Discovering the units in language cognition: From empirical evidence to a computational model

Jinbiao Yang

(杨金彪)

Funding body
This research was funded by the Max Planck Society for the Advancement of Science (www.mpg.de/en) as an International Max Planck Research School for Language Sciences Ph.D. Fellowship awarded to Jinbiao Yang (grant period 2017–2021).

International Max Planck Research School (IMPRS) for Language Sciences
The educational component of the doctoral training was provided by the International Max Planck Research School (IMPRS) for Language Sciences. The graduate school is a joint initiative between the Max Planck Institute for Psycholinguistics and two partner institutes at Radboud University – the Centre for Language Studies, and the Donders Institute for Brain, Cognition and Behaviour. The IMPRS curriculum, which is funded by the Max Planck Society for the Advancement of Science, ensures that each member receives interdisciplinary training in the language sciences and develops a well-rounded skill set in preparation for fulfilling careers in academia and beyond. More information can be found at www.mpi.nl/imprs

The MPI series in Psycholinguistics
Initiated in 1997, the MPI series in Psycholinguistics contains doctoral theses produced at the Max Planck Institute for Psycholinguistics. Since 2013, it includes theses produced by members of the IMPRS for Language Sciences. The current listing is available at www.mpi.nl/mpi-series

© 2022, Jinbiao Yang
ISBN: 978-94-92910-40-0
Cover design by Mengdi Yang & Jinbiao Yang
Printed and bound by Ipskamp Drukkers, Enschede

All rights reserved. No part of this book may be reproduced, distributed, stored in a retrieval system, or transmitted in any form or by any means, without prior written permission of the author. The research reported in this thesis was conducted at the Max Planck Institute for Psycholinguistics, in Nijmegen, the Netherlands.

Ontdekking van de eenheden in taalcognitie: Van empirisch bewijs naar een computationeel model

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen
op gezag van de rector magnificus prof. dr. J.H.J.M. van Krieken,
volgens besluit van het college voor promoties
in het openbaar te verdedigen op

Dinsdag 20 september 2022
om 12.30 uur precies

door

Jinbiao Yang
geboren op 31 december 1990
te Fuyang (China)

Promotoren:

Prof. dr. Antal van den Bosch

Copromotoren:

Dr. Stefan L. Frank

Manuscriptcommissie:

Prof. dr. Mirjam Ernestus

Prof. dr. Fermín Moscoso del Prado Martín

Prof. dr. Inbal Arnon (Hebrew University of Jerusalem)

Dr. Tianlin Wang (University at Albany, SUNY)

Dr. Lars Meyer (Max Planck Institute for Human Cognitive and Brain Sciences)

Contents

"In the matter of language, people have always been satisfied with ill-defined units."

— De Saussure, 1916/1959, p. 111

1 | General introduction

1.1. The discreteness of cognition

The world may be continuous, but our cognitive apparatus that deals with the world is at least partially discrete. George A. Miller, in his famous paper (1956), introduced the capacity limitation for processing the information as 7 ± 2 *chunks*. Though a later study (Cowan, 2001) updated the number to 4 ± 2, they did not change the key point of Miller's theory, that the unit of memory is a *chunk*. A chunk may consist of many smaller units, but it can be recognized as a whole pattern. For example, a chess expert can remember realistic positions of the pieces of a whole chess board in several seconds (De Groot, 1946). The expert has great experience of the chess board patterns, so she/he memorized the whole board as a few chunks or even a single chunk instead of memorizing the chess pieces one by one. Even though most people cannot memorize the chess board as one chunk, everyone can recognize portions of the world by a few chunks. This ability is called *conceptualization*.

The ability of conceptualization is necessary to deal with the huge amount of information received from vision, hearing, and other senses in a human's daily life. In vision alone, the retina can transmit about 10 million bits to the brain per second (Koch et al., 2006). Our brains are powerful, but processing all the details of the received information is still impossible. Instead, received information will be clustered into discrete chunks that carry different concepts, such as blue sky, hot sand, and the sound of waves. The number of concepts is much smaller than the number of all received details, therefore humans are able to recognize the board world with the limited processing capacity of working memory.

The concept chunks are the entries or cues to meaning. They are basic units in our cognition. It is worth noting that *basic* does not mean *smallest*. There can be many conceptual levels: a forest consists of trees; a tree consists of a canopy, a stem, and roots; a canopy consists of leaves, branches; and so on and so forth. The large hierarchy of concepts makes it hard to define the lowest level of concepts, so it is not possible to construct concepts from the *smallest* units. *Basic* units, instead, refer to the units that take priority during understanding, and the priority depends on the interpreter's experience. For example, most people will recognize a cat as *cat* rather than the superordinate concept *mammal* or the subordinate concept *Russian Blue*, and a cat owner may move their prior level to different breeds of cats(Hajibayova, 2013). Owing to the priority of the basic units in conceptual levels, people can reduce the complicated hierarchy to a single concept. In this thesis, I investigate the basic units of language comprehension.

1.2. The discreteness of language cognition

Language is a part of the human world. So as in the cognition of the world, discreteness can also be seen in the cognition apparatus dealing with language. The world may be continuous, but language is intrinsically discrete since it evolved from human's discrete cognitive actions rather than the natural environment. Units in languages include phonemes, gestures, signs, and other symbols. If these simple symbols are used alone, their expressive capabilities are far from adequate for the current communication of people. However, as Wilhelm von Humboldt summarized language as a system which "makes infinite use of finite means" (translated and quoted by Noam Chomsky (2016)), humans invent different syntactic roles for symbols so the symbols can be combined to form a short sentence, and a sentence can nest inside another sentence to generate a longer sentence. The recursion of language allows language users to build infinite expressions from an initially limited set of concepts. In other words, not only is language inherently discrete, the evolution of language has also taken advantage of its discrete nature.

The discrete nature of language is apparent in its spoken and written form. Sentences and their internal clauses are usually separated by speech pauses, and the voice stress given to certain syllables can indicate the separations between words, clauses, or other linguistic units. Almost all current writing systems have punctuations and delimiters (such as spaces). Punctuations sometimes help to disambiguate the meaning of the text. For example, in case we remove punctuations from "*Most of the time, travellers worry about their luggage.*", the sentence "*Most of the time travellers worry about their luggage*" turns to be Sci-Fi. Also, current European and Arabic writing systems usually put spaces in the text to indicate the boundaries of words. The spaces play a similar role of disambiguation as punctuations. With punctuations, spaces, and other dividers, language users can distinguish different entities and understand the materials more easily.

However, the discreteness in language forms differs from the discreteness in language cognition. The dividers in the form are explicit and perceptible. But a lot of discreteness in language cognition is implicit and can be recognized only by experience. Syllable stresses do not mark all the separations between linguistic units. A robot voice may even lack stress cues and become monotonous, while people can still understand it without much effort. The written styles of Ancient Greek and Classical Latin, scriptio continua or continuous script (Nagy, 2009, p. 5; Rhodes, 2013), do not use dividers like space or punctuation. Classical Chinese text has a similar style, and was used until about one hundred years ago. Modern Chinese added punctuations, but still has no space between words.

In sum, perceptual dividers may facilitate understanding, but they only partially mark the discrete units of language cognition. Therefore, the discreteness of language cognition cannot be deduced merely from the perceptual cues, which leaves the open challenge of discovering the building blocks in language cognition. These building blocks are the genuine language units that we memorize, perceive, and produce in our daily life. I will name them *cognitive units* below.

1.3. From linguistic units to cognitive units

Ferdinand de Saussure, in his famous edited works *Course in General Linguistics,* indicated the importance of units in language science:

"... just as the game of chess is entirely in the combination of the different chess pieces, language is characterized as a system based entirely on the opposition of its concrete units. We can neither dispense with becoming acquainted with them nor take a single step without coming back to them" (De Saussure, 1916/1959, p. 107).

Various branches of linguistics take some form of *units* as the basis of their theoretical frameworks. In the following paragraphs, I will go through different types of linguistic units that researchers have used and discuss their relations with our desired cognitive units.

Although Saussure admitted the difficulty of seizing language units in practice, he said the word "bears a rough resemblance to the unit and has the advantage of being concrete" (Saussure 1959, page 113). Indeed, words are the most generally accepted units of language, either in daily use, such as dictionary lookup and word count, in linguistics, such as sentence parsing (Chomsky, 1957), in computational linguistics, such as word embedding (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013; Pennington, Socher, & Manning, 2014), or in psycholinguistics, such as lexical decision task (Meyer & Schvaneveldt, 1971). The reason behind the wide acceptance of word units is as Saussure said: they are *concrete*. The spaces and punctuations make words conspicuous in most texts, so it is easy for the reader to locate them as units.

However, words are not always so conspicuous. For example, fluent speech smooths the pronunciation of neighboring words into a continuum, rather than marking the pronunciation of each word separately; and as I have mentioned in Section 1.2, some writing systems lack perceptual word dividers, so even the definitions of *words* in those systems remain ambiguous. Moreover, we cannot say that words are cognitive units even in the writing systems which have conspicuous words. For example, infrequent morphologically complex words (e.g., *allyship*) or long words (e.g., *pseudohypoparathyroidism*) should probably be decomposed during understanding because of their unfamiliarity

and composition into meaningful parts. For those cases, morphemes, that are defined as the linguistic form which bears no partial phonetic-semantic resemblance (Bloomfield, 1933, p. 161) or the minimal meaning-bearing units in a language (Chomsky, 1953), may serve as the better candidates for cognitive units. But if morphemes serve as the only units in cognition, the number of tokens would increase (as would the number of operations required), and many opaque compounds (e.g., *strawman*) would not be understood as a whole.

Still, words and morphemes are the prevalent units in mainstream linguistics. It may be because they are all defined as the *minimal* units though under different criteria: words are the minimal units in most syntactic theories or the minimal units between two spaces in most European texts, and morphemes are the minimal units bearing meaning. Taking only the *minimal* units in language formalization is a type of reductionism, and the reductive approach can simplify formalization. For example, words and morphemes are more distinguishable than phrases and clauses, and words and morphemes are also of fewer types than phrases and clauses, so the units based on *words + morphemes* require simpler definitions and rules in formalization.

However, the advantage in formalization does not mean these units alone can describe cognition. For instance, there are a lot of idioms of which the figurative meanings are different from their literal meanings. The figurative meanings of idioms must be assessed holistically, and that means idioms should be included as cognitive units. When Chomsky attempted to formalize the language systems into *generative grammar*, he considered only grammar, which is the set of rules composing the minimal units into sentences, and regarded meaning and pragmatics as secondary (Barman, 2012). His approach simplified his formal system of language and facilitated the analysis of "regular" language, but this system treats idioms as irregular exceptions (Almohizea, 2016). While generative grammar became mainstream linguistics from the middle of the last century, its neglect of idioms and other shortcomings kept drawing criticisms. Some of these criticisms spawned *usage-based linguistics*, which regards all linguistic knowledge emerged from embodied experience and language use in the real world. Usage-based

linguistics naturally allows idioms to be cognitive units, since idioms are observable products of language use.

Usage-based linguistics softens the traditional, rigid definitions of language units and makes the language units flexible in terms of sizes and structures. Given this, one might assert that usage-based linguistics should be able to describe cognitive units better than formal linguistics in the style of Chomsky. Having said that, usage-based linguistics is not developed in a unified frame, but is developed in parallel theories, such as Cognitive Grammar (R. W. Langacker, 1987), Cognitive Construction Grammar (Lakoff, 1987, p. 467; Goldberg, 1995; Goldberg, 2006), and Radical Construction Grammar (Croft, 2001). But at least, they share the same terminology *construction* to describe the language units.

We may wonder whether constructions are our expected cognitive units. In general, construction means form-meaning or form-function pairing, but its definition has been evolving several times. In the early days of usage-based linguistics, Lakoff argued in "Linguistic Gestalts" (1977) that the meaning of an expression may be more than the additive meanings of its parts, and Langacker said a construction contains two or more symbolic components (1987, p. 82). They considered constructions to be *complex* structures and serve as supplements to the minimal units. However, since the 1990s, most researchers in usage-based linguistics, such as Goldberg (1995; 2006; 2013), Croft and Cruse (2004), and even Langacker himself (2017), used the notion of construction also for minimal units. This extension integrated morphemes, words, idioms, etc., which used to be described at different linguistic levels, into a neat concept.

Another conceptual change of construction is on the constraint of "unpredictability" or "non-decomposability". Lakoff and Goldberg's Cognitive Construction Grammar is the most popular genre in construction grammar, and its timeline illustrates this conceptual change. According to Goldberg in her first book (1995, p. 4), constructions should not be strictly predictable from their component parts; in a later book (2006, p. 5), she privileged a condition: "patterns are stored as constructions even if they are fully predictable as long as they occur with sufficient frequency". In a more recent book chapter (2013), she loosened the constraint further and refined the definition of construction

as “conventional, learned form-function pairings at varying levels of complexity and abstraction”. The cancellation of “unpredictability”, together with the cancellation of “complexity”, allows *any* expression, such as “-ly”, “dog”, “I don’t know”, or “he is”, to be the constructions.

There is abundant evidence from psycholinguistic research showing that the units in language cognition indeed span from morphemes (Fiorentino, Naito-Billen, Bost, & Fund-Reznicek, 2014; Kaczer, Timmer, Bavassi, & Schiller, 2015; Koester & Schiller, 2011; Leminen, Smolka, Duñabeitia, & Pliatsikas, 2019; MacGregor & Shtyrov, 2013) to multi-word expressions (Arnon & Snider, 2010; Bannard & Matthews, 2008; Jackendoff, 2002). It seems, therefore, that the term *constructions*, in linguistics, approximates the cognitive units I talk about in this thesis. However, the definition of constructions still does not fully satisfy what I want of cognitive units, because the constraint “conventional, learned” in the new definition of construction (Goldberg, 2013) creates a circular argument if we ask for a discriminative statement rather than a descriptive statement. A common approach to get around this plight is to interpret “conventional, learned” as a matter of concrete frequency. But we should note that the frequency effect cannot explain everything about cognitive units: as an instance for low frequency, people may immediately memorize the name of a new friend; as an instance for high frequency, people may forget something that was once familiar.

In summary, the word, the morpheme, the idiom, as well as other classical linguistic units, only describe part of cognitive units; the construction, as a type of usage-based linguistic unit, represents the notion of cognitive unit best among the linguistic units, but still not perfectly. It remains necessary to study cognitive units from a variety of perspectives.

1.4. The research questions of cognitive units

I have argued that cognitive units are rooted in language usage in the above section, but also argued that the existing concepts in current usage-based linguistics failed to give

a formal, definite description of cognitive units. To study what cognitive units are, we should examine the reason for that failure.

The linguistic units assumed by current formal theories, e.g., words and morphemes, lack flexibility, while they are well-defined and can be easily identified in any language material. In contrast, usage-based linguistics discards some of the rigidity of formal linguistics. But units in usage-based linguistic theories are not defined consistently yet, and outsider adopters from e.g. psycholinguistics or computational linguistics cannot easily decide on a *suitable* definition given the different variants of theories. Furthermore, even if we had decided on a definition, the definition is still ambiguous for identifying the units in given language material: The current typical solution for usage-based linguists is to handcraft a vocabulary with linguistic intuition, but this is not a generic solution for defining the units in a general cognitive model. For these reasons, even psycholinguistic studies, which require cognitive reality and plausibility, still often adopt traditional linguistic units in their analyses.

In the current thesis, I will not dig into the divergence between different linguistic theories. Rather, I will attempt to discover cognitive units, adopting the flexibility of the usage-based view. I am interested in formalizing the underlying principles that are analogous to principles allowing humans to learn and use cognitive units. More specifically, I will explore three questions about cognitive units:

1. **What will be learned as cognitive units during language acquisition?**

   Traditional studies of language acquisition usually talk about learning words. Yet, words may not be the units of learning; perhaps the learning of cognitive units should be the more generic research goal. Cognitive units may be very different language chunks, but not just any language chunk is a Cognitive unit (otherwise, the mental lexicon would be unlimited). The flexibility of cognitive units implies that humans are able to learn whether an arbitrary chunk in the language input is a Cognitive unit without prior knowledge.

   Accordingly, I propose the first research question on cognitive units: what will be learned as cognitive units during language acquisition? In Section 1.3, I have mentioned that the definition of construction "conventional, learned form-function

pairings at varying levels of complexity and abstraction" (Goldberg, 2013) closely represents cognitive units. But the prerequisite "*conventional or learned*" in the definition is too vague since it does not answer anything about what will be learned. This issue leads to the requirement of a formal definition of cognitive units.

Furthermore, the above question implies *how to learn them* because cognitive units are the results of learning. The two questions are two sides of the same coin, but it is also interesting to investigate more specific aspects of the learning.

2. **How to segment language input into cognitive units?**

   We can hardly read a sentence instantly. Instead, we segment the sentence into smaller cognitive units. A similar effort is sentence parsing, which has been well-studied. However, it is different from cognitive units segmentation, because sentence parsing analyzes the structure of complete sentences, while cognitive units segmentation may operate on partial sentences as sentences are incrementally processed. In other words, sentence parsing in the end requires the input to be fully available, after which the parse can be established post-hoc, while cognitive units segmentation should be modeled as an online task that does not require the input to be finished or pre-segmented.

   Moreover, cognitive units may vary in length or form, and they are not always highlighted by perceptual cues in the language input. In order to understand continuous language, readers must rely on their experience to segment the language input into cognitive units. In other words, the cognitive function for Cognitive unit segmentation in the language input is based on the mental lexicon that stores learned cognitive units.

   In this thesis, I will investigate and model the mechanism of cognitive units segmentation. The answer would not only contribute as a psycholinguistic theory, but also provide a formal solution for sentence segmentation in usage-based linguistics.

3. **How to process segmented cognitive units?**

   A natural follow-up to the cognitive unit segmentation is the processing of the segmented result. Through the processing of the cognitive units and their relations, language input can be fully understood. However, the investigation of this process is very complicated since it involves many subprocesses (e.g., semantic access and syntactic composition of these units) and the interaction between them. In this thesis, I attempt to develop a concise framework to describe the processing of cognitive units. More specifically, I will use the mechanism for scheduling cognitive units of various grain sizes as the entry point for this framework.

   If only one Cognitive unit is segmented at each time step, there is no question of which one to process. However, more cognitive units could be segmented at the same time in the case that these units form a hierarchical tree. For example, if *mail*, *box*, and *mail box* are all stored in the mental lexicon, the whole form and its components may all be segmented and activated. The multiple candidates complicate subsequent processing: would the reader process the cognitive units at different levels simultaneously, process only some of them, or process some of them in advance? This is precisely the scheduling question asked in this section.

1.5. The methodologies

Language is the product of human cognition. However, traditional studies of language usually start from language *per se*. Chomsky, who claims linguistics is a branch of cognitive psychology (2006, p. 1), built his theories within the scope of language structure, using as the cornerstone of his theory a controversial hypothesis -- the innate grammar ability in the human brain. Even the schools of usage-based linguistics, which do not use Chomsky's hypothesis and ground themselves in cognitive plausibility (which is why they are also commonly referred to *cognitive linguistics*), focus on the language phenomena to prove their consistency and completeness rather than collecting cognitive evidence to prove their cognitive reality.

In investigating our research questions, I will start from psycholinguistic methods, i.e., designing experiments to collect human behavioral or neural responses to language material. By doing so, we may obtain the empirical positive or negative evidence for our hypotheses on cognitive units, and eventually, we may infer some key mechanisms of language cognition that deal with these units. Specifically, this thesis will involve the following experimental techniques:

1. Behavioral approaches

   The philosophy behind behavioral experiments is that people's behaviors reflect the underlying cognitive procedure. Not any cognitive procedure will result in overt or measurable behavior, but we may be able to design a targeted task to force the cognitive procedure to show a behavior. The **Lexical decision task (LDT)** is such a task in psycholinguistics. In detail, LDT measures how quickly the experiment participants classify the designed stimuli as words or non-words. By comparing the reaction time between two types of stimuli that are different on a linguistic attribute X, we can estimate how X affects the processing time.

   LDT requires reporting the decision of each stimulus, so the reading should be interrupted often. In contrast, **eye-tracking** can collect data during natural reading. This technique measures people's eye movement, so it requires no extra response, such as in LDT, from the participants. An eye fixation usually reflects attention at the fixation point. Using eye tracking instruments, researchers can record and analyze the latencies, durations, and locations of fixations while reading, to understand when, to what extent, and where the language content is processed.

2. Neuroimaging

   Cognitive procedures may not always result in overt behaviors, but they are certainly reflected by neural activities. Neuroimaging, therefore, can provide more detail of our interested cognition procedure. Some imaging techniques, such as functional magnetic resonance imaging (fMRI) and

positron emission tomography (PET), have high spatial resolution and facilitate the discovery of the link between brain areas to cognitive functions. Some imaging techniques, such as **electroencephalography (EEG)** and magnetoencephalography (MEG), have high temporal resolution and facilitate the discovery of the temporal dynamics of cognitive functions.

The studies of this thesis will use EEG since the research questions care about the temporal dynamics more than the brain areas of Cognitive unit processing. The electrical activity in response to the processing will conduct to the scalp, and EEG can record the electrical activity by multiple electrodes attached to the scalp. Even though the EEG signal reflects the brain activity with extremely short lag, which is much better than the behavioral responses that also involve the time for motor planning and muscle movement, we should notice that EEG cannot reflect all cognitive activities and the EEG signal is complicated for analysis. Thus, EEG is a complement, not a replacement, to the behavioral techniques.

In addition to the psycholinguistic methods, which mainly provide scattered evidence for building our hypothesis of the processing mechanism related to cognitive units, I will also build a learning **computational model** to formalize the hypothesized mechanism. Within computational models that learn from data, there are two main branches: *supervised* models and *unsupervised* models. *Supervised* means the data to feed into the model are labeled as ground truth. Ground truths are taught by others. In human learning, supervised ground-truth learning is rare, and most learning is from our own experience of the world. Learning in the absence of labeled ground truth is called *unsupervised* learning.

In case we had the ground truth of cognitive units, a supervised computational model could fit itself to the data and then predict the cognitive units in new cases. In reality, we lack a ground truth. More importantly, children learning a language have no access to the ground truth either. Only unsupervised computational models, therefore, are acceptable for cognitive units discovery. Though an unsupervised computational model does not need labeled data as ground truth, it requires the learning target to guide the

discovery from unlabeled data. In order to make the model cognitively plausible, I will build on a number of general cognitive principles to design the learning target.

An unsupervised model does not require ground truth to fit. However, it also means that it is difficult to know whether the unsupervised model reflects cognitive reality if it is not tested on empirical evidence. In order to evaluate the cognitive reality of our unsupervised model, we will let the model do a psycholinguistic task that the model should not have knowledge of. In our case, the task will be to predict eye fixations during reading since our unsupervised model does not aim to simulate the eye movements of humans, nor it is trained on eye-fixation data. If the model succeeds, we could say the model does conform to the cognitive reality of reading.

1.6. The roadmap of the thesis

This thesis presents the discovery of the processing units in language cognition. The discovery contains two parts: the first part (Chapter 2, 3, and 4) describes laboratory studies that provide empirical evidence to a theoretical model of the processing of cognitive units; the second part (Chapter 5 and 6) introduces a computational model, that formalizes the learning and processing of cognitive units. The computational model is based on the theoretical model.

In Chapter 2, I will introduce a toolbox, *EasyEEG*, which was developed for our subsequent analysis of EEG data. EasyEEG includes several multivariate pattern analysis (MVPA) methods to precisely examine the temporal dynamics of our interested cognitive mechanisms. The MVPA methods require no background knowledge of the ERP components and the sensor locations. This advantage is critical to the studies in the next two chapters since the studies about cognitive units are exploratory, which lacks the background knowledge. In Chapter 3 and 4, I will present the behavioral and EEG evidence for our hypotheses about cognitive units, and propose a two-stage workflow to describe the processing of cognitive units.

In Chapter 5, I will present an unsupervised computational model, *Less-is-Better* (*LiB*). This model can segment any given text into sequences of units. The plausibility of

the model-derived units as *cognitive units* should be tested on a realistic cognitive task namely, by the task related to empirical human behavior. Therefore, in Chapter 6, I will present the usage of the LiB model to predict eye fixations during reading under the hypothesis that reading units are cognitive units.

In the final chapter, I will summarize what I have found about cognitive units, discuss the implications of the findings and the theoretical connections to other domains, and propose some future work.

Part I.

Part One: Laboratory Experiments

2 | Group-Level Multivariate Analysis in EasyEEG Toolbox: Examining the Temporal Dynamics using Topographic Responses[1]

Abstract

Electroencephalography (EEG) provides high temporal resolution cognitive information from non-invasive recordings. However, one of the common practices – using a subset of sensors in ERP analysis makes it difficult to provide holistic and precise dynamic results. Selecting or grouping subsets of sensors may also be subject to selection bias and multiple comparisons, and is further complicated by individual differences in the group-level analysis. More importantly, changes in neural generators and variations in response magnitude from the same neural sources are difficult to separate, which limits the capacity for testing different aspects of cognitive hypotheses. We introduce EasyEEG, a toolbox that includes several multivariate analysis methods to directly test cognitive hypotheses based on topographic responses that include data from all sensors. These multivariate methods can investigate effects in the dimensions of response magnitude and topographic patterns separately using data in the sensor space, and therefore enables assessing neural response dynamics. The concise workflow and the modular design provide user-friendly and programmer-friendly features. Users of all levels can benefit from the open-source, free EasyEEG toolbox to obtain a straightforward solution for efficient processing of EEG data and a complete pipeline from raw data to final results for publication.

[1]Yang, J., Zhu, H., & Tian, X. (2018). Group-Level Multivariate Analysis in EasyEEG Toolbox: Examining the Temporal Dynamics Using Topographic Responses. *Frontiers in Neuroscience, 12*, 468.

2.1. Introduction

Electroencephalography (EEG) is a suitable non-invasive measure for investigating the temporal dynamics of mental processing because of its high temporal resolution and cost-effectiveness. The event-related potential (ERP) is the most common way to reflect neural response dynamics in the temporal domain. However, ERP analyses are mostly based on responses in individual sensors or an average of a group of selected sensors. This "selecting sensors" analysis method is not optimal, because it faces various challenges (Tian & Huber, 2008; Tian, Poeppel, & Huber, 2011). For example, only relying on data in a few sensors cannot easily differentiate between changes in the distribution of neural sources versus changes in the magnitude of neural sources. Moreover, selecting sensors may introduce subjective bias during the selection processes, and sometimes data in different sensors may derive inconsistent or even contradicting results. Unless all possible sensor selections have been tested, readers will not know whether the reported effects are robust across sensors or sensor groups. Running statistical tests among multiple (groups of) sensors is subject to multiple comparisons, and hence increases the risk of type I error (false positives) or type II error (false negatives that could be induced by correction methods). Furthermore, the ERP analysis heavily depends on identifying ERP components. However, data in a few sensors cannot fully represent the spatial and temporal features of components, which makes the estimation of components' response magnitude and latency difficult and incomplete. Last, individual differences in spatial and temporal characteristics caused by anatomical and functional differences across subjects further complicate the analysis, which makes group-level analysis even more opaque. Therefore, most of the time, it is difficult to get a precise and holistic view of temporal dynamics by using "selected sensors" in ERP analyses.

These problems may be solvable by using information from all available sensors. Two approaches can be taken. The first one is to localize neural sources by projecting all sensors' information back to the source space (source localization). The advantage is that additional information about source spatial distribution can be estimated together with their temporal dynamics. Numerous source localization methods, such as dipole modeling, sLoreta(Pascual-Marqui, 2007), Beamforming (Van Veen & Buckley, 1988), and MNE(Grech et al., 2008), have been proposed and built in software packages such as BESA, EEGLab, Brainstorm, NutMEG, SPM, Fieldtrip, MNE-Python. However, source localization is an ill-posed problem – an infinite number of solutions can be obtained from the mixture of recordings. Therefore, many assumptions have to be met and sophisticated procedures and careful manipulation have to be followed in order to obtain meaningful source localization results. Moreover, these localization methods work best with magnetoencephalography (MEG) that has better spatial resolution. EEG signals, in contrast, are highly distorted by the skull. High-density EEG systems and realistic head models that are estimated by individual anatomical MRI scans are required to achieve

acceptable results of EEG source localization. However, these high-cost systems and MRI scans may be not feasible for many researchers.

The second approach is to work with all "raw" data in the sensor space. Compared with methods with dependent variables from individual sensors or averages of selected sensors, this approach that relies on information from multiple sensors is called multivariate analysis. Basically, multivariate analysis in EEG uses the topographical patterns of sensors, and tries to differentiate response patterns among conditions at each given time point. If differences, either in response magnitude, topographic patterns, or latency were detected across a timespan, we can infer that different mental processes and their temporal dynamics under distinct conditions. This multivariate approach aims to directly test cognitive hypotheses by using data in all sensors (Tian & Huber, 2008; Tian et al., 2011) and by-passing source localization in case the location information of cortical activities was not the primary research interest of the study. Note that performing the source localization by solving the inverse problem is the only way in EEG and MEG studies to directly address questions regarding location at the brain level. Scalp data and topographic patterns reflect the response dynamics at the sensor level and can be used as indicators of modulation by experimental manipulation.

In this paper, we introduce the EasyEEG toolbox (`https://github.com/ray306/EasyEEG`), in which several multivariate analyses are included for processing EEG sensor data and testing cognitive hypotheses. To our knowledge, a few EEG analysis software packages (Delorme et al., 2011; Gramfort et al., 2013; Groppe, Urbach, & Kutas, 2011; Pernet, Chauveau, Gaspar, & Rousselet, 2011) have already included several multivariate analysis methods for data in the sensor space. For example, LIMO EEG (Pernet et al., 2011) aims to test the effects at all sensors and all time points by a set of statistical tools such as ANOVA, ANCOVA and Hierarchical General Linear Models along with multiple comparison corrections; Mass Univariate ERP Toolbox (Groppe et al., 2011) applies univariate tests (e.g., t-test) in each sensor over time points with multiple comparison correction; the Donders Machine Learning Toolbox [2] supports single-trial analysis with several machine learning methods built in, and MNE-Python (Gramfort et al., 2013) makes use of a machine learn package named Scikit-Learn (Pedregosa et al., 2011) to see the decoding performance over the temporal or spatial domain. Those toolboxes and the new toolbox EasyEEG share the same goal which is to investigate the temporal neural dynamics using all data in all sensors. The uniqueness of EasyEEG toolbox is that the included multivariate methods are carried out over the explicit measures that reflect the topographic patterns across all sensors. It offers a straightforward and intuitive approach to efficiently test cognitive hypotheses.

The design principle of this toolbox is to be both user friendly and programmer friendly. We therefore separated the procedure of EEG data analysis into several steps, and made each step into an independent module with concise input/output interfaces.

[2] `https://github.com/bahramisharif/DMLT`

In each module, common important but tedious operations that involve complicated programming details have been encapsulated into several simple commands. Various multivariate group analysis methods have been built in with single-line commands. Users simply need a descriptive dictionary to snip the data and one line of concatenated commands to perform all analyses and visualize the results. After knowing only a few commands, all users, regardless of programming experience, could start their analysis within a few minutes. Moreover, the open-source nature of this toolbox enables and supports users to add more algorithms for the EEG data analysis. EasyEEG has encapsulated many APIs for the programmers. Researchers who want to introduce a new analysis method need only pay attention to the core logic of that method, but leave out the trivial details, such as reshaping data and plotting, from the programming. And even for deep learning applications to EEG data, EasyEEG also provides a concise interface. In general, it offers a clear way to perform group level statistical tests to directly investigate cognitive hypotheses. We introduce how to use this package in the next section.

2.2. Workflow and Methods

The general analysis workflow in EasyEEG involves four stages:

1. Import the preprocessed data. EasyEEG currently (v.0.8.3) supports epoched data generated from MNE and EEGLAB;

2. Define a dictionary (a Python syntax) to describe the analysis target (e.g., conditions, sensors, temporal durations, and/or any comparison between two groups), then extract the data by a function "extract()" with the definition as the parameter;

3. Apply one of four analysis functions (e.g. `tanova()`) introduced in this paper. For algorithms that require long processing time, the computation process can be seen in a progress bar showing used time and estimated remaining time to finish; The computation function will yield a special data structure named *AnalyzedData*;

4. Visualize and output the results. *AnalyzedData* includes the analysis name (in the *analysis_name* attribute), the analysis result (in the *data, annotation*, or *supplement* attribute), and the parameters for visualization (in the *default_plot_params* attribute). Researchers cannot only examine the *p*-values or other information, but also customize the visualization parameters for different figures.

More details can be found in EasyEEG's online documentation (`http://easyeeg.readthedocs.io/en/latest/`).

We introduce a procedure that includes four multivariate methods for testing cognitive hypotheses using information in topographic patterns. An open dataset of face perception (Wakeman & Henson, 2015) is used to demonstrate this procedure and methods.

The first two methods are to combine univariate approaches with topographic information to estimate the spatial extent of experimental effects (*distribution of significant sensors*) and the overall temporal dynamics of experimental effects (dynamics of global field power, *GFP*). These analyses can make the connection with the common practice of ERP analysis. The next two methods are to implement multivariate analyses, introducing in this paper *topographic analysis of variance* (*TANOVA*) and *pattern classification* that takes holistic topographic information into account to perform group-level statistics and investigate the dynamics of response patterns.

1. Distribution of significant sensors

The spatial extent of experimental effects can be estimated by the number and distribution of sensors that are significantly different between conditions. This analysis is done by performing statistical tests, such as paired t-test, on response amplitudes between two conditions in each sensor at all given time points or windows, and counting the number of the sensors that have significant results. In this way, we can quantify the spatial difference in terms of response amplitude between two topographies. By examining differences across timepoints, we can estimate the temporal dynamics of underlying neural processes that are reflected in topographies.

2. Dynamics of global field power (GFP)

Global field power (GFP) was introduced by Lehmann and Skrandies (Lehmann & Skrandies, 1980). It is calculated with the following equations:

$$\begin{aligned} \mathrm{GFP}_t &= \sqrt{\tfrac{1}{n} \cdot \textstyle\sum_{i=1}^{n} u_i^2} \\ u_i &= U_i - \bar{u} \\ \bar{u} &= \tfrac{1}{n} \cdot \textstyle\sum_{i=1}^{n} U_i \end{aligned}$$

where t is the target timepoint n is the number of sensors in the montage; U_i is the measured potential of the ith senosr (for a given condition at a given time point t); and u_i is the average-referenced potential of the ith electrode.

Basically, GFP is a summary statistics of response magnitude from all sensors on a topography, which is in the form of variance of response magnitude and mathematically equals the root mean square (RMS) of all mean-referenced sensor values. GFP reflects the overall energy fluctuation of distributed electric potentials across all sensors at a specific time point. Therefore, it is a good way to summarize and visualize the temporal dynamics of the whole brain activity. Nevertheless, researchers need to be cautious that the essence of GFP is a non-linear transformation. Therefore, when researchers apply GFP to group-averaged ERP, the outcome is not the same as the average of individual GFPs. Variances between subjects have a major effect on group-averaged GFP.

The group-level statistical analysis of GFP can be addressed by many common approaches (time point by time point, area measures, peak measures etc.). We provide the time-point-wise approach in EasyEEG. For comparison between any two conditions, we take every subject's data from every temporal window with defined duration of interest from both conditions and apply a paired t-test. Thus we get the *p*-value that indicates the level of significance across all sensors in successive temporal windows.

3. Topographic Analysis of Variance (TANOVA)

Topographies reflect underlying neural processes. Comparing pattern similarity between topographies in different conditions can reveal distinct mental processes and hence directly test cognitive hypotheses. TANOVA is a statistical analysis on a measure of similarity between topographies. This topographic similarity measure, called "angle measure" (Tian & Huber, 2008), quantifies the topographic pattern similarity by a high-dimensional angle between two topographies. More specifically, the multivariate topographic patterns across all sensors are represented in high-dimensional vectors $\overrightarrow{A}$ and $\overrightarrow{B}$ for two conditions A and B, where the number of dimensions is the number of sensors. The topographic similarity between the two conditions is quantified by the cosine value of the angle θ that can be obtained by the following equation:

$$\cos(\theta) = \frac{\overrightarrow{\mathrm{A}} \cdot \overrightarrow{\mathrm{B}}}{|\overrightarrow{\mathrm{A}}||\overrightarrow{\mathrm{B}}|}$$

The cosine value is an index of spatial similarity between two conditions, where a value of 1 represents identical patterns and a value of -1 represents exactly opposite patterns. Moreover, because this index is normalized by the response magnitude of both conditions, it has the advantage that it is unaffected by the magnitude of responses.

The statistical analysis of the "angle measure" is a non-parametric statistical test, termed topographic analysis of variance (TANOVA) (Brunet, Murray, & Michel, 2011; Murray, Brunet, & Michel, 2008). The critical step in TANOVA is to generate a null distribution. In EasyEEG (0.8.4.1), we provided three different strategies to generate the null distribution of the angle measure cosine values.

Strategy 1:

1. Put all subjects' data into one pool regardless of experimental conditions.
2. Shuffle the pool and randomly re-assign the condition label for each trial (data permutation).
3. Calculate the group-averaged ERPs for each new labeled condition.
4. Calculate the topographic similarity angle measure (cosine value of angle θ) between the new group-averaged ERPs.

5. Repeat the former steps (1)-(4) 1000 times (suggested by Manly, 2006).

Strategy 2:

1. Perform data permutation within subject. That is, shuffle and re-label the trials for each subject.
2. Calculate the group averaged ERPs for each new labeled condition.
3. Calculate the topographic similarity angle measure (cosine value of angle θ) between the new group-averaged ERPs.
4. Repeat the former steps (1)-(3) 1000 times.

Strategy 3:

1. Calculate the ERPs for each condition and subject.
2. Perform data permutation within subject for ERPs. That is, re-label the ERPs for each subject.
3. Calculate the spatial topographic similarity angle measure (cosine value of angle θ) between the new group-averaged ERPs.
4. Repeat the former steps (1)-(3) 1000 times.

Strategy 1 is used by many researchers (Brunet et al., 2011; Lange, Perret, & Laganaro, 2015; Murray et al., 2008). However, it loses subject's information by mixing all subjects' data into one pool. In contrast, Strategy 2 permutes the data at the within-subject level. Both Strategies 1 & 2 may be time-consuming and computational demanding. Therefore, Strategy 3 has the advantage of reducing computing complexity and processing duration (can be done within 1-2 minutes). However, Strategy 3 also has the limitation that it loses trial information by averaging trials at the first step. Regardless of the procedure, we find that the results from three strategies are similar and stable when the repetition times are beyond 1000 times (see details in the next section). Thus we suggest that Strategy 3 can be used as a pilot test to have a quick check of results, and Strategy 2 for further validation.

After determining the null distribution, a comparison is made between the actual topographic similarity angle measure and the null distribution. The *p*-value is determined by finding the rank position of that actual cosine value in the generated null distribution. It reveals how significant the similarity between two topographic response patterns in different conditions are in a chosen time window.

4. Pattern classification

Although TANOVA is good at measuring the significance of the topographic variance at a given moment, it is insensitive to the fluctuation over time. We introduce a pattern classification method in EasyEEG to capture topographic dynamics. Moreover, pattern classification can collaboratively take advantage of all aspects of information in topographies, compared with GFP and TANOVA that only emphasize response magnitude and energy distribution, respectively.

This pattern classification method is in the framework of supervised machine learning. The collection of magnitudes of all sensors at a time point composes a sample, and the corresponding condition category is the label of the sample. After a classifier is trained by mapping the samples in a dataset to their labels, the classifier is used to infer the labels of samples in a new dataset for testing.

