Lexicalization Loop. The core component of the model is the lexicalization loop that implements the iterative generation of the sentence, at each step attempting to add new words to the sentence (by attaching them to one of the existing words). The loop includes three nested cycles: an outer cycle over top node lexicalization attempts (performed through semantic search), an intermediate cycle over the existing words in the sentences (referred to as ‘expansion roots’) in search of the best node to attach the new word, and an inner cycle over syntactic connections of the expansion root in search of the best word to add to the sentence (referred to as ‘expansion leaf’). The words found by either semantic or syntactic search are accepted or rejected by the sentence generation engine depending on their meaning evaluation. In practical terms, the lexicalization process starts with an attempt to apply semantic search to the top concept. When the word is found, instead of trying to repeat the semantic search for the next concept in the tree (avoiding an expensive operation), the model tries to apply the syntactic search to the word it has just found. The found word is assumed to form a phrase with the first word and this phrase is evaluated for matching the meaning of the message.

Meaning Comparison. The model is built on the assumption that any phrase or sentence can be evaluated in terms of how well it expresses the meaning of the message. Generally speaking, any compositional semantics can be plugged in to the model, but for the purposes of the simulation presented in this paper a simple approach was taken: each concept in the message is deemed to be expressed by each word in the sentence. The extent to which a word expresses the meaning of the concept depends, however, on the relative positions of the concept and the word in the corresponding tree hierarchies. For example, the top concept is best expressed by the head word of the sentence, the immediate children of the top concept are best expressed by the immediate children of the head word, etc. Mathematically, the evaluation is expressed by the distance between the nodes in the trees multiplied by the semantic association strength between the concept and the word. Note that even though this process is based on semantic associations it does not involve semantic search and therefore is not cognitively expensive.

Semantic Distance Threshold. Meaning comparison (referred to as semantic distance evaluation) defines the next step of the lexicalization process with three possible decisions. First, if the phrase that is built with the word just found exceeds some semantic distance threshold (i.e., is close enough to the meaning of the message), then the lexicalization process is terminated. The semantic distance threshold is considered to be the key parameter set by the cognitive control processes depending on communicative intentions and available cognitive resources. Since the sentence generation process is costly, speakers may not bother to look for a precise lexicalization and may instead settle on some simple even if not very accurate representation of the message. The semantic distance threshold is the main independent variable used in simulations described below. Second, if the newly found word improves the quality of expression significantly, but does not reach the semantic distance threshold, the word is accepted into the sentence (i.e., the sentence is expanded), but the lexicalization loop continues with the new iteration at the middle cycle by choosing a new expansion root among all words in the sentence including the one that was just added. The choice of the new expansion root is made based on the position of the word in the syntactic tree and the number of existing children (i.e., words that have been accepted at previous steps). Naturally, it is assumed that the higher the word in the hierarchy and the fewer children it already has the higher the probability of it being selected for expansion, but in the more general case it may be considered a characteristic of stylistic preferences of the speaker.

The improvement in semantic distance that is required for the model to accept a word is the second independent variable manipulated in the simulation below. It is referred to as the incremental threshold (as opposed to the total threshold for the sentence acceptance as described above). It affects sentence production in the following way. A high incremental threshold requires the model to look for a precise word (semantically very close to the concept it expresses). Such a search takes longer (on a per word basis) and produces shorter sentences (as the total threshold is covered in a smaller number of steps). The lower the incremental threshold the more verbose is the model. The sentences it produces are long and complex in their syntactic structure, but individual words and phrases bear less relevant semantic content. The two extremes can be said to resemble Broca and Wernicke aphasia respectively.

The lexicalization loop terminates either when a satisfactory expression is found or when all words have been tested and none passed the threshold, or when some limit of time is reached.