The pattern classification method aims to obtain topographic differences among conditions at all timepoints to reveal the topographic changes over time. The general procedure works as follows:

1. Data in each condition in a specific time point or window are extracted to form a sample. Samples in the time points or windows of interest from two conditions form a dataset for each subject.

2. The pattern classification is done separately for each subject, so that we can obtain the classification results of all subjects at a given time point or window.

    a) Each dataset is divided into a training set and a test set. The samples in the training set are used to train the classifier, and then the samples in the test set are used to evaluate the trained classifier (get a classification score).

    b) Repeat step 2a for all time points and average the scores.

    c) Repeat step 2a and step 2b for each subject.

3. Compare the classification scores of all subjects with the chance level 0.5 for a classification (where the two alternatives are equally frequent) with the permutation test (Pitman, 1937). the p-value can be obtained to indicate whether topographies in two conditions are significantly different at a given time point or window.

4. Repeat the steps 2 and 3 at successive time points or windows, so that dynamics across time can be obtained.

Any supervised machine learning model can be used as a classifier. One should notice, however, that the classifier model determines the capacity of inferring the functional relationship between samples and their labels. The biggest issue for discovering the relationship is the number of available trials in the EEG data. In general, an EEG

experiment generates fewer than hundreds of trials per subject. If we attempt to infer a complex functional relationship from only a few hundred samples, the result can hardly generalize to other samples (the problem of "overfitting"). One solution is to keep the balance between the trial counts and the complexity of the functional relationship. For example, Logistics Regression (Cox, 1958) is a linear model, which can provide a simple functional relationship without much tuning of hyperparameters. We adopted the Logistics Regression algorithm as the default classifier model. Depending on different situations and needs, users can easily switch to other supervised machine learning algorithms such as Naive Bayes or Support Vector Machine in EasyEEG. Because sometimes the sample size in two labels might be unbalanced, we adopted Area Under Curve (AUC) as the classification score (King et al., 2013). To make the classification score more robust, the algorithm will be applied several times to different partitions of the samples (Cross Validation; Arlot and Celisse, 2010).

The simple classifier models can reduce overfitting, but the functional relationship they are able to catch may also be too simple to represent the real relationship. That is, some complicated topographic pattern differences won't be recognized by the model (the problem of "under-fitting"). The solution for under-fitting is to increase the complexity of classifier models, which tends to cause overfitting. Therefore, we need to find a fine balance using an appropriate regularization model (e.g., Krogh and Hertz, 1992; Prechelt, 1998; Hinton, Srivastava, Krizhevsky, Sutskever, and Salakhutdinov, 2012) or a special deep model that is designed for few samples (e.g., Kimura, Ghahramani, Takeuchi, Iwata, and Ueda, 2018). Should one need to customize, all these extra optimizations can be easily added to the existing function by the programming interface provided in the toolbox.

2.3. Examples and Results

2.3.1. Data for this Tutorial

Data used for this tutorial are an open dataset of EEG responses to face stimuli (available at `https://openfmri.org/dataset/ds000117/`; Wakeman and Henson, 2015). The face stimuli comprise 300 grayscale photographs (half from famous people and half from non-famous people) that are matched and cropped to show only the face. In addition, there are 150 grayscale photographs of scrambled faces that are generated by taking the 2D-Fourier transform of either famous or non-famous faces, permuting the phase information, and then inverse-transforming back into the image space. Subjects were required to make a judgment about how symmetric they regard each face stimulus by pressing a key, while EEG signals were recorded. The EEG data was acquired from 16 healthy subjects at 1100 Hz sampling rate in a light magnetically shielded room using a

70-channel Easycap EEG cap (based on the EC80 system: http://www.brainlatam.com/manufacturers/easycap/ec80--185). Full details about the experimental design and data acquisition can be found in (Wakeman & Henson, 2015).

2.3.2. Processing Pipeline

All raw data were first preprocessed by MNE-Python with a standard script (see Supplementary Code Snippet 2) and saved in the ".h5" format. Epochs were chosen from −200 ms pre-stimulus to 600 ms post-stimulus onset, were baseline corrected based on the pre-stimulus period, and band-pass filtered from 0.1 to 30 Hz. Epochs that contain artifacts were excluded based on a $\pm 100 \mu V$ rejection criterion.

We demonstrate scripts for applying four analysis methods and their outcomes as follows (the entire script was running in a Jupyter notebook[3]). The runtime environment for the following examples was based on EasyEEG 0.8.4.1, Python 3.6 64 bit, Ubuntu 16.04.1.

Load data and define the analysis target

First, we define a dictionary that contains information for further analysis. The descriptive dictionary "target" is composed of two components: conditions and timepoints. To make the comparison between conditions, we add "&" between conditions as the operation symbol and use "X vs X" as the annotation. Because all analyses are based on all sensors, we do not need to define the channels. The duration of each epoch is 0-600 ms.

Code Snippet 2.1: **The data loading and analysis target definition.**

```
target = {'conditions':'S vs F:Scrambled&Famous,\
                        S vs U:Scrambled&Unfamiliar,\
                        U vs F:Unfamiliar&Famous',
          'timepoints':'0~600'}
e = epoch.extract(target)
```

EasyEEG provides a simple and easy way to complete the loading and extraction process by calling the "load_epochs()" and "extract()" functions. Data is extracted for further analysis by passing the descriptive dictionary "target" to the "extract()" function, and is saved in the variable "e".

[3] https://github.com/ray306/EasyEEG/blob/master/tests/(Demo)%20Face%20perception.ipynb

Distribution of significant sensors

By applying the function "`topography()`", we can perform the distribution of significant sensors analysis. Specifically, we define successive time windows of every 100 ms. The distribution results are saved in the variable "result". By calling the function *"plot()"*, we can visualize the results (Fig. 2.1). Sensors that show significant differences between two conditions are circled in white (Fig. 2.1a). The function "*significant_channels_count()"* can be used to more clearly illustrate the temporal dynamics by the number of significant sensors. The results are saved in the variable "sig_ch_count" and depicted in Fig. 2.1b that displays the number of significant sensors across time. The color scale represents the number of significant sensors.

Code Snippet 2.2: **Apply the *Distribution of significant sensors* analysis.**

```
# the topographies of difference
topo = e.topography(win_size='100ms')
topo.plot()

# the dynamics of the count of the significant sensors
sig_ch_count = e.significant_channels_count(win_size='5ms')
sig_ch_count.plot()
```

Figure 2.1 shows that the differences between conditions "Famous" (F) and "Scrambled" (S), as well as the difference between conditions "Unfamiliar" (U) and "Scrambled" (S), are significant in sensors above frontal, central, bilateral parietal-occipital areas. These differences start around 200 ms (180 ms in sensor count results in Fig. 2.1b). The comparison between conditions "Famous" (F) and "Unfamiliar" (U), however, only shows significant differences at the latencies of 300-400 ms and 500-600 ms. From 300 ms to 400 ms, only about 10 sensors above parietal and right-lateral occipital areas show significant differences. From 500 ms to 600 ms, around 25 sensors above middle frontal and bilateral occipital areas show significant differences. These differences are weaker compared to the comparisons between face and non-face conditions. See Supplementary Result 1 and 2 for the summary of sensor magnitude, *p*-values, and the count of significant sensors. See supplementary attachment[4] for the raw data.

GFP

The function "`GFP()`" can be used to obtain the GFP. Computation of GFP can be done within a few seconds. We set the "*compare*" parameter to be "True" to enable statistical analysis between any two conditions. With the function 'plot()', the results of GFP can be visualized.

[4]https://www.frontiersin.org/articles/10.3389/fnins.2018.00468/fullsupplementary-material

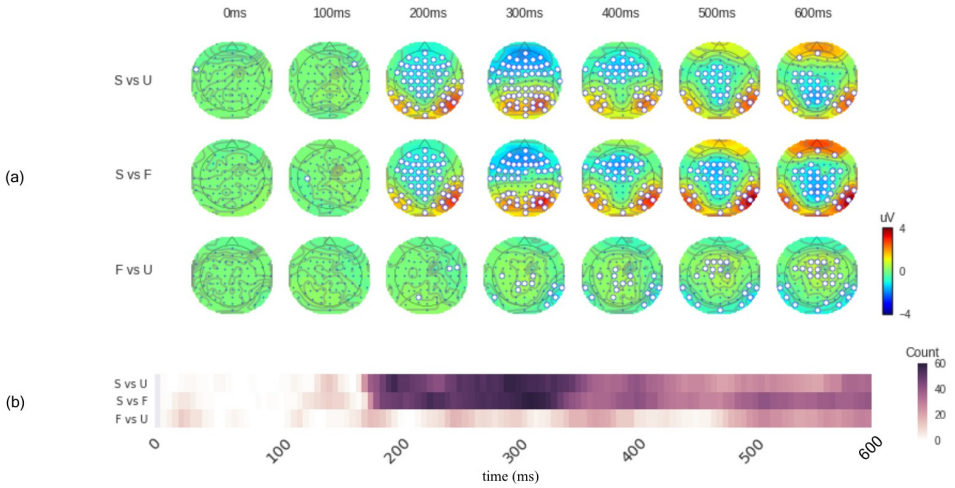

Figure 2.1.: **Results of *distribution of significant sensors* analysis**: (a) Topographies of response differences between conditions across time. Each row contains topographies for a given comparison at different time points. Sensors that show significant response magnitude differences are circled in white. The color on the topography represents the response magnitude differences. The conditions in each comparison are listed on the left: S for scrambled condition, F for famous face condition, and U for unfamiliar face condition. (b) The number of significant sensors across time. The color scale represents the number of significant sensors. The conditions of comparison are listed at the left side of the figure. Labels are the same as in (a). The difference between face perception conditions (F and U) and scrambled (S) condition is significant in sensors above frontal, central, bilateral parietal-occipital areas, starting around 180 ms. The comparison between face perception conditions (F vs U), however, only shows significant differences at the latencies of 300-400 ms and 500-600 ms. Refer to main text for detailed results.

Code Snippet 2.3: **Apply the *GFP* analysis.**

```
scripts = [{'conditions': 'Scrambled,Famous',
            'timepoints': '0~600'},
           {'conditions': 'Scrambled,Unfamiliar',
            'timepoints': '0~600'},
           {'conditions': 'Unfamiliar,Famous',
            'timepoints': '0~600'}]

# do the three analyses independently
for idx,script in enumerate(scripts):
```

```
gfp = epochs.extract(script).GFP(compare=True)
gfp.plot()
```

As shown in Figure 2.2, the condition "Scrambled" (S) begins to be significantly different from the conditions "Famous" (F) or "Unfamiliar" (U) around 140 ms. A small significant difference is found between conditions "Scrambled" (S) and "Unfamiliar" (U) at 500-600 ms, whereas the comparison between conditions "Scrambled" (S) and "Famous" (F) shows weak but significant difference at 400-600 ms. For comparison between conditions "Famous" (F) and "Unfamiliar" (U), significant differences are at 220-260 ms (most at 240 ms), 300-400 ms (most at 400 ms), and 500-600 ms (most at 600 ms). See Supplementary Result 3 for the summary of the GFP powers and the *p*-values over time. See supplementary attachment[5] for the raw data.

TANOVA

The function *"tanova()"* is for performing TANOVA analysis. Data was averaged in every 5 milliseconds defined by the parameter "win_size". The number of repetitions for creating the null distribution was set to 1000 times as defined by the parameter "shuffle". Different strategies for creating the null distribution can be defined by the parameter "strategy". The computation time is about 60 times slower in Strategy 1 and Strategy 2 than in Strategy 3 (about 1 minute using our system). The output of the *"tanova()"* function is a series of *p*-values. We corrected the *p*-values by accepting consecutive significant data points which are longer than 20 ms (Lange et al., 2015) using a command "*correct(method='cluster')*". Users can also use other solutions for multiple comparisons correction such as FDR (Benjamini & Hochberg, 1995) by replacing the value of parameter "method".

Code Snippet 2.4: **Apply the *TANOVA* analysis.**

```
t_result = e.tanova(win_size='5ms',shuffle=1000,strategy=1) #
    ↪ change value of the parameter 'strategy' to 2 or 3 for
    ↪ Strategy 2 or Strategy 3
t_result.correct(method='cluster').plot()
```

The results from Strategy 1 and Strategy 2 are highly similar. The topographic response patterns in condition "Scrambled" starts to be significantly different from those in the conditions "Famous (F) / Unfamiliar (U)" after 170 ms ($p < 0.01$). For comparison between conditions "Famous" and "Unfamiliar", most time after 470 ms shows a significant difference ($p < 0.01$) except from 530 ms to 560 ms. The results from Strategy 3 mostly agree with those from Strategies 1 and 2, with one noticeable exception at 180 ms for the comparison between two face perception conditions. The results from all

[5] https://www.frontiersin.org/articles/10.3389/fnins.2018.00468/fullsupplementary-material

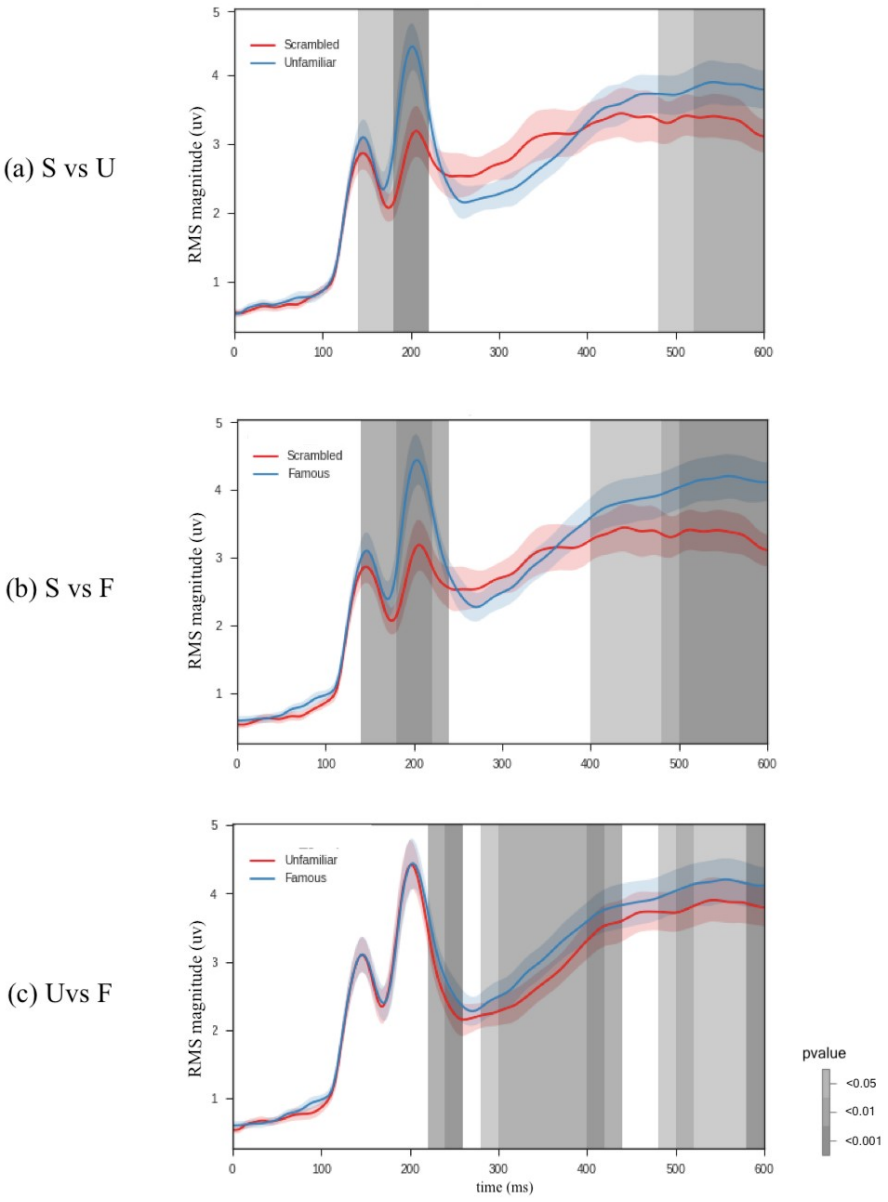

Figure 2.2.: **Results of *GFP* analysis.** Each color line represents the GFP of the corresponding condition. Condition labels are the same as Fig. 2.1. The shaded areas around each line depict the standard error of the mean. The grayscale vertical bars stand for the results of statistical analysis. Grayscale represents the significance level, and location represents the latencies of significant effects. (a) & (b) The condition "Scrambled" (S) begins to be significantly different from the face perception conditions around 140 ms. Differences are also significant at some later latencies. (c) For comparison between two face perception conditions, significant differences are observed starting around 220 ms, later than those in comparisons between face and non-face conditions in (a) and (b). Some later significant differences are also observed. Refer to main text for detailed results.

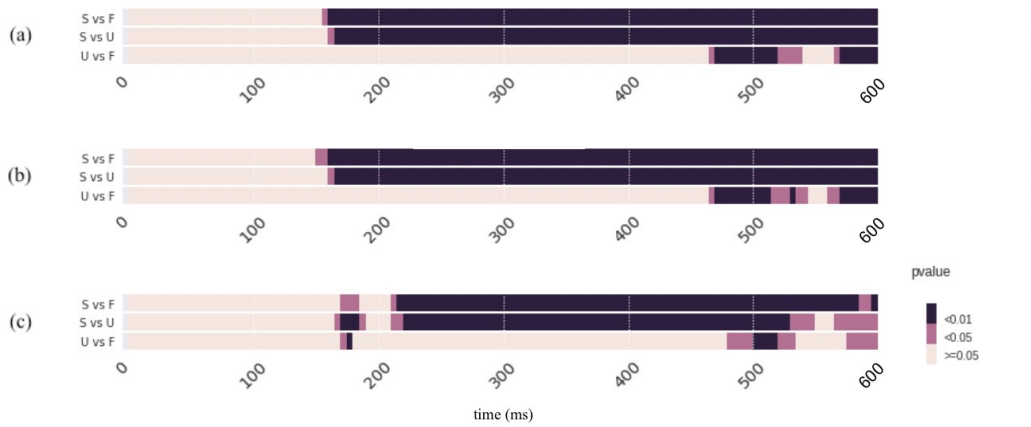

Figure 2.3.: **Results of *TANOVA* analysis.** The results are represented as *p*-values across time. Color represents the significance level, with darker color for smaller *p*-values. Condition labels are the same as in Fig. 2.1. (a-c) The results obtained by applying different strategies for computing null distributions in the non-parametric tests. These results are similar to each other. The topographic response patterns in condition "Scrambled" starts to be significantly different from those in the face perception conditions after 170 ms and lasts until the end of the epoch. For comparison between face perception conditions (F vs U), significant pattern differences are obtained after 470 ms. Results from Strategy 3 have an exception that all three comparisons show significant differences for a short time period around 180 ms. Refer to main text for detailed results.

three comparisons show significant differences for a short time period around 180 ms ($p < 0.01$ for comparison "Scrambled vs Unfamiliar" and comparison "Unfamiliar vs Famous"; $p < 0.05$ for comparison "Scrambled vs Famous"). See Supplementary Result 4 for the summary of the *p*-values of TANOVA over time. See supplementary attachment[6] for the raw data.

Pattern classification

The function 'classification' is for performing pattern classification analysis. The default classifier is a logistic regression classifier. Data was averaged in every 5 milliseconds defined by the parameter "win_size='5ms' ". The parameters "test_size=0.3" and "fold=25" indicate that 30% of data were randomly selected as the test set and the rest are in the training set in each fold (data splitting iteration) and the number of folds is 25 in the cross validation.

[6]https://www.frontiersin.org/articles/10.3389/fnins.2018.00468/fullsupplementary-material

Code Snippet 2.5: **Apply the *Pattern classification*.**

```
c_result = e.classification(win_size='5ms',fold=25,test_size=0.3)
c_result.correct(method='cluster').plot()
```

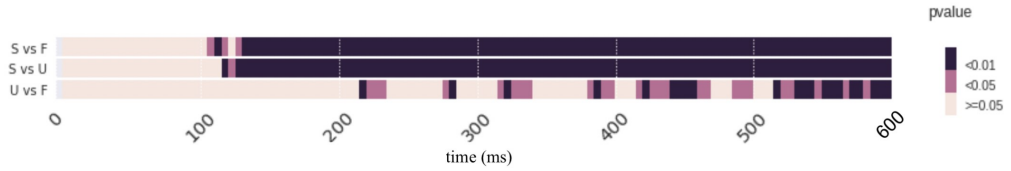

Figure 2.4.: **Results of *Pattern classification* analysis.** Pattern classification results are represented as p-values across time. Color represents the significance level. Condition labels are the same as in Fig. 2.1. Both face perception conditions show differences ($p < 0.01$) from the scrambled condition as early as around 120 ms. Differences between two face perception conditions are scattered across the timespan. Refer to the main text for detailed results.

Figure 2.4 depicts the pattern classification results as p-values across time. The condition "Scrambled" starts to be significantly different from condition "Famous (F) or Unfamiliar (U)" after 120 ms. The conditions "Unfamiliar" and "Famous" show sparse differences across time. More specifically, results show that at around 220 ms, 280 ms, 330 ms, 380 ms, 410-450 ms, and 510-600 ms, there are significant differences between these two conditions ($p < 0.05$). See Supplementary Result 5 for the summary of the scores of the p-values of Pattern classification over time. See supplementary attachment[7] for the raw data.

The function "classification()" also allows researchers to use an external model such as a deep learning model (Abadi et al., 2016; Chollet & Others, 2015), see Supplementary Code Snippet 3 for an example.

2.4. Discussion

EEG provides high temporal resolution information that reflects cognitive processes. However, common ERP methods using partial information in selected sensors make it difficult to obtain a precise and comprehensive temporal dynamics across the system. In contrast, source localization may estimate the distribution of neural generators and their dynamics. However, sophisticated procedures, various assumptions, as well as high demand on data quality and computational power may make localization methods not

[7] https://www.frontiersin.org/articles/10.3389/fnins.2018.00468/fullsupplementary-material

practical for some users. In the EasyEEG toolbox, we offer multivariate analyses that use EEG topographical patterns of sensors to obtain holistic system-level dynamic information without projecting back to the source space. Different types of analyses that take distinct yet related perspectives help users infer different aspects of temporal dynamics by differentiating response patterns and magnitude across time. Main functions and other necessary steps have been packed in this toolbox, so that users can easily use them. Moreover, the highly flexible, compatible and expandable design in programming are also ideal for advanced users. Our EasyEEG toolbox offers a practical, efficient and complete pipeline from raw data to publication for EEG research to directly test cognitive hypotheses.

This paper introduces four methods included in EasyEEG, which take information from all sensors of a topography to investigate neural dynamics. These methods target different aspects of topographic information and separately evaluate topographic patterns and response magnitude across time. The first method, the distribution of significant sensors analysis, can provide the spatial extent of effects by observing the spatial configuration and counting the number of sensors that have significant differences among conditions. The sample results show greater spatial extent and higher number of significant sensors in both face perception conditions, compared with the scrambled condition, starting around 180 ms (Fig. 2.1). These results indicate that the distribution of significant sensors can grossly identify the dynamics of neural processing in different conditions. The second method, GFP analysis, provides an indicator of overall energy variation among all sensors. The sample results show that the face perception conditions start to differ from the scrambled condition around 140 ms, whereas response magnitudes differ between face perception conditions (famous vs unfamiliar) starting around 220 ms. These latency differences in response magnitude reveal that the general face perception occurs earlier, and specific face identification occurs later.

The third method, the TANOVA analysis, provides a way to quantify and statistically test pattern similarity between topographies. The sample results show that the response topographic patterns in face perception conditions start to differ from those in the scrambled condition around 170 ms (Fig. 2.3). These results indicate that distinct processes for face perception emerge around 170 ms. In contrast, topographic responses in two face perception conditions remain the same until around 470 ms. These results indicate that similar sensor patterns mediate the perception of famous and unfamiliar faces during the early perceptual processes. The differences start around 470 ms, which could be because the effects of familiarity induce additional neural processes for famous faces compared with the processes for unfamiliar faces. The fourth method, the pattern classification, uses self-adaptive algorithms and takes advantage of all information regarding response magnitude and patterns of topographies to investigate neural dynamics. The sample results show that both face perception conditions show differences from the scrambled condition as early as around 120 ms. Differences between two face per-

ception conditions are scattered across the timespan. These results indicate that the pattern classification method can reveal response magnitude and pattern differences as the classification results between two face conditions, as well as provide additional information such as magnitude and pattern interaction, indicated by the detection of early differences between scrambled and face conditions.

These four methods are complementary to each other and can provide information at different levels to overcome limitations of individual methods. Users can use them in combination to obtain a comprehensive picture of their data. For example, the distribution of significant sensors was obtained by individually testing response magnitude differences in each sensor. Without correction, this is subject to the multiple comparisons problem. We use this result to provide a general and direct visualization of data and dynamic results, similar to the common practice in fMRI research that uses "$p < 0.05$ uncorrected" for visualizing results.

The observed significant sensors distribution differences, as demonstrated in the face perception sample, can be caused either by response magnitude changes or a change of neural generators that is reflected in topographic patterns. We use the GFP and TANOVA to further test the magnitude and pattern differences among conditions, respectively. The GFP results show magnitude differences between two face perception conditions starting around 220 ms, whereas TANOVA results show pattern difference starting until 440 ms. These results from two methods collaboratively suggest that response magnitude in the same neural sources is firstly different between perceiving famous and unfamiliar faces, and later distinct neural generators are involved for processing familiarity. In the comparisons between face and scrambled conditions, both GFP and TANOVA analyses reveal differences starting around 170 ms, suggesting both neural generators and their magnitude differ when processing faces versus non-faces.

The pattern classification analysis gives the combination of magnitude and topographic differences, and can be used to verify and "double-check" the results in both GFP and TANOVA. In the sample results, the latencies of significant results in the classification agree with the combination of results in GFP and TANOVA in comparisons between face and non-face conditions, as well as between face perception conditions. Moreover, the pattern classification can provide more information than GFP or TANOVA methods alone. This additional information is likely to arise from the interaction between the response magnitude and patterns. For example, the early differences between face and scrambled conditions is only detected using pattern classification.

Based on the features of the four methods and their complementary nature, we recommend the following procedure. Users can follow all or part of this procedure based on their research goals to obtain topographic and response magnitude dynamics.

1. Perform basic pre-processes such as noise reduction, baseline correction, filtering using other available toolboxes such as MNE Python.

2. Load the pre-processed data "`EasyEEG.io.load_epochs('path')`", define conditions and comparisons, and extract the data epochs of interests "`extract()`".
3. Obtain the distribution of significant sensors "`topography()`" for a direct and intuitive visualization "`plot()`" of effects.
4. Test the overall magnitude differences "`GFP().plot()`".
5. Test the topographic pattern differences "`tanova().plot()`".
6. Perform pattern classification "`classification().plot()`" to verify the results from step 3 to 5.

By following the above 6 steps, users can visually inspect their data and effects, obtain statistical analysis results at the group-level regarding response magnitude and topographic patterns, and have a verification of obtained results from the perspective of pattern classification and machine learning. EasyEEG provides the realization of these steps and a complete pipeline from raw EEG data, to generating figures, to statistical testing for publication.

The results obtained by EasyEEG are consistent with those from other analysis approaches. A mass univariate General Linear Model (GLM) was applied on the same face perception dataset (Wakeman & Henson, 2015). Their results suggested that faces and scrambled conditions significantly differed from around 160 ms and last to the end of the epoch (600 ms), with differences in the sensors over fronto-central and lateral parieto-occipital areas, which are very consistent with our results (Fig. 2.1, 2.2 & 2.4). In the comparison between two face perception conditions, they found a single cluster over mid-frontal electrodes from 520–620 ms (Wakeman & Henson, 2015), which also agrees with our TANOVA results (Fig. 2.3). These consistent results obtained by different approaches and toolboxes demonstrate the reliability of our methods and EasyEEG.

Besides the reliability, EasyEEG can obtain additional results and provide more insights. The most important one is separating response magnitude effects from topographic pattern changes. As in our results, GFP and TANOVA analyses reveal differences in response magnitude but not in topographic patterns between two face perception conditions, whereas both magnitude and patterns differ between face and scrambled conditions. These results highlight the advantage and capacity of EasyEEG on testing different aspects of hypotheses. Moreover, EasyEEG provides an unbiased omnibus measure using information of all sensors in topographies, which overcomes individual spatial and temporal differences and facilitates group-level analyses.

EasyEEG shares some attributes with other existing toolboxes of multivariate analyses, yet has distinct features. For instances, the Mass Univariate ERP Toolbox applies the univariate test at each of all sensors, and reduces the multiple comparison pollution by different correction methods (Groppe et al., 2011), whereas EasyEEG takes the topographical pattern of sensors directly with multivariate approaches so that it can better

avoid the multiple comparison problems than the univariate tests. LIMO EEG utilizes the hierarchical general linear model for multivariate data (Pernet et al., 2011). The Donders Machine Learning and MNE-Python offer an interface to Scikit-Learn for retrieving the classification score (Gramfort et al., 2013).

EasyEEG offers great convenience and outstanding compatibility. The most common difficulty of using various software packages is how to get your own EEG data working in that toolbox. EasyEEG has a solution by reducing programming demands for customized algorithms. First, the complicated and tedious data extraction operations are replaced by calling built-in extraction functions with a descriptive dictionary. Researchers are only required to understand the structure of extracted EEG data. Second, EasyEEG makes extraction and combination of data in multiple sections/blocks automatic. In this way, users avoid the tedious and error-prone repetitive steps. Third, the proposed multivariate analysis methods have been implemented in simple command lines. Users can specify the intended analysis and parameters in one place and obtain the final results. Thus, researchers can focus more on their experiments and selection of core algorithms and methods, and obtain quick results to test their hypotheses.

EasyEEG also provides great flexibility and expandability for advanced users. Should researchers want to examine different aspects of data or to apply some other customized algorithms, they only need to modify a small portion of the current scripts to quickly create new computational or visualization algorithms based on a resilient data structure and a number of well-written application programming interfaces (APIs).

Besides the introduced multivariate analysis methods, we aim to include more analysis methods in EasyEEG to investigate neural dynamics, and increase the reliability of these methods. More specifically, we plan to integrate more machine learning models for EEG data analysis and pattern classification methods. Moreover, we aim to increase the efficiency and expandability of EasyEEG by designing more programming APIs for developers.

There are several limitations of the current version of our toolbox. First, methods included in our toolbox work best with the activation widely distributed among all sensors. However, if the effects are focused in several electrodes, the effect size could be reduced by the summary of topography, especially in the GFP analysis. Second, the multivariate methods rely on the topographies in the sensor space to infer the relation between neural sources of different conditions. The mapping between sources and topographies could be complicated. For example, two different neural sources, in theory, could generate the same pattern, although it is highly unlikely. If this situation occurred, our toolbox would derive incorrect results. Moreover, the topography-based analysis can find differences of neural sources between conditions but it cannot further separate whether the differences are induced by the changing of source location or the orientation of the same source. All these limitations are induced by the cost-effectiveness tradeoff. While methods in our toolbox can offer direct and easy ways to test psychological and neu-

roscientific hypotheses, we sacrifice the ability to precisely test aspects of underlying neural sources. Therefore, users should choose different methods based on their own questions and needs. Third, only four multivariate methods are built into the current version of toolbox. We aim to integrate more features in the future, such as deep learning techniques, to increase the power of our toolbox, meet broader requirement of users and provide solutions to wider ranges of questions.

In summary, EasyEEG provides simple, flexible and powerful methods that can be used to directly test cognitive hypotheses based on topographic responses. These multivariate methods can investigate effects in the dimensions of response magnitude and topographic patterns separately using data in the sensor space, therefore enabling assessing neural response dynamics without sophisticated localization. Python-based algorithms provide concise and extendable features of EasyEEG. Users of all levels can benefit from EasyEEG and obtain a straightforward solution to efficiently handle and process EEG data and a complete pipeline from preprocessing to statistical testing and to result visualization.

3 | How do we segment text? Two-stage chunking operation in reading[1]

Abstract

Chunking in language comprehension is a process that segments continuous linguistic input into smaller chunks that are in the reader's mental lexicon. Effective chunking during reading facilitates disambiguation and enhances efficiency for comprehension. However, the chunking mechanisms remain elusive, especially in reading given that information arrives simultaneously yet the written systems, such as Chinese, may not have explicit cues for labeling boundaries. What are the mechanisms of chunking that mediates the reading of the text that contains hierarchical information? We investigated this question by manipulating the lexical status of the chunks at distinct levels in four-character Chinese strings, including the two-character local chunk and four-character global chunk. Human participants were asked to make lexical decisions on these strings in a behavioral experiment, followed by a passive reading task when their electroencephalography (EEG) was recorded. The behavioral results showed that the lexical decision time of lexicalized two-character local chunks was influenced by the lexical status of the four-character global chunk, but not vice versa, which indicated the processing of global chunks possessed priority over the local chunks. The EEG results revealed that familiar lexical chunks were detected simultaneously at both levels and further processed in a different temporal order: the onset of lexical access for the global chunks was earlier than that of local chunks. These consistent results suggest a two-stage operation for chunking in reading: the simultaneous detection of familiar lexical chunks at multiple levels around 100 ms followed by recognition of chunks with global precedence.

Significance Statement

The learners of a new language often read word by word. However, why can proficient readers read multiple words at a time? The current study investigates how we efficiently segment a complicated text into smaller pieces and how we process these pieces. Participants read Chinese strings with different structures while their key-press responses and brain EEG signals were recorded. We found that texts were quickly (about 100 ms from their occurrences) segmented to varied sizes of pieces, and larger pieces were then processed earlier than small pieces. Our results suggest that readers can use existing knowledge to efficiently segment and process written information.

[1]Yang, J., Cai, Q., & Tian, X. (2020). How do we segment text? Two-stage chunking operation in reading. *eNeuro, 7*(3).

3.1. Introduction

Reading is arguably unique for human. However, how we process written texts remains elusive. For instance, how can we comprehend a complex sentence? A sentence consists of many letters/characters that form a hierarchical structure of text chunks (e.g., morphemes, words, and phrases). Readers need to incrementally segment a complex sentence into smaller chunks that map onto their mental lexicon. This process is termed as text *chunking* (Gobet, Lloyd-Kelly, & Lane, 2016; Reali & Christiansen, 2007). What are the small chunks during chunking? How do we process the chunks? To answer those questions, this study investigated the cognitive procedure of text chunking.

Words and their sub-level (morphemes) are usually assumed as the basic units in reading models in psycholinguistics and computer science (Coltheart, Rastle, Perry, Langdon, & Ziegler, 2001; McClelland & Rumelhart, 1981; Taft, 2013). However, eye-tracking studies suggested we can perceive the text longer than a word at one time (Rayner, 1998). Our working memory also allows us to remember familiar multiple words (Miller, 1956). Even more, multi-word expressions can be stored in our mental lexicons (Arnon & Snider, 2010; Siyanova-Chanturia, Conklin, Caffarra, Kaan, & van Heuven, 2017). These studies suggest the multi-word representations and the beyond-word processing are feasible. Moreover, relying on larger chunks effectively reduces the cognitive load while processing sentences: fewer chunks to be interpreted and integrated (Blache & Rauzy, 2012; Ellis, 2003; Krishnamurthy, 2003). Furthermore, the semantic combination of constituents can be different from holistic meaning (Goldberg, 1995). One extreme example is idioms, as the metaphors of an idiom can be distinct from their literal meanings of smaller constituents. Multi-word representation is required in certain contexts to avoid ambiguity. Therefore, multi-word chunks, as well as word chunks, could be the units during chunking.

What is the relation between the processes of word chunks and multi-word chunks during chunking? The studies of compound words (a single lexical entity but consists of more than one root morphemes, e.g., “flagship”) may offer hints. According to the dual-route models of compound-word processing, both the whole word and its constituents are processed at the same time or are selected to process each level flexibly (Andrews, Miller, & Rayner, 2004; Blache, 2015; Koester, Gunter, & Wagner, 2007; MacGregor & Shtyrov, 2013; Semenza & Luzzatti, 2014). In a similar vein, we hypothesized that all the familiar lexical chunks, no matter which level it is, could be processed simultaneously. More specifically, the detection of chunks would be the first step in chunking, and the detection of chunks at multiple levels would occur at the same time, as the early lexical familiarity checking assumed in the E-Z reader model (Reichle, Rayner, & Pollatsek, 2003).

How does the multi-level operation unfold in the chunking process? Which level has the priority after being detected? The word superiority effect indicates that the

recognition of letters within words is better than letters in nonwords or stand-alone letters (Reicher, 1969). It suggests that the word has priority over the letter in reading. Similarly, the processing priority of global chunks can reduce the steps of integration and avoid the ambiguity to enhance the efficiency of language processing (Blache & Rauzy, 2012; Ellis, 2003; Krishnamurthy, 2003). Generalizing from the word superiority effect, we hypothesized that global chunks take priority over the parts and would be initiated first in the processing stage after detection.

In this study, we used Chinese four-character strings to investigate the chunking operation in reading. The Chinese written system is an ideal model for observing multi-level chunking because Chinese does not have explicit word boundaries. Each Chinese character is a basic lexical unit with a similar length. Four characters can form two levels of chunks: chunks with two characters (hereafter called the *local level chunks*) and a chunk with four characters (hereafter called the *global level chunk*). The lexicality was manipulated at both levels so that four types of stimuli were included (phrase, idiom, random words, and random characters). In the behavioral experiment, we investigated the interaction between the global and local chunks in reading by a lexical decision task at different levels of chunks. Moreover, an EEG experiment was carried out to investigate the temporal dynamics of detection and recognition stages in the multi-level chunking operation.

3.2. Materials and Methods

3.2.1. Participants

Twenty-one healthy native Chinese speakers (10 males, mean age 21 years, range 18-30 years) with normal or corrected-normal vision participated in both behavior and EEG experiments for financial compensation. Five participants who produced extensive EEG artifacts were excluded from EEG analysis. Hence, a total of sixteen participants were included in the EEG study. The experiments were approved by the Research Ethics Committee of East China Normal University. Written informed consents were obtained from all participants before the experiments.

3.2.2. Stimuli

All stimuli are four-Chinese-character strings. Two factors are included when designing these stimuli. The first factor is the *chunk size* that contains two levels -- a global size of 4 characters and a local size of 2 characters. The second factor is *lexicality* (word or nonword) at each chunk size. These two factors are fully crossed and yield four types of stimuli. We denote *chunk size* using upper case letters -- 'G' for global and 'L' for local, and use lower case letters for lexicality in each chunk size ('w' for word and 'n'

	Local word	*Local nonword*
Global word	***GwLw*:** lexicalized compound phrase composed of two two-character words.	***GwLn*:** lexicalized compound phrases that consist of four independent mono-morphemic characters (Chinese idioms `Chengyu').
	e.g. "希 腊 神 话 " (pinyin: xī là shén huà), translation: Greek mythologies. "希 腊 " and "神 话 " means "Greek" and "mythologies" respectively	e.g. "以 逸 待 劳 " (pinyin: yǐ yì dài láo), translation: wait for the exhausted enemy at your ease. "以 逸" and "待 劳 " are not words.
Global non-word	***GnLw*:** non-lexicalized compound phrase composed of two two-character words.	***GnLn*:** random character string – nonwords at both levels.
	e.g. "存 款 电 脑 " (pinyin: cún kuǎn diàn nǎo), translation: deposit-computer. "存 款 " and "电 脑 " means "deposit" and "computer" respectively.	e.g. "投 其 顾 此 " (pinyin: tóu qí gù cǐ), a nonsense phrase. "投 其 " and "顾 此 " are not words either.

Table 3.1.: **Stimuli description.**

for nonword). For example, 'GnLw' stands for the condition of stimuli that are four-character nonwords at the global level made of two two-character words at the local level. Note that the stimuli in 'GwLn' are Chinese idioms so called 'Chengyu'. They are lexicalized compound phrases that consist of four independent mono-morphemic characters. None of the two characters within 'Chengyu' can form a common word, whereas the four characters together form an idiomatic expression. Chengyu generally expresses the gist or moral message of myths, stories, or historical events from which they were derived. Therefore, the meaning of a chengyu usually surpasses the sum of the meanings from the four characters. The four types of stimuli are listed in Table 3.1.

We selected and created all stimuli with the following steps. We extracted the *GwLw* and *GwLn* stimuli from a database of Sogou Pinyin (`https://www.sogou.com/labs/resource/w.php`) and a database of Chinese characters (CharDB: data version: 0.98.1; program version: 0.97.2; https://chardb.cls.ru.nl/). All the *GwLw* and *GwLn* stimuli sat-

isfied the following criteria at the global level: 1) noun (the part of speech is determined by a record of the lexicon of Jieba v0.36: `https://github.com/fxsjy/jieba`); 2) high-frequency (the frequency was determined by the database of the Sogou Pinyin, and high-frequency meant frequency above 3,000); and 3) no duplicative characters (e.g., "高高兴兴", translation: happy). Moreover, the *GwLw* stimuli satisfied the following criteria at the local level: 1) both two-character words were nouns, and 2) high-frequency words. Moreover, the lexicality of *GwLn* stimuli at the local level was verified by checking if the first two or last two characters' combination did not exist in the Sogou Pinyin database. These selection criteria made the *GwLw* and *GwLn* stimuli consistent in all aspects except the lexical status at the local level.

The *GnLw* stimuli were created by randomly pairing two different two-character words, the Lw in GnLw and GwLw follow the same criteria. So the only difference between the *GwLw* and *GnLw* was the lexical status at the global level. Finally, the *GnLn* stimuli were created by randomly mixing four different characters, and none of the first or last two characters' combinations existed in the Sogou Pinyin database. Characters used in all stimuli have log frequency ranging from 3.011 to 5.344, with stroke counts ranging from 4 to 13. The character's log frequency was determined by the Subtitle Database (Cai & Brysbaert, 2010).

The distinction between word and nonword was further controlled by familiarity. Twelve participants who were not in the main experiment were asked to rate the familiarities of either the entire four-character or the constituents of two-character strings as being words or not. The rating range was from 1 to 5, where 1 stands for unfamiliar strings/nonwords and 5 for familiar words. The strings that were rated from 2 to 4 were removed and only the stimuli that were either very familiar words or very unfamiliar nonwords remained in a pool. Eighty stimuli in each condition were randomly selected from the pool and used in this study.

3.2.3. Procedure behavioral experiment

In each trial, participants were first asked to focus on a cross presented at the center of the screen. After 400 ms, the fixation cross disappeared, and a four-character string was shown until response. A line also appeared either under the entire four-character string or under two-character string (the first or the last two characters). Participants were asked to make a lexical decision about the underlined string, either the entire string (global task henceforth) or the first or last two-character string (local task henceforth) by pressing either "F" or "J" on the keyboard as fast as possible. Participants had a maximum of 3 s to respond. Responses and reaction time were collected. Response keys were counterbalanced across participants. The intertrial intervals were randomly selected from a range from 800 to 1000 ms.

Four stimuli types (*GwLw*, *GwLn*, *GnLw*, and *GnLn*) were fully crossed with task types (global task vs local task) and yielded eight conditions; 320 trials were included in this experiment. Half of the trials were randomly selected and used in the global task and the other half in the local task. The order of conditions was randomized. The experimental presentation was programmed in a Python package, Expy, which is a software for presenting and controlling psychological experiments (`https://expy.readthedocs.io/`).

Behavioral data analysis

All participants had response accuracy exceeding 85%, and the average accuracy was 92%. No participant's data were excluded. Trials with incorrect responses were removed before analysis. We applied repeated measures three-way ANOVA (analysis of variance) on the reaction time data with factors of global-level lexicality, local-level lexicality, and task, followed by planned *t* tests for testing specific hypotheses.

3.2.4. Procedure EEG experiment

The same group of subjects participated in the EEG experiment. The EEG experiment shared the same stimuli list with the behavioral experiment, but both the procedure and the task are different. First, the display of each character string lasted for 300 ms. Participants were asked to read the underlined parts of the stimuli (to keep their attention on the stimuli), but they did not perform any lexical decision task. We used all 320 strings with 80 for each stimuli type in the global task and repeated once in the local task. Moreover, 320 four-symbol strings were included as the visual baseline in the EEG experiment. The symbols in a symbol string trial were randomly sampled with replacement from four symbols (□, △, ◇, and ○). Underlines were included in the symbol trials similar to those in the global and local tasks in experimental trials. Half of the trials were randomly selected and used in the global task and the other half in the local task. To guarantee participants' attention on the stimuli, we randomly inserted strings of digits for 100 ms, and participants were asked to report the underlined digits by pressing number buttons on a keyboard. About 48 number-report trials were presented to each participant.

EEG recording

EEG signals were recorded with a 32-electrode active electrodes system (actiChamp system, Brain Products GmbH). FP1 and FP2 were used to monitor vertical eye movements. Electrode impedances were kept below 10 $k\omega$. Data were continuously recorded in single DC mode. Data were sampled at 500 Hz, online referenced to the Cz.

EEG data analyses

EEG data were preprocessed using EEGLAB (version 13.5.4b; (Delorme & Makeig, 2004)). Data were bandpass filtered (0.1–30 Hz, Hamming windowed sinc FIR filter), and re-referenced to the average reference. The preprocessed data were epoched between –200 and 800 ms relative to the onset of strings and baseline-corrected using the 200-ms prestimulus period. The trials with eye blinks were rejected if the amplitude within the 1000-ms epoch exceeded ±50 μV. Remaining trials with apparent noise were rejected manually. Approximately 15% of trials were rejected. Five participants who produced a large number of artifacts or showed continuous α waves were excluded from further analysis. Epochs in each condition were averaged and used to creat an event-related potentials (ERPs). Root mean square (RMS) responses were also calculated as a geometric mean of all channels.

In addition to the univariate analyses on ERPs and RMS responses, our analysis used the topographic patterns or distributions across all sensors rather than the response amplitude in selected groups of sensors, as it can provide more holistic and unbiased information. Such multivariate methods can collectively reflect spatial and temporal information and offer more power to test psychological and neuroscience hypothesis by overcoming problems such as individual differences, sensor selection and reference selection in EEG (Murray et al., 2008; Tian & Huber, 2008; Tian et al., 2011; X. Wang, Zhu, & Tian, 2019; Yang, Zhu, & Tian, 2018). Two multivariate-based methods [clustering and topographic ANOVA (TANOVA)] and one mass-univariate method (analysis of the topographic distribution of amplitude differences) were applied as:

CLUSTERING A clustering method on the ERP topographic responses was implemented first. This unsupervised machine learning method groups data across all conditions by forming temporal clusters based on the similarity of their topographical patterns. This clustering method is a data-driven method, in which it explores the pattern similarity in topographies in all conditions. The clustering algorithm organizes data at different time points into distinct clusters, so that we can explore the temporal dynamics of pattern changes. Moreover, if one considers the clusters reflecting different processing stages, this analysis can identify the processing stages in each condition and display the temporal differences of any specific stage among conditions. We used K-means, the most popular algorithm for clustering. The clustering analysis is an omnibus test about neural dynamics, which can detect clusters and set up the time windows of interest for the following analyses.

The procedure of clustering algorithm covers three steps: (1) averaged EEG data across all participants to get ERPs at each time point for each condition; (2) defined ERPs at each time point in each condition as a sample, and the amplitudes of 32 electrodes were used as features in a sample; (3) K-means algorithm was conducted at all

samples. The K-means algorithm is data driven. The target cluster number (K) can range from the minimum of one to the number of data points. We assumed that the baseline period before stimuli onset involved rest or random cognitive processes. The topographies would be consistent random patterns that were different from later sequences of topographies induced by stimuli. If the K-means separated the baseline period into more than one cluster, it was most likely overfitting. Therefore, we set the criterion of getting two clusters at the baseline period as a stopping point for increasing the number of clusters. That is, the K-means algorithm was conducted at all samples. The number of clusters was initially two and increased until the clustering result included more than one cluster at the baseline stage.

ANALYSIS OF THE TOPOGRAPHIC DISTRIBUTION OF AMPLITUDE DIFFERENCES We calculated the response amplitudes across all sensors at a given time window. The changes in the amplitude differences can reflect the processing dynamics. Especially, the earliest time point that shows significant differences would indicate the temporal onset of interest. Moreover, the spatial extent of experimental effects can be estimated by the distribution of sensors in which the amplitude differences are observed. At a window size of 20 ms, we checked the significant electrodes (Yang et al., 2018) to obtain the distribution differences between conditions. Because there were 32 comparisons in each time window, the p values were corrected by false discovery rate (FDR). The primary purpose of this analysis was to identify the possible onset timing of any amplitude differences between conditions. However, the power of detecting the onset using a measure of amplitude could be small. To reduce the Type II error, we did not apply corrections across time so that we can reduce the chance of missing the exact onset time of amplitude differences.

TANOVA We further investigate the patterns of topographies by considering all sensors at the same time to infer the differences in underlying neural processes across conditions. A single index was calculated to indicate topographic information. Mathematically, each topography can be viewed as an n-dimensional vector, where the n equals the number of sensors. The divergence between the topographies of two experimental conditions can be quantified by the cosine value of the high-dimensional angle between two vectors (Tian & Huber, 2008). The cosine distance has a range from 0 to 2, where 0 stands for identical topographies and 2 exactly opposite patterns. Note that the cosine distance represents the similarity between the response patterns in topographies and is free from the difference of response magnitude because the measure of cosine distance is normalized by the vector length. To statistically test the cosine distance between topographies and to infer the underlying neural processing in different conditions, we applied an algorithm named TANOVA (Brunet et al., 2011; Lange et al., 2015; Murray et al., 2008; Tian & Huber, 2008; Tian et al., 2011). In TANOVA, the null hypothesis distri-

bution is generated by shuffling the condition labels, and we here shuffled the condition labels on the subjects' ERPs using the EasyEEG toolbox (Yang et al., 2018); strategy 2, shuffle times: 1000, window size: 10 ms). Furthermore, the temporal clusters in the TANOVA results were identified by a precluster threshold of 0.1 and were tested by cluster-based permutation with a corrected threshold of 0.05 (Maris & Oostenveld, 2007).

3.2.5. Results

Behavioral experimental results

Reaction time was subject to repeated measures three-way ANOVA with the factors of global-level lexicality, local-level lexicality, and task. The main effect of global-level lexicality was significant ($F_{1,20} = 10.83$, $p < 0.01$), suggesting that participants took longer time to identify global-level nonwords than global-level words. The main effect of local-level lexicality is significant ($F_{1,20} = 6.30$, $p = 0.02$), suggesting participants took longer time to identify local-level nonwords than local-level words. However, the main effect of task is not significant ($F_{1,20} = 1.07$, $p = 0.31$), suggesting different tasks that require participants to respond to either global or local chunks have a similar level of difficulty. More importantly, all three two-way interactions are significant, global-level lexicality × local-level lexicality ($F_{1,20} = 57.11$, $p < 0.001$), global-level lexicality × task ($F_{1,20} = 16.02$, $p < 0.001$), and local-level lexicality × task ($F_{1,20} = 57.11$, $p < 0.001$).

Planned *post hoc t* tests were further conducted in each factor to specify the observed significant interactions. First, we examined how global information affects processing at the local level (Fig. 3.1*A*). In the local task, the reaction time in *GnLw* was significantly longer than that in *GwLw* ($t_{20} = 5.71$, $p < 0.001$, difference = 55 ms), suggesting that the nonwords at the global level significantly slowed down the lexical decision of words at the local level. Moreover, the reaction time of *GwLn* was significantly longer than *GnLn* in the local task ($t_{20} = 3.62$, $p = 0.002$, difference = 54 ms), suggesting that the words at the global level also slowed down the lexical decision of nonwords at the local level. Second, we examined how the local chunk could affect processing at the global level (Fig. 3.1*B*). In the global task, we did not find a significant difference between reaction time to *GwLw* and *GwLn* ($t_{20} = 1.00$, $p = 0.32$; Fig. 3.1*B*, left), suggesting that the lexical status of local chunks cannot affect the lexical decision of words at the global chunks. These results collaboratively suggest that processing at the global level may take priority.

Another comparison on how the local chunk could affect processing at the global level revealed that the reaction time of *GnLw* was significantly longer than that of *GnLn* in the global task ($t_{(20)} = 6.69$, p <0.001, difference = 112 ms; Fig. 3.1*B*, right). This result suggests that the decisions of global chunks and local chunks could be parallel

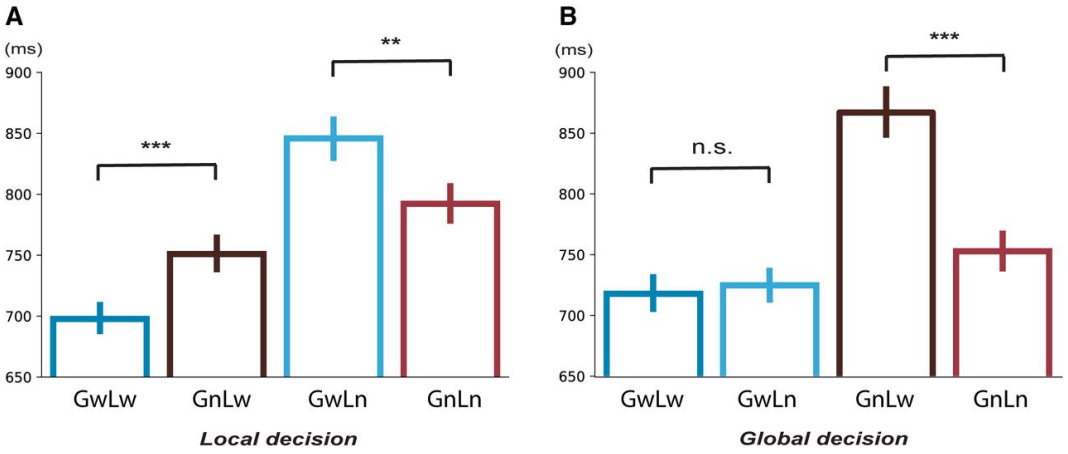

Figure 3.1.: **Results of the behavioral experiment.** ***A***, ***B***, Reaction time results in the global and local tasks, respectively. In each plot, condition labels are provided along the *x*-axis. Error bars represent ±1 SEM (standard error of the mean). Each planned paired test was represented by the line linking two bars; n.s., not significant; $^{**}p < 0.01$, $^{***}p < 0.001$.

when the global decision took too long. The lexical information at the local level may leak through to the processing of global chunks and influence the decision of nonwords. The behavioral results demonstrate a unidirectional influence from the global level to the local level when the task targets are words and interactions between levels when the task targets are nonwords that need more time to make a decision about. That is, global chunks may take priority in lexical processing. Whereas the processing of global and local chunks could be parallel when global decision took too long, the lexical information at one level may leak through to the process of the other level and influence the decision of nonwords. We further test the processing dynamics in an EEG experiment.

To examine and exclude the possible effects of the underline position, we extracted the trials with two-character underlines and ran a repeated measures three-way ANOVA with the factors of global-level lexicality, local-level lexicality, and underline position. The underline position showed neither main effect ($F_{(1,20)} = 1.61$, $p = 0.22$), nor interaction effect with global-level lexicality ($F_{(1,20)} = 0.01$, $p = 0.92$), nor interaction effect with local-level lexicality ($F_{(1,20)} = 0.38$, $p = 0.54$). These results suggest that positions of stimuli that were relevant to the task did not affect response speed.

EEG results

Clustering revealed distinct stages of chunking We first conducted the clustering analysis to explore the dynamics of ERP responses. The clustering algorithm aimed to separate the continuous ERP responses into distinct stages based on common features

observed across time. As shown in Fig. 3.2, the clustering results were reliable as similar clusters were observed continuously in each condition.

More importantly, clear temporal profiles were revealed by the clustering analysis in all conditions. First, the same cluster was observed in the baseline period until around 80 ms after stimulus onset, as well as at the end of epochs (∼600 ms after onset) among all types of stimuli. The clustering in these periods was presumably because few cognitive processes that relate to the stimuli or task were available or manifested in the ERP topographies. Second, a novel cluster (the second cluster) appeared after 80 ms across five conditions. The clustering spanned in similar latencies as N1/P2 components, presumably reflecting visual processing. However, different dynamics were observed across conditions after 200 ms. The third cluster appeared earlier in the symbol condition with a much shorter duration than the four experimental string conditions in which the third cluster appeared around 250 ms and lasted ∼90 ms. Moreover, in the symbol condition, the third cluster was immediately followed by the sixth cluster that did not appear until 500 ms in the four experimental string conditions (except in *GwLw* condition around 430 ms). The early start and long-lasting sixth cluster in the symbol condition was accompanied by the absense of the first and second clusters that appeared around 320 ms and lasted untill 450 ms in other conditions.

More interestingly, a 15 ms period around 130 ms was labeled as the cluster baseline that was grouped in the baseline and the end of epoch periods. This formed a short gap that broke the early processing into two stages. The clustering results set up the time windows of interest for the following analyses. We focused on the components in early timing to further investigate the underlying processes of chunking operation.

Chunk detection in the earliest stage To test the hypothesis about the lexical detection in the earliest stage, we conducted analyses to investigate the lexicality effects at the global and local chunk levels. First, we applied repeated measures one-way ANOVA on the ERPs in P3 and P4, as well as RMS waveforms calculated using all channels (Fig. 3.3*A*,*B*). We did not find any significant amplitude differences among the four conditions. Next, compared with the responses to *GnLn* strings that contained no lexical chunks at either level, the topographical pattern of response amplitudes in conditions that include lexical chunks (*GwLw*, *GwLn*, and *GnLw*) did not show any significant differences in any sensors after multiple comparison correction (Fig. 3.3*C*). However, the difference topographies showed distinctive patterns of amplitude distribution (90–130 ms, higher on the left frontal area and lower on occipital area). These results suggest that lexical detection could induce changes in the configuration of neural sources, rather than in response amplitude. Therefore, we investigated the topographic patterns to infer the different configurations of neural processing across conditions.

The analysis of TANOVA revealed significant differences between the topographies of *GnLn* and response patterns in *GwLw*, *GwLn*, and *GnLw* conditions (Fig. 3.3*D*, high-

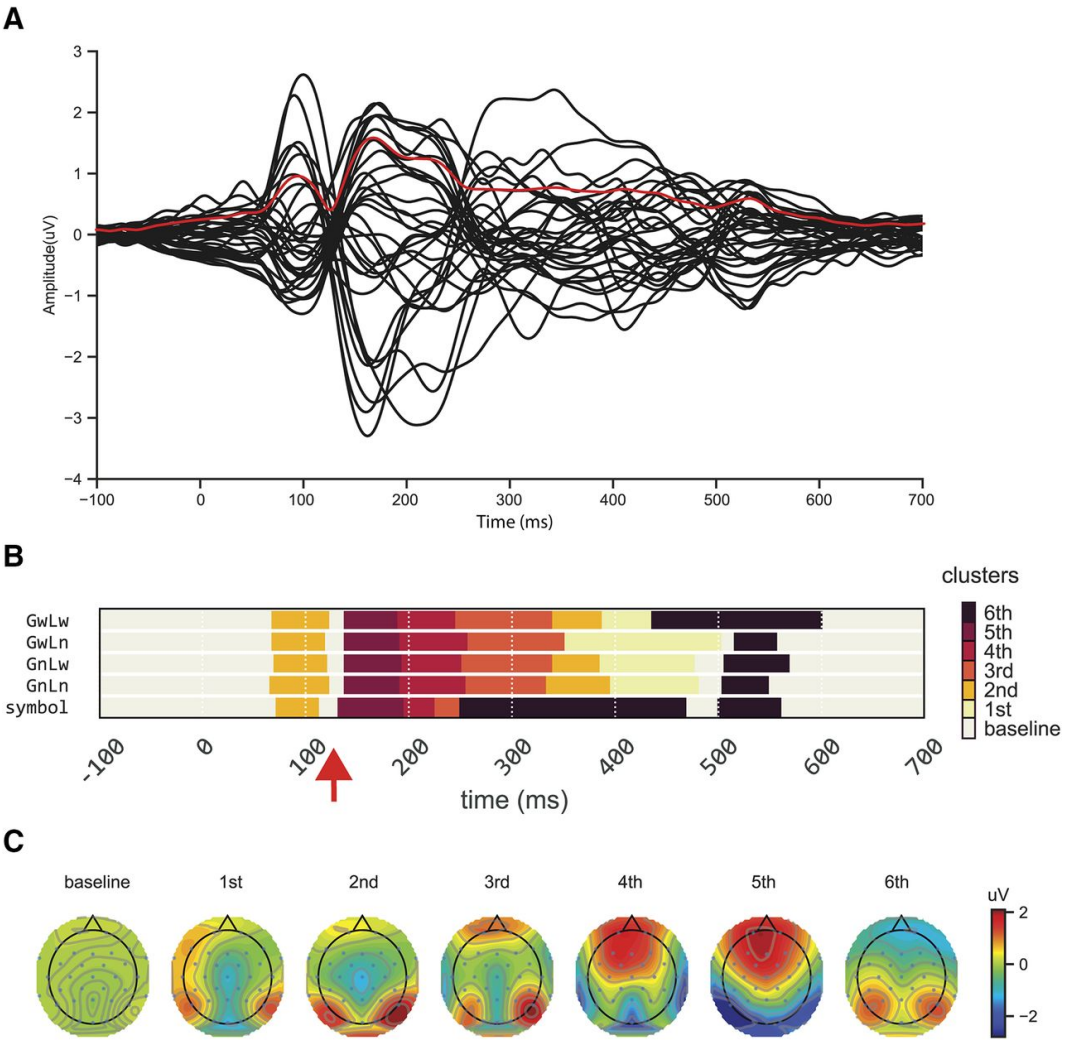

Figure 3.2.: **The dynamics of ERP responses and clustering results.** ***A***, Averaged ERPs waveform responses of all conditions from 32 electrodes (black lines), and RMS waveform response across all electrodes (red line). ***B***, Temporal clustering results of topographies for four conditions (*GwLw*, *GwLn*, *GnLw*, and *GnLn*) and a baseline symbol condition (symbol). Different colors represent distinct clusters. Samples in the same color but at different time points indicate that they are grouped into the same cluster, sharing similar features but occurring at different times. The temperature of colors represents the rank of the cluster distance relative to the cluster baseline (cluster defined by the baseline period). Approximately 80 ms after stimulus onset, a novel cluster (the second cluster) appears at the same time across five conditions, followed by another new cluster. However, in the symbol condition, the third cluster (∼250 ms) appears earlier with a much shorter duration than four-character string conditions. ***C***, The topographies of each cluster.

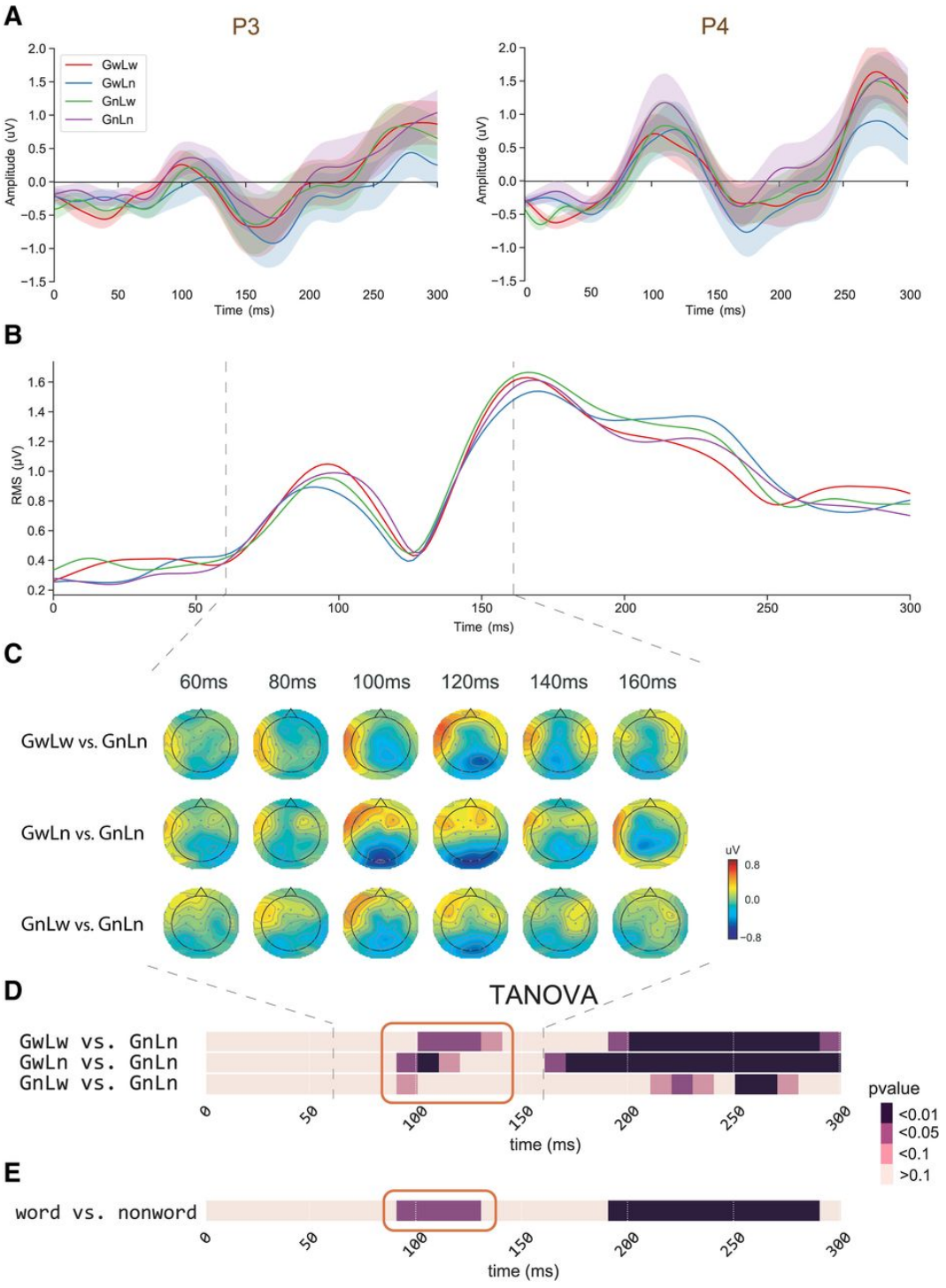

Figure 3.3.: **The effects of lexicalized chunks revealed in the paired comparisons between *GnLn* condition and the other three lexical conditions (*GwLw*, *GwLn*, and *GnLw*). *A***, ERP waveform responses in the representative channels P3 and P4. One-way ANOVA did not reveal any response amplitude differences among conditions. ***B***, RMS waveform responses of all channels. No amplitude difference was found. ***C***, Topographical comparisons of response amplitude. Each row shows a comparison across time. The color scheme depicts the differences in response amplitude between conditions. No significant difference was found on the electrodes after the multiple comparison correction (FDR). ***D***, The temporal dynamics of TANOVA on paired comparisons (uncorrected). The red boxes highlight the earliest latency when the significant differences were observed. All three conditions show evidence of early lexical detection. ***E***, The temporal dynamics of TANOVA on the comparison between *GnLn* and the average of three lexical conditions, corrected by temporal clustering analysis with a corrected threshold of 0.05 (Maris & Oostenveld, 2007). The lexicality effects emerge around 100 ms.

lighted in the red box). The differences were first observed at 90 ms after stimulus onset. The differences were most substantial in the *GwLn-GnLn* comparison as the significant level at $p < 0.01$ for the following 20 ms. The pattern differences were also observed in the *GwLw-GnLn* comparison, but the significant differences started later at 100 ms and lasted for 30 ms. The *p* value of the 90- to 100-ms time bin in the *GnLw-GnLn* comparison was 0.0539, and the relative Bayes factor (BF10) was 2.3371, which indicates weak evidence in favor of H1 (Held & Ott, 2018). The TANOVA of the comparison between *GnLn* and the average of the other three lexical conditions showed significant differences around 110 ms (Fig. 3.3*E*, highlighted in the red box). TANOVA revealed that topographic differences occurred before 130 ms between lexicalized chunks at either level and the non-lexicalized *GnLn* condition. These results suggested that the early-stage process related to the detection of lexicality. The global level information may facilitate the detection because the effect size ranked from biggest to smallest in the order *GwLn*, *GwLw*, and *GnLw*.

We also observed significant differences in later responses. *GwLn* has a significant cluster starting at 150 ms, followed by *GwLw* that has a significant cluster starting around 190 ms. *GnLw* does not have a significant late cluster until 230 ms. These latency differences at a later stage suggested that the lexical processes could first occur at the global level. We further investigate these dynamics in the next section.

Processing of chunks at different levels We analyzed response amplitude and applied pattern analyses between conditions with different lexical status either at the global level or at the local level to test the dynamics of chunk processing. We compared ERPs from the P3 and P4 channels for words and nonwords, separately at the global and local levels (Fig. 3.4*A*,*B*), as well as RMS from all channels (Fig. 3.4*C*). Paired *t* tests were applied to these data. We found that the lexicality effects occurred as early as 160 ms at the global level, but much later around 250 ms at the local level. These effects are in selected parietal channels but are absent in the RMS, suggesting the effects are narrowly distributed, which is consistent with the distribution results in Figure 3.4*D*. When comparing the strings that contained a global-level word (*GwLw*, *GwLn*) with the strings that contained global-level nonword (*GnLw*, *GnLn*), the significant differences were observed in the electrodes over the middle parietal and left frontal-temporal regions (Fig. 3.4*D*, indicated by white points) starting around 170 ms (Fig. 3.4*D*, red arrow). When comparing the strings that contained a local-level word (*GwLw*, *GnLw*) with the strings that contained local-level nonword (*GwLn*, *GnLn)*, the significant differences in response amplitudes started much later at the latency around 230 ms in the electrodes over frontal and parietal-occipital regions (Fig. 3.4*D*, blue arrow).

The TANOVA results in Figure 4*E* further showed that response patterns between distinct global-level lexical status were statistically significantly different began around 160 ms after the stimulus onset (Fig. 3.4*E*, red arrow), and the significant pattern differences

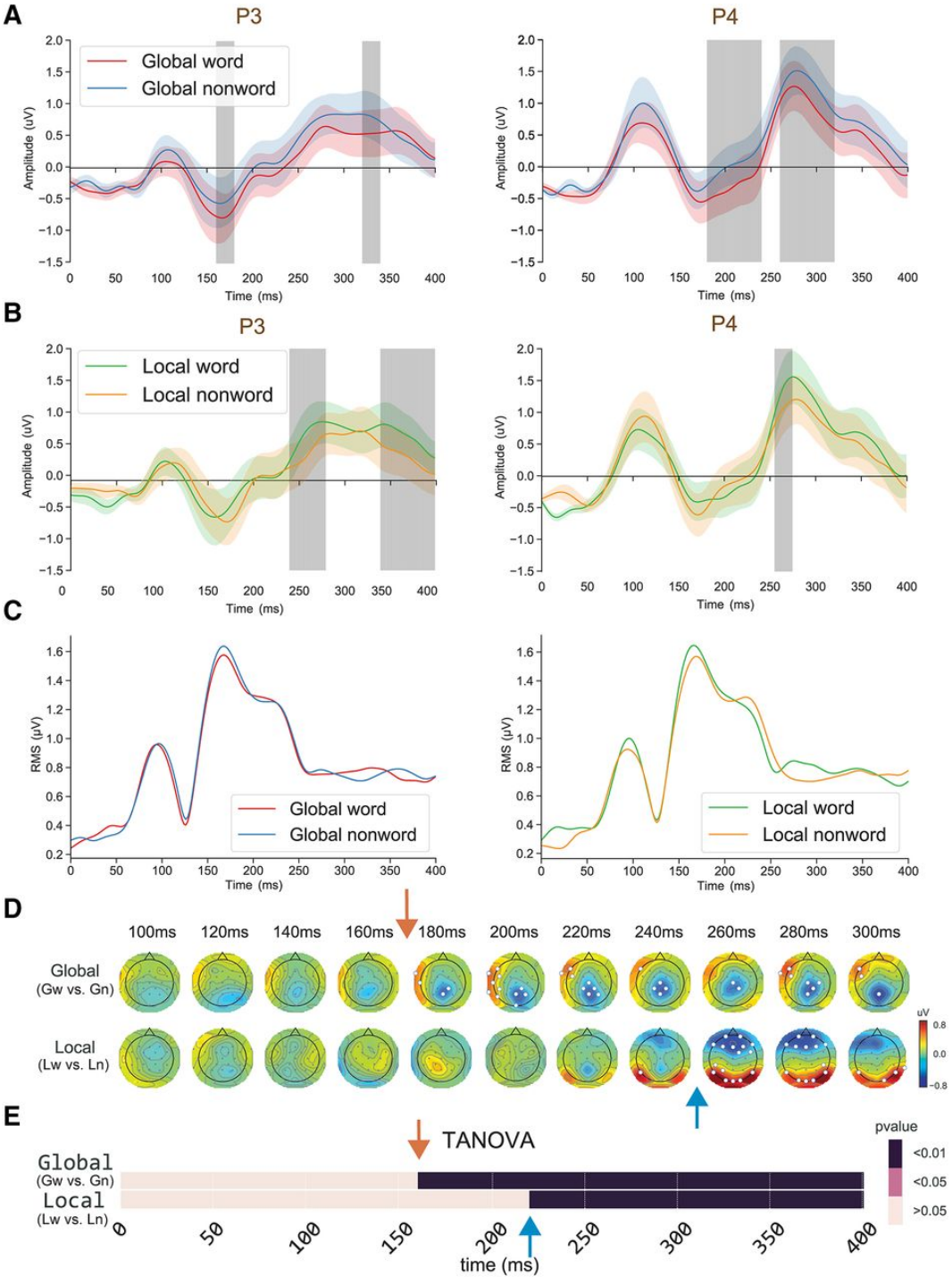

Figure 3.4.: **The processing dynamics of chunks at global and local levels.** ***A***, ERP waveform responses of the global lexicality effect in the representative channels P3 and P4; *t* tests revealed the effects occurred around 160 ms. The shaded area indicates $p < 0.05$. ***B***, ERP waveform responses of local lexicality effect in the representative channels P3 and P4. The effects occurred around 250 ms, later than the global lexicality effect in ***A***. The shaded area indicates $p < 0.05$. ***C***, RMS waveform responses of all channels on global lexicality and local lexicality comparisons. No amplitude difference was found. ***D***, Analysis of response amplitude in topographical comparisons between different lexical status at the global level (upper row) and at the local level (lower row) across time. The color scheme depicts the differences in response amplitude between conditions, and the white points superimposed on the topographies indicate the electrodes that showed significant differences after multiple comparison correction (FDR). ***E***, The temporal dynamics of TANOVA results. The results showed a distinct starting time of significant response pattern differences between different lexical status at the global and local levels. The red arrows in all plots indicate the earliest latency of difference in the global level comparison, and the green arrows indicate the earliest latency of difference in the local level comparison. The results were corrected by a temporal clustering analysis with a corrected threshold of 0.05(Maris & Oostenveld, 2007).

at the local level began around 220 ms (Fig. 3.4*E*, blue arrow), consistent with the observation in Figure 4*A*,*D*, as well as the observations of late differences in Figure 3*D*.

3.2.6. Discussion

This study investigated the processing dynamics for written texts that included different levels of chunks, such as word and phrase. With the stimuli of Chinese four-character strings that contain multiple grain-size language chunks, the behavioral results showed that the recognition of lexicalized local chunks was affected by the lexical status of global chunks, but not vice versa. These results suggest that the processing of chunks at the global level is prioritized over the processing of local ones during reading. Moreover, the earliest EEG responses showed distinct patterns between lexicalized and non-lexicalized chunks, and the latency of successive EEG responses was faster when processing chunks at the global level than that for local chunks. These consistent behavioral and electrophysiological results suggest that two distinct stages successively operate in the early stage of reading for the detection of potential chunks and further processes operated on the detected chunks at multiple levels.

Detection of chunks at 100 ms

In the clustering results (Fig. 3.2), a "temporal gap" was observed in the early EEG reading responses, which separated the processing from 80 to 200 ms into two distinct clusters, suggesting different neural bases and possible distinct functions. Furthermore, the response patterns of the earliest cluster around 100 ms were modulated by the lexical status of chunks at both global and local levels (Fig. 3.3). These findings are consistent with the early lexical familiarity checking mechanism proposed in the E-Z reader model (Reichle et al., 2003). Language chunks and their lexical status should be checked before accessing the semantics. In other words, the familiar lexical chunks are detected before subsequent processes (e.g., semantic retrieval). This is especially important in the writing system that lacks explicit boundaries for lexicalized chunks/phrases, such as written Chinese. Our results suggest such lexical checking/detection can occur early in the reading process around 100 ms and extend to multiple chunk levels.

What factor enables this early chunk detection in reading? Top-down mechanisms have been proposed to account for perceptual and cognitive functions, such as the application of prior knowledge or prediction of the global shape information in object recognition (Bar, 2003; Bar et al., 2006; Panichello, Cheung, & Bar, 2012). The detection of language chunks at multiple levels during reading involved the left frontal regions and occipital regions (Fig. 3.3*A*), similar to the top-down modulation by the early feed-forward projection of low spatial frequency information (Bar et al., 2006). In previous research, high-frequency words can be easily detected and recognized (Ellis, 2002; Monsell, 1991). The transparency (MacGregor & Shtyrov, 2013) and decompos-

ability (Abel, 2003; Vannest, Polk, & Lewis, 2005) also affect the mental encoding of complex words, phrases, and idioms. However, individual differences in reading may make the perception of these physical attributes vary across individuals. Therefore, the factor that leads to the early chunk detection is likely to be the familiarity of these attributes. The effect of familiarity has been demonstrated in improving language retrieval (Bannard & Matthews, 2008; Zheng, Li, & Xiao, 2015). In this study, we controlled the familiarity by only using stimuli that were rated at the extreme degree of familiarity, either very familiar words or strange nonwords. We speculate that the familiarity of lexical-orthographic features (such as frequency and decomposability) is the criterion of chunk detection, and it can apply simultaneously at both global and local levels during early reading processes.

The priority of processing global chunks

Our behavioral results demonstrated that the processing of a global chunk significantly affected the lexical decision of lexicalized local chunks. In contrast, the local chunks had no impact on the lexical decision of the lexicalized global chunk. The unidirectional effect suggested that the processing of global level chunks had priority over the processing of their constituents. EEG further provided evidence supporting the temporal hierarchy in processing global and local chunks. The EEG results showed that the processing of global chunks started around 160 ms, while the onset of local chunk processing was much later ($\sim$ 220 ms). These EEG results, together with our behavioral data, demonstrated that after the simultaneous chunk detection at both levels, the processing of different sizes of lexical chunks began at different times: the processing of global chunks preceded that of local chunks.