Language. The model is designed to work with any language that can be described according to the specification described above (concepts inventory, lexicon, semantics, language model). Generally, the model is designed to be used with some manually encoded fragment of a natural language, but for the purposes of the simulation presented in this paper an artificially designed language was used. This allows easier quantification of relationship between the message and the sentence. The language was created as follows: a set of concepts was generated by combining 2 letter syllables with an added final letter. So, each 'concept' contained N vowels and N+1 consonants. The concepts are not supposed to be related to letters or phonemes in any way, but for convenience of presentation the letter-based algorithm of concept generation proved to be useful. As an example, DEREXIG, LUMAK, etc., are concepts built according to the rules. For each concept, a set of words were generated by randomly replacing certain numbers of letters. For example,, for the concept DEREXIG the word 'darexin' could be generated by replacing 2 letters. Semantic distance between the words and the concepts is a function of number of letter replacements. The lexicon that was generated in this way is then sampled to decrease its size and to emulate variation in availability of words matching each concept.

Simulations

The purpose of the simulations is to demonstrate the capabilities of the model by testing the trade-off hypothesis. A mini language of 100 concepts and 150 words was generated according to the algorithm described above. The language was used to generate a corpus of 10 messages with 5 concepts each and the same 3-tier hierarchical structure (equivalent, for example, to the structure of a sentence 'a big black dog barks up a tree'). Four parameters of the model were used as independent variables: semantic distance threshold (maximum for sentence acceptance), incremental threshold (maximum for a word acceptance), semantic noise (random fluctuation in semantic search), and syntactic noise (random fluctuations in syntactic search). The number of levels for each factor used in the simulation was 7, 9, 5, 5 respectively, giving a total 1575 combinations of parameters (parameter space points).

For each point of the parameter space 5 attempts to generate a sentence for each message in the corpus were undertaken, giving a total of 78,750 attempts, 41,466 of which resulted in successful generation of a sentence.

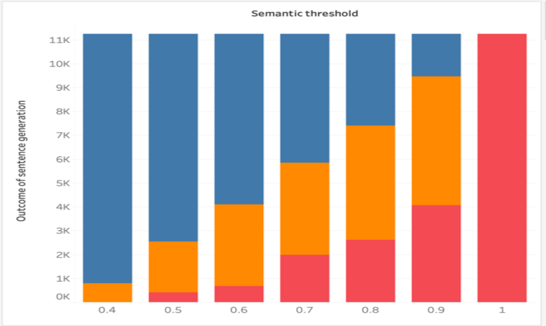

Figure 2. X-axis: semantic threshold per concept, Y-axis: number of attempts to generate sentence. Blue bar = failure to generate sentence, red bar = one-word sentence accepted, orange bar = multi word sentence accepted

Figure 2 shows, as expected, that when the threshold is too strict (equivalent of trying too hard to find the perfect words) the model often fails to generate a sentence, while when it is too low trivial one-word sentences are always accepted. Thus, the realistic range of thresholds (given the specific language and other settings of the model) producing results resembling the real-life speech is in the range from 0.6 to 0.8.

The stricter the threshold the more words the model has to try before the sentence becomes good enough, and the longer and deeper the sentence becomes (see Figure 3). If the semantic threshold is to be interpreted as a measure of accuracy, the results indicate that higher accuracy requires more complex sentences.

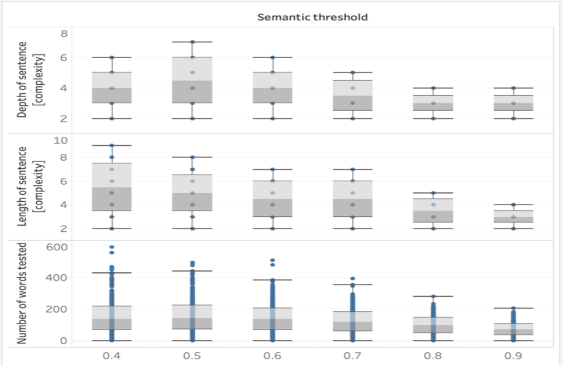

Figure 3. X-axis: semantic threshold per concept, Y-axis: top pane = depth of the sentence (number of hierarchy levels) of the sentence, middle pane = length of the sentence (number of words), bottom pane = number of words tested during lexicalization of a given sentence. Boxes in the box plot indicate the range for 50% of the data points.