The priority of global information has been demonstrated in many cognitive domains. Gestaltism (Dewey, 2018; Heider, 1977) considers the global more informative than the aggregation of its parts. In vision, the global precedence effect (Navon, 1977) suggests that recognizing a scene is hierarchical and global processing has priority over local processing. In contrast, local processing is subject to the top-down reevaluation and integration into global processing. Similarly, the top-down facilitation of visual object recognition also implies that the activation of high-level information will be faster than the lower-level information (Bar, 2003; Bar et al., 2006). In linguistics, the word superiority effect (Reicher, 1969) suggests that the processing of a word at the global level interacts with letter identification (McClelland & Rumelhart, 1981). This study further demonstrates the influences of phrases on words. Our results expand previous research and suggest that the global-priority mechanism can be applied across multiple levels in a hierarchical manner in the linguistic context. The priority of global chunks is consistent with the information theory (Shannon, 1948): larger chunks contain more context information and less internal entropy, which can prevent ambiguity.

Paralleled processing of chunks at both levels

The behavioral results revealed that the judgment of a non-lexicalized phrase at the global level was more difficult when the task-unrelated chunks were familiar words at the local level. These results indicate that local processing might be initiated before the finish of global processing. The EEG results further support that processing at both levels temporally overlapped: the response patterns of processing global chunks continued after the start of local processing response patterns (Fig. 3.4). This observation of partially temporal overlap in the processing of part-whole hierarchies is consistent with simultaneous processing mechanisms implemented in connectionist networks (Hinton, 1990). A scheduler could control the participation of processing at different levels. Should processing a chunk exceed expected duration, the processing of chunks at other levels would occur. Moreover, the topographic patterns showed left lateralization for processing chunks at the global level, whereas both hemispheres engaged in processing chunks at the local level (Fig. 3.4), suggesting possible anatomic differences that mediate the partial temporal paralleled processes at both levels.

Chunking in a broader cognitive perspective

Various cognitive functions can exert a top-down influence on early perceptual responses. For example, attention is one of the most common functions that modulate early perceptual responses, such as increasing the response gain in the visual (Fries, Reynolds, Rorie, & Desimone, 2001), auditory (Poghosyan & Ioannides, 2008), and somatosensory (Steinmetz et al., 2000) domains. The current study offers a new top-down influence in a linguistic context. The lexicality/accessibility of the character combination determines the way of chunking and recombination of characters to form representations at both global and local scales. Such reconstruction of representations may modulate the early visual responses in reading.

Top-down influence provides a common framework that links among cognitive systems. For example, orofacial motion alters speech perception, such as in the McGurk effect (McGurk & MacDonald, 1976), and shortens latency of early auditory responses (Van Wassenhove, Grant, & Poeppel, 2005). Speech articulation dampens the auditory responses to speech feedback (Houde, Nagarajan, Sekihara, & Merzenich, 2002) and modulates the sensitivity to auditory stimuli via motor-to-sensory transformation (S. Li, Zhu, & Tian, 2020; Ma & Tian, 2019; Tian, Ding, Teng, Bai, & Poeppel, 2018; Tian & Poeppel, 2010, 2013, 2015; Tian, Zarate, & Poeppel, 2016). The current study provides evidence supporting that the language system can penetrate and influence visual processing.

Chunking, which deducts combinatory representations into more basic linguistic units for processing, plays a crucial role in language comprehension. Previous studies suggest that linguistic chunking arguably occurs in complex morphology such as decom-

posing compounds into morphemes – the smallest linguistic unit that carries meaning (Fiorentino & Poeppel, 2007; Stockall & Marantz, 2006). The current study further demonstrates that phrases can be segmented into smaller linguistic units based on lexicality at both global and local levels. Our results bridge chunking in morphology with chunking in sentences based on semantics and syntax (Ding, Melloni, Zhang, Tian, & Poeppel, 2016), and even higher linguistic levels such as paragraphs or an entire text based on formal structures and conceptual flow (Teng et al., 2020). A complete picture of chunking operation across all levels of linguistic hierarchy emerges.

The two-stage processing suggested by our results may contribute to the debate regarding the accessible units in complex words (Giraudo & Dal Maso, 2016). Some studies suggest that morphologic decomposition occurs only in semantically transparent morphologic pairs (e.g., hunter–HUNT; Meunier & Longtin, 2007). In contrast, other studies found semantically-opaque but morphologically-complex words (e.g., corner = corn + er) also show decomposition (M. H. Davis, 2004; J. T. Devlin, Jamison, Matthews, & Gonnerman, 2004; Gold & Rastle, 2007). That is, the surface morpheme-like unit that could be an interface between form and meaning is accessible regardless of the semantic relation between the global level and its constituents (J. T. Devlin et al., 2004; Gold & Rastle, 2007). Our results are more consistent with the latter view and suggest that this surface morpheme-like unit could be detected automatically as long as it is available. Specifically, these results show that bi-character words, which are bi-morpheme units, are also automatically decomposed from phrases, suggesting that the surface morpheme-like unit in the decomposition is not limited to the basic linguistic morphemes. Furthermore, the access of the surface morpheme-like unit has been localized over the occipito-temporal and left inferior frontal regions (J. T. Devlin et al., 2004; Gold & Rastle, 2007; Meinzer, Lahiri, Flaisch, Hannemann, & Eulitz, 2009; Pliatsikas, Wheeldon, Lahiri, & Hansen, 2014), which is consistent with our EEG topographic pattern (Fig. 3.3*C*). Orthographic typicality and lexicality modulate reading responses around 100 ms (Faísca, Reis, & Araújo, 2019; Hauk et al., 2006), which is also consistent with the detection timing in our observations.

The "global first" principle in different levels of accessible units has been observed in morphology (Bybee, 1995), letter detection (Han, Yund, & Woods, 2003), and general vision (Chen, 1982; B. Wang, Zhou, Zhuo, & Chen, 2007). All these results are consistent with our findings that processing global level information possesses priority (Fig. 3.4). Last, the discrepancy between global and local information affects ERP responses as early as 250 ms (Han, Fan, Chen, & Zhuo, 1999), suggesting a possible initiation time of parallel processing that is consistent with our results (Fig. 3.4).

The chunking operation is universal among sensory modalities for processing information that is beyond cognitive capacity. However, the nature of stimuli among sensory modalities may differentiate the possible neural mechanisms that mediate chunking. For example, linguistic information unfolds over time in speech, whereas the informa-

tion can be available at the same time in reading (e.g., visual field and reading span). Therefore, temporal processing such as neural oscillations might be a potential dominant mechanism for chunking in the auditory domain with neural entrainment to acoustic features (such as prosodic cues and speech envelope; (H. Luo & Poeppel, 2007)), top-down rhythmic and melodic template (Di Liberto et al., 2020; Nozaradan, Peretz, Missal, & Mouraux, 2011), semantic and syntactic cues (Ding et al., 2017, 2016), as well as structures and formats of language (Teng et al., 2020). However, in the visual domain, additional spatial information can be available at the same time. Chunking is more likely based on the template from higher hierarchy, such as an orthographic template in global/local letters (Kimchi, 1992) and mental representation of lexicality in the current study.

Based on all results, we tentatively put forward a workflow of processing multiple-level information in reading (Fig. 3.5). The segmentation occurs in an early and short time window, and possible chunks at all levels are detected based on the familiarity of lexical-orthographic features (detection stage). The chunks at each level are further processed with distinct temporal characteristics (processing stage). Specifically, the processing of global chunks possesses priority over the local chunks, while the processing of local chunks can launch before global chunk processing finishes. Hence, the processes of chunks at two levels have a partial temporal overlap that enables interaction across levels before final integration.

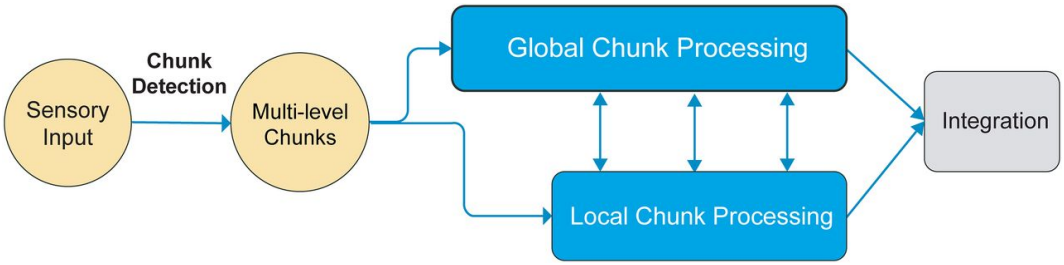

Figure 3.5.: **Schematic diagram of proposed two-stage chunking operation in reading.**

Because our primary goal was to test the relation between lexicality and chunking at different levels, we controlled the lexical-orthographic features such as the number of strokes, and frequencies. Theoretically, lexical access arguably occurs earlier than semantic processing. It is more likely that lexical factors are the primary factors mediating the effects that we observed. Semantic attributes could be another factor influencing the late process of chunking. It would be of interest to study semantics in chunking and obtain a complete understanding. Moreover, we investigated the computational dynamics of chunking in reading by testing the response latencies. EEG is one of the optimal tools to test the dynamics and latency, but not an optimal tool for inferring the spatial location of sources. The spatial distribution in topography is a distorted and incom-

plete representation of underlying neural sources because the topography is most likely a manifestation of a mixture from multiple neural sources. To avoid confusion, we only take advantage of changes in topographies across time or across conditions to infer the neural dynamics (Tian & Huber, 2008; Tian et al., 2011; X. Wang et al., 2019; Yang et al., 2018). Nevertheless, the location of the chunking operation is another aspect of interest. We plan to use fMRI for further investigation.

3.3. Conclusion

The current study investigated the chunking mechanism in reading. Consistent behavioral and EEG results suggested that multiple levels of chunks are realized via two distinct stages of chunking in the early time course of reading. The first stage detects lexicalized chunks at all levels of grain-size. In the second stage, in contrast, the processing at the global level precedes the local level and later results in a parallel and interactive process. This study revealed the rich dynamics of chunking operation during reading, which provides the starting computation for comprehension of hierarchical language systems.

4 | Rapid familiarity detection for text chunks in surrounding text

Abstract

I attempted to replicate the study in Chapter 3 using another behavioral experiment and another EEG experiment. The stimuli used in Chapter 3 were presented in isolation, rather than within the surrounding text. However, in natural reading environments, the cognitive units are usually surrounded by more text. To test whether the findings in Chapter 3 will generalize to a more natural reading environment, the stimuli in replication experiments used rare Chinese characters (as surrounding text) to fill a four-character Chinese string (as the reading target). The results of Chapter 3 were partially replicated: behavioral experiments in the original and new studies showed similar results; both EEG experiments found early detection of available cognitive units, but the new study neither replicated nor falsified the global priority effect of cognitive units. Several possible explanations for the absence of global precedence effects will be discussed in Section 4.4.

4.1. Introduction

Readers need to segment the reading material into chunks/units that are small enough for cognitive processing, but the segmentation mechanism remains under debate. As a part of this debate, many psycholinguists (e.g., Bybee, 1995; Fiorentino et al., 2014; Sinclair, 1991; Taft & Forster, 1976) argued whether a compound word (e.g. "neuroscience") is a single unit or consists of component units (e.g., "neuro" and "science") during cognitive processing. According to dual-route models (Andrews et al., 2004; Koester et al., 2007; MacGregor & Shtyrov, 2013; Semenza & Luzzatti, 2014), both the compound and its components can be processed. If we extend our view to not only compound words but all units in sentences, the disagreement becomes even sharper. Words seem to be the most obvious units because many writing systems explicitly mark word boundaries (e.g., by spaces), consequently, segmenting into word units seems easy and natural. However, psycholinguists found that morphemes, which are the minimal meaning-form pairs, can be the processing units especially when the multimorphemic words are too complicated or too infrequent for being recognized as single units (Andrews et al., 2004; Fiorentino et al., 2014; Koester et al., 2007; MacGregor & Shtyrov, 2013; Semenza & Luzzatti, 2014; Taft & Forster, 1976). Frequent phrases and idioms, which are multi-word expressions, can also be stored in our mental lexicon and be processed as units (Arnon, 2015; Arnon & Priva, 2013; Arnon & Snider, 2010). All those findings support the idea that cognitive units in language are not at a specific linguistic level but can be across many levels.

The flexibility of cognitive units asks for a cognitive mechanism to organize the processing. Readers can understand the meaning of a symbol sequence only after they detect the cognitive units in the sequence. The E-Z reader model for eye movements during reading (Rayner, 2009; Reichle et al., 2013, 2003; Reichle & Sheridan, 2015; Reichle, Tokowicz, Liu, & Perfetti, 2011) describes such a procedure: the identification of a word begins with the identification of its orthographic form. The orthographic identification is named the *familiarity check* or *L1* in the model; it is the first stage of lexical access and it provides information for the planning of the next saccade/further analysis. The familiarity detection is only shallow lexical access so it should occur at an early stage. For example, the ERP component N1, which occurs around 100 ms after stimulus onset, is sensitive to the orthographic familiarity of words (Araújo, Faísca, Bramão, Reis, & Petersson, 2015). EEG's high temporal resolution also provides researchers with a tool to test the more precise time course of familiarity detection. Reichle et al. (Reichle et al., 2011) predicted that the familiarity check of peripherally displayed letter strings starts from 150∼178 ms after the first fixation based on their EOG/EEG data and E-Z reader model. Dufau et al. (Dufau, Grainger, Midgley, & Holcomb, 2015) claimed that the effect of word frequency emerges between 120 to 160 ms. By checking the multivariate pattern analysis (MVPA) scores on the time course, Ling et al. (Ling, Lee, Armstrong, &

Nestor, 2019) found the discrimination of a word's orthographic features arises soon after 100 ms from the word's appearance. These findings did not provide a very consistent timing and we can still wonder if the detection can occur even earlier.

To complete lexical access, advanced analysis of the unit, such as semantic access, needs to follow the familiarity detection. In the detection stage, multiple familiar units may be detected at the same time (e.g. *hot*, *dog*, and *hotdog*) and some of these units may take priority over others. Studies have found the recognition of letters within words is faster than recognition of stand-alone letters (Reicher, 1969) and the recognition of words within grammatically correct word sequences is more accurate than in agrammatical scrambled sequences (Snell & Grainger, 2017). These behavioral findings already show that the language elements can be facilitated by meaningful context and suggest processing priority for larger text chunks.

In Chapter 3, I described a two-stage procedure for the chunking operation. The study used Chinese four-character strings as the stimuli. The four characters comprised either a two-word phrase, an idiom, a random word pair, or a meaningless string (see Table 3.1 for details and examples). There were two experiments in the study: a behavioral and an EEG experiment.

In the behavioral experiment, the participants were asked to make lexical decisions on either the first half, last half, or the whole string. The reaction time (RT) showed the half-string lexical decision on phrases was significantly faster than on random word pairs, which suggests that the global lexical status can affect the process of local lexical status; but there was no significant difference in the whole-string lexical decision RT between phrases and idioms, which suggests that the local lexical status cannot affect the process of global lexical status in return. Together, these results suggest there exists global precedence during lexical/familiarity access.

In the EEG experiment, the participants were not asked to perform a task for the trials with stimuli but just pay attention to the stimuli. Their brain responses over time were compared across the stimuli conditions. The comparison between the only non-word stimuli (random character sequences) and the other stimuli showed a significant difference around 100 ms after the stimulus display. That showed the latency of familiarity detection during reading is around 100 ms. What's more, significant differences between the global-word group (phrases + idioms) and the global-nonword group (random word pairs + random character sequences) occurred around 160 ms after the stimuli onset. At around 220 ms, differences occurred between the local-word group (phrases + random word pairs) and the local-nonword group (idioms + random character sequences). In other words, the effect of global lexicality arose after 160 ms and the effect of local lexicality arose after 220 ms. Together, the EEG results showed that the reader could detect the familiar lexical chunks around 100 ms after seeing the text and then recognize the lexical chunks; the recognition of global familiar chunks is earlier than the local familiar chunks.

The experimental results in Chapter 3 suggested a processing flow for the cognitive units. The stimuli used in Chapter 3 (and also in other studies that investigate the recognition of words/compounds/phrases) are presented in isolation rather than in surrounding text. However, cognitive units, in a natural reading environment, are commonly surrounded by more text. This text blurs the outer boundaries of the cognitive units so the reader may be distracted by surrounding text while detecting or recognizing the targets. Therefore, the isolated stimuli may make the evidence insufficient to be generalized to natural reading. In order to overcome this limitation, I tested whether the two-stage procedure for the chunking operation described in the previous chapter still occurs with the distraction of the surrounding text.

In the current study, the stimuli were 6-char strings, consisting of the original 4-char strings surrounded by extra characters. The extra characters in the stimuli blur the boundary(ies) of the original 4-character target, and to some extent simulate the natural reading in this way. The study in this chapter partially replicated the results in Chapter 3: the behavioral experiments in the original and current studies show similar results; for the EEG experiments, both studies found the early effect of the familiar vs. unfamiliar chunks, but the current study did not replicate the global-first effect in the original study. In the subsequent part of this chapter, I will show the details of the current experiments and discuss the similarities and differences in the results.

4.2. Methods

4.2.1. Participants

In total, 33 healthy native Chinese speakers (14 males, mean age 26 years, range 20 – 35 years) with normal or corrected-normal vision participated in one or both of the experiments for financial compensation. The behavioral experiment recruited 20 participants and the EEG experiment also recruited 20 participants (seven participants attended both experiments). No participants produced extensive EEG artifacts that required their exclusion. The experiments were approved by the Ethics Assessment Committee Humanities of Radboud University. Written consent forms were obtained from all participants before the experiments.

4.2.2. Materials and experiments

The current experimental design basically followed the design of the previous experiment (please see Section 3.2 since the current study was intended to replicate the findings in Chapter 3. However, the current study was also intended to test whether the two-stage procedure for the chunking operation described in the previous chapter still occurs with the distraction by surrounding text. Therefore, the reading targets are still

the four-character Chinese string with different global/local familiarity, but each target was padded by two extra characters as the simulation of the surrounding text. The extra characters were extremely rare, thereby preventing them from forming meaningful chunks that would make participants pay attention to the unexpected meaningful chunk and affect the following analyses. In order to prevent it, I used extremely rare characters[1] as the extra characters. In the behavioral experiment, there was a line under the local/global part of the reading target in each stimulus. If the trial asked the local decision, the underline covered the left (or right) two characters of the reading target, and two extra characters were added to the left (or right) side of the reading target. If the trial asked the global decision, the underline covered all four characters of the reading target, and one extra character was added to each side of the reading target. In this way, the underlined text (i.e., decision target) is always in the center of the six-character stimulus so participants need not move their eye fixations (which may cause unexpected cognitive processing). As simple examples, the stimulus may be "考试成绩" or "考试成绩" (note that 考试成绩 is a Chinese phrase and "" and "" are two rare characters).

In the EEG experiment, one extra character was always added to each side of the reading target. In addition, unlike the underlined stimuli in Chapter 3, there was no underlining under any text in any stimulus in order to make the stimuli look more like in a natural reading environment. As a simple example, the stimulus may be "考试成绩". There were some attention-checking trials in both the previous and current EEG experiments since the passive reading task still required the participants to pay attention to the stimulus. In the previous design, the participant needed to recognize the digital number; in the current design, the participant needed to recognize two Chinese numbers shown in the middle of four extra characters, which makes the attention-checking trial look like a normal trial. As a simple example, the stimulus may be "五三" (note that "五" and "三" are two Chinese numbers and the rest are rare characters).

I used the same condition labels as in the previous study (*GwLw*, *GwLn*, *GnLw*, and *GnLn*; see Table 3.1 for explanation) to represent the type of stimulus.

4.2.3. Behavioral data analysis

All participants had response accuracy exceeding 73%, and the average accuracy was 92%. No participant's data were excluded. Trials with incorrect responses or with reaction time exceeding three standard deviations for each participant were removed before analysis. We applied t-tests for testing specific hypotheses.

[1]The EXTRA characters were randomly selected from the characters with no frequency information in Chardb (https://chardb.cls.ru.nl/) and filtered by the stroke number (ranging from 4 to 13 as in the previous study).

4.2.4. EEG recording

EEG signals were recorded with a 32-electrode active electrodes system (actiChamp system, Brain Products GmbH). Electrode impedances were kept below 10kV. Data were continuously recorded in single DC mode. Data were sampled at 2000 Hz, online referenced to Cz for all participants. While the previous experiment used no specific sensor to record the EOG, the current experiment used three sensors: TP10 and FT10 were placed at the outer canthus of the participants' left and right eyes for monitoring horizontal eye movements (hEOG), TP9 was placed at infraorbital regions of the left eye for monitoring vertical eye movements (vEOG). Because of the specific EOG recording, the participants in the current experiment (unlike those from the previous experiment) were not instructed to rigidly prevent blinks during the trial. As a result, more blinks were made than in the previous study.

4.2.5. EEG data analyses

Due to the high number of blinks in the current experiment, I estimated the linear relationship between the recorded VEOG and HEOG and the signal amplitudes on other sensors by the method of least squares and then removed the eye-movement artifact from the signal. In addition, the lower limit of the band filter was 1 Hz, because EOG artifact estimation required the DC offset to be as small as possible in the time window covering a normal blink. Except for the artifact removal, all preprocessing steps were the same in both experiments.

After the preprocessing, the EEG data were analyzed with a similar workflow as in the previous chapter, which includes two multivariate-based methods [clustering and topographic ANOVA (TANOVA)] and one mass-univariate method (analysis of the topographic distribution of amplitude differences). Please see Section 3.2.4 for the details of these analysis methods.

4.3. Results

4.3.1. Behavioral experiment

Planned *t*-tests were conducted on the comparisons that were consistent with the comparisons in the previous study: In the local task, the reaction time on the *GnLw* items was significantly longer than that on *GwLw* ($t_{20} = 3.02$, $p < .01$, difference = 18 ms Fig. 4.1A, left), suggesting that the nonwords at the global level significantly slowed down the lexical decision of words at the local level. In the global task, we did not find a significant difference between reaction time on *GwLw* and *GwLn* ($t_{20} = 0.72$, $p = 0.47$, difference = 10 ms; Fig. 4.1B, left), suggesting that the lexical status of local chunks

cannot affect the lexical decision of words at the global chunks. Moreover, the reaction time of *GwLn* was significantly longer than of *GnLn* in the local task ($t_{20} = 3.96$, $p < 0.001$, difference = 70 ms; Fig. 4.1B, right); *GnLw* was significantly longer than that *GnLn* in the global task ($t_{20} = 6.39$, $p < 0.001$, difference = 164 ms; Fig. 4.1B, right).

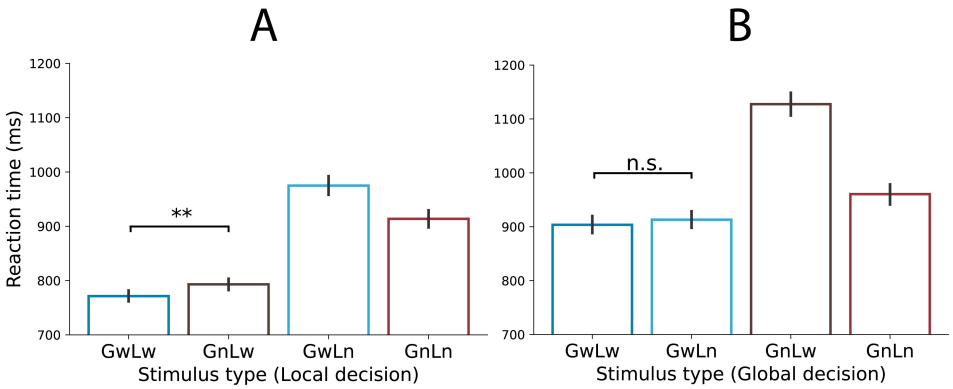

Figure 4.1.: **Results of the behavioral experiment.** Reaction time results in the global (A) and local (B) tasks, respectively. In each plot, condition labels are provided along the x-axis. Error bars represent ±1 SEM (standard error of the mean). Each planned paired test is represented by the line linking two bars; n.s., not significant; ** $p < 0.01$.

Even though the reaction times are shown in the current study are longer in the previous study (about 100 ms longer in the local decision and about 200 ms longer in the global decision; see Figure 4.1 and Figure 3.1), all the comparisons resulted in similar patterns as in the previous study. These similar patterns indicate that even with the distraction by the extra padding, the current results further confirm the priority of the global chunk over the local chunk, and the possible overlapping processing of the global and the local chunks.

4.3.2. EEG experiment

Clustering

As in the previous study, we first conducted clustering analysis to explore the dynamics of ERP responses (Fig. 4.2). All conditions show a standalone cluster about 100 ms after the stimuli onset and a baseline cluster following it. These patterns matched the previous finding and again suggested an early stage and a later stage during processing.

Analysis of the topographic distribution of amplitude differences

As in the previous study, we demonstrated the topographical patterns of response amplitude on different conditions and different time points. Fig. 4.3A clearly shows early

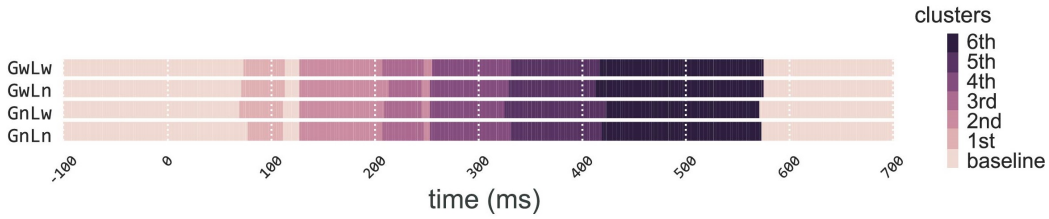

Figure 4.2.: **Temporal clustering results of topographies for four conditions (*GwLw*, *GwLn*, *GnLw*, and *GnLn*).** Samples in the same color but at different time points indicate that they are grouped into the same cluster, sharing similar features but occurring at different times. The temperature of colors represents the rank of the cluster distance relative to the cluster baseline (cluster defined by the baseline period).

differences (with 50~100 ms) between the *GnLn* condition and the other three lexical conditions (*GwLw*, *GwLn*, and *GnLw*). This finding was already explained as the detection of familiar chunks in the previous study, but the current results showed an earlier and clearer effect. For the later stage (after 100 ms), however, both the effects of global processing and local processing are absent in the current study (Fig. 4.3B), which was different from the previous study.

TANOVA

As in the previous study, we conduct the TANOVA method to examine the difference between conditions. The TANOVA result and the topographical demonstrations in the current study are consistent: The difference between the *GnLn* condition and the other three lexical conditions (*GwLw*, *GwLn*, and *GnLw*) are early and clear (Fig. 4.4A), but the difference of two Gw conditions vs. two Gn conditions, and two Lw conditions vs. two Ln conditions are absent (Fig. 4.4B).

4.4. Discussion

The main goal for the current study is to test whether the two-stage procedure for the chunking operation described in Chapter 3 still occurs with distraction by surrounding text. The two-stage processing includes a detection stage and a recognition stage. Generally speaking, the current study replicated the effect of the first stage (detection); the effect of the second stage (recognition) is shown in the behavioral experiment but not in the EEG experiment. I will discuss the two stages separately below.

Most studies related to lexical processing regard recognition as a standalone function. Only a few talk about a pre-recognition stage. For example, E-Z reader (a model of eye movements during reading) assumes a rapid familiarity check of cognitively available

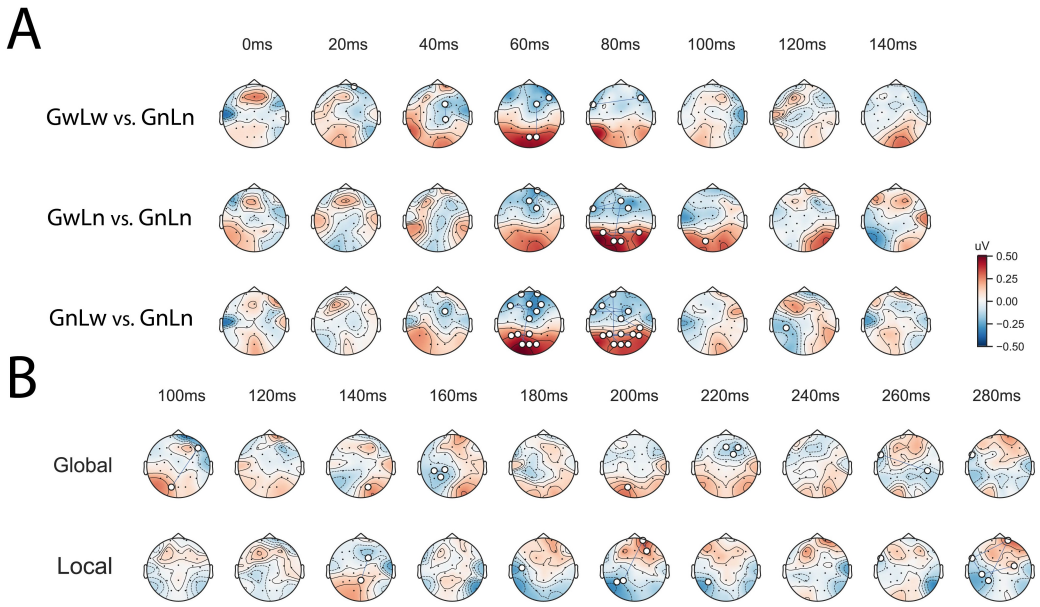

Figure 4.3.: **Topographical comparisons of response amplitude. Each row shows a comparison across time.** The color scheme depicts the differences in response amplitude between conditions. **A**. The processing dynamics of lexicalized chunks are revealed in the paired comparisons between *GnLn* condition and the other three lexical conditions (*GwLw*, *GwLn*, and *GnLw*. **B**. The processing dynamics of chunks at global and local levels are revealed by two *Gw* conditions vs. two *Gn* conditions, and two *Lw* conditions vs. two *Ln* conditions.

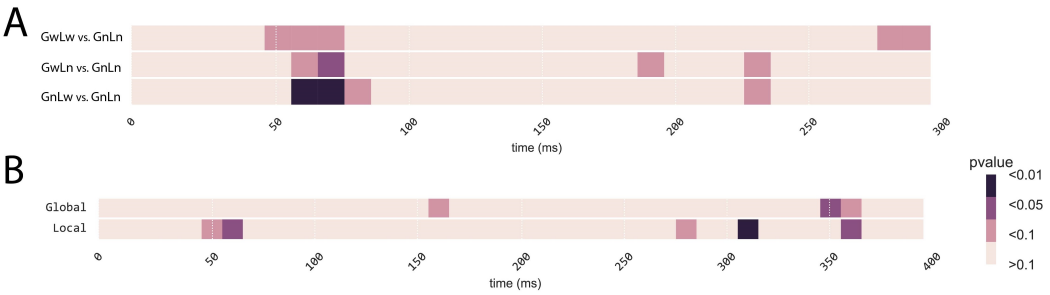

Figure 4.4.: **The temporal dynamics of TANOVA results. A**. The processing dynamics of lexicalized chunks are revealed in the paired comparisons between *GnLn* condition and the other three lexical conditions (*GwLw*, *GwLn*, and *GnLw*. **B**. The processing dynamics of chunks at global and local levels are revealed by two *Gw* conditions vs. two *Gn* conditions, and two *Lw* conditions vs. two *Ln* conditions.

content before the lexical recognition (Reichle & Sheridan, 2015). A detection stage is theoretically necessary before the recognition of flexible targets since "where are the targets?" should be answered before "what are the targets?".

The EEG results in the current study provide more evidence for the detection stage: it shows an early and brief significant difference between the processing of familiar and unfamiliar chunks, and the timing of the differences across global/local levels are similar. The current finding replicated the early effect of familiarity chunks in the previous study. Furthermore, the stimuli in the previous study were four-character strings so the global level of the stimulus had explicit boundaries on both sides, which means the detection of the global units in the experiment was easier than in a natural reading environment. The current finding is based on the stimuli with extra characters padding to the reading target, which blurs the boundary(ies) of even global units. Therefore, the current finding confirmed the detection stage under a more realistic condition.

The detection stage described in the last and the current chapters is similar to the "familiarity check" described in E-Z reader: this stage detects all familiar chunks. The familiarity of these chunks suggests that they are stored in the reader's mental lexicon and ready to be recognized. Moreover, the earliness and the briefness of this stage showing in the EEG result suggest that the detection just pre-activates the recognizable chunks and does not involve deeper processing, such as semantic access. Dissociating the processing into two stages is a strategy to save energy cost since it avoids attempts to recognize unrecognizable chunks. In addition, the detection stage shown in the current EEG experiment is (about 40 ms) earlier than in the previous EEG experiment even though the current experiment contained distraction. I do not have a persuasive explanation, thus future work on the earliness is necessary.

After the detection stage, the pre-activated chunks wait for recognition. However, the pre-activated units may be distributed at different linguistic levels. The processing allocated to the global units and to the local units may be unequal. The processing priorities for global/local units are tested in both the behavioral and EEG experiments.

The current behavioral experiment tested the priority in a more naturalistic reading environment because the extra characters in the stimuli blur the target boundaries which are explicit in the original study. Even with the distraction characters, the reaction times in the current behavioral experiment show a pattern consistent with the results in Chapter 3: the behavioral results show that the lexical status of the global string influences the lexical decision of local lexicalized strings, but not vice versa. Additionally the lexical status of the global/local string influences the lexical decision of local/global unlexicalized strings. The consistent results in both behavioral experiments confirm the priority of global recognizable units over local units during reading; and if the processing of global units is not finished before the processing of local units, the two types of processes interact.

The behavioral experiments show that global units are recognized prior to local units. However, those studies did not clarify whether the priority is for the *temporal* order (i.e., processing global unit earlier than local unit) or the *resource usage* (i.e., processing global

and local units at the same time, but global unit is given more cognitive resources so its processing is faster).

The EEG result shows no significant continuous difference between global familiar/unfamiliar chunks, nor between local familiar/unfamiliar chunks. In other words, the effects of both global processing and local processing (that had been found in Chapter 3) are absent in the current study. Therefore, we cannot observe the temporal order of the global and local processing.

The result fails to replicate the temporal priority of global chunking during reading. However, the result does not falsify the temporal priority of the global chunk since the temporal priority of the local chunk is also absent, not to mention the current behavioral experiment has replicated the priority of the global chunk. If we take the premise that the absent effect is due to the difference in the previous and the current experimental design, it is most likely that the current design reduces the attention of participants to the reading targets. For example:

1. Distracting character: The extra characters were padded to the reading targets to serve as vanilla padding. However, they might strongly distract the attention that should be paid to the reading target, especially because the extra characters were rare and therefore might surprise participants.

2. Narrow attention field: Even though both EEG experiments inserted some number-reporting trials to maintain the participants' attention during the passive reading task, the trials in current experiment required the response to only the middle-two characters (3rd + 4th characters), whereas the trials in previous experiment required the response to the entire strings.Therefore, it is possible that participants ignore the recognition of either the global reading target (centered 4-character string) or local reading target (2nd + 3rd characters or 4th + 5th characters).

3. Missing guidance: The EEG experiment of Chapter 3 used the underline. The underline may have helped focus attention during recognition and enhanced the recognition effect as a result.

As explained above, the absent effect of the recognition stage may be related to reduced attention to the reading target. On the contrary, the replicated effect of the detection stage is insensitive to attention since the detection is automatic. In any further work that aims to investigate the temporal priority of cognitive units at different levels, the experimental design must make sure the participants direct enough attention to the reading targets.

Part II.

Part Two: Computational modeling

5 | Less is Better: A cognitively inspired unsupervised model for language segmentation[1]

Abstract

Language users process utterances by segmenting them into many *cognitive units*, which vary in their sizes and linguistic levels. Although we can do such unitization/segmentation easily, its cognitive mechanism is still not clear. This paper proposes an unsupervised model, *Less-is-Better* (LiB), to simulate the human cognitive process with respect to language unitization/segmentation. LiB follows the principle of least effort and aims to build a lexicon which minimizes the number of unit tokens (alleviating the effort of analysis) and number of unit types (alleviating the effort of storage) at the same time on any given corpus. LiB's workflow is inspired by empirical cognitive phenomena. The design makes the mechanism of LiB cognitively plausible and the computational requirement light-weight. The lexicon generated by LiB performs the best among different types of lexicons (e.g. ground-truth words) both from an information-theoretical view and a cognitive view, which suggests that the LiB lexicon may be a plausible proxy of the mental lexicon.

[1]Yang, J., Frank, S. L., & van den Bosch, A. (2020). Less is Better: A cognitively inspired unsupervised model for language segmentation. *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, 33–45.

5.1. Introduction

During language comprehension, we cannot always process an utterance instantly. Instead, we need to segment all but the shortest pieces of text or speech into smaller chunks. Since these chunks are likely the cognitive processing units for language understanding, we call them *cognitive units* in this paper. A chunk may be any string of letters, characters, or phonemes that occurs in the language, but which chunks serve as the cognitive units? Traditional studies (Chomsky, 1957; Taft, 2013, for example) often use words as the units in sentence analysis. But speech, as well as some writing systems such as Chinese, lack a clear word boundary. Even for written languages which use spaces as word boundaries, psychological evidence indicates that the morphemes, which are sub-word units, in infrequent or opaque compound words take priority over the whole word (Fiorentino et al., 2014; MacGregor & Shtyrov, 2013); at the same time, some supra-word units such as frequent phrases and idioms are also stored in our long-term mental lexicon (Arnon & Snider, 2010; Bannard & Matthews, 2008; Jackendoff, 2002). The evidence suggests that the cognitive units can be of different sizes; they can be words, or smaller than words, or multi-word expressions.

Despite the flexible size of the cognitive units, and the lack of overt segmentation clues, infants are able to implicitly learn the units in their caregivers' speech, and then generate their own utterances. Arguably, children's language intelligence allows them to build their own lexicons from zero knowledge about the basic (cognitive) units in the particular language the child is learning, and then use the lexicon to segment language sequences. Can we mimic this ability of a human language learner in a computer model? This question is often phrased as the task of unsupervised segmentation. Several types of computational models or NLP algorithms have been proposed for segmentation, taking different approaches:

- **Model the lexicon**: A straightforward basis for segmentation is to build a lexicon. One of the lexicon-building algorithms, Byte pair encoding (BPE) (Sennrich, Haddow, & Birch, 2016), is popular for NLP preprocessing. It iteratively searches for the most common n-gram pairs and adds them into the n-gram lexicon. Some other models such as the Chunk-Based Learner (McCauley & Christiansen, 2019) and PARSER (Perruchet & Vinter, 1998) are also based on the local statistics of tokens (e.g., token frequency, mutual information, or transitional probability).
- **Model the grammar**: Some studies attempted to analyze the grammar patterns of sentences and then parse/segment the sentences based on these patterns. To find the optimal grammar, de Marcken (1996) used Minimum Description Length, and Johnson and Goldwater (2009) used the Hierarchical Dirichlet Process.
- **Model the sequences**: Recurrent neural networks and its variations are able to learn the sequential patterns in language and to perform text segmentation

(Chung, Ahn, & Bengio, 2017; Kawakami, Dyer, & Blunsom, 2019; Z. Sun & Deng, 2018; Zhikov, Takamura, & Okumura, 2013).

In general, lexicon models capture only the local statistics of the tokens so they tend to be short-sighted at the global level (e.g. long-distance dependencies). The other two types of models, in contrast, learn how the tokens co-occur globally. Yet, the ways grammar models and sequence models learn the global information makes them more complicated and computing-intensive than the lexicon models.

In this paper we propose a model that builds a lexicon, but does so by using both local and global information. Our model is not only a computational model but also a cognitive model: it is inspired by cognitive phenomena, and it needs only basic and light-weight computations which makes it cognitively more plausible than the grammar- and sequence-learning models mentioned above. We show that our model can effectively detect the cognitive units in language with an efficient procedure. We also show that our model can detect linguistically meaningful units. We further evaluate our model on traditional word segmentation tasks.

5.2. The Less-is-better Model

5.2.1. Cognitive principles

We want our system to mimic human cognitive processes of language unitization/segmentation by simulating not only the behavioral output, but also the cognitive mechanism. We designed such a computational model by emulating three cognitive phenomena: the principle of least effort, larger-first processing, and passive and active forgetting.

The principle of least effort: The essence of the model is a simple and natural cognitive principle: the principle of least effort (Zipf, 1949), which says human cognition and behavior are economic; they prefer to spend the least effort or resources to obtain the largest reward. Since a language sequence can be segmented into different sequences of language chunks, we assume the cognitive units are the language chunks in the sequence which follow the principle of least effort.

Larger-first processing: As we mentioned, any language chunk may be the cognitive unit, short or long. A broadly known finding is that global/larger processing has priority over local/smaller processing for visual scene recognition; an effect named “global precedence" (Navon, 1977). This follows from the principle of least effort: the larger the units we process, the fewer processing steps we need to take. For visual word processing, the word superiority effect (Reicher, 1969) shows the precedence of words over recognizing letters. Recent work (Snell & Grainger, 2017; Yang, Cai, & Tian, 2020) extends global precedence to the level beyond words, and also shows that we do not process

only the larger units: smaller units also have a chance to become the processing units when processing larger units does not aid comprehension. In other words, cognitive units may be of any size, but the larger have priority.

Passive and active forgetting: To mimic human cognition, the model should have a flexible memory to store and update information. Forgetting is critical to prevent the accumulation of an extremely large number of memory engrams. It has been commonly held that forgetting is merely the passive decay of the memory engram over time, but recent studies put forward that forgetting can also be an active process (R. L. Davis & Zhong, 2017; Gravitz, 2019). Passive forgetting by decay can clean up the engrams that are no longer used in our brains. However, our brains may sometimes need to suppress counter-productive engrams immediately. Active forgetting may thus be called upon to eliminate the unwanted engram's memory traces, which enhances the memory management system (R. L. Davis & Zhong, 2017; Oehrn et al., 2018).

5.2.2. General idea

We assume the cognitive units are the chunks in the language sequence which follow the principle of least effort (Section 5.2.1). In other words, the less information we need to encode the language material, the better cognitive units we have. This less-is-better assumption grounds our model, so we named it Less-is-Better, or LiB for short.

The LiB model accepts any sequence S of atomic symbols s: $S = (s_1, s_2, \ldots)$, as the input. A collection of S forms a document D and all D together form the training corpus. S can be segmented into chunk tokens $(c_1, \ldots, c_N)$, where each chunk is a subsequence of S: $c = (s_i, \ldots, s_j)$ and N is the number of chunk tokens in S. The segmentation is based on a lexicon L (Fig. 1) where all chunk types are stored in order. The ordinal number of chunk type c in L is denoted $\Theta(c)$, and $|L|$ is the number of chunk types in L.

Let $I(c)$ be the amount of information (the number of encoding bits) required to identify each chunk type in L, that is, $I(c) = \log_2 |L|$, and $I(S)$ be the amount of information required for the input S, then: $I(S) = I(c)N$. Our model aims to minimize the expected encoding information to extract the cognitive units in any S, which means minimizing $E[I(S)]$, which is accomplished by simultaneously reducing $|L|$ (smaller $|L|$ means lower $I(c)$) and $E[N]$ (the expected number of chunk tokens in S). In practice our model:

1. Starts with an empty L;
2. Randomly selects a D from the corpus and analyzes the S in D;
3. Adds previously unseen symbols s as (atomic) chunk types to L;
4. Recursively combines adjacent chunk tokens into new chunk types, reducing $E[N]$ but increasing $|L|$;
5. Removes less useful types from L, reducing $|L|$;
6. Repeats steps 2 to 5 for a predetermined number of epochs.

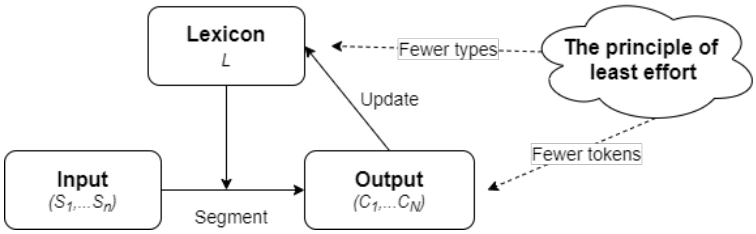

Figure 5.1.: **Information flow in the LiB model.**

The LiB model can segment any string S into a sequence of chunks $(c_1, ..., c_N)$ based on the lexicon L. The chunk types in L are ordered based on their importance inferred from the segmentation. The lexicon quality and the segmentation result mutually affect each other: LiB learns from its own segmentation results and updates L accordingly, then improves its next segmentation (Figure 5.1). The bootstrap procedure makes the model unsupervised.

5.2.3. Implementation

Segmentation

Larger-first selection: An S can be segmented in different ways. For example, if both "going" and "goingto" are in L, and the given S is "goingtorain", then the first chunk token can be "going" or "goingto". The Larger-first principle (Section 5.2.1) dictates that LiB takes the largest substring of S that matches a chunk type in L (in the example case, it is "goingto"), i.e. greedy matching, and selects it as a chunk token (segment). If there is no chunk type in L that matches the current S, the first symbol s becomes the selected chunk token.

Chunk evaluation: In most cases, selecting larger chunk tokens will reduce the number of tokens N in S, but in some cases it will not. Let us continue the example we gave: If "goingtor", "a", "in", and "rain" are also in L, the largest chunk token becomes "goingtor", resulting in the segmentation "goingtor/a/in". If "goingto" had been selected, this would result in "goingto/rain". Hence, selecting the largest chunk type resulted in a larger N. The average chunk token sizes of the two segmentations are 5.5 and 3.6 letters, respectively.

In order to test whether the selected chunk type c reduces N, LiB compares the proposed segmentation to the segmentation that results if c is not in L, i.e., if the second largest chunk type in L is selected instead of c. In case L cannot provide a second largest chunk token, there is no evaluation and c is selected directly. Otherwise, c is evaluated

as “Good" if it results in fewer chunk tokens or in the same number of tokens but with lower total ordinal numbers (i.e., chunks that are higher up in the lexicon):

$$\texttt{segment}(S, L) : S \rightarrow (c_1, c_2, \ldots, c_N)$$
$$\texttt{segment}(S, L - c) : S \rightarrow (c'_1, c'_2, \ldots, c'_{N'})$$
$$\texttt{evaluate}(c) = \begin{cases} \begin{cases} \text{Good} & \text{if } N < N' \\ \text{Bad} & \text{if } N > N' \end{cases} & \text{if } N \neq N' \\ \begin{cases} \text{Good} & \text{if } \sum_{i=1}^{N} \Theta(c_i) \leq \sum_{i=1}^{N'} \Theta(c'_i) \\ \text{Bad} & \text{if } \sum_{i=1}^{N} \Theta(c_i) > \sum_{i=1}^{N'} \Theta(c'_i) \end{cases} & \text{if } N = N' \end{cases}$$

If $\texttt{evaluate}(c)$ is Good, c is selected; otherwise, the second largest chunk token is selected.

Lexicon update

Memorizing: LiB learns new chunks from the segmentation results. There are two types of new chunks in the results: unknown symbols $s \notin L$ and concatenations of known chunks (c_i, c_{i+1}) (with $c_i \in L$ and $c_{i+1} \in L$) that occur consecutively in S. L starts empty, learns the symbol chunks, then the smallest chunks construct larger chunks and the larger chunks construct even larger chunks. Thus, L can contain chunks in different sizes.

The number of all (c_i, c_{i+1}) in the training corpus can be enormous, and most of them are infrequent chunks. In order to reduce the lexicon size $|L|$, LiB will memorize all s, but not all (c_i, c_{i+1}). To recognize the frequent chunks, a strategy is to count all chunks' occurrences and delete the infrequent ones (Perruchet & Vinter, 1998). However, this strategy asks for storing all chunks at the beginning, which is memory inefficient for both a brain and a computer. Thus, LiB adopts a sampling strategy: The model samples from all possible (c_i, c_{i+1}) tokens in the current S and memorizes only the tokens which were sampled at least twice. The probability of sampling a chunk pair is the sampling probability α. The sampling strategy is implicitly sensitive to the chunk token frequency in the text. It makes sure that even without explicit counting, higher-frequency chunks have a higher probability to be memorized. The at-least-twice strategy is not cognitively inspired but heuristic; it helps to prevent memorization of many arbitrary chunks.

Re-ranking and active forgetting: To avoid storing the frequencies of all possible chunk types, and to be more efficient, LiB bypasses explicit frequency counting of chunk types. Instead, LiB encodes the types' importance by their ordinals $\Theta(c)$ in L – the lower the more important. The importance reflects not only the frequency but also the principle of least effort (preference for fewer tokens and fewer types). In general, newly

memorized chunk types are less frequent than known chunk types, so new chunk types are appended to the tail of L. The ordinals of known chunk types also need to be adjusted after new training text data comes in. The chunk evaluation we described in Section 5.2.3 is not only for segmentation, but also for importance re-ranking. The "good" chunk types, which result in fewer chunk tokens in S, will move closer to the lexicon head (i.e., lower ordinal); The "bad" chunk types, which result in more chunk tokens in S, will move closer to the lexicon tail, i.e., they get a higher ordinal number. The updated $\Theta(c)$ of a chunk type is relative to its previous ordinal $\Theta'(c)$ in L:

$$\Theta(c) = \begin{cases} \lfloor \Theta'(c)(1-\Delta) \rfloor & \text{if } c \text{ is good} \\ \lfloor \Theta'(c)(1+\Delta) \rfloor & \text{if } c \text{ is bad} \end{cases}$$

where $0 < \Delta < 1$ is the re-ranking rate. In case the updated $\Theta(c) > |L|$, c will be deleted from L.

Passive forgetting: Obviously, the re-ranking also influences other chunk types whose ordinals are between $\Theta(c)$ and $\Theta'(c)$. So even though the sampling strategy of the memorizer may add a few infrequent chunk types into L, the re-ranker will move them closer to the tail of L. Those chunk types, as well as the "bad" chunk types, are "junk chunks" which increase $I(c)$. The passive forgetter removes them from L to reduce $I(c)$.

The junk chunk types tend to be at the tail of L, but the tail may also store some non-junk types. A cognitive strategy to avoid deleting them is *waiting* for more evidence. So instead of deleting these types immediately, LiB uses a soft deleting strategy: after each training epoch, LiB will select the last $\omega|L|$ (at least one) chunk types in L and assign them a probation period τ. Here, ω is the forgetting ratio and τ is the remaining time until deletion; it is initialized at τ_0 and decreases by one after each training epoch (LiB analyzes one document D in each training epoch). Once the probation time is over, when $\tau = 0$, the chunk is forgotten (i.e., removed from L). If a chunk type was evaluated as "good" during its probation period, its probation is cancelled. The c that occur in fewer documents are more likely to be forgotten.

5.3. Model Training

We trained the LiB model on both English and Chinese materials (Table 5.1). The English material is **BR-phono**, which is a branch of the Brent corpus (Bernstein-Ratner, 1987), containing phonetic transcriptions of utterances directed at children. We used it for testing segmentation of spoken language. LiB accepts the document as an input batch in each training epoch but the utterances in the BR-phono corpus have no document boundaries. We randomly sampled 200 utterances (without replacement) from BR-phono to form one document and repeated this 400 times to create 400 documents

for model training. The Chinese materials are taken from Chinese Treebank 8.0 (**CTB8**) (Xue et al., 2013), which is a hybrid-domain corpus (news reports, government documents, magazine articles, conversations, web discussions, and weblogs). As preprocessing, we replaced all the Roman letters and Arabic numbers with [X], and regarded all punctuation as sequence boundaries.

Corpus	Documents	Sentences	Word tokens	Word types
BR-phono	400	9,790	33,399	1,321
CTB8	3,007	236,132	1,376,142	65,410
MSR	/	18,236	89,917	11,728
PKU	/	15,492	88,327	12,422

Table 5.1.: **The training and test corpus statistics after preprocessing.** MSR and PKU are the (Chinese) test corpora which are mentioned in Section 5.4.5. Word units are pre-segmented in the CTB8, MSR, and PKU corpora.

In order to examine the unsupervised performance of LiB, all spaces in the corpora were removed before training. We trained LiB on BR-phono and on CTB8 separately. The parameter settings are shown in Appendix 7.4.4. The example segmentations with increasing number of training epochs are shown in Appendix 7.4.4. The related code and preprocessed corpora are available online[2].

5.4. Model Evaluation

5.4.1. Subchunks

After training, we evaluated the chunk units in the training corpora from two information-theoretical views that bear a relation to cognitive processing: description length and language model surprisal. We also examined the performance of LiB on word segmentation tasks. However, since LiB can learn new chunks from the concatenation of known chunks, the learned chunks are not only words, but also possible multi-word expressions. For the word segmentation task, we want to know the words in those multi-word expressions, so we had LiB find the subchunks $c^\flat$, which are the chunks inside the original chunks (e.g., "you" and "are" inside "youare"), and regarded the subchunks as the words. LiB defines the subchunk by searching all the potential chunk sequences in the original chunk (c_{raw}) and selecting the sequence with lowest sum of ordinals unless c_{raw} has the lowest sum:

[2] `https://github.com/ray306/LiB`

$$(c_1^\flat, \ldots, c_n^\flat) = \arg\min_{(c_1,\ldots,c_n)} \left(\sum_i \Theta(c_i) \right), \text{where } (c_1, \ldots, c_n) = c_{raw}$$

$$\text{Subchunk(s) of } c_{raw} = \begin{cases} (c_1^\flat, \ldots, c_n^\flat) & \text{if } \max_i(\Theta(c_i^\flat)) < \Theta(c_{raw}) \\ c_{raw} & \text{otherwise} \end{cases}$$

5.4.2. Qualitative evaluation

Since the LiB lexicon is ordered, we may examine the head of the trained lexicons (Table 5.2), which are the highest-ranked chunk units. They show that LiB appears to learn common words and collocations. Among the learned units we observe some collocations (e.g., “that’sa") which are not linguistic phrases. The lexicon of LiB trained on CTB8 shows that the high-ranked Chinese chunk units are usually bigrams (Appendix 7.4.4). The middle and the tail of the trained lexicons are also shown in Appendix 7.4.4. We present examples of chunk and subchunk segmentation results in Table 5.3. The results show the chunk units include common collocations, while the subchunk units are very close to the linguistic words.

Corpus	**Top 50 entries (translated) in Lexicon**
BRphono	the, yeah, you, what, wanna, can you, two, and, that’s, okay, four, now, it, they’re, he’s, in, look, with, you want, who, he, that, all, your, here, i think, put, that’s a, what’s, you can, his, my, see, you wanna, no, is that, high, whose, this, good, there’s, very, see the, its a, is it, alright, this is, are you, ing, have
CTB8	haven’t, China, we, economics, already, kid, but, education, can, now, government, country, a, these, self, can’t, if, journalist, today, they, although, require, tech, process, this, Xinhua News Agency, wish, issue, is, mainland, because, some, and, all are, so, now, may, Taiwan, should, political, development, also is, also is, society, such, via, continue, isn’t, Shanghai, ’s

Table 5.2.: **Transliterations/translations into English of the top 50 entries in the lexicons.** The original results of BRphono are in phonemic characters, and the original results of CTB8 are the Chinese characters. For completeness, in Appendix 7.4.4 we repeat these results with the original results included.

Corpus	Level	Segmentation
BRphono	Input	allrightwhydon'tweputhimawaynow
	Chunks	allright·whydon't·we·puthimaway·now
	Subchunks	all·right·why·don't·we·put·him·away·now
	Words	all·right·why·don't·we·put·him·away·now
CTB8	Input	这个出口信贷项目委托中国银行为代理银行
	Chunks	这个·出口信贷·项目·委托·中国银行·为·代理·银行
	Subchunks	这个·出口·信贷·项目·委托·中国·银行·为·代理·银行
	Words	这·个·出口·信贷·项目·委托·中国·银行·为·代理·银行

Table 5.3.: **Example segmentations of strings in the two corpora.** BRphono's results are transcribed into English words for ease of presentation.

5.4.3. Description length evaluation

LiB provides two types of new units to segment language: **LiB chunks** are the raw segmentation result of LiB, and **LiB subchunks** are the subchunks inside LiB chunks. In order to examine the encoding efficiency of LiB chunks and LiB subchunks, we compared the description lengths (DL) on different segmentations. The DL is the number of bits required to represent the corpus; it sums the number of bits required to encode the lexicon and the number of bits required to encode the corpus when segmented by the lexicon (Zhikov et al., 2013):

$$\begin{aligned} DL(total) &= DL(lexicon) + DL(corpus) \\ &= -\sum_{i=1}^{\#s} Freq(s_i)\log_2 P(s_i) - \sum_{j=1}^{\#u} Freq(u_j)\log_2 P(u_j) \end{aligned} \tag{5.1}$$

Here, $\#s$ denotes the number of unique symbols s in L (either as a single-symbol chunk or as part of a larger chunk); $Freq(s_i)$ and $P(s_i)$ are the occurrence count and ratio of s_i in L; $\#u$ denotes the number of unique units u in the corpus; $Freq(u_j)$ and $P(u_j)$ are the occurrence count and ratio of u_j in the corpus.

As benchmarks, we used **Symbol** (the indivisible units; in our two corpora, phonemes and characters respectively), **Word** (the words presegmented in the corpora), and **BPE subword** (the Byte Pair generated by SentencePiece (Kudo & Richardson, 2018) with default parameters setting). The DL result (Table 5.4) shows that LiB chunks result in shortest DL; they minimze the information; they are the most concise encodings.

5.4.4. Language model evaluation

Besides the DL, which compares the information efficiencies of different lexicons, we are also interested in whether the LiB lexicon can reflect the mental lexicon. We lack a

		Segmentation				
Corpus	**Evaluation metric**	**Symbol**	**BPE subword**	**Word**	**LiB subchunk**	**LiB chunk**
BRphono	Token length	1	2.8	2.9	2.9	3.6
	Lexicon size	50	5,574	1,321	1,119	1,869
	DL(lexicon)	**<1**	173	28	24	47
	DL(corpus)	490	278	262	258	**233**
	DL(total)	490	451	289	282	**281**
CTB8	Token length	1	1.4	1.7	1.7	1.9
	Lexicon size	4,697	7,980	65,410	24,763	39,320
	DL(lexicon)	**57**	133	1,767	621	1,153
	DL(corpus)	21,864	18,229	15,669	16,188	**15,602**
	DL(total)	21,921	18,362	17,436	16,809	**16,755**

Table 5.4.: **Average token lengths, lexicon sizes, and the DL results of different types of segmentation on the two corpora.** The unit of Average Length is phoneme (BRphono) or Chinese character (CTB8). The unit of DL is kilobit.

ground truth of what is in the putative mental lexicon. However, we can regard natural language material as a large-scale result of human language use and language behavior. Trained on a very large corpus, a recent study by Brown et al. (2020) shows that Language Models (LMs) can closely predict human performance on various language tasks. LMs capture the probabilistic constraints in natural language and perform the tasks by making predictions, which is a fundamental cognitive function (Bar, 2007). So, by measuring the prediction surprisal in the corpus segmented by different lexicons, we can evaluate different lexicons from a cognitive view, and we presume that the lexicon that gets the best LM performance is a better approximation of the mental lexicon.

Many studies have shown that word surprisal is positively correlated with human word-reading time (Monsalve, Frank, & Vigliocco, 2012; Smith & Levy, 2013) and size of the N400 component in EEG (Frank, Otten, Galli, & Vigliocco, 2015). From the cognitive principle of least effort, it follows that readers try to minimize reading time. Hence, it follows that readers would try to find lexical units such that total surprisal is also minimized.

Surprisal, defined as $-\log_2(P(w|\text{context}))$, is not comparable between models with different segmentations. Instead we use bits per character (BPC) (Graves, 2013), which is average surprisal/$|c|$, where $|c|$ is the average chunk length over the whole test set. We tested the segmentations[3] on both bigram and trigram language models and the

[3]The code of the BPC calculations was modified from a Github project: `https://github.com/joshualoehr/ngram-language-model`. We kept all tokens during training.

		Segmentation				
Corpus	**Model**	**Symbol**	**BPE subword**	**Word**	**LiB subchunk**	**LiB chunk**
BRphono	2-gram	1.539	0.695	0.677	0.649	**0.548**
	3-gram	0.950	0.390	0.405	0.378	**0.335**
CTB8	2-gram	2.466	1.932	1.617	1.668	**1.452**
	3-gram	1.404	0.827	0.806	0.748	**0.626**

Table 5.5.: **Bits-per-character scores on different segmentations.**

results show that the corpora represented by LiB chunks achieve the lowest surprisal (Table 5.5).

5.4.5. Word segmentation evaluation

As we already illustrated in Table 5.3, subchunk units tend to be close to linguistic words. We thus tested LiB subchunks as a resource for word segmentation. To evaluate LiB on English word segmentation, we compared LiB with Adaptor Grammar (AG) (Johnson & Goldwater, 2009), which achieves state-of-the-art performance on the segmentation task of BR-phono. AG requires grammar construction rules that encode prior linguistic knowledge. These rules presuppose knowledge about unigrams only, or unigrams+collocations, or unigrams+collocations+syllables. This yields three versions of AG. Table 5.6a shows that AG(syllable), whose rules carry extra linguistic knowledge (Johnson & Goldwater, 2009), achieves the highest score. The score of LiB is higher than AG(unigram) and slightly lower than AG(collocations), the two versions of AG comparable to our approach. AG(syllable) presumes knowledge that our model does not have (and that could possibly benefit LiB).

In the Chinese segmentation task. we compared LiB with three popular word segmentation toolboxes: Jieba[4], THULAC (M. Sun, Chen, Zhang, Guo, & Liu, 2016), and pkuseg (R. Luo, Xu, Zhang, Ren, & Sun, 2019). These toolboxes are supervised, learning the ground truth (word boundaries) during training. For comparison, we also modified a supervised LiB (LiB(sup)) for the word segmentation task. LiB(sup) skips the training phase. Instead, it counts all the ground-truth words in the training set and adds them as the chunk types to L. The higher the frequency of a type in the training set, the smaller its ordinal in L. We trained and tested the models on CTB8. To test the generalization performance of the models in the word segmentation task, we also test the training result on two additional corpora: MSR and PKU (Table **??**) provided by the Second International Chinese Word Segmentation Bakeoff (Emerson, 2005). The segmentation rules are slightly different among MSR, PKU, and CTB8. MSR and PKU are

[4] `https://github.com/fxsjy/jieba`

news domain, which is different from CTB8. MSR and PKU were preprocessed in the same way as CTB8.

Table 5.6b shows that the scores of the unsupervised original version of LiB are lower than the supervised models[5], but the scores of the supervised version of LiB are close to the supervised models and are even higher on MSR. Due to the low out-of-vocabulary (OOV) rate of MSR (Emerson, 2005), the good performance on MSR shows that the lexicon is important for LiB. The only difference between the two versions of LiB is in their lexicons: the original LiB learned the lexicon from zero and the supervised LiB directly uses the ground-truth words in its lexicon. It shows that the segmentation module in LiB is appropriate for the word segmentation task.

[a]

Model	**Scores**
AG (unigram)	56
AG (collocations)	76
AG (syllable)	**87**
LiB subchunk	71

[b]

	Test set scores		
Model	**CTB8**	**MSR**	**PKU**
Jieba	87.1	82.8	87.1
THULAC	94.6	83.5	89.1
pkuseg	**95.7**	83.7	**89.7**
LiB subchunk	76.1	78.7	78.9
LiB(sup) chunk	94.7	**84.5**	88.3

Table 5.6.: **Token F1 scores (%) of segmentations.** [a] the scores on BR-phono by three versions of Adaptor Grammar (AG) and LiB subchunks. [b] the scores of Jieba, THULAC, PKUSEG, LiB subchunks, and LiB(sup) chunks. LiB(sup) represents the supervised adaptation of LiB.

5.5. Conclusions and Future Work

This paper presented an unsupervised model, LiB, to simulate the human cognitive process of language unitization/segmentation. Following the principles of least effort, larger-first processing, and passive and active forgetting, LiB incrementally builds a lexicon which can minimize the number of unit tokens (alleviating the effort of analysis) and unit types (alleviating the effort of storage) at the same time on any given corpus. Moreover, it is able to segment the corpus, or any other text in the same language, based on the induced lexicon. The computations in LiB are light-weight, which makes it very efficient. The LiB-generated lexicon shows optimal performances among different types

[5]The scores of Jieba, THULAC, and pkuseg are provided by `https://github.com/lancopku/pkuseg-python`

of lexicons (e.g., ground-truth words) both in terms of description length and in terms of statistical language model surprisal, both of which are associated with cognitive processing. The workflow design and the computation requirement make LiB cognitively plausible, and the results suggest that the LiB lexicon may be a useful proxy of the mental lexicon.

Future work will be to allow skip-gram units in the lexicon. Skip-grams may help to capture longer-distance dependencies, and further lessen the cognitive effort by reducing the number of unit types/tokens. Furthermore, as the word segmentation results of the current LiB are not ideal, we hypothesize that skip-gram units may also benefit the detection of infrequent named entities (e.g., the skip-gram “Mr._said" helps to detect “Mortimer" in “Mr.Mortimersaid") and thus improve the word segmentation performance. Other future work includes a LiB variant that accepts speech input and a semi-supervised LiB variant that uses semantic knowledge (e.g., word embeddings) to enhance the language unitization.

6 | Unsupervised text segmentation predicts eyefixations during reading[1] [2]

Abstract

Words typically form the basis of psycholinguistic and computational linguistic studies about sentence processing. However, recent evidence shows the basic units during reading, i.e., the items in the mental lexicon, are not always words, but could also be sub-word and supra-word units. To recognize these units, human readers require a cognitive mechanism to learn and detect them. In this paper, we assume eye fixations during reading reveal the locations of the cognitive units, and that the cognitive units are analogous with the text units discovered by unsupervised segmentation models. We predict eye fixations by model-segmented units on both English and Dutch text. The results show the model-segmented units predict eye fixations better than word units. This finding suggests that the predictive performance of model-segmented units indicates their plausibility as cognitive units. The Less-is-Better (LiB) model, which finds the units that minimize both long-term and working memory load, offers advantages both in terms of prediction score and efficiency among alternative models. Our results also suggest that modeling the least-effort principle for the management of long-term and working memory can lead to inferring cognitive units. Overall, the study supports the theory that the mental lexicon stores not only words but also smaller and larger units, suggests that fixation locations during reading depend on these units, and shows that unsupervised segmentation models can discover these units.

[1] Yang, J., van den Bosch, A., & Frank, S. L. (2022). Unsupervised text segmentation predicts eye fixations during reading. *Frontiers in Artificial Intelligence, 5*:731615.

[2] All code and datasets involved in modelling and experimentation are available at `https://github.com/ray306/LiB-predicts-eye-fixations`.

6.1. Introduction

Language researchers may easily agree that an utterance comprises a sequence of “units", but it is not easy to come to an agreement on what these units are. The units can be words, phonemes, morphemes, phrases, etc. from a linguistic perspective (Jackendoff, 2002); or unigrams, bigrams, trigrams, etc. from a statistical perspective (Manning & Schütze, 1999). In this paper, we take a *cognitive perspective* and aim to identify the cognitive units that play the role of building blocks in human language processing.

Words seem to be the most generally accepted units, perhaps because the spaces in written European languages steer us towards implicitly assuming that individual words are the most distinctive elements of sentences. Pollatsek and Rayner (1989) summarized ten key questions for the cognitive science of reading; nearly half of them are about words. Another case in point is that there are many models of visual word recognition, such as the Interactive Activation model (McClelland & Rumelhart, 1981), the Triangle model (Plaut, McClelland, Seidenberg, & Patterson, 1996), and the Dual Route Cascaded model (Coltheart et al., 2001). Even when considering sentence-level processing, researchers tend to take words as the basic units in their studies; this is the case for the classical studies relevant to the garden-path model which describes how the reader analyzes the grammatical structure of sentence from the serial input of words (Frazier, 1987; Frazier & Rayner, 1982), for the E-Z reader model which explains how the attributes of words guide eye movements during reading (Reichle, Pollatsek, Fisher, & Rayner, 1998), and for the discovery of the N400 component in brain activity which responds to semantically anomalous words (Kutas & Hillyard, 1980). Word units are also assumed for more recent studies such as those that map brain activity to processing of each word of a sequence (Brennan & Hale, 2019; Brennan et al., 2012; Ding et al., 2016) as well as studies that compare the statistical attributes of words in sentences with the cognitive and neural response to the words (Frank et al., 2015; Frank & Willems, 2017; Mahowald, Fedorenko, Piantadosi, & Gibson, 2013).

Though words are often used as psycholinguistic units, morphemes, which are defined as the smallest meaning-bearing units in a language (Chomsky, 1953), are usually the basic units in linguistic analysis. The central role of morphemes in linguistics also influenced some psycholinguists to consider the mental operations of morphologically complex words. In a recent review, Leminen et al. (2019) analyzed more than 100 neuroimaging studies of inflected words (e.g., walk-ed), derived words (e.g., dark-ness), and compounds (e.g., walk-man). As they summarized, most studies of the processing of derivational/inflectional morphology agree that such complex words are decomposed during processing; but studies of the processing of compound words show inconsistent results: some support the access of constituent morpheme units (Fiorentino et al., 2014; Koester & Schiller, 2011), some support the access of whole-word units (Stites, Feder-

meier, & Christianson, 2016), and some support mixed access of both (Kaczer et al., 2015; MacGregor & Shtyrov, 2013; Yang, Cai, & Tian, 2020)

In addition to subword units, the cognitive system can also make use of supra-word units. Some studies provide indications that supra-words such as frequent phrases and idioms (e.g., “I don’t know") are stored in our long-term mental lexicon (Arnon & Snider, 2010; Bannard & Matthews, 2008; Jackendoff, 2002), implying that supra-words can be processed directly. Baayen (2007) has argued that the mental lexicon involves storage (of the wholes) and computation (of the combinatorial rules), and that they counterbalance each other. Yang, Cai, and Tian (2020) also considered the counterbalancing, arguing that storing more supra-words in our mental lexicon could reduce the cognitive load of computation since larger units (e.g., “I am|going to" vs. “I|am|going|to") result in fewer processing steps (e.g., two retrievals + one combination vs. four retrievals + three combinations). Taken together, this diverse evidence shows that cognitive units exist at various linguistic levels, and that cognitive units have a wide range of possible lengths.

The flexibility of cognitive units implies that there is no clear or uniform perceptual salience of the units during reading, since a cognitive unit may be a sub-word or a supra-word that is not surrounded by two dividers (i.e., spaces), let alone the fact that in some writing systems (e.g., Chinese) there are no dividers for words. However, readers must be able to segment language input into cognitive units in order to access the meaning of the units and understand the input. Thus, our cognitive system must have a mechanism to quickly locate the cognitive units in language input for subsequent recognition. In fact, our eye movements in daily tasks may indicate the existence of this mechanism, since eye movements include many fixations which land neither randomly nor uniformly, but primarily on the targets of salience, information, or interest in the scene we see (Buswell, 1935; Henderson, 2011). So it is with reading: eye movements are controlled to skip some words, especially when the words are high-frequency function words (Rayner, Well, Pollatsek, & Bertera, 1982).

The flexibility of cognitive units also implies that it is hard for language learners to decide on the basis of perceptual cues whether or not a particular morpheme, word, or arbitrary string is a cognitive unit. Humans must have the ability to learn the cognitive units from their own experience, or in machine learning terms: unsupervised. To understand the human ability to learn and to identify the cognitive units, we need a model that is unsupervised and cognitively plausible. We here introduce the Less-is-Better (LiB) model (Yang, Frank, & van den Bosch, 2020) as a candidate.

The LiB model is inspired by one intrinsic aspect of our nature: the principle of least effort. George Kingsley Zipf, who proposed the principle, explained it as “[the human agent] will strive to solve his problems in such a way as to minimize the total work that he must expend in solving both his immediate problems and his probable future

problems." (Zipf, 1949, p. 1). Limiting ourselves to purely cognitive tasks, we here interpret his words as:

a. *To reduce the number of* ***processing units*** *in* ***both current and prospective working memory***.

The cognitive load refers to the demand not only on working memory, but also on long-term memory (see Fig. 6.1a below), so we here extend the principle of least effort to:

b. *To reduce the number of* ***stored units*** *in* ***long-term memory***.

The LiB model regards the cognitive units as the language chunks that require the least effort during language processing, and the above two goals can be operationalised as: *a. to reduce the number of unit* ***tokens*** *in all potential texts*, and *b. to reduce the number of unit* ***types*** *in long-term memory (mental lexicon)*. There is a trade-off between the two goals. The former goal will prefer combining adjacent chunks into larger chunks, such as phrases, to reduce the number of tokens. If this process would be unrestricted, it would lead to units being so large as to represent the entire text with only one unit token. This would result in an extremely large lexicon memory that will not generalize to future use, as its units are unlikely to recur. To prevent this from happening, the latter goal will remove low-frequency chunk types from memory. The two goals counterbalance each other during learning and make the result in line with the least-effort requirement.

The current study aims to evaluate how similar the units segmented by unsupervised word segmentation models are to cognitive units. Although we lack a gold standard for cognitive units, eye movements during reading, specifically the eye fixations, may provide information about them. Taking words as units of analysis of eye-tracking data, studies have reported that fixation positions frequently fall at (or close to) the center of a word when the word is fixated only once (X. Li, Liu, & Rayner, 2011; Paterson, Almabruk, McGowan, White, & Jordan, 2015; Rayner, Sereno, & Raney, 1996), but some words are fixated more than once (Cop, Dirix, Drieghe, & Duyck, 2017; Hyönä & Olson, 1995; Kliegl, Grabner, Rolfs, & Engbert, 2004; Rayner & McConkie, 1976), while some short words are not fixated upon (Brysbaert & Vitu, 1998; Kerr, 1992). Taking multiword sequences as units of analysis of eye-tracking data, formulaic sequences (e.g., “as a matter of fact") get fewer fixations than non-formulaic sequences (e.g., “it’s a well-known fact") (Underwood, Schmitt, & Galpin, 2004). In light of these empirical findings, we hypothesize that eye fixations are a proxy to the location of the cognitive building blocks of the text, that is, the cognitive units.

Our main goal in the current study is not to predict or explain eye fixations, but to validate the model as a cognitive model by quantifying its ability to predict eye fixations. The current model aims to be as simple as possible by using only unannotated

text. Therefore, in this study, we will not include properties that can improve fixation prediction but are outside the scope of the model (e.g., semantics).

We use the units segmented by LiB in a corpus to predict the locations of the eye fixations in the same or a different corpus. If the LiB units indeed predict eye fixations, this suggests both that the LiB units are similar to the cognitive units and that the cognitive units are located by the eye fixations. In other words, cognitive units may be considered a latent factor driving both eye fixations and the discovery of units by LiB, and the extent to which the LiB units predict eye fixations reflects their plausibility as cognitive units. Then we evaluate the similarity between the LiB units and the hypothesized cognitive units during human reading by comparing the eye fixations predicted by LiB with the observed eye fixations extracted from an eye-movement corpus. As the design and the training of the model are independent of the eye movements (i.e., the model is not fitted on the eye-tracking data), any overlap found between model predictions and eye movements is caused by properties of the model itself and not by spurious patterns discovered in the eye-movement data.

Two other segmentation models are also evaluated for comparison: Chunk-Based Learner (McCauley & Christiansen, 2017), and Adaptor Grammar (Goldwater, Griffiths, & Johnson, 2009). We also compare to two word-based baselines: one that assumes the cognitive units are equal to words, and one that assumes the cognitive units are determined by the word length. The models are introduced in more detail below. In the comparisons, we will demonstrate that the segmentation models outperform the baselines, and show the advantages of LiB in various aspects.

6.2. Methods

6.2.1. The Less-is-Better Model

The information workflow of the LiB model consists of an interaction loop between the text segmentation module (blue box with solid line; Fig. 6.1a) and the lexicon update module (orange box with dotted line; Fig. 6.1a). We briefly characterise the model; more detail is given in Yang, Frank, and van den Bosch (2020). The model has a lexicon L which is an ordered set of unit types u. In each epoch, the segmentation module (Fig. 6.1b) segments the input, which is a sequence of symbols $(s_1, s_2, \ldots, s_n)$ (in the current simulations, symbols are characters, excluding spaces) into a sequence of a minimal number of unit tokens $(u_1, \ldots, u_N)$, where each u token is a subsequence of the input; $u = (s_i, \ldots, s_j)$, and each u is in the current L. The update module (Fig. 6.1c) then updates L according to the output $(u_1, \ldots, u_N)$, meaning that some new unit types are created and added to L to decrease the number of tokens in future inputs, and some current unit types are removed to decrease the number of types in L.

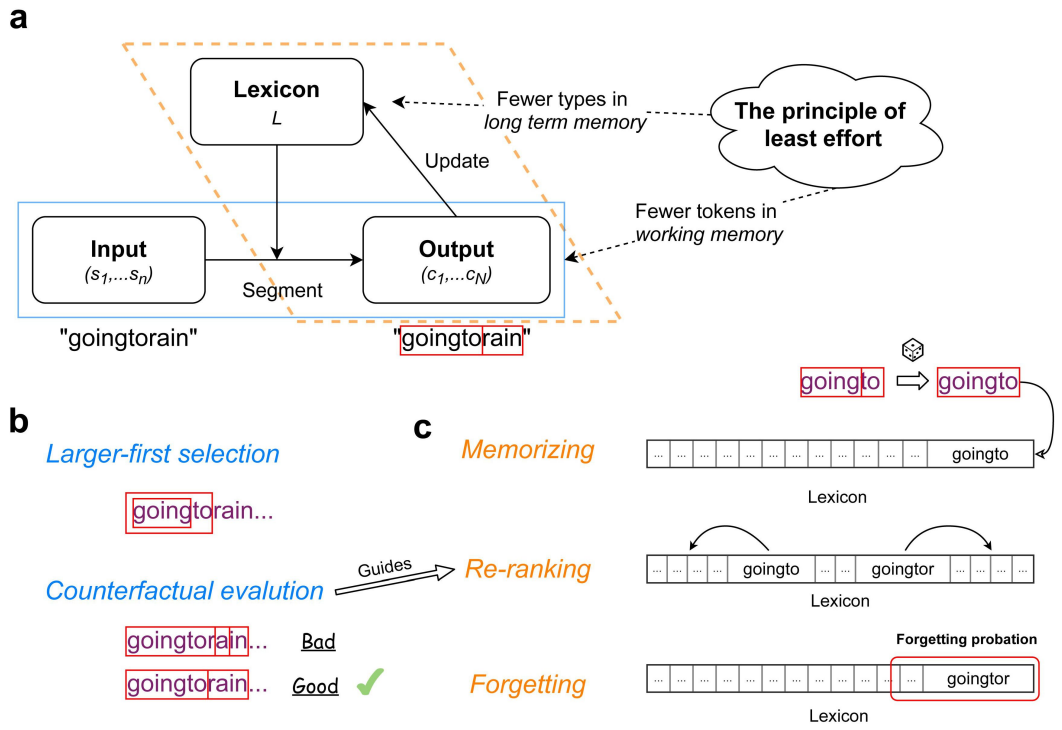

Figure 6.1.: **Illustration of the LiB model**: a) information flow in the LiB model; b) the mechanisms in the text segmentation module; c) the mechanisms in the lexicon update module.