The incremental threshold can be considered as a proxy for speech style and can to some extent be used to distinguish speakers oriented towards short, meaningful, precise words as opposed to 'talkative' speakers preferring to speak a lot, but not necessarily clearly or to the point. Curiously, as shown in Figure 4, a higher incremental threshold (corresponding to less talkativeness) leads to simpler, but more accurate and fluent speech (shorter sentences).

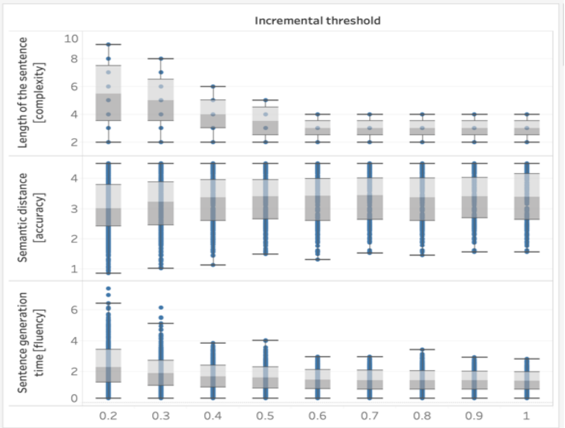

Figure 4. X-axis: incremental threshold, Y-axis: top pane = length of the sentence, middle pane = semantic distance, bottom pane = generation time.

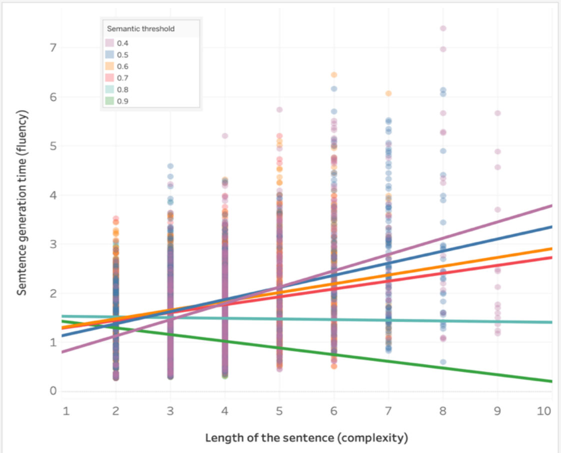

Figure 5. X-axis: length of the sentence (complexity), Y-axis = sentence generation time (fluency). The points of the scatter plot represent individual sentence generation instances. The color indicates the semantic threshold. The trend lines show relationship between the semantic distance and sentence generation time for different thresholds.

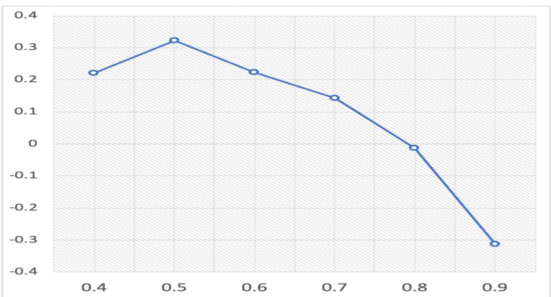

Figure 6: The slope of the regression between the generation time (fluency) and length of the sentence (complexity) (vertical axis) as a function of semantic distance threshold (horizontal axis).

Testing the Trade-off Hypothesis. The trade-off hypothesis, described in the introduction may be tested by regressing sentence generation time against sentence length, for different values of sentence acceptance threshold. As is shown in Figures 5 and 6, the slope of the resultant regression line decreases as the sentence acceptance threshold increases. This supports the trade-off hypothesis, in that if the speaker pursues higher accuracy (by recruiting attentional resources to achieve lower semantic distance), then production of more complex sentences demands disproportionally longer time (i.e., it significantly depresses fluency). In contrast, if the semantic threshold is high (which is equivalent of willingness to accept lower accuracy of the produced sentence), the speaker can increase the complexity of the language produced without a significant increase in production time. In other words, if one of the variables is sacrificed (e.g., accuracy is set to be low), then the speaker need not sacrifice fluency to achieve complexity. However, if accuracy is set high, the speaker will have to sacrifice either fluency or complexity. Thus, within the model all three variables can't be improved at the same time, and there is always trade-off.