To reduce the number of u tokens during the segmentation, if u types of different sizes in L match the current input, the largest u type has the priority to be selected as the u token (*Larger-first selection*; Fig. 6.1b). Then LiB evaluates the u token by segmenting the following input and counting the segmented tokens. LiB segments the current input as if the largest u type does not exist (*counterfact*) so the second-largest u type will also be evaluated. In case the largest u type causes more u tokens in the input than the second-largest u type, the largest u type is evaluated as *Bad* (otherwise as *Good*) and the second-largest u type is selected instead (*Counterfactual evaluation*; Fig. 6.1b).

L is empty at the beginning, and all the symbols $s_1, \ldots, s_n$ in the input are unknown to the model. Those symbols will be memorized as the first batch of new u types in L. Adjacent u tokens in the input can become a larger u token by concatenation of the two original tokens, and the adjacent larger u tokens can become an even larger u token. LiB randomly samples the combinations of segmented u tokens and memorizes the sampled combinations as new u types (*Memorizing*; Fig. 6.1c). These new u types go to L immediately and can be used for further segmentations and combinations, so LiB learns online. The sampling strategy achieves similar results as tracking the frequencies of each u type and dropping the low-frequency ones, since the u types with higher fre-

quencies are more likely to be sampled. However, compared to frequency tracking, LiB's sampling strategy consumes markedly less resources of memory and operation.

Although no statistical information of the *u* types is recorded, LiB indicates a *u* type's likelihood of being a cognitive unit by the type's rank in the Lexicon. A newly memorized *u* type is appended to the end of *L*, which means it has the lowest likelihood of being a cognitive unit, because the new *u* type might merely be an accidental concatenation of two *u* tokens. Besides the memorizing order, the order of *L* also depends on *Chunk evaluation*: after the evaluation, a *Good u* is moved forward and a *Bad u* is moved backward in *L* (*Re-ranking*; Fig. 6.1c).

The *Re-ranking* pushes the chunks *u* that were evaluated as *Bad* as well as very infrequent chunks (that never had the opportunity to be evaluated) backward in *L*. This means the end of *L* contains not just newly memorized *u* but also junk *u* (infrequent *u* and *Bad u*). To clean up only the junk *u*, all *u* at the end of *L* enter a probationary period. In case a *u* was evaluated as *Good* during the probation, its probation is canceled; otherwise, the chunk is removed from *L*. By such a mechanism (*Forgetting*; Fig. 6.1c) LiB can reduce the number of *u* types and keep a small size *L*.

6.2.2. Other models for evaluation

Firstly we introduce a frequentist computational model named Chunk-Based Learner (CBL; McCauley & Christiansen, 2019), which aims to simulate human incremental language acquisition. CBL also has its cognitive basis: frequency-based learning. In detail, CBL processes naturalistic linguistic input as sequences of chunks. Initially, each word is a chunk. Then CBL calculates the backward transitional probabilities (BTPs) between the chunks. If the BTP of a chunk-pair rises above the average of all tracked BTPs, the chunk-pair will be grouped as a new chunk and be replaced by the new chunk in further processes. CBL in this way implements the incremental learning of multi-word units. Some words will not be combined into larger chunks, and thus the lexicon of CBL will contain both word units and multi-word units.

Bayesian models can be seen as an alternative to frequentist models, and the "Bayesian coding hypothesis" also argues that humans behave Bayesian (Knill & Pouget, 2004). Adaptor Grammar (AG; Johnson, Griffiths, & Goldwater, 2007) is a word segmentation model based on a Bayesian framework. Like the other models we compare, it aims to segment the tokens from the input in an unsupervised way. The AG model represents each input sequence from the corpus as a multi-level tree structure with a predefined number of levels. Although different trees can represent the same sequence, AG assumes there is an optimal tree. The Hierarchical Dirichlet Process, which is a nonparametric Bayesian approach to group the observed data hierarchically (Teh, Jordan, Beal, & Blei, 2006), is used to find the *optimal* trees that fit the input sequences. Notwithstanding AG

is usually used for word segmentation, syllabification, and other linguistic applications (Johnson, 2008; Johnson & Goldwater, 2009; Johnson et al., 2007; Zhai, Boyd-Graber, & Cohen, 2014), in the current study we investigate whether the unsupervised nature of the model can help to discover the cognitive units.

Besides the segmentation models that can generate non-word units, we set two baselines that are completely word-based. The first baseline (*Word-by-Word*) simply assumes that the cognitive units are equal to words. As we mentioned above, words are the commonly accepted units in many studies so it is worth investigating whether words or the model-produced cognitive units can better predict eye fixations.

Another baseline (*Only-Length*) implements the assumption that the number of fixations on a word is determined by the length of the word. Different from the *Word-by-Word* baseline, the *Only-Length* baseline uses the knowledge of observed eye fixations. *Only-Length* groups the words with the same number of letters together and shuffles the numbers of fixations within each group. Only the distributions related to the word lengths persist in this baseline so the prediction will not be influenced by frequency, morphology, position, or other non-length information.

6.2.3. Eye fixation data

The eye fixation data is extracted from the Ghent Eye-Tracking Corpus (GECO) corpus[3] (Cop et al., 2017). GECO contains three sets of eye-tracking data: fourteen English monolinguals reading the English novel *The Mysterious Affair at Styles* by Christie (1920/2008) (monolingual set); nineteen Dutch (L1)–English (L2) bilinguals reading the same novel (L2 set); and the same bilinguals reading the Dutch translation of the novel (title in Dutch: *De zaak Styles*) (L1 set). The English monolingual group read the full English novel and the bilingual group read either the first half of the novels in English and the second half in Dutch, or vice versa. For the evaluation in the current study, we discard the L2 set since it is not native-language reading.

The GECO datasets provide two types of eye fixation data: *first-pass fixation count* and *total fixation count*. The first-pass fixation includes only the initial reading (until any fixation on another word) within each word and the total fixation includes also the re-reading (after regression) within each word. Most of the regressions reflect post-lexical language processing (Reichle, Warren, & McConnell, 2009), and others may reflect oculomotor error or difficulty associated with the identification of words (Vitu & McConkie, 2000). These processes are beyond the scope of the segmentation models we evaluated, since the segmented cognitive units are for planning what to process rather than post-hoc adaptation. That being so, we evaluate only the first-pass fixation count.

[3] `https://expsy.ugent.be/downloads/geco/`

6.2.4. Corpora

Both the English and the Dutch GECO corpora are used for model training in the current study. Since the material presented to the participants are in multiple lines, and the last word in a line and the first word in the next line are too far apart to be perceived as a cognitive unit, we break any sentence that appears across different lines into separate sequences. Two other corpora also serve as training material but only in the generalizability test of the models. One of them is Corpus of Contemporary American English (COCA; Davies (2008). We used a sample dataset of COCA which is free for the public[4]. Although the sample dataset is only a small part of the complete COCA corpus, it is more than one hundred times larger than the English material in GECO. The other additional training corpus is SoNaR (Oostdijk, Reynaert, Hoste, & Schuurman, 2013), a 500-million-word reference corpus of contemporary written Dutch from a wide variety of text types. The complete corpus is very large so we selected the *book* subset of SoNaR. The corpus sizes are shown in Table 6.1.

Language	*Corpus*	*Sentences*	*Word tokens*	*Word types*
English	GECO	13,491	57,170	5,316
	COCA (sample)	1,745,060	9,451,421	140,553
Dutch	GECO	13,407	60,836	5,859
	SoNaR (books)	3,308,337	22,802,170	272,865

Table 6.1.: **The corpora statistics after preprocessing.**

The text from all corpora was converted to lowercase. Dutch characters with accents (*diacritical characters*) were replaced by their unaccented counterparts (e.g., ë → e). All punctuation (except the apostrophe as a part of possessive) was used as a divider between the input sequences and then were replaced by a space. Finally, all sentences have a space added at the end to make sure that all word tokens end with a space.

6.2.5. Evaluation

To evaluate the units segmented by the different models against the eye fixation counts on each word from GECO, we predict the eye fixation count from the segmentation models and from the word-based baselines, and then compare the predicted eye fixation counts per word with the observed eye fixation counts.

For the segmentation models, the eye fixation counts are predicted in the following procedure:

1. **Training the models**:

[4] `https://www.corpusdata.org/coca/samples/coca-samples-db.zip`

- **The LiB model**: In each training epoch, a 200-sentence batch is randomly extracted from the corpus text (batch-based update in LiB reduces the computing cost) and then fed into the model. When training on GECO, which is rather small, the batch extraction is with replacement and the training stops when the number of input encoding bits[5] no longer decreases. When training on the large-scale corpus, the batch extraction is without replacement, and the training will stop when there is no training material left. The hyperparameter settings in the current study follow the previous LiB study (Yang, Frank, & van den Bosch, 2020) except the setting of *probation period*[6].
- **The CBL model**: Different from LiB and AG which regard the input as a sequence of characters, CBL regards the input as a sequence of words, so it preprocesses the input into words based on the spaces. There is no change in the training stage of the original code of McCauley and Christiansen's (2019) implementation[7].
- **The AG model**: The simplest grammar tree in AG starts with characters, then processes words, and then sentences. The model tends to under-segment without an intermediate level of collocations (Johnson & Goldwater, 2009), so the AG grammar tree used in the current study is: character(s) → word, word(s) → collocation, collocation(s) → sentence. Besides the design of the grammar tree, we also follow the hyperparameter settings of Johnson and Goldwater's (2009) experiment[8].

2. **Segmenting the text into units**:
   - **The LiB model**: Each sequence of the GECO corpus is fed to the trained model to be segmented into the units existing in the trained lexicon. The segmentation is guided by *Larger-first selection* and *Counterfactual evaluation*. No new cognitive units will be memorized in this stage.
   - **The CBL model**: The original implementation simulates children's incremental learning so the lexicon is empty at the start of training. This means a group of words may be a unit at the end of segmenting the GECO corpus but not at the beginning. To keep the segmentation consistent in the test material, as in the other models, our implementation of CBL learns the

[5]The metric is the product of $\log_2 |\text{lexicon}|$ (the cost of storing unit types) and $\frac{1}{\text{average chunk length}}$ (the cost of computing unit tokens).

[6]To simplify the model, we removed a regularizer that only memorizes the chunk tokens that appear more than twice in a single epoch in the LiB model described by Yang, Frank, and van den Bosch (2020). We reduced the *probation period* to an arbitrary value of 3, since the original setting will cause the lexicon to grow too fast after removing the regularizer.

[7]`https://github.com/ray306/LiB-predicts-eye-fixations/tree/main/other_models/cbl_modified.py`

[8]`https://github.com/ray306/LiB-predicts-eye-fixations/tree/main/other_models/AdapterGrammar/ag_geco/Makefile`

training corpus thoroughly and then segments the test corpus again with a fixed lexicon.

- **The AG model**: AG learns the parsing rules during training. When the model applies the rules to the test corpus, each sequence will be processed into a hierarchical structure which contains the *character* level, the *word* level, the *collocation* level, and the *sentence* level. We extracted the units at the *word* level (the *AG-word* units) and at the *collocation* level (the *AG-collocation* units).

3. **Predicting the number of fixations on each word**:

   We assume that reading is based on the cognitive units and the fixation positions are the centers of the cognitive units, at least if the entire unit is within the perceptual span. We ignore the perceptual span for now since we want to evaluate the models totally free of prior limitations. We calculate the predicted number of fixations on a word as the number of cognitive units centered on the word. For example, the predicted fixation position of the unit *I have* is between *h* and *a*, so the predicted fixation number is zero on the word *I* (we can also say *I* is skipped in this case) and one on the word *have*. The predicted fixation number of the word *neuroscience* is two if it is segmented into *neuro* and *science*.

   To investigate the possible effect of perceptual span limitations, we also evaluate LiB while considering the perceptual span. We set different upper limits on the unit length in the model and expect there to be a maximum length that is optimal for the prediction of fixation counts and that equals the perceptual span.

For the word-based baselines, the eye-fixation numbers are predicted differently:

- The *Word-by-Word* baseline predicts exactly one fixation on each word;
- The *Only-Length* baseline groups the words with the same length together and randomly shuffles the observed number of fixations on the words within each group. Hence, the predicted number of fixations of a word is actually the observed number of fixations of another word with the same length.

The last step in the evaluation is comparing the predicted number of eye fixations with the observed number of eye fixations on each word. The F1 metric (Equation 6.1) is commonly used for evaluating a binary classification model based on the predictions made for the positive class (Van Rijsbergen, 1979).

$$\text{F1} = \frac{\text{True positive}}{\text{True positive} + 0.5 \times (\text{False positive} + \text{False negative})} \tag{6.1}$$

However, both the observed number and our predicted number of eye fixations are not binary, in other words, the metric must work for multi-label data. Because of the very imbalanced distribution of the fixation counts, we choose weighted F1 as the measure of prediction accuracy[9]. The weighted F1 calculates the binary F1 metric, which is shown by Equation 6.1, for each label (fixation count), and finds their average weighted by the number of true instances for each label.

6.3. Results

6.3.1. Qualitative comparison

Firstly we provide some segmentation examples generated by the different models (Table 6.2). In general, short and frequent collocations tend to form individual units (e.g., *to do*), and some of those collocations are not in a syntactic phrase (e.g., *i was*); long words tend to be divided into subword units (e.g., *uitnodigde* segmented as *uit*|*nodig*|*de*). Each model has its own characteristics: the CBL model learns no subword units; the AG-word model always over-segments the text; the LiB units and the AG-collocation units are generally similar.

6.3.2. Unit-length comparison

The segmentation examples show the models' output are markedly different from each other even though they are all unsupervised models. To investigate in more detail how the models' outputs differ and relate to eye fixations, we first look at the average of the unit token lengths of the models and the observed eye fixations. The GECO dataset does not provide the locations but only the number of eye fixations in any word, so we do not know every interval between each two eye fixations. Instead, we infer the locations of eye fixations from their counts in each word and then calculate the lengths of the eye fixation units by assuming eye fixations are located in the middle of the units. Fig. 6.2 shows that the average unit length of the observed eye fixations is clearly longer than the average length of space-delimited words, and is close to the average unit length of LiB. Among the unsupervised models, only AG-word shows even shorter unit length than the linguistic words. Moreover, Dutch units are in general slightly longer than English units, except in the AG models.

6.3.3. The distributions of predicted and actual fixation counts

[9] `https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html`

		Language	
Sample	*Model*	*English*	*Dutch*
1	Input	i was trying to make up my mind what to do	was ik nog aan het overleggen wat ik zou gaan doen
	LiB	i was |trying to |make |up |my mind |what |to do	was ik |nog |aan het |over|leggen |wat ik |zou gaan |doen
	CBL	i |was trying |to make |up |my mind |what |to do	was |ik nog |aan |het overleggen |wat |ik zou gaan |doen
	AG-word	i |was |try|ing |to |make |up |my |mind |what |to |do	was |ik |nog |aan |het |over|leg|gen |wat |ik |zou |gaan |doen
	AG-collocation	i was |trying to |make |up |my mind |what |to do	was ik |nog |aan het |over|leggen |wat ik |zou gaan |doen
2	Input	and it ended in his inviting me down to styles to spend my leave there	en het eind van 't liedje was dat hij mij uitnodigde mijn verlof door te brengen op styles
	LiB	and it |ended |in his |invi|ting |me |down to |styles to |sp|end |my |leave |there	en |het eind |van 't |li|e|d|je was |dat hij mij |uitnodig|de |mijn |verlof |door te brengen |op styles
	CBL	and |it ended |in |his inviting |me |down |to |styles to spend |my |leave |there	en |het eind |van |'t liedje |was |dat hij |mij uitnodigde |mijn verlof |door |te brengen |op styles
	AG-word	and |it |end|ed |in |his |invit|ing |me |down |to |styl|es to |spend||my |leav|e |there	en |het |eind |van |'t |lie|d|je |was |dat |hij |mij |uit|nodig|de |mijn |ver|lo|f |door |te breng|en |op |styles
	AG-collocation	and |it |ended |in his |inviting |me |down to |styles to |spend my |leave |there	en |het eind van |'t |lied|je |was |dat |hij mij |uitnodig|de |mijn |verlof |door |te brengen |op styles

Table 6.2.: **Segmentation examples from different models in English and Dutch.**

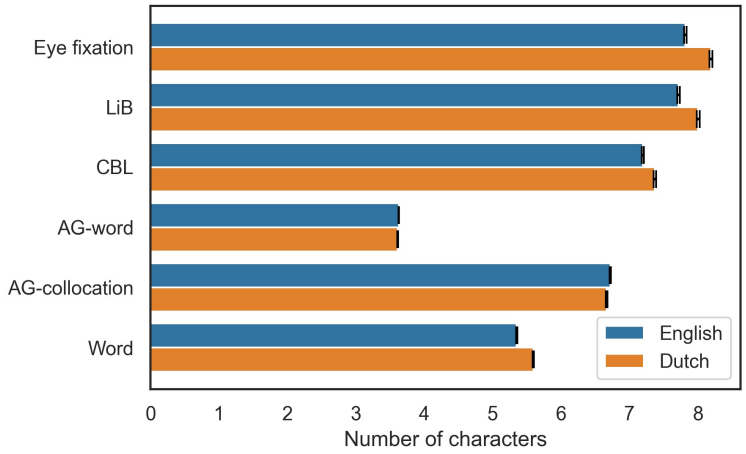

Figure 6.2.: **The average token lengths of the observed eye fixation units, the model-segmented units, and linguistic words in English and Dutch texts.** The error bars represent 99% confidence intervals.

Next, we predicted the number of eye fixations on each word token from the segmentation of the models. We display the joint distribution of the predictions and observed eye fixations (Fig. 6.3). The CBL model's output has only very few subword units (indicated by *More*, meaning more than one predicted fixation on the word). In fact, CBL itself does not output any subword units, but there are some hyphenated tokens in GECO (e.g., *forty-five*), which are processed as multiple words by the models while GECO (and so the evaluation) regards them as single words. AG-word has only very few supraword units (indicated by *Skip*, meaning no fixation at the word). Compared to the distributions of other models' predictions, the distribution of LiB's prediction is most similar to the distribution of observed fixations on both the English and the Dutch dataset. Furthermore, the surface area of the circle in the confusion matrix (Fig. 6.3) shows that *One* (exactly one fixation at the word) predictions match the observed data most often for all models, and that *More* predictions match the observed data the least.

6.3.4. The F-scores of model predictions

The unit lengths and fixation distributions displayed above (Figs. 6.2 and 6.3) provide an overview of the differences between the predicted eye fixations by the different models and the observed eye fixations. Next, we quantitatively evaluate the similarity between the predicted and observed eye fixations by their weighted F1 scores.

Table 6.3 shows that three of the four segmentation models outperform the word-based baselines in the eye-fixation prediction tasks. The *Only-Length* baseline, which predicts by only the word length, is better than the *Word-by-Word* baseline, and close to the segmentation models. Out of the four models, LiB and AG-collocation produce

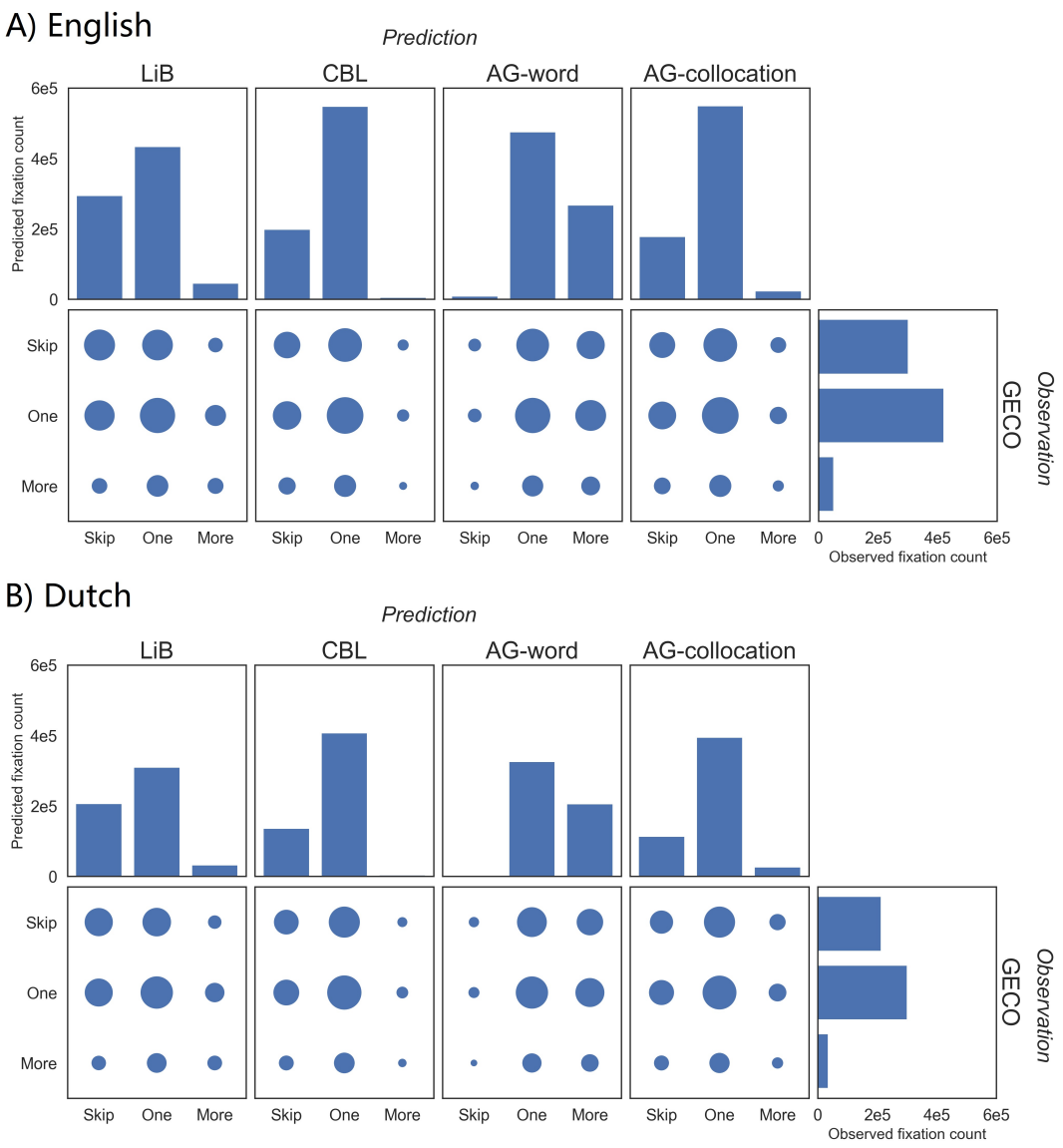

Figure 6.3.: **Distributions of the counts of the (predicted) eye fixations on the English (a) and Dutch (b) corpora.** Firstly we define three labels of the fixation counts (*Skip*: 0, *One*: 1, and *More*: >1). The histograms present the distribution of three labels; specifically, the vertical histograms present the predictions of the models and the horizontal histogram presents the observations in GECO. The scatter plots present the confusion matrix between the model predictions and the GECO observations; the surface area of each circle indicates the item count of the matching instance.

the best predictions and AG-word produces the worst predictions, worse than the word-based baselines.

Model	*English*	*Dutch*
LiB	53.06	**51.87**
CBL	52.20	50.04
AG-word	30.10	28.95
AG-collocation	**53.35**	51.45
Word-by-Word	38.32	38.68
Only-Length	50.82	50.57

Table 6.3.: **Evaluations of models/baselines in different languages.** All the scores are the weighted F1 metric between the predicted eye fixations and the observed eye fixations. Bold font indicates the highest score across models.

6.3.5. The effect of unit-length limitation

Next, we evaluate LiB under different limitations of unit length, which captures possible perceptual limitations. The sharp rise of the prediction scores with increasing maximum length quickly levels off (Fig. 6.4). The prediction scores even slightly decrease after the peaks: the optimal maximum unit lengths (indicated by the arrows in Fig. 6.4) are 16 for English (F1 = 53.84) and 13 for Dutch (F1 = 53.16).

6.3.6. Training on non-GECO corpus

The eye fixation data are from the GECO corpora, which are also the model training corpora for the results above. To test the generalizability of the models, we evaluate the models trained on the non-GECO large scale corpora and compare the results with the models trained on the GECO corpora[10]. Table 6.4 shows that training on the non-GECO corpora improves the prediction of eye fixations compared to training on the GECO corpora themselves. This is the case for both LiB and CBL, although the predictions by LiB remain the most accurate. CBL shows high time efficiency since its training is based on words rather than characters (as in LiB and AG). However, CBL, compared with LiB, shows a higher *relative* increase in training time with the same increase of the training materials. Moreover, CBL tracks the frequencies of all words and the backward transitional probabilities of all word pairs, which causes the sharp growth of the lexicon on the large corpora.

[10]The training of the AG model on large scale corpus is not feasible because of its very low time efficiency (more than 10 hours on GECO).

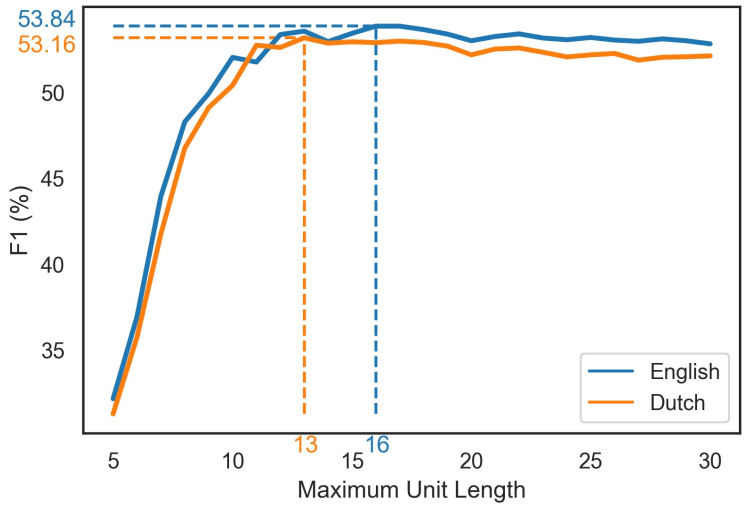

Figure 6.4.: **The prediction scores with different LiB unit length limitations.** The blue/orange dotted lines indicate the peak scores and the corresponding maximum unit length in the English/Dutch simulations, respectively.

Model	*Training corpus*	*English*			*Dutch*		
		Training time	*Lexicon size*	*F1 Score*	*Training time*	*Lexicon size*	*F1 Score*
LiB	GECO	2min31s	15,867	53.06	2min38s	17,525	51.87
	Other	24min51s	97,872	**53.46**	72min5s	143,665	**53.72**
CBL	GECO	1s	29,268	52.28	1s	33,248	50.04
	Other	1min24s	2,051,239	53.30	3min23s	3,782,605	51.71

Table 6.4.: **Comparison of training times and F1 scores (%) between different models and different training corpora.** *Other* means the training corpus is COCA for the English task and SoNaR for the Dutch task. The *Lexicon size* of CBL is the sum count of its stored unigrams and backward transitional probabilities between the unigrams. Bold font indicates the highest score across models.

6.4. Discussion

In this study we have shown how to predict eye fixations on text by unsupervised segmented cognitive units. Conversely, we evaluate these units by their predictions of eye fixations. In particular, we tested three segmentation models: the LiB model, the CBL model, and the AG model. We also compared them with two word-based baselines: assuming that reading is word by word; and assuming that we can predict the number of fixations on each word by the length of the word. Firstly, we found eye fixations can

be better explained by the cognitive-unit-based models than by the word-based models, and both the LiB and AG models predicted the fixations best among the cognitive-unit-based models. Secondly, the predictions are robust between two languages (English and Dutch). Lastly, we found the LiB and CBL models can predict eye fixations on a different corpus, and large-scale training material improves the prediction.

6.4.1. From word-based to cognitive-unit-based reading theories

The evaluations in the current paper show that eye fixations during reading can be predicted by unsupervised text segmentation models (Fig. 6.4; Tables 6.3 and 6.4). These results suggest a cognitive-unit-based eye-movement planning in the oculomotor system. Eye fixation during reading is not arbitrary nor guided by purely orthographic cues such as spaces and punctuations, so the reader's oculomotor system must plan the fixations by using both orthographic cues in the text and top-down knowledge.

Traditional theories of fixation-planning regard words as reading units. To explain the fixations which are not word by word (*fixating words more than once* or *skipping words*), it is usually assumed that a word's lexical attributes (e.g., frequency, predictability, and length) can help to decide whether to refixate or skip the word. An example of such a theory is the E-Z reader model (Reichle et al., 1998; Reichle & Sheridan, 2015; Reichle et al., 2009), which is one of the most popular eye movement models. It assumes that our visual system can preview the text to the right of the current fixation and make use of the lexical attributes of the next word to plan the next fixation position.

Different from the traditional word-based theories, we regard *cognitive* units as reading units and assume that most of the first-pass fixations are at the center of each reading unit (we here ignore the limitation of perceptual span for simplicity and will get back to it later). Based on this assumption, the fixation-planning task may be approximated as a cognitive-unit segmentation task, just as we did in this study. The cognitive-unit-based predictions (from the *LiB*, *CBL*, and *AG* models) are generally better than the word-based predictions (*Word-by-Word* and *Only-Length*) (Table 6.3). *Word-by-Word* assumes reading proceeds with one fixation per word, but the baseline's poor performance undermines this assumption. *Only-Length* assumes that the number of fixations on a word is determined solely by the word's length. It scores higher than *Word-by-Word* showing that longer words tend to be fixated more often (which is already well known). Importantly, *Only-Length* predictions are still worse than the cognitive-unit-based predictions, even though its predictions use the distributions of observed eye fixation, which the segmentation models are ignorant to. To sum up, the cognitive-unit-based approaches can outperform the word-based approaches with less information and even when allowing for unrealistically long units (i.e., longer than the visual span). Therefore, cognitive-

unit-based reading can be seen as a new, and arguably better candidate for explaining eye movement during reading.

The evaluation results are also consistent between English and Dutch (Figs. 6.2, 6.3, and 6.4; Tables 6.3 and 6.4), which shows the validity of the models and the theory of cognitive-unit-based eye movement are not limited to a particular language. To collect more evidence of whether they are indeed language-independent, it would be interesting to run the same study on Chinese, where we may find even better results because there are no perceptual cues (spaces) that guide eye-movements in addition to the cognitive units. However, there is currently no publicly available large-scale Chinese eye-tracking corpus.

It must be noted that the perceptual span of our eyes is limited, so a cognitive unit should get more fixations when it exceeds the span. Perceptual span is not of concern to any segmentation model, but we can examine perceptual span anyway by limiting the unit length in the LiB model. If the maximum unit length is shorter than the real perceptual span, the predicted fixation location would be biased to the left for the cognitive units whose length are between the limitation and the perceptual span; if the maximum unit length is longer than the real perceptual span, the prediction would be biased to the right for the cognitive units whose length exceeds the perceptual span; the optimal maximum unit length should reflect the best fixation prediction. The results did show the best prediction scores when we limit the unit length to 16 in the English prediction task and to 13 in the Dutch prediction task (Fig. 6.4), which is close to the finding that the perceptual span extends to 14-15 letter spaces to the right of fixation (Rayner, 1998).

This finding does not mean that there really is a maximum unit length in cognition. The current study does not aim to improve the prediction of eye fixations: the eye-fixation prediction task in this study only serves to cognitively evaluate the units segmented by the models. For this reason, we needed to prevent any prior information about eye movements (e.g., oculomotor constraints or linguistic knowledge) from "contaminating" the eye-fixation prediction task, which is why we did not include such prior knowledge in the models. However, in possible future work which explicitly aims to predict or explain the eye fixations, we may include the constraints from the physiological system and the linguistic attributes of reading material in the linking hypothesis between segmentation and eye fixation. For example,the attention distribution in the perceptual span is actually asymmetrical (Reilly & Radach, 2006), which causes the optimal viewing position and preferred viewing location to be slightly to the left of the middle of a token (McConkie, Kerr, Reddix, & Zola, 1988; Rayner, 1979) rather than the exact center as we assume in this study. If the fixation fails to land on the ideal location in a token, it will trigger an immediate re-fixation for correction (Nuthmann, Engbert, & Kliegl, 2005). Another example is that high-level linguistic attributes of tokens can

also influence eye fixation by mediating the tokens' predictability (Balota, Pollatsek, & Rayner, 1985; Ehrlich & Rayner, 1981; Warren & McConnell, 2007).

The concept of cognitive unit does have some connections to models of eye movement control in reading. The E-Z reader model assumes that the attention of reading shifts word by word, so it is categorized as a "serial attention shift" (SAS) model (Engbert, Nuthmann, Richter, & Kliegl, 2005; Reichle et al., 2009). An important alternative model is the SWIFT model, which assumes that parallel activation of multiple words over the fixated region is also possible. This model is categorized as a "gradient by attention guidance" (GAG) model (Engbert et al., 2005; Reichle et al., 2009). Although E-Z reader and SWIFT are still word-based models, so neither supports the processing of multi-word groups as single units, the latter model shares with the LiB model the idea that multiple words can be activated in parallel. Besides, human processing of cognitive units may involve two stages (familiarity detection and recognition; Yang, Cai, and Tian (2020)) which are similar to the two stages (familiarity check and lexical access) assumed by the E-Z reader model (Reichle et al., 2009).

6.4.2. Cognitive units from different models/motivations

We have seen the advantage of cognitive-unit-based predictions for explaining eye fixation during reading (Table 6.3). The question then turns to which model best segments the cognitive units from text, that is, which model more accurately predicts the eye fixations. The answer is also in Table 6.3: LiB is on par with AG-collocation, AG-word performs the worst, and CBL is in between. Moreover, LiB (unlike AG-word and AG-collocation) predicts longer fixation distances on Dutch than on English, in accordance with the observed pattern (Fig. 6.2). The performance differences between the models may reflect differences between *how our cognition defines the units* and *how the model defines the units*.

The CBL model follows the notion that both children and adults can learn multi-word sequences from words, and that learning is based on the transitional probabilities (McCauley & Christiansen, 2017). The units in CBL are words and multi-word sequences, not subwords. However, McCauley and Christiansen (2019) also admit that learning directly from individual words is unrealistic for children. In addition, both CBL and LiB tend to memorize frequent units, but only LiB also forgets the units that are frequent but increase the numbers of types or tokens and therefore violate the principle of least effort. Thus, in the current study, the higher performance of LiB over CBL may be attributed to the character-based learning and the principle of least effort.

The learning in AG takes another approach: it tries to infer the *optimal* (in the Bayesian view) tree structures to represent the given language material (Johnson, 2008). The model clusters the symbols in the corpus in a hierarchical way so its output units are shown at the middle level(s) of the hierarchy (Johnson & Goldwater, 2009). The similar

performance of AG-collocation and LiB (Table 6.3) arises in spite of their very different structures and workflows. Although AG is based on Bayesian estimation whereas LiB is based on cognitive assumptions, both models aim to optimize the lexicon and the segmentation, and thereby learn supra-word and subword units (Fig. 6.3), which may result in their similar performance. Another interesting phenomenon is that the AG-collocation units show much better performance than the AG-word units in the fixation prediction task (Table 6.3). Since AG-word finds only very few supraword units and many subword units (Fig. 6.3), the higher performance of AG-collocation and LiB suggests that language cognition prefers larger units.

The motivation underlying the LiB model is the least-effort principle: LiB regards the text chunks fitting the least-effort requirement as the cognitive units during reading. This motivation follows William of Ockham's (1287–1347) law of parsimony, which is also known as *Occam's razor*. The law of parsimony for cognition is applicable since cognitive resources are limited. This motivation also follows Zipf's (1949) argument that all human behavior can be systematized under the Principle of Least Effort (PLE). Although neither Occam's razor nor PLE is tangible and quantifiable enough for a computational model, LiB implements their philosophy by interpreting least effort (in language processing) as less use of both working memory (the number of cognitive unit tokens) and long-term memory (the number of cognitive unit types).

The balance between working and long-term memory can be seen as the balance between computation and storage, which is still under debate. Chomsky and Halle (1968) believed complex words are generated from simpler forms. Baayen (2007) criticized this generative theory, because in that case the balance of storage and computation is shifted totally to the maximization of computation and the minimization of storage. He, in turn, claimed the importance of storage, but did not provide a measure of the two. Minimum description length (MDL; Rissanen, 1978) fills the blank to some extent: MDL describes both storage and computation by their required encoding bits and so MDL unifies the two parts. Yang, Frank, and van den Bosch (2020) showed that LiB also minimizes description length of a corpus compared to some other models. MDL assigns storage and computation the same weights. However, they are in different cognitive systems (long-term memory versus working memory) and may have different cognitive processing costs. These costs may also depend on individual differences. In the LiB model, these differences can be reflected in hyperparameters.

Moreover, the cognitive units should be generalizable if we want them to be practical. The reading experience of an educated adult relies to a large extent on language materials. It is meaningless if the language users learn the cognitive units from some piece of language materials but cannot use them on new material. Fortunately, a task-independent but large-scale corpus can help to discover cognitive units that are at least as usable as those from the task-specific corpus (Table 6.4). This finding demonstrates the training generalizability of the segmentation models and the external validity of the

trained cognitive units. Besides the better performance, it is also worth noting that the time and memory costs of LiB on large training data are reasonable because LiB only requires simple computations (compared with Bayesian computation) and a small lexicon (compared with tracking all unit frequencies or even bigram transitional probabilities). The saving of time and storage suggest that the LiB lexicon is in itself actively trying to optimise towards a saturation point, or to converge towards a set of *good* cognitive units.

6.4.3. Room for improvement of cognitive unit discovery

The ability to predict eye fixations demonstrates the cognitive reality of the concept *cognitive unit*, but cognitive units can do more than predict the eye fixations. Those units by definition are the building blocks of human language processing. They may serve as better operational units in computational linguistics, psycholinguistics, language education, translation, and so on. As an example in computational linguistics, the corpus segmented into LiB's cognitive units shows more concise description and lower N-gram language model perplexity than when words form the units (Yang, Frank, & van den Bosch, 2020). All in all, it is still worth seeking ways to improve the discovery of cognitive units.

Although the hyperparameters for training LiB in the current study had almost the same values as in a previous LiB study (Yang, Frank, & van den Bosch, 2020), which is unrelated to eye-fixation prediction and thereby avoids the double-dipping issue, we still want to decouple LiB from its hyperparameters to discover the cognitive units shared by most users of a language or the cognitive units that reflect the shared thoughts in multiple languages. CBL is an exemplar of such decoupling because it has no hyperparameters and its built-in parameter (the frequency threshold for constructing a chunk) is adjusted according to the running average of the chunk frequencies. We intend to also make the hyperparameters adaptive in the future LiB model. Alternatively, we may aim to make LiB into a dissipative system (a system that can reach a steady state when it interacts with the environment), more self-organized and insensitive to the initial hyperparameters.

Decoupling LiB from its hyperparameters enhances the generality of the model. On the opposite side, the model can be tuned specifically to simulate the individual properties of a human agent; for example, the unique lexicon of a person with aphasia, or the change of a child's mental lexicon during language acquisition. Introducing more hyperparameters related to individual cognitive differences may help to discover idiosyncractic cognitive units. Possible relevant hyperparameters could be the perceptual span and the balance between long-term memory and working memory that we have discussed above, and other empirical knowledge of physiology.