Conclusions

We have presented a model of sentence production that is based on three major assumptions:

1. There is no single word matching the meaning of an abstract concept, rather there is a set of words matching the concept to a certain degree.
2. Sentences can be composed through the combination of semantic and syntactic search. Semantic search is more expensive than syntactic search and both searches are more expensive than evaluation of a word or sentence meaning.
3. The selection of words (lexicalization, referring expression generation) depends not only on communicative intentions, speech acts, or situational factors, but also on faculties made available by language. The choice can be made in particular because the word is easy to use or because the speaker likes it, and not because it helps to identify the object in the context in the best way.

By replicating some of the natural language statistical characteristics the results demonstrate the viability of the approach to the sentence generation taken in the model. The simulation, however, can only be taken as a small scale example of the model. Testing with a larger data set and extended parameter space and, for example, more complex messages with variable hierarchical structure, is required. The most interesting development of the model, however, would be in its application to a natural (rather than artificially constructed) language. This could be either based on a fragment of language describing a particular scene (encoded by hand), or via an application of existing word embedding data sets (such as GloVe or Word2Vec) repurposed as semantic association matrices. The model also requires further theoretical justification, especially since it steps into an area actively developing within AI and deep learning research. The work described in this paragraph is critical because the underlying assumptions (stated above) remain contentious in both the language production literature and the computational linguistics literature.

References

Bates, E., & MacWhinney, B. (1987). Competition, variation, and language learning. In *Mechanisms of language acquisition* (pp. 157–193). Lawrence Erlbaum Associates, Inc.

Chang, F. (2002). Symbolically speaking: A connectionist model of sentence production. *Cognitive Science*, *26*(5), 609–651.

Chang, F., Dell, G. S., & Bock, K. (2006). Becoming syntactic. *Psychological Review*, *113*(2), 234–272.

Cooper, R. P., Ruh, N., & Mareschal, D. (2014). The Goal Circuit Model: A Hierarchical Multi-Route Model of the Acquisition and Control of Routine Sequential Action in Humans. *Cognitive Science*, *38*(2), 244–274.

Cooper, R., & Shallice, T. (2000). Contention scheduling and the control of routine activities. *Cognitive Neuropsychology*, *17*(4), 297–338.

Dell, G. S. (1986). A spreading-activation theory of retrieval in sentence production. *Psychological Review*, *93*(3), 283–321.

Fodor, J. A. (1983). *The Modularity of Mind*. MIT Press.

Foster, P., & Skehan, P. (1996). The Influence of Planning and Task Type on Second Language Performance. *Studies in Second Language Acquisition*, *18*(3), 299–323.

Geeraerts, D., Kristiansen, G., & Peirsman, Y. (Eds.). (2010). *Advances in cognitive sociolinguistics*. Mouton de Gruyter.

Housen, A., Kuiken, F., & Vedder, I. (Eds.). (2012). *Dimensions of L2 performance and proficiency: Complexity, accuracy and fluency in SLA*. John Benjamins Pub. Co.

Levelt, W. J. M. (1989). *Speaking: From intention to articulation*. The MIT Press.

Meyer, A., Wheeldon, L., & Krott, A. (2007). *Automaticity and Control in Language Processing*. Psychology Press.

Norman, D. A., & Shallice, T. (1986). *Attention to action*. Springer.

Oppenheim, G. M., Dell, G. S., & Schwartz, M. F. (2010). The dark side of incremental learning: A model of cumulative semantic interference during lexical access in speech production. *Cognition*, *114*(2), 227–252.

Robinson, P., & Gilabert, R. (2007). Task complexity, the Cognition Hypothesis and second language learning and performance. *IRAL - International Review of Applied Linguistics in Language Teaching*, *22*(1).

Roelofs, A. (1997). The WEAVER model of word-form encoding in speech production. *Cognition*, *64*(3), 249–284.

Salimi, A., & Dadashpour, S. (2012). Task Complexity and Language Production Dilemmas (Robinson's Cognition Hypothesis vs. Skehan's Trade-off Model). *Procedia - Social and Behavioral Sciences*, *46*, 643–652.