Lastly, we should note that the prediction scores of different models vary within a narrow range. Also, altering training material from GECO to 100 times larger corpora did not lead to an F1-score improvement of more than two percentage points. The reason for the apparent performance ceiling could be that the current LiB model, as well as the CBL model and the AG model, discover only the frequent units. Some infrequent units can also be cognitive units: for example, people may immediately memorize the name of a never-heard city in a breaking news since the name is salient in the context. The current LiB model is not sensitive to such contextual semantic and pragmatic information.

6.5. Conclusion

The current study demonstrates the advantage of cognitive-unit-based reading theories over traditional word-based reading theories by using an eye-fixation prediction task. Among the computational implementations of cognitive-unit-based reading as unsupervised word segmentation, the LiB model shows good performance and high efficiency, and indicates that least effort in both working memory and long-term memory may play an important role during language learning and processing. Overall, the study supports the theory that the mental lexicon stores not only words but also smaller and larger units, suggests that fixation locations during reading depend on these units, and shows that unsupervised segmentation models can discover these units.

7 | General discussion

"Linguistics accordingly works continuously with concepts forged by grammarians without knowing whether or not the concepts actually correspond to the constituents of the system of language. But how can we find out? And if they are phantoms, what realities can we place in opposition to them?" -- (De Saussure, 1916/1959, p. 110)

7.1. Summary of the findings

My investigations into cognitive units of language consist of two parts: 1. The laboratory studies provided empirical evidence to a theoretical model describing the processing of cognitive units; 2. The computational model simulated the mechanisms of learning of, and segmentation into, cognitive units.

7.1.1. How readers process cognitive units nearby a point of gaze: the two-stage workflow

I investigated the processing dynamics for different cognitive units in the language hierarchy at a point of gaze. The first experiment, described in Chapter 3, was a behavioral lexical decision task whose stimuli were four-character Chinese strings that are short enough to be perceived in one eye fixation. With four-character strings, we were able to manipulate the recognizability[1] of the global string (four characters) and two local strings (two characters each).

We measured the reaction time of participants in the global/local lexical decision task. The behavioral results showed that the lexical decision of local lexicalized strings were influenced by the lexical status of the global string, but not vice versa; the lexical decision of an unlexicalized string at any level was influenced by the lexical status of the other level. The task result suggested the higher priority for processing global recognizable units over local units; and if the process of global units had not been finished before the process of global units, the two types of processes would interact.

Then we modified the test into an EEG test to observe the dynamics of processing (Chapter 3). The EEG result showed that the processing workflow was divided into

[1] In a pre-test, we rated the wordness of the two/four-character strings. But besides the two-character words, four-character idioms and familiar compounds are also considered as words in Chinese. In order to disambiguate the notion in a general sense, we use *recognizability* instead of *wordness* here and below.

two stages: after about 100 ms from stimulus onset, the recognizable units, no matter whether they are global or local, are processed in a short time window; the global units will be processed again at about 160 ms, and the local units at about 220 ms.

According to those results, we proposed a two-stage workflow to describe how readers process the text nearby a point of gaze

- *Detection stage*: within 100 ms, the reader can detect different text chunks (e.g., a phrase"*blue sky*", and its components "*blue*" and "*sky*") simultaneously near a point of gaze, as long as the reader's mental lexicon has stored them as cognitive units, and regardless of their complexities. This stage pre-activates the potential cognitive units to make them entries for subsequent recognition;

- *Recognition stage*: the pre-activated units may be distributed at different linguistic levels. The resources allocated to them are unequal: global units (e.g., "*blue sky*") will be recognized earlier than local units (e.g., "*blue*" and "*sky*"). However, their recognitions can occur in parallel and may interact.

I also attempted to replicate the study in Chapter 3 using another behavioral experiment and another EEG experiment. The stimuli used in Chapter 3 were presented in isolation, rather than within the surrounding text. However, in natural reading environments, the cognitive units are usually surrounded by more text. To test whether the findings in Chapter 3 will generalize to a more natural reading environment, the stimuli in replication experiments used rare Chinese characters (as surrounding text) to fill a four-character Chinese string (as the reading target). The results of Chapter 3 were partially replicated: behavioral experiments in the original and new studies showed similar results; both EEG experiments found early detection of available cognitive units, but the new study neither replicated nor falsified the global priority effect of cognitive units. Several possible explanations for the absence of global precedence effects have been discussed in Section 4.4.

7.1.2. How readers learn and segment cognitive units from text: the *Less-is-Better* model

The behavioral and EEG findings not only covered a cognitive mechanism, but also confirmed that cognitive units are flexible in terms of size and linguistic level. But how do humans learn the flexible units, and how do humans identify the flexible units in continuous language input? In order to provide an answer, I computationally modeled the human ability to learn and segment cognitive units (Chapter 5).

There is no ground truth of cognitive units provided to human language users, and therefore the current model does not rely on labeled data, i.e., the model is unsupervised. An unsupervised model requires a learning objective to guide the discovery from

unlabeled data. To design such a learning objective, I hypothesize that language users would attempt to make their language cognition more efficient, and cognitive units are the adaptive result of the attempt. In other words, *cognitive units are the units that can minimize the cognitive load*. More specifically, the objective of the model is to minimize long-term memory load (the numbers of unit types) and working memory load (the numbers of unit tokens) simultaneously. I named the model *Less-is-Better* (LiB), and I will explain the logic of the objective in more detail in the next section.

With the objective of minimizing the numbers of unit types and tokens, the LiB model consists of a *Memorizer* and a *Forgetter*. *Memorizer* will merge some adjacent units in the corpus into new (longer) units[2] and store them in a Lexicon. With longer units, the number of unit tokens in text decreases while the number of types increases. In contrast, *Forgetter* will remove "junk" (less useful) units from the Lexicon to reduce the number of unit types. The junk units may be unit types that will increase the number of unit tokens in the corpus[3], as well as some unit types that do not occur frequently in the corpus. *Memorizer* and *Forgetter* counterbalance each other and finally converge to a (relatively) steady state, and the Lexicon at the steady state consists of the units that are close to the minimization objective. The model derived units that are flexible, such as "you", "you can" and "ly", which is in line with my general views about cognitive units. However, more qualitative evaluations are needed.

In Chapter 5, I evaluated the model-derived units computationally. In terms of description length (a metric in information theory) and bits-per-character (a metric of language model quality), the LiB units display advantages over units composed of characters, words, and units generated by the Byte Pair Encoding (BPE) algorithm.

The computational advantages cannot provide empirical evidence for the plausibility of the model-derived units as cognitive units. Ideally, evaluation of the plausibility requires knowing what the true cognitive units are. However, there is no means to measure cognitive units in our mental lexicon directly. Therefore, I made two hypotheses to examine the relationship between the model-derived units and cognitive units: 1. LiB units reflect cognitive units; 2. the locations of cognitive units in text steer eye fixations during reading. The eye-fixation is an empirical human behavior and is independent of the design and training of the LiB model. In that sense, only cognitive units can link the model predictions and the eye fixations. Thus, if we can use the LiB units to predict the locations of eye fixations during reading, the above two hypotheses will be supported at the same time. In Chapter 6, I tested this idea on a dataset that consists of both text and eye-fixation data, and found the LiB units offered advantages both in terms of pre-

[2]For examples, "g" and "o" -> "go" "going" and "to" -> "going to".

[3]For example, in case the Lexicon has already stored "go ", "goi", "ing", and their letters, the string "going" can be segmented to "go/ing " or "goi/n/g". Because the shorter unit "go" results in fewer units in the segmentation, the longer unit "goi" will be regarded as a junk unit type.

diction score and efficiency among the alternatives. These positive results support the hypothesis that the LiB units are indeed analogous with cognitive units.

7.2. Linking the findings: The need of cognitive economy / the principle of least effort

In theory, the two-stage workflow in reading describes how readers detect and recognize cognitive units nearby a point of gaze in text. The LiB model computationally describes how readers learn and segment cognitive units from a lot of text. Although they describe different aspects of Cognitive unit processing, we could notice that both of them reflect cognitive economy in their processing.

The two-stage workflow describes economic scheduling of processing. Once a stimulus is displayed, the reader will first attempt to detect the potential cognitive units in the stimulus. The detection requires only the surface information (i.e., form) of those units, so the activation of those units is also shallow, fast, and low-cost at this stage. In contrast, it would be redundant for the reader to fully process all potential cognitive units from the beginning, as I will explain under *Synchronous* strategy below.

The detection stage prepares multiple cognitive units for recognition. Then there might be different strategies to schedule the recognition of these units:

- *Local only*: recognizing only minimal units. This would hamper comprehension of opaque expressions such as idioms;
- *Global only*: recognizing only maximal units. This would hamper comprehension of unfamiliar expressions;
- *Local first*: recognizing minimal units before maximal units. This might be ambiguous in case of opaque expressions, since both the metaphorical meaning and the literal meaning would be accessed;
- *Global first:* recognizing maximal units before minimal units. This would not hamper comprehension. In case the processing of the global units take too much time, the processing of the local units would catch up and assist the comprehension of the global units;
- *Synchronous*: recognizing different units simultaneously. In case of opaque expressions, this has the same shortcoming as the *Local first* strategy; in case of transparent expressions, accessing meaning that exists at different levels is redundant.

In the comparison of strategies above, we can see that the larger-first strategy in the recognition stage, which is also supported by the empirical evidence, will reduce the pro-

cessing load on the smaller units, which may be redundant for the final comprehension. Therefore, both stages of unit processing reveal the economy of our reading ability.

Chomsky's linguistic theory has been updated several times. In the most recent update (i.e., Minimalism) he also claimed that the design of language is determined by the "principle of economy" (1995, p. 168). He did not clarify the source of such an economy, but usage-based linguists might agree and explain that language use is adapted to be more economic under our needs. In general, it appears that the current form of language (e.g. dependency length and word length) is economic for our language system (Futrell, Mahowald, & Gibson, 2015; Piantadosi, Tily, & Gibson, 2011). That is to say, the current form of language reflects the principle of least effort in our cognition, and we can use this principle to investigate language, as we did when designing the LiB model.

From Aristotle's preference of "the demonstration which derives from fewer postulates or hypotheses" in his *Posterior Analytics*, to William of Ockham's (1287–1347) law of parsimony (also known as Occam's razor), all tried to reveal the simplicity of nature. The principle of least effort inherited those ideas and extended to the parts of human nature -- behaviors and cognition. Loh Seng Tsai stated the animal "tends to select that [behavior] which involves the least expenditure of energy" (1932). Tsai's statement can also be inferred from the common idea of evolution: the species/genes/behaviors that are more efficient for the current environment are more likely to survive. However, "the least expenditure of energy" in Tsai's statement implies a pitfall: it in general means the current effort for the behavior rather than the future effort raised by the behavior. Therefore, George Kingsley Zipf, who formally proposed the *Principle of Least Effort*, explained the principle as "[the person] will strive to solve his problems in such a way as to minimize the total work that he must expend in solving both his immediate problems and his probable future problems." (Zipf, 1949, p. 1). Zipf's emphasis on both "immediate" and "probable future" effort perfected Tsai's statement.

Language is a part of human behavior and cognition, so I considered language as a system that obeys the principle of least effort, and cognitive units are the result of applying the principle of least effort to language. It opened the door to discover the flexible cognitive units in a formal way. However, Zipf's principle of least effort is more like a metaphysical proposition, since "effort" is hard to measure. To solve this issue and build a computational model, I define "effort" as the load on both working memory and long-term memory. In other words, the principle of least effort turns out to be:

- Minimizing the load of **both current and prospective working memory**;
- Minimizing the load of **long-term memory**.

More specifically for language, the above two goals should be operationalized as:

- Minimizing the number of processing units in all potential texts (i.e., **unit tokens**);
- Minimizing the number of stored units in the lexicon (i.e., **unit types**).

By defining cognitive units as the language chunks that satisfy the minimization of the numbers of unit tokens and types, the theory of cognitive units becomes formal, and the computational model for discovering cognitive units becomes feasible.

Of course, working memory and long-term memory are only parts of our language-cognition system. These memories store the unit tokens and types, which represent the morphological aspect of language, but language also involves syntax, semantics, and pragmatics. Therefore, according to the view that cognitive units are the result of applying the principle of least effort to language, we should also minimize the complexity of dependency relations, the steps of meaning encoding and decoding, the social pressure, etc., in order to obtain cognitive units. But as I have shown in Chapter 3 and Section 7.1.1, the detection of cognitive units can be based their morphological form. This is why the units learned by the LiB model, which only takes into account working and long-term memory, can be close to cognitive units and predict eye fixations in reading. However, any model is a simplification of reality. In case we require more precise cognitive units, we need to minimize more objectives in the LiB model.

7.3. The current answers to the research questions

Language is discrete, but the cognitive units of language are not simply words or any other linguistic units that have been well studied. To obtain a more clear picture of cognitive units and their processing, I have posed three questions in Section 1.4, and the research findings I presented in this thesis provide the corresponding answers to different extents.

7.3.1. What will be learned as cognitive units?

Cognitive units may be any form of linguistic chunk: word, morpheme, idiom, etc. They do not need to be indicated by perceptible dividers, or by an external dictionary. But which linguistic chunks will be learned as cognitive units? The question is related to the definition of cognitive units. Cognitive units cannot be defined by their forms because their forms are too flexible; instead, the previous chapters answered the question from a functional view: **cognitive units are the units that minimize the cognitive load**. In the usage-based view, cognitive units are the product of language usage. Inefficient units are more likely to be abandoned during usage, so which units will be retained in memory and which will be forgotten will follow the principle of least effort.

How to implement the learning of cognitive units in human's cognitive system? Since we know little of the mechanism, one may simply attribute the black-box mechanism to a specific module. If such a module does exist, the aim of the minimization requires the module to be able to measure the cognitive load of other modules in the language system. In contrast, the LiB model proposes that cognitive units can emerge from the interaction of different general cognitive modules (e.g., storage, computation, emotion, etc.) each trying to minimize its own load[4]. In that sense, cognitive units rely only on general cognitive modules, so no specific module is needed. Furthermore, the flexible forms of cognitive units can be explained as the unpredictable interactions among the modules, rather than intricate rules programmed in a single module.

The states of different cognitive modules can play different roles for learning cognitive units. For example, the capability of the long-term memory limits the number of unit types, the capability of working memory affects the number of unit tokens and the complexity of unit dependencies. Besides the cognitive modules, the language experience is also an important factor in the learning. We cannot expect an uneducated person to have the same mental lexicon (of cognitive units) with a literary giant, nor can we even expect a person to have the same mental lexicon in childhood and adulthood.

Even with the same lexicon, the *working* cognitive units may be different under different environments. For example, while speaking to a child, an adult may tend to use fewer unit types but more unit tokens than when addressing another adult. Therefore, we should notice that cognitive units are not fixed as linguistic units. If we need to estimate the mental lexicon of an individual, shared by a population, or shared under an environment, we must take the individual/population's attributes (e.g., age, nativeness, or with/without language disorder) and the usage environment into consideration.

It is also worth noting that cognitive units are not just the result of frequency, even though frequency is the most convenient standard to determine the flexible units. But let us take these cases into consideration: 1. the individual letters are the most frequent units, but they may not be cognitive units; 2. people may immediately memorize a newly-heard name as a cognitive unit. These cases suffice to support the idea that cognitive units involve *more* than frequency. And this *more* is, as I said above, the emerging result of the counterbalancing of different cognitive modules.

In summary, cognitive units are the units that help to minimize the cognitive load. Cognitive units are not pre-defined by linguistics, but are the emerging result of different cognitive modules each trying to minimize their own load. Although for research we can estimate the general cognitive units, the genuine cognitive units may vary between individuals or between settings.

[4]We should notice that the counterbalancing of different cognitive modules does not lead to the global minimum of cognitive load.

7.3.2. How to segment language input into cognitive units?

Cognitive units are not perceptually highlighted in language input, but as we discussed above, cognitive units can be learned unsupervisedly. The learned cognitive units are stored in the mental lexicon, and the mental lexicon will guide the segmentation that online discretizes language input into cognitive units.

Since cognitive units are independent of linguistic levels, the segmentation into cognitive units may involve different complexities. Such segmentation links to a controversial debate between the composition view and the whole-form view in psycholinguistics (Fiorentino et al., 2014; Koester & Schiller, 2011; Leminen et al., 2019; Stites et al., 2016). The composition view argues that only morphemes, the minimal meaning-bearing units, are stored in the mental lexicon, and the processing of language is based on the composition of morphemes. The whole-form view argues that frequent compound words are represented and processed as units, so the units of different complexities are equally stored in the mental lexicon (Arnon & Snider, 2010; Siyanova-Chanturia et al., 2017), and the composition is redundant for these complex but frequent compound words (Blache, 2015; Ellis, 2003; Krishnamurthy, 2003).

The concept of Cognitive unit naturally supports the whole-form view in the segmentation since the traditional linguistics units of varying sizes are integrated into cognitive units, which is a single level. In Chapter 3, the EEG experiment result shows that the lexical units at different levels are detected simultaneously, which provides empirical evidence for the whole-form view. We can assume such segmentation physiologically acts like an attractor network, which will lead the input to the closest attractor (stable pattern). The learning of cognitive units creates a different attractor for each unit in the network, and when the network perceives the language input containing one or more cognitive units, the attractors of these cognitive units will be activated in parallel.

The timing of the segmentation is also a crucial question. In case meaning was involved for segmentation, the segmentation might be more accurate, but it also means the segmentation would take until semantic processing is finished. In contrast, if only form information is involved, the segmentation can be achieved quickly since the access of semantic information should rely on the access of form information. The EEG result in Chapter 3 supports early and fast (after about 100 ms and lasting less than 50 ms) segmentation. Because this time window is too early and short for semantic analysis, it is more plausible for visual form analysis, which is at the lower level of cognitive processing. The LiB model, which utilizes only form information, can also learn the units close to cognitive units. Therefore, the segmentation strategy that relies on only form matches the principle of least effort, and benefits the speed needed for online segmentation. Moreover, the decoupling of form and meaning is also more economical for the meaning processing after segmentation, which I will discuss more in the next question (see Section 7.3.3).

In summary, the segmentation of language input is fast enough to support online processing. Within the 100 ms after gazing at a point, the cognitive units which are stored in the reader's mental lexicon and displayed in the gazing text can be detected. The detection is quick because it only pre-activates the form of those cognitive units.

7.3.3. How to process the segmented cognitive units?

The raw form of language input is usually variable and complex, so understanding it as a whole is difficult. But by segmenting the language input into the combination of cognitive units, readers can understand the input based on the knowledge of these cognitive units. An important issue here is that cognitive units may form hierarchical treelets (e.g., *[[blue] [sky]]*) instead of flat sequences (e.g., *blue | sky*). The processing order of a hierarchy is uncertain: it may be top-down, bottom-up, or other ways I have discussed in Section 7.2. To deal with the uncertainty, the language system of humans should also be able to arrange the processing of cognitive units. The experiments in Chapter 3 suggest that the schedule of the processing is top-down, that the processing of the global units in a hierarchy will start earlier than of the local units, and then the processing of both units will be in parallel.

Top-down processing has been studied for a long time in many domains. The earliest idea may come from holism in many kinds of philosophies, such as Tai Chi and Aristotle's "*The whole is more than the sum of its parts*". The latter also influenced Max Wertheimer's Gestaltism theory in the early 20th century (see (Brett & Michael, 2017)). In the study of vision, Navon (1977) found the global precedence effect , which suggests that the processing of the global level is faster than of the local level while recognizing a scene. The priority of global level was also found in psycholinguistics as the word superiority effect (Reicher, 1969), that the letter in words can be better recognized than the letter presented alone or in nonwords. The finding in Chapter 3 further extends the superiority to phrases, which suggests the superiority effect as a generalized effect of varying global language units.

The studies of top-down processing usually focus on the priorities for global units, and Chapter 3 further elucidates the reason for the priority - the earlier processing onset of global units. This finding is consistent with Bar et al.'s (2006) study; they found the visual top-down effect results from the coarse representation of input (i.e., global information) that projects from early visual areas rapidly through the dorsal pathway, while the fine representation (i.e., local information) propagates relatively slower along the ventral pathway. My finding and Bar's finding share the same logic, both for faster processing: the processing of the coarse representation/global unit may quickly detect the gist of the object, and so the processing of the detail representation/local units is not needed. Even if a coarse representation/global unit leaves uncertainty, it still narrows

the predictions of the object and relieves the processing load of the detailed representation/local units.

The priority of the global cognitive units does not only benefit the speed of processing, but also reduces the ambiguity in processing. Some of the global units are "opaque" (e.g., "hot dog"), which means the composition of the meaning of local units cannot be simply combined to determine the meaning of the global unit. Taking "*The whole is more than the sum of its parts*" more seriously, the direct access of any global unit may obtain a more appropriate meaning than compositing the local units. For example, even the word "blue sky" is not just a literal combination of "blue" and "sky", it also metaphorically suggests an ideal situation.

Moreover, as I mentioned before, the economy in the processing benefits from the decoupling of form and meaning. The segmentation involves only the form, not the meaning of different units. The advantage of this strategy is that the processing of the local unit can be omitted when the meaning of the global unit can be processed quickly, and it also avoids ambiguity if the global unit and the composition of the local unit have different meanings.

In summary, the global and local cognitive units in a hierarchical treelet are processed parallel, but the global cognitive units are processed earlier. This processing schedule may speed up processing and reduce ambiguity.

7.4. Extending the notion of cognitive units to different fields

7.4.1. Cognitive units and grammar

Grammar refers to the roles of abstract language units and the structural constraints between them. Parsing is the application of grammar to find the structure of a sentence. The current LiB model does something similar to parsing: It can get the sequence of units (e.g., *I like blue sky -> [I like] [blue sky]*), and also the treelets belong to those units by segmenting the units again (e.g., *[I like] [blue sky] -> [[I] [like]] [[blue] [sky]]*). However, the mechanism of the current LiB model does nothing related to grammar. This is just because the current LiB model aims to be as simple as possible to show its theoretical value[5]; it concerns only the unit tokens and types.

However, the LiB model can be adapted to represent grammar. A future version of LiB may not only combine the adjacent units (N-gram), but also the nonadjacent units (skip-gram), into a new unit. The skip-gram units can capture the long-distance relationship between tokens (e.g., "*too_to_*", "*is_by_*", and "*the_.*"). The skip-gram units

[5] "Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful." (Box and Draper, 1987, p. 74)

can also benefit the detection of infrequent entities (e.g., the skip-gram "*Mr._said*" helps to detect "*Mortimer*" in "*Mr.Mortimersaid*"). Furthermore, skip-gram units integrate the grammatical structure into the lexicon, so the function of cognitive units is closer to Cognitive grammar (R. W. Langacker, 1987), which regards the grammar and lexicon as a continuum.

7.4.2. Cognitive units and Natural Language Processing

Natural Language Processing (NLP) studies how to program computers to process human language. As do humans, computers need to discretize language input into units (tokenization). The traditional tokenization approach is to use words as the units, but there are too many infrequent words. Accepting all of them would raise an extremely large lexicon, which is inefficient for processing. A solution is to segment rare words into smaller units, such as characters[6] or subwords from Byte-Pair-Encoding (BPE) (Sennrich et al., 2016). However, some units larger than words (e.g., "*machine learning*", "*kick the bucket*") can improve semantic tasks since their global forms contain extra information.

Maybe the Cognitive-unit tokenization is a better solution. cognitive units include units are, within, or beyond words; and the features of cognitive units - fewer tokens and fewer types - benefit the computation cost of NLP. In Chapter 5, I have shown the advantage of cognitive units on simple language models. It is worth evaluating cognitive units on the giant language models, which usually cost enormous computation resources, such as BERT (J. Devlin, Chang, Lee, & Toutanova, 2018) and GPT-3 (Brown et al., 2020).

Cognitive units may also be a better solution for machine translation. As I mentioned above, the meaning of some multi-word units cannot be deduced from the meanings of the individual words. And some words in a language have no corresponding words in another language. For example, the English word "*pragmatism*" corresponds to the Chinese phrase "实用主义", the English phrase "*open source*" corresponds to the Chinese word "开源". A common solution to the asymmetry is to add an auxiliary memory that stores these asymmetric translation pairs, but it is a manual task that takes time and is also error-prone. Instead, "*pragmatism*", "实用主义", "*open source*", and "开源" may all be induced as cognitive units, and the LiB model can segment them out from sentences for the next translation.

7.4.3. Cognitive units and information theory

Shannon (1951) has estimated the entropy of English. Although his method is rough (asking human subjects to guess upcoming characters in a short passage) from a modern point of view, he showed that the information in a language can be measured. Not only

[6] https://karpathy.github.io/2015/05/21/rnn-effectiveness/

do different languages have different entropies, different types of units in a language also have different entropies. In Chapter 5, I have shown that a language has lower entropy when it is encoded by cognitive units than by characters, words, or BPE subwords. This indicates the high efficiency of cognitive units in terms of information theory (and also validates that cognition is an efficient machine).

However, I should note some gaps (maybe small, but they exist) between the language in cognition and the language in information theory. Although minimal description length (MDL) is a common information-theoretical tool to measure the complexity of an object, there is a shortcoming of MDL to describe human language system: MDL measures both lexicon and corpus in bits and simply sums their bits; that means their weights are equal, but the weights in the cognitive modules representing lexicon and corpus may not equal. Another gap is in the encoding/decoding of language. Information theory usually sees decoding as the symmetric operation of encoding. But in the real world, the encoders and decoders in language communication are always asymmetric. This issue might be solved by a deeper investigation of information processing, such as Logical depth proposed by Charles H. Bennett (1995), which emphasizes information *value* ("*the value of a message is the amount of mathematical or other work plausibly done by its originator, which its receiver is saved from having to repeat.*") more than information entropy.

7.4.4. Cognitive units and complex systems

If a concept is formalized, much of the ambiguity in the concept can be avoided. The concept of cognitive units is deeply rooted in the usage-based perspective. Usage-based linguistics emphasizes the flexible usage of language and to some extent opposes the formal limitations of language. It therefore may seem as if usage-based linguistics is to a lesser extent formalized. Because of this relative lack of formalization, theories of usage-based linguistics are often difficult to analyze language quantitatively[7].

For formalizing the theory related to a phenomenon, it is more straightforward to fit a descriptive model from the statistical pattern of the phenomenon than to build a generative model using the mechanisms behind the phenomenon. However, this approach has a weakness: many types of models will treat rare events as exceptions/noises/errors, while they are just as real and productive as more frequent events (van den Bosch & Daelemans, 2013). In case we want to include the rare events, the model might be overcomplicated; not to say it might be impossible to observe all rare events. Chomskyan linguistics argues against descriptive linguistics, and provides a generative approach to formalize language and simplify the analysis of language. However, I argue that Chomskyan linguistics is still not flexible enough: it allows only a limited class of units (i.e.,

[7]There are a few exceptions such as Fluid construction grammar (Steels & De Beule, 2006) and Memory-Based Learner (Daelemans & van den Bosch, 2005).

words/morphemes); it cannot deal with idiomatic units, let alone the flexible cognitive units I am proposing.

From the usage-based perspective, language involves not only the symbolic language, but also the human cognitive modules, and even the environment. Therefore, language is a *complex system*, which refers to the dynamical system containing multiple modules that interact with each other. An important attribute of complex systems is that complex phenomena may emerge from the complex interactions of modules while the working rules of the modules are simple. In other words, formalizing complex language phenomena in complex systems may be simple. An example of complex system is the Less-is-Better model introduced in the current thesis: the model simulates the interactions of several cognitive modules and the language input, and can discover the flexible cognitive units unsupervisedly.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... Zheng, X. (2016). TensorFlow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 265–283.

Abel, B. (2003). English idioms in the first language and second language lexicon: A dual representation approach. *Second Language Research*, *19*(4), 329–358. doi: 10.1191/0267658303sr226oa

Almohizea, M. I. (2016). The placement of idioms in traditional and non-traditional approaches. *International Journal of Language and Linguistics*, *3*(5).

Andrews, S., Miller, B., & Rayner, K. (2004). Eye movements and morphological segmentation of compound words: There is a mouse in mousetrap. *Eur. J. Cogn. Psychol.*, *16*(1-2), 285–311. doi: 10.1080/09541440340000123

Araújo, S., Faísca, L., Bramão, I., Reis, A., & Petersson, K. M. (2015). Lexical and sublexical orthographic processing: An ERP study with skilled and dyslexic adult readers. *Brain Lang.*, *141*, 16–27. doi: 10.1016/j.bandl.2014.11.007

Arlot, S., & Celisse, A. (2010). A survey of cross-validation procedures for model selection. *Stat. Surv.*, *4*, 40–79. doi: 10.1214/09-SS054

Arnon, I. (2015). What can frequency effects tell us about the building blocks and mechanisms of language learning? *J. Child Lang.*, *42*(2), 274–7; discussion 316–22. doi: 10.1017/S0305000914000610

Arnon, I., & Priva, U. C. (2013). More than words: The effect of multi-word frequency and constituency on phonetic duration. *Lang. Speech*, *56*(Pt 3), 349–371. doi: 10.1177/0023830913484891

Arnon, I., & Snider, N. (2010). More than words: Frequency effects for multi-word phrases. *J. Mem. Lang.*, *62*(1), 67–82. doi: 10.1016/j.jml.2009.09.005

Baayen, R. H. (2007). Storage and computation in the mental lexicon. In G. Jarema & G. Libben (Eds.), *The mental lexicon: Core perspectives* (pp. 81–104). Amsterdam: Elsevier. doi: 10.1163/9780080548692_006

Balota, D. A., Pollatsek, A., & Rayner, K. (1985). The interaction of contextual

constraints and parafoveal visual information in reading. *Cogn. Psychol.*, *17*(3), 364–390. doi: 10.1016/0010-0285(85)90013-1

Bannard, C., & Matthews, D. (2008). Stored word sequences in language learning: The effect of familiarity on children's repetition of four-word combinations. *Psychol. Sci.*, *19*(3), 241–248. doi: 10.1111/j.1467-9280.2008.02075.x

Bar, M. (2003). A cortical mechanism for triggering top-down facilitation in visual object recognition. *J. Cogn. Neurosci.*, *15*(4), 600–609. doi: 10.1162/089892903321662976

Bar, M. (2007). The proactive brain: using analogies and associations to generate predictions. *Trends Cogn. Sci.*, *11*(7), 280–289. doi: 10.1016/j.tics.2007.05.005

Bar, M., Kassam, K. S., Ghuman, A. S., Boshyan, J., Schmid, A. M., Schmidt, A. M., ... Halgren, E. (2006). Top-down facilitation of visual recognition. *Proc. Natl. Acad. Sci. U. S. A.*, *103*(2), 449–454. doi: 10.1073/pnas.0507062103

Barman, B. (2012). The linguistic philosophy of noam chomsky. *Philos. Prog.*, 103–122. doi: 10.3329/pp.v51i1-2.17681

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: A practical and powerful approach to multiple testing. *J. R. Stat. Soc. Series B Stat. Methodol.*, *57*(1), 289–300. doi: 10.1111/j.2517-6161.1995.tb02031.x

Bennett, C. H. (1995). Logical depth and physical complexity. In *Computerkultur* (pp. 207–235). Vienna: Springer Vienna. doi: 10.1007/978-3-7091-6597-3_8

Bernstein-Ratner, N. (1987). The phonology of parent-child speech. *Children's language*, *6*(3).

Blache, P. (2015). Hybrid parsing for human language processing. In *Natural language processing and cognitive science* (pp. 9–20). Venice, Italy: Libreria Editrice Cafoscarina.

Blache, P., & Rauzy, S. (2012). Robustness and processing difficulty models. A pilot study for eye-tracking data on the french treebank. In *24th international conference on computational linguistics* (p. 21). Mumbai, India.

Bloomfield, L. (1933). *Language*. Unwin University Books.

Box, G. E. P., & Draper, N. R. (1987). Empirical model-building and response surfaces. *Wiley series in probability and mathematical statistics.*, *669*.

Brennan, J., & Hale, J. T. (2019). Hierarchical structure guides rapid linguistic

predictions during naturalistic listening. *PLoS One*, *14*(1), e0207741. doi: 10.1371/journal.pone.0207741

Brennan, J., Nir, Y., Hasson, U., Malach, R., Heeger, D. J., & Pylkkänen, L. (2012). Syntactic structure building in the anterior temporal lobe during natural story listening. *Brain Lang.*, *120*(2), 163–173. doi: 10.1016/j.bandl.2010.04.002

Brett, K. D., & Michael, W. (2017). *Max wertheimer and gestalt theory*. Routledge.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... Amodei, D. (2020). Language models are Few-Shot learners. *arXiv preprint arXiv:2005.14165*.

Brunet, D., Murray, M. M., & Michel, C. M. (2011). Spatiotemporal analysis of multichannel EEG: CARTOOL. *Comput. Intell. Neurosci.*, *2011*, 813870. doi: 10.1155/2011/813870

Brysbaert, M., & Vitu, F. (1998). Word skipping: Implications for theories of eye movement control in reading. In G. Underwood (Ed.), *Eye guidance in reading and scene perception* (pp. 125–147). Amsterdam: Elsevier Science Ltd. doi: 10.1016/B978-008043361-5/50007-9

Buswell, G. T. (1935). *How people look at pictures: A study of the psychology and perception in art* (Vol. 198). Oxford, England: University of Chicago Press.

Bybee, J. (1995). Regular morphology and the lexicon. *Lang. Cogn. Process.*, *10*(5), 425–455. doi: 10.1080/01690969508407111

Cai, Q., & Brysbaert, M. (2010). SUBTLEX-CH: Chinese word and character frequencies based on film subtitles. *PLoS One*, *5*(6). doi: 10.1371/journal.pone.0010729

Chen, L. (1982). Topological structure in visual perception. *Science*, *218*(4573), 699–700. doi: 10.1126/science.7134969

Chollet, F., & Others. (2015). *Keras.* Retrieved from `https://keras.io`

Chomsky, N. (1953). Systems of syntactic analysis. *J. Symbolic Logic*, *18*(3), 242–256. doi: 10.2307/2267409

Chomsky, N. (1957). *Syntactic structures*. Mouton.

Chomsky, N. (1995). *The minimalist program*. MIT Press.

Chomsky, N. (2006). *Language and mind* (third ed.). Cambridge University Press.

Chomsky, N. (2016). Minimal computation and the architecture of language. *Chin. Semiot. Stud.*, *12*(1), 13–24. doi: 10.1515/css-2016-0003

Chomsky, N., & Halle, M. (1968). *The sound pattern of English*. The MIT Press.

Christie, A. (1920/2008). *The mysterious affair at styles*. Urbana, Illinois: Project

Gutenberg.

Chung, J., Ahn, S., & Bengio, Y. (2017). Hierarchical multiscale recurrent neural networks. In *5th international conference on learning representations, ICLR 2017*. Toulon, France.

Coltheart, M., Rastle, K., Perry, C., Langdon, R., & Ziegler, J. (2001). DRC: A dual route cascaded model of visual word recognition and reading aloud. *Psychol. Rev.*, *108*(1), 204–256. doi: 10.1037/0033-295X.108.1.204

Cop, U., Dirix, N., Drieghe, D., & Duyck, W. (2017). Presenting GECO: An eyetracking corpus of monolingual and bilingual sentence reading. *Behav. Res. Methods*, *49*(2), 602–615. doi: 10.3758/s13428-016-0734-0

Cowan, N. (2001). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behav. Brain Sci.*, *24*(1), 87–114; discussion 114–85. doi: 10.1017/s0140525x01003922

Cox, D. R. (1958). The regression analysis of binary sequences. *J. R. Stat. Soc. Series B Stat. Methodol.*, *20*(2), 215–242.

Croft, W. (2001). *Radical construction grammar: Syntactic theory in typological perspective*. Oxford University Press. doi: 10.1093/acprof:oso/9780198299554.001.0001

Croft, W., & Alan Cruse, D. (2004). *Cognitive linguistics*. Cambridge University Press. doi: 10.1017/CBO9780511803864

Daelemans, W., & van den Bosch, A. (2005). *Memory-Based language processing*. Cambridge University Press. doi: 10.1017/CBO9780511486579

Davies, M. (2008). *Corpus of Contemporary American English (COCA)*. Retrieved from `https://www.english-corpora.org/coca/` (Accessed: 2021-3-19)

Davis, M. H. (2004). Units of representation in visual word recognition. *Proc. Natl. Acad. Sci. U. S. A.*, *101*(41), 14687–14688. doi: 10.1073/pnas.0405788101

Davis, R. L., & Zhong, Y. (2017). The biology of forgetting — a perspective. *Neuron*, *95*(3), 490–503. doi: 10.1016/j.neuron.2017.05.039

De Saussure, F. (1916/1959). Course in general linguistics (W. Baskin, trans.). *New York: Philosophical Library*.

De Groot, A. D. (1946). Het denken van den schaker [thought and choice in chess]. *Amsterdam: Noord Hollandsche*.

Delorme, A., & Makeig, S. (2004). EEGLAB: An open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *J. Neurosci. Methods*, *134*(1), 9–21. doi: 10.1016/j.jneumeth.2003.10.009

Delorme, A., Mullen, T., Kothe, C., Akalin Acar, Z., Bigdely-Shamlo, N., Vankov, A., & Makeig, S. (2011). EEGLAB, SIFT, NFT, BCILAB, and ERICA: New tools for advanced EEG processing. *Comput. Intell. Neurosci.*, *2011*, 130714. doi: 10.1155/2011/130714

de Marcken, C. (1996). *Unsupervised language acquisition* (Unpublished doctoral dissertation). Massachusetts Institute of Technology, Cambridge, MA, USA.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (pp. 4171–4186). Stroudsburg, PA, USA: Association for Computational Linguistics. doi: 10.18653/v1/N19-1423

Devlin, J. T., Jamison, H. L., Matthews, P. M., & Gonnerman, L. M. (2004). Morphology and the internal structure of words. *Proc. Natl. Acad. Sci. U. S. A.*, *101*(41), 14984–14988. doi: 10.1073/pnas.0403766101

Dewey, R. A. (2018). *Gestalt Psychology | in Chapter 04: Senses.* Retrieved from `http://www.psywww.com/intropsych/ch04-senses/gestalt-psychology.html` (Accessed: 2020-1-19)

Di Liberto, G. M., Pelofi, C., Bianco, R., Patel, P., Mehta, A. D., Herrero, J. L., ... Mesgarani, N. (2020). Cortical encoding of melodic expectations in human temporal cortex. *Elife*, *9*. doi: 10.7554/eLife.51784

Ding, N., Melloni, L., Yang, A., Wang, Y., Zhang, W., & Poeppel, D. (2017). Characterizing neural entrainment to hierarchical linguistic units using electroencephalography (EEG). *Front. Hum. Neurosci.*, *11*, 481. doi: 10.3389/fnhum.2017.00481

Ding, N., Melloni, L., Zhang, H., Tian, X., & Poeppel, D. (2016). Cortical tracking of hierarchical linguistic structures in connected speech. *Nat. Neurosci.*, *19*(1), 158–164. doi: 10.1038/nn.4186

Dufau, S., Grainger, J., Midgley, K. J., & Holcomb, P. J. (2015). A thousand words are worth a picture: Snapshots of Printed-Word processing in an Event-Related potential megastudy. *Psychol. Sci.*, *26*(12), 1887–1897. doi: 10.1177/0956797615603934

Ehrlich, S. F., & Rayner, K. (1981). Contextual effects on word perception and eye movements during reading. *Journal of Verbal Learning and Verbal Behavior*, *20*(6), 641–655. doi: 10.1016/S0022-5371(81)90220-6

Ellis, N. C. (2002). FREQUENCY EFFECTS IN LANGUAGE PROCESSING: A

review with implications for theories of implicit and explicit language acquisition. *Studies in Second Language Acquisition*, *24*(2), 143–188. doi: 10.1017/S0272263102002024

Ellis, N. C. (2003). Constructions, chunking, and connectionism: The emergence of second language structure. In *The handbook of second language acquisition* (pp. 63–103). Oxford, UK: Blackwell Publishing Ltd. doi: 10.1002/9780470756492.ch4

Emerson, T. (2005). The second international Chinese word segmentation bakeoff. In *Proceedings of the fourth SIGHAN workshop on Chinese language processing*.

Engbert, R., Nuthmann, A., Richter, E. M., & Kliegl, R. (2005). SWIFT: A dynamical model of saccade generation during reading. *Psychol. Rev.*, *112*(4), 777–813. doi: 10.1037/0033-295X.112.4.777

Faísca, L., Reis, A., & Araújo, S. (2019). Early brain sensitivity to word frequency and lexicality during reading aloud and implicit reading. *Front. Psychol.*, *10*, 830. doi: 10.3389/fpsyg.2019.00830

Fiorentino, R., Naito-Billen, Y., Bost, J., & Fund-Reznicek, E. (2014). Electrophysiological evidence for the morpheme-based combinatoric processing of English compounds. *Cogn. Neuropsychol.*, *31*(1-2), 123–146. doi: 10.1080/02643294.2013.855633

Fiorentino, R., & Poeppel, D. (2007). Compound words and structure in the lexicon. *Lang. Cogn. Process.*, *22*(7), 953–1000. doi: 10.1080/01690960701190215

Frank, S. L., Otten, L. J., Galli, G., & Vigliocco, G. (2015). The ERP response to the amount of information conveyed by words in sentences. *Brain Lang.*, *140*, 1–11. doi: 10.1016/j.bandl.2014.10.006

Frank, S. L., & Willems, R. M. (2017). Word predictability and semantic similarity show distinct patterns of brain activity during language comprehension. *Language, Cognition and Neuroscience*, *32*(9), 1192–1203. doi: 10.1080/23273798.2017.1323109

Frazier, L. (1987). Sentence processing: A tutorial review. *Attention and performance 12: The psychology of reading.*, *12*(707), 559–586.

Frazier, L., & Rayner, K. (1982). Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cogn. Psychol.*, *14*(2), 178–210. doi: 10.1016/0010-0285(82)90008-1

Fries, P., Reynolds, J. H., Rorie, A. E., & Desimone, R. (2001). Modulation of

oscillatory neuronal synchronization by selective visual attention. *Science*, *291*(5508), 1560–1563. doi: 10.1126/science.1055465

Futrell, R., Mahowald, K., & Gibson, E. (2015). Large-scale evidence of dependency length minimization in 37 languages. *Proc. Natl. Acad. Sci. U. S. A.*, *112*(33), 10336–10341. doi: 10.1073/pnas.1502134112

Giraudo, H., & Dal Maso, S. (2016). The salience of complex words and their parts: Which comes first? *Front. Psychol.*, *7*, 1778. doi: 10.3389/fpsyg.2016.01778

Gobet, F., Lloyd-Kelly, M., & Lane, P. C. R. (2016). What's in a name? The multiple meanings of "chunk" and "chunking". *Front. Psychol.*, *7*, 102. doi: 10.3389/fpsyg.2016.00102

Gold, B. T., & Rastle, K. (2007). Neural correlates of morphological decomposition during visual word recognition. *J. Cogn. Neurosci.*, *19*(12), 1983–1993. doi: 10.1162/jocn.2007.19.12.1983

Goldberg, A. E. (1995). *Constructions: A construction grammar approach to argument structure*. University of Chicago Press.

Goldberg, A. E. (2006). *Constructions at work: The nature of generalization in language*. Oxford University Press.

Goldberg, A. E. (2013). Constructionist approaches. In T. Hoffmann & G. Trousdale (Eds.), *The oxford handbook of construction grammar*. Oxford University Press. doi: 10.1093/oxfordhb/9780195396683.013.0002

Goldwater, S., Griffiths, T. L., & Johnson, M. (2009). A bayesian framework for word segmentation: Exploring the effects of context. *Cognition*, *112*(1), 21–54. doi: 10.1016/j.cognition.2009.03.008

Gramfort, A., Luessi, M., Larson, E., Engemann, D. A., Strohmeier, D., Brodbeck, C., ... Hämäläinen, M. (2013). MEG and EEG data analysis with MNE-Python. *Front. Neurosci.*, *7*(7 DEC), 267. doi: 10.3389/fnins.2013.00267

Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Gravitz, L. (2019). The importance of forgetting. *Nature*, *571*, S12–S14.

Grech, R., Cassar, T., Muscat, J., Camilleri, K. P., Fabri, S. G., Zervakis, M., ... Vanrumste, B. (2008). Review on solving the inverse problem in EEG source analysis. *J. Neuroeng. Rehabil.*, *5*, 25. doi: 10.1186/1743-0003-5-25

Groppe, D. M., Urbach, T. P., & Kutas, M. (2011). Mass univariate analysis of event-related brain potentials/fields i: A critical tutorial review. *Psychophysiology*, *48*(12), 1711–1725. doi: 10.1111/j.1469-8986.2011

.01273.x

Hajibayova, L. (2013). Basic-level categories: A review. *J. Inf. Sci. Eng.*, *39*(5), 676–687. doi: 10.1177/0165551513481443

Han, S., Fan, S., Chen, L., & Zhuo, Y. (1999). Modulation of brain activities by hierarchical processing: a high-density ERP study. *Brain Topography*, *11*(3), 171–183. doi: 10.1023/a:1022244727182

Han, S., Yund, E. W., & Woods, D. L. (2003). An ERP study of the global precedence effect: The role of spatial frequency. *Clin. Neurophysiol.*, *114*(10), 1850–1865. doi: 10.1016/s1388-2457(03)00196-2

Hauk, O., Patterson, K., Woollams, A., Watling, L., Pulvermüller, F., & Rogers, T. T. (2006). [q:] when would you prefer a SOSSAGE to a SAUSAGE? [a:] at about 100 msec. ERP correlates of orthographic typicality and lexicality in written word recognition. *J. Cogn. Neurosci.*, *18*(5), 818–832. doi: 10.1162/jocn.2006.18.5.818

Heider, G. M. (1977). More about hull and koffka. , *32*(5), 383a. doi: 10.1037/0003-066X.32.5.383.a

Held, L., & Ott, M. (2018). On p-values and bayes factors. *Annu. Rev. Stat. Appl.*, *5*(1), 393–419. doi: 10.1146/annurev-statistics-031017-100307

Henderson, J. M. (2011). Eye movements and scene perception. In S. P. Liversedge, I. Gilchrist, & S. Everling (Eds.), *The oxford handbook of eye movements*. Oxford University Press. doi: 10.1093/oxfordhb/9780199539789.013.0033

Hinton, G. E. (1990). Mapping part-whole hierarchies into connectionist networks. *Artif. Intell.*, *46*(1), 47–75. doi: 10.1016/0004-3702(90)90004-j

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Houde, J. F., Nagarajan, S. S., Sekihara, K., & Merzenich, M. M. (2002). Modulation of the auditory cortex during speech: An MEG study. *J. Cogn. Neurosci.*, *14*(8), 1125–1138. doi: 10.1162/089892902760807140

Hyönä, J., & Olson, R. K. (1995). Eye fixation patterns among dyslexic and normal readers: Effects of word length and word frequency. *J. Exp. Psychol. Learn. Mem. Cogn.*, *21*(6), 1430–1440. doi: 10.1037/0278-7393.21.6.1430

Jackendoff, R. (2002). What's in the lexicon? In S. Nooteboom, F. Weerman, & F. Wijnen (Eds.), *Storage and computation in the language faculty* (pp. 23–58). Dordrecht: Springer Netherlands. doi: 10.1007/978-94-010-0355-1

_2

Johnson, M. (2008). Using Adaptor Grammars to identify synergies in the unsupervised acquisition of linguistic structure. In *Proceedings of ACL-08: HLT* (pp. 398–406). Columbus, Ohio.

Johnson, M., & Goldwater, S. (2009). Improving nonparameteric bayesian inference: Experiments on unsupervised word segmentation with adaptor grammars. In *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics* (pp. 317–325). Boulder, Colorado: Association for Computational Linguistics.

Johnson, M., Griffiths, T. L., & Goldwater, S. (2007). Adaptor grammars: A framework for specifying compositional nonparametric bayesian models. In B. Scholkopf, J. Platt, & T. Hoffman (Eds.), *Advances in Neural Information Processing Systems 19: Proceedings of the 2006 Conference* (Vol. 19). MIT Press.

Kaczer, L., Timmer, K., Bavassi, L., & Schiller, N. O. (2015). Distinct morphological processing of recently learned compound words: An ERP study. *Brain Res.*, *1629*, 309–317. doi: 10.1016/j.brainres.2015.10.029

Kawakami, K., Dyer, C., & Blunsom, P. (2019). Learning to discover, ground and use words with segmental neural language models. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 6429–6441). Stroudsburg, PA, USA: Association for Computational Linguistics. doi: 10.18653/v1/P19-1645

Kerr, P. W. (1992). *Eye movement control during reading: The selection of where to send the eyes* (Unpublished doctoral dissertation). University of Illinois at Urbana-Champaign.

Kimchi, R. (1992). Primacy of wholistic processing and global/local paradigm: a critical review. *Psychol. Bull.*, *112*(1), 24–38. doi: 10.1037/0033-2909.112.1.24

Kimura, A., Ghahramani, Z., Takeuchi, K., Iwata, T., & Ueda, N. (2018). Imitation networks: Few-shot learning of neural networks from scratch. *arXiv preprint arXiv:1802.03039*.

King, J. R., Faugeras, F., Gramfort, A., Schurger, A., El Karoui, I., Sitt, J. D., ... Dehaene, S. (2013). Single-trial decoding of auditory novelty responses facilitates the detection of residual consciousness. *Neuroimage*, *83*, 726–738. doi: 10.1016/j.neuroimage.2013.07.013

Kliegl, R., Grabner, E., Rolfs, M., & Engbert, R. (2004). Length, frequency, and

predictability effects of words on eye movements in reading. *Eur. J. Cogn. Psychol.*, *16*(1-2), 262–284. doi: 10.1080/09541440340000213

Knill, D. C., & Pouget, A. (2004). The bayesian brain: The role of uncertainty in neural coding and computation. *Trends Neurosci.*, *27*(12), 712–719. doi: 10.1016/j.tins.2004.10.007

Koch, K., McLean, J., Segev, R., Freed, M. A., Berry, M. J., 2nd, Balasubramanian, V., & Sterling, P. (2006). How much the eye tells the brain. *Curr. Biol.*, *16*(14), 1428–1434. doi: 10.1016/j.cub.2006.05.056

Koester, D., Gunter, T. C., & Wagner, S. (2007). The morphosyntactic decomposition and semantic composition of German compound words investigated by ERPs. *Brain Lang.*, *102*(1), 64–79. doi: 10.1016/j.bandl.2006.09.003

Koester, D., & Schiller, N. O. (2011). The functional neuroanatomy of morphology in language production. *Neuroimage*, *55*(2), 732–741. doi: 10.1016/j.neuroimage.2010.11.044

Krishnamurthy, R. (2003). Language as chunks, not words. In M. Swanson & K. Hill (Eds.), *JALT2002: Conference proceedings: waves of the future* (pp. 288–294). Tokyo, Japan: The Japan Association for Language Teaching.

Krogh, A., & Hertz, J. A. (1992). A simple weight decay can improve generalization. In J. E. Moody, S. J. Hanson, & R. P. Lippmann (Eds.), *Advances in neural information processing systems 4* (pp. 950–957). Morgan-Kaufmann.

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 conference on empirical methods in natural language processing: System demonstrations* (pp. 66–71). Brussels, Belgium: Association for Computational Linguistics. doi: 10.18653/v1/D18-2012

Kutas, M., & Hillyard, S. A. (1980). Reading senseless sentences: brain potentials reflect semantic incongruity. *Science*, *207*(4427), 203–205. doi: 10.1126/science.7350657

Lakoff, G. (1977). Linguistic gestalts. In W. Beach, S. Fox, & S. Philosoph (Eds.), *Papers from the thirteenth regional meeting, chicago linguistic society, april 14-16* (pp. 236–287). University of Chicago, Chicago, Illinois.

Lakoff, G. (1987). *Women, fire, and dangerous things: What categories reveal about the mind*. University of Chicago Press.

Langacker, R. (2017). *Ten lectures on the basics of cognitive grammar*. BRILL. doi: 10.1163/9789004347458

Langacker, R. W. (1987). *Foundations of cognitive grammar: Theoretical prerequisites*. Stanford University Press.

Lange, V. M., Perret, C., & Laganaro, M. (2015). Comparison of single-word and adjective-noun phrase production using event-related brain potentials. *Cortex*, *67*, 15–29. doi: 10.1016/j.cortex.2015.02.017

Lehmann, D., & Skrandies, W. (1980). Reference-free identification of components of checkerboard-evoked multichannel potential fields. *Electroencephalogr. Clin. Neurophysiol.*, *48*(6), 609–621.

Leminen, A., Smolka, E., Duñabeitia, J. A., & Pliatsikas, C. (2019). Morphological processing in the brain: The good (inflection), the bad (derivation) and the ugly (compounding). *Cortex*, *116*, 4–44. doi: 10.1016/j.cortex.2018.08.016

Li, S., Zhu, H., & Tian, X. (2020). *Distinct neural signals in speech preparation differentially modulate auditory responses.* doi: 10.1101/2020.01.14.905620

Li, X., Liu, P., & Rayner, K. (2011). Eye movement guidance in Chinese reading: Is there a preferred viewing location? *Vision Res.*, *51*(10), 1146–1156. doi: 10.1016/j.visres.2011.03.004

Ling, S., Lee, A. C. H., Armstrong, B. C., & Nestor, A. (2019). How are visual words represented? Insights from EEG-based visual word decoding, feature derivation and image reconstruction. *Hum. Brain Mapp.*, *40*(17), 5056–5068. doi: 10.1002/hbm.24757

Luo, H., & Poeppel, D. (2007). Phase patterns of neuronal responses reliably discriminate speech in human auditory cortex. *Neuron*, *54*(6), 1001–1010. doi: 10.1016/j.neuron.2007.06.004

Luo, R., Xu, J., Zhang, Y., Ren, X., & Sun, X. (2019). PKUSEG: A toolkit for multi-domain Chinese word segmentation. *arXiv preprint arXiv:1906.11455*.

Ma, O., & Tian, X. (2019). Distinct mechanisms of imagery differentially influence speech perception. *eNeuro*, *6*(5). doi: 10.1523/ENEURO.0261-19.2019

MacGregor, L. J., & Shtyrov, Y. (2013). Multiple routes for compound word processing in the brain: Evidence from EEG. *Brain Lang.*, *126*(2), 217–229. doi: 10.1016/j.bandl.2013.04.002

Mahowald, K., Fedorenko, E., Piantadosi, S. T., & Gibson, E. (2013). Info/information theory: speakers choose shorter words in predictive contexts. *Cognition*, *126*(2), 313–318. doi: 10.1016/j.cognition.2012.09.010

Manly, B. F. J. (2006). *Randomization, bootstrap and monte carlo methods in biology, third edition*. New York: Chapman and Hall/CRC. doi: 10.1201/9781315273075

Manning, C. D., & Schütze, H. (1999). *Foundations of statistical natural language*

processing. Cambridge, MA, USA: MIT Press.

Maris, E., & Oostenveld, R. (2007). Nonparametric statistical testing of EEG- and MEG-data. *J. Neurosci. Methods*, *164*(1), 177–190. doi: 10.1016/j.jneumeth.2007.03.024

McCauley, S. M., & Christiansen, M. H. (2017). Computational investigations of multiword chunks in language learning. *Top. Cogn. Sci.*, *9*(3), 637–652. doi: 10.1111/tops.12258

McCauley, S. M., & Christiansen, M. H. (2019). Language learning as language use: A cross-linguistic model of child language development. *Psychol. Rev.*, *126*(1), 1–51. doi: 10.1037/rev0000126

McClelland, J. L., & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: I. an account of basic findings. *Psychol. Rev.*, *88*(5), 375–407. doi: 10.1037/0033-295x.88.5.375

McConkie, G. W., Kerr, P. W., Reddix, M. D., & Zola, D. (1988). Eye movement control during reading: I. the location of initial eye fixations on words. *Vision Res.*, *28*(10), 1107–1118. doi: 10.1016/0042-6989(88)90137-x

McGurk, H., & MacDonald, J. (1976). Hearing lips and seeing voices. *Nature*, *264*(5588), 746–748. doi: 10.1038/264746a0

Meinzer, M., Lahiri, A., Flaisch, T., Hannemann, R., & Eulitz, C. (2009). Opaque for the reader but transparent for the brain: neural signatures of morphological complexity. *Neuropsychologia*, *47*(8-9), 1964–1971. doi: 10.1016/j.neuropsychologia.2009.03.008

Meunier, F., & Longtin, C.-M. (2007). Morphological decomposition and semantic integration in word processing. *J. Mem. Lang.*, *56*(4), 457–471. doi: 10.1016/j.jml.2006.11.005

Meyer, D. E., & Schvaneveldt, R. W. (1971). Facilitation in recognizing pairs of words: Evidence of a dependence between retrieval operations. *Journal of Experimental Psychology*, *90*(2), 227–234. doi: 10.1037/h0031564

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems* (Vol. 26). Curran Associates, Inc.

Miller, G. A. (1956). The magical number seven plus or minus two: some limits on our capacity for processing information. *Psychol. Rev.*, *63*(2), 81–97.

Monsalve, I. F., Frank, S. L., & Vigliocco, G. (2012). Lexical surprisal as a general predictor of reading time. In *Proceedings of the 13th conference of the euro-*

pean chapter of the association for computational linguistics (pp. 398–408).

Monsell, S. (1991). The nature and locus of word frequency effects in reading. In *Basic processes in reading: Visual word recognition* (pp. 148–197). Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Murray, M. M., Brunet, D., & Michel, C. M. (2008). Topographic ERP analyses: A step-by-step tutorial review. *Brain Topogr.*, *20*(4), 249–264. doi: 10.1007/s10548-008-0054-5

Nagy, G. (2009). *Performance and text in ancient greece*. Oxford University Press. doi: 10.1093/oxfordhb/9780199286140.013.0037

Navon, D. (1977). Forest before trees: The precedence of global features in visual perception. *Cogn. Psychol.*, *9*(3), 353–383. doi: 10.1016/0010-0285(77)90012-3

Nozaradan, S., Peretz, I., Missal, M., & Mouraux, A. (2011). Tagging the neuronal entrainment to beat and meter. *J. Neurosci.*, *31*(28), 10234–10240. doi: 10.1523/JNEUROSCI.0411-11.2011

Nuthmann, A., Engbert, R., & Kliegl, R. (2005). Mislocated fixations during reading and the inverted optimal viewing position effect. *Vision Res.*, *45*(17), 2201–2217. doi: 10.1016/j.visres.2005.02.014

Oehrn, C. R., Fell, J., Baumann, C., Rosburg, T., Ludowig, E., Kessler, H., … Axmacher, N. (2018). Direct electrophysiological evidence for prefrontal control of hippocampal processing during voluntary forgetting. *Curr. Biol.*, *28*(18), 3016–3022.e4. doi: 10.1016/j.cub.2018.07.042

Oostdijk, N., Reynaert, M., Hoste, V., & Schuurman, I. (2013). The construction of a 500-million-word reference corpus of contemporary written Dutch. In P. Spyns & J. Odijk (Eds.), *Essential speech and language technology for Dutch* (pp. 219–247). Springer, Berlin, Heidelberg. doi: 10.1007/978-3-642-30910-6_13

Panichello, M. F., Cheung, O. S., & Bar, M. (2012). Predictive feedback and conscious visual experience. *Front. Psychol.*, *3*, 620. doi: 10.3389/fpsyg.2012.00620

Pascual-Marqui, R. D. (2007). Discrete, 3D distributed, linear imaging methods of electric neuronal activity. Part 1: Exact, zero error localization. *arXiv preprint arXiv:0710.3341*.

Paterson, K. B., Almabruk, A. A. A., McGowan, V. A., White, S. J., & Jordan, T. R. (2015). Effects of word length on eye movement control: The evidence from Arabic. *Psychon. Bull. Rev.*, *22*(5), 1443–1450. doi: 10.3758/s13423-015-0809-4

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, *12*(Oct), 2825–2830.

Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543). Doha, Qatar. doi: 10.3115/v1/D14-1162

Pernet, C. R., Chauveau, N., Gaspar, C., & Rousselet, G. A. (2011). LIMO EEG: A toolbox for hierarchical linear modeling of ElectroEncephaloGraphic data. *Comput. Intell. Neurosci.*, *2011*, 831409. doi: 10.1155/2011/831409

Perruchet, P., & Vinter, A. (1998). PARSER: A model for word segmentation. *J. Mem. Lang.*, *39*(2), 246–263. doi: 10.1006/jmla.1998.2576

Piantadosi, S. T., Tily, H., & Gibson, E. (2011). Word lengths are optimized for efficient communication. *Proc. Natl. Acad. Sci. U. S. A.*, *108*(9), 3526–3529. doi: 10.1073/pnas.1012551108

Pitman, E. J. G. (1937). Significance tests which may be applied to samples from any populations. *Supplement to the Journal of the Royal Statistical Society*, *4*(1), 119–130. doi: 10.2307/2984124

Plaut, D. C., McClelland, J. L., Seidenberg, M. S., & Patterson, K. (1996). Understanding normal and impaired word reading: Computational principles in quasi-regular domains. *Psychol. Rev.*, *103*(1), 56–115. doi: 10.1037/0033-295x.103.1.56

Pliatsikas, C., Wheeldon, L., Lahiri, A., & Hansen, P. C. (2014). Processing of zero-derived words in English: An fMRI investigation. *Neuropsychologia*, *53*, 47–53. doi: 10.1016/j.neuropsychologia.2013.11.003

Poghosyan, V., & Ioannides, A. A. (2008). Attention modulates earliest responses in the primary auditory and visual cortices. *Neuron*, *58*(5), 802–813. doi: 10.1016/j.neuron.2008.04.013

Pollatsek, A., & Rayner, K. (1989). Reading. In M. I. Posner (Ed.), *Foundations of cognitive science* (pp. 401–436). 55 Hayward St., Cambridge, MA, United States: MIT Press. doi: 10.7551/mitpress/3072.003.0003

Prechelt, L. (1998). Early stopping - but when? In G. B. Orr & K.-R. Müller (Eds.), *Neural networks: Tricks of the trade* (pp. 55–69). Berlin, Heidelberg: Springer Berlin Heidelberg. doi: 10.1007/3-540-49430-8_3

Rayner, K. (1979). Eye guidance in reading: fixation locations within words. *Perception*, *8*(1), 21–30. doi: 10.1068/p080021

Rayner, K. (1998). Eye movements in reading and information processing: 20

years of research. *Psychol. Bull.*, *124*(3), 372–422. doi: 10.1037/0033-2909.124.3.372

Rayner, K. (2009). *Eye movements and attention in reading, scene perception, and visual search* (Vol. 62). doi: 10.1080/17470210902816461

Rayner, K., & McConkie, G. W. (1976). What guides a reader's eye movements? *Vision Res.*, *16*(8), 829–837. doi: 10.1016/0042-6989(76)90143-7

Rayner, K., Sereno, S. C., & Raney, G. E. (1996). Eye movement control in reading: A comparison of two types of models. *J. Exp. Psychol. Hum. Percept. Perform.*, *22*(5), 1188–1200. doi: 10.1037/0096-1523.22.5.1188

Rayner, K., Well, A. D., Pollatsek, A., & Bertera, J. H. (1982). The availability of useful information to the right of fixation in reading. *Percept. Psychophys.*, *31*(6), 537–550. doi: 10.3758/bf03204186

Reali, F., & Christiansen, M. H. (2007). Word chunk frequencies affect the processing of pronominal object-relative clauses. *Q. J. Exp. Psychol.*, *60*(2), 161–170. doi: 10.1080/17470210600971469

Reicher, G. M. (1969). Perceptual recognition as a function of meaningfulness of stimulus material. *J. Exp. Psychol.*, *81*(2), 275–280. doi: 10.1037/h0027768

Reichle, E. D., Liversedge, S. P., Drieghe, D., Blythe, H. I., Joseph, H. S. S. L., White, S. J., & Rayner, K. (2013). Using E-Z reader to examine the concurrent development of eye-movement control and reading skill. *Dev. Rev.*, *33*(2), 110–149. doi: 10.1016/j.dr.2013.03.001

Reichle, E. D., Pollatsek, A., Fisher, D. L., & Rayner, K. (1998). Toward a model of eye movement control in reading. *Psychol. Rev.*, *105*(1), 125–157. doi: 10.1037/0033-295X.105.1.125

Reichle, E. D., Rayner, K., & Pollatsek, A. (2003). The E-Z reader model of eye-movement control in reading: Comparisons to other models. *Behav. Brain Sci.*, *26*(4), 445–76. doi: 10.1017/s0140525x03000104

Reichle, E. D., & Sheridan, H. (2015). E-Z reader: An overview of the model and two recent applications. In *The oxford handbook of reading* (pp. 277–290). Oxford University Press. doi: 10.1093/oxfordhb/9780199324576.013.17

Reichle, E. D., Tokowicz, N., Liu, Y., & Perfetti, C. A. (2011). Testing an assumption of the E-Z reader model of eye-movement control during reading: using event-related potentials to examine the familiarity check. *Psychophysiology*, *48*(7), 993–1003. doi: 10.1111/j.1469-8986.2011.01169.x

Reichle, E. D., Warren, T., & McConnell, K. (2009). Using E-Z reader to model

the effects of higher level language processing on eye movements during reading. *Psychon. Bull. Rev.*, *16*(1), 1–21. doi: 10.3758/PBR.16.1.1

Reilly, R. G., & Radach, R. (2006). Some empirical tests of an interactive activation model of eye movement control in reading. *Cogn. Syst. Res.*, *7*(1), 34–55. doi: 10.1016/j.cogsys.2005.07.006

Rhodes, D. (2013). *Conditional random field latin word segmenter* (Tech. Rep.). Tech. rep., University of Stanford.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*(5), 465–471. doi: 10.1016/0005-1098(78)90005-5

Semenza, C., & Luzzatti, C. (2014). Combining words in the brain: The processing of compound words. introduction to the special issue. *Cogn. Neuropsychol.*, *31*(1-2), 1–7. doi: 10.1080/02643294.2014.898922

Sennrich, R., Haddow, B., & Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 1715–1725). Berlin, Germany: Association for Computational Linguistics. doi: 10.18653/v1/P16-1162

Shannon, C. E. (1951). Prediction and entropy of printed English. *Bell System Technical Journal*, *30*(1), 50–64. doi: 10.1002/j.1538-7305.1951.tb01366.x

Sinclair, J. (1991). *Corpus, concordance, collocation*. Oxford University Press.

Siyanova-Chanturia, A., Conklin, K., Caffarra, S., Kaan, E., & van Heuven, W. J. B. (2017). Representation and processing of multi-word expressions in the brain. *Brain Lang.*, *175*, 111–122. doi: 10.1016/j.bandl.2017.10.004

Smith, N. J., & Levy, R. (2013). The effect of word predictability on reading time is logarithmic. *Cognition*, *128*(3), 302–319. doi: 10.1016/j.cognition.2013.02.013

Snell, J., & Grainger, J. (2017). The sentence superiority effect revisited. *Cognition*, *168*, 217–221. doi: 10.1016/j.cognition.2017.07.003

Steels, L., & De Beule, J. (2006). A (very) brief introduction to fluid construction grammar. In *Proceedings of the third workshop on scalable natural language understanding* (pp. 73–80). New York City, New York: Association for Computational Linguistics.

Steinmetz, P. N., Roy, A., Fitzgerald, P. J., Hsiao, S. S., Johnson, K. O., & Niebur, E. (2000). Attention modulates synchronized neuronal firing in primate somatosensory cortex. *Nature*, *404*(6774), 187–190. doi:

10.1038/35004588

Stites, M. C., Federmeier, K. D., & Christianson, K. (2016). Do morphemes matter when reading compound words with transposed letters? Evidence from Eye-Tracking and Event-Related Potentials. *Lang Cogn Neurosci*, *31*(10), 1299–1319. doi: 10.1080/23273798.2016.1212082

Stockall, L., & Marantz, A. (2006). A single route, full decomposition model of morphological complexity: MEG evidence. *Ment. Lex.*, *1*(1), 85–123. doi: 10.1075/ml.1.1.07sto

Sun, M., Chen, X., Zhang, K., Guo, Z., & Liu, Z. (2016). *Thulac: An efficient lexical analyzer for chinese.* Retrieved from `https://github.com/thunlp/THULAC`

Sun, Z., & Deng, Z.-H. (2018). Unsupervised neural word segmentation for Chinese via segmental language modeling. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 4915–4920). Brussels, Belgium: Association for Computational Linguistics. doi: 10.18653/v1/D18-1531

Taft, M. (2013). *Reading and the mental lexicon*. Psychology Press.

Taft, M., & Forster, K. I. (1976). Lexical storage and retrieval of polymorphemic and polysyllabic words. *Journal of Verbal Learning and Verbal Behavior*, *15*(6), 607–620. doi: 10.1016/0022-5371(76)90054-2

Teh, Y. W., Jordan, M. I., Beal, M. J., & Blei, D. M. (2006). Hierarchical dirichlet processes. *J. Am. Stat. Assoc.*, *101*(476), 1566–1581. doi: 10.1198/016214506000000302

Teng, X., Ma, M., Yang, J., Blohm, S., Cai, Q., & Tian, X. (2020). Constrained structure of ancient Chinese poetry facilitates speech content grouping. *Curr. Biol.*, *0*(0). doi: 10.1016/j.cub.2020.01.059

Tian, X., Ding, N., Teng, X., Bai, F., & Poeppel, D. (2018). Imagined speech influences perceived loudness of sound. *Nature Human Behaviour*, *2*(3), 225–234. doi: 10.1038/s41562-018-0305-8

Tian, X., & Huber, D. E. (2008). Measures of spatial similarity and response magnitude in MEG and scalp EEG. *Brain Topogr.*, *20*(3), 131–141. doi: 10.1007/s10548-007-0040-3

Tian, X., & Poeppel, D. (2010). Mental imagery of speech and movement implicates the dynamics of internal forward models. *Front. Psychol.*, *1*(OCT), 166. doi: 10.3389/fpsyg.2010.00166

Tian, X., & Poeppel, D. (2013). The effect of imagination on stimulation: The functional specificity of efference copies in speech processing. *J. Cogn.*

Neurosci., *25*(7), 1020–1036. doi: 10.1162/jocn_a_00381

Tian, X., & Poeppel, D. (2015). Dynamics of self-monitoring and error detection in speech production: Evidence from mental imagery and MEG. *J. Cogn. Neurosci.*, *27*(2), 352–364. doi: 10.1162/jocn_a_00692

Tian, X., Poeppel, D., & Huber, D. E. (2011). TopoToolbox: Using sensor topography to calculate psychologically meaningful measures from event-related EEG/MEG. *Comput. Intell. Neurosci.*, *2011*, 674605. doi: 10.1155/2011/674605

Tian, X., Zarate, J. M., & Poeppel, D. (2016). Mental imagery of speech implicates two mechanisms of perceptual reactivation. *Cortex*, *77*, 1–12. doi: 10.1016/j.cortex.2016.01.002

Tsai, L. S. (1932). *The laws of minimum effort and maximum satisfaction in animal behavior*. National Research Institute of Psychology.

Underwood, G., Schmitt, N., & Galpin, A. (2004). The eyes have it: An eye-movement study into the processing of formulaic sequences. In N. Schmitt (Ed.), *Formulaic sequences: Acquisition, processing and use* (Vol. 9, pp. 153–172). Amsterdam: John Benjamins Publishing Company. doi: 10.1075/lllt.9.09und

van den Bosch, A., & Daelemans, W. (2013). Implicit schemata and categories in memory-based language processing. *Lang. Speech*, *56*(Pt 3), 309–328. doi: 10.1177/0023830913484902

Vannest, J., Polk, T. A., & Lewis, R. L. (2005). Dual-route processing of complex words: new fMRI evidence from derivational suffixation. *Cogn. Affect. Behav. Neurosci.*, *5*(1), 67–76. doi: 10.3758/CABN.5.1.67

Van Rijsbergen, C. J. (1979). *Information retrieval*. London: Butterworth and Co.

Van Veen, B. D., & Buckley, K. M. (1988). Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Magazine*, *5*(2), 4–24. doi: 10.1109/53.665

Van Wassenhove, V., Grant, K. W., & Poeppel, D. (2005). Visual speech speeds up the neural processing of auditory speech. *Proceedings of the National Academy of Sciences*, *102*(4), 1181–1186.

Vitu, F., & McConkie, G. W. (2000). Regressive saccades and word perception in adult reading. In A. Kennedy, R. Radach, D. Heller, & J. Pynte (Eds.), *Reading as a perceptual process* (pp. 301–326). North-Holland/Elsevier Science Publishers. doi: 10.1016/B978-008043642-5/50015-2

Wakeman, D. G., & Henson, R. N. (2015). A multi-subject, multi-modal human neuroimaging dataset. *Sci Data*, *2*, 150001. doi: 10.1038/sdata.2015.1

Wang, B., Zhou, T. G., Zhuo, Y., & Chen, L. (2007). Global topological dominance in the left hemisphere. *Proc. Natl. Acad. Sci. U. S. A.*, *104*(52), 21014–21019. doi: 10.1073/pnas.0709664104

Wang, X., Zhu, H., & Tian, X. (2019). *Revealing the temporal dynamics in non-invasive electrophysiological recordings with topography-based analyses.* doi: 10.1101/779546

Warren, T., & McConnell, K. (2007). Investigating effects of selectional restriction violations and plausibility violation severity on eye-movements in reading. *Psychon. Bull. Rev.*, *14*(4), 770–775. doi: 10.3758/bf03196835

Xue, N., Zhang, X., Jiang, Z., Palmer, M., Xia, F., Chiou, F.-D., & Chang, M. (2013). *Chinese treebank 8.0.* Retrieved from `https://catalog.ldc.upenn.edu/LDC2013T21` doi: 10.35111/wygn-4f57

Yang, J., Cai, Q., & Tian, X. (2020). How do we segment text? two-stage chunking operation in reading. *eNeuro*, *7*(3). doi: 10.1523/ENEURO.0425-19.2020

Yang, J., Frank, S. L., & van den Bosch, A. (2020). Less is better: A cognitively inspired unsupervised model for language segmentation. In *Proceedings of the workshop on the cognitive aspects of the lexicon* (pp. 33–45). Online: Association for Computational Linguistics.

Yang, J., Zhu, H., & Tian, X. (2018). Group-Level multivariate analysis in EasyEEG toolbox: Examining the temporal dynamics using topographic responses. *Front. Neurosci.*, *12*, 468. doi: 10.3389/fnins.2018.00468

Zhai, K., Boyd-Graber, J., & Cohen, S. B. (2014). Online adaptor grammars with hybrid inference. *Transactions of the Association for Computational Linguistics*, *2*, 465–476. doi: 10.1162/tacl_a_00196

Zheng, Z., Li, J., & Xiao, F. (2015). Familiarity contributes to associative memory: The role of unitization. *Advances in Psychological Science*, *23*(2), 202. doi: 10.3724/SP.J.1042.2015.00202

Zhikov, V., Takamura, H., & Okumura, M. (2013). An efficient algorithm for unsupervised word segmentation with branching entropy and MDL. *Information and Media Technologies*, *8*(2), 514–527. doi: 10.11185/imt.8.514

Zipf, G. K. (1949). *Human behavior and the principle of least effort* (Vol. 573). Oxford, England: Addison-Wesley Press.

Appendix

Materials for Chapter 2

Code Snippets

1. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#1-pre-processing-script-for-mne-python
2. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#2-load-multiple-fif-epoch-files-and-save-as-one-h5-file
3. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#3-an-example-for-using-an-external-classifier-convolutional-neural-network

Output Results

1. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#1
2. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#2
3. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#3
4. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#4

5. https://github.com/ray306/EasyEEG/wiki/Appendix-for-paper-%22Group-Level-Multivariate-Analysis-in-EasyEEG-Toolbox:-Examining-the-Temporal-Dynamics-using-Topographic-Responses%22#5

Materials for Chapter 5

Training parameter settings

Since BR-phono is a child-directed speech corpus, its chunk types are usually very common, and so they often have much higher document ratios than CTB8 chunks. We use a lower τ_0, which is related to document ratio, to balance the corpus difference. The number of training epochs for CTB8, which is large-scale, was set to a higher number than for BR-phono. The epochs numbers are well beyond the convergence points. α and Δ mainly affect the training speed, while ω and τ_0 mainly affect $|L|$. The current parameter settings may not be optimal for end tasks such as word segmentation; in preliminary experiments we optimized for speed[1].

Corpus	α	Δ	ω	τ_0	**epochs**
BR-phono	0.25	0.2	0.0001	10	5,000
CTB8				500	50,000

Table .1.: **The parameter settings in the training on two corpora.** α is the sampling probability, Δ the re-ranking rate, ω the forgetting ratio, τ_0 the probation period.

[1]The training of BR-phone costs 57 s and the training of CTB8 costs 31 min 55 s. The code is written in pure Python 3.7 and ran on a single core of Intel Core i5-7300HQ.

Segmentations with increasing number of training epochs

The progression in chunking over training epochs before convergence (Table .2) shows LiB can learn some word chunks even in the very early epochs. Also, Table .2 illustrates that convergence is reached well before the preset number of runs.

Corpus	Epoch	Segmentation
BRphono	0	Olr9tW9dontwipUthIm6wenQ
	1	O·l·r·9·t·W·9·don·t·w·i·pUt·h·I·m·6·w·e·nQ
	2	Ol·r·9t·W·9·dont·wi·pUt·h·I·m·6·we·nQ
	10	Olr9t·W9·dont·wi·pUt·hIm·6we·nQ
	100	Olr9t·W·9dont·wi·pUthIm6we·nQ
	1,000	Olr9t·W·9dont·wi·pUthIm6we·nQ
CTB8	0	这个出口信贷项目委托中国银行为代理银行
	1	这·个·出·口·信·贷·项·目·委·托·中·国·银·行·为·代·理·银·行
	2	这·个·出·口·信·贷·项·目·委·托·中国·银·行·为·代·理·银·行
	10	这·个·出口·信·贷·项·目·委·托·中国·银·行·为·代·理·银·行
	100	这个·出口·信·贷·项目·委·托·中国·银·行·为·代·理·银·行
	1,000	这个·出口·信贷·项目·委·托·中国·银行·为·代·理·银行
	10,000	这个·出口信贷·项目·委托·中国银行·为·代理·银行

Table .2.: **Example segmentations of strings in the two corpora with increasing number of training epochs.** See Table 5.3 for the correct word-level segementation.

Top, middle and tail entries in lexicon

Corpus	Entries in Lexicon
BRphono (Top 50)	D6 **the**, y& **yeah**, yu **you**, WAt **what**, wan6 **wanna**, k&nyu **can you**, tu **two**, &nd **and**, D&ts **that's**, oke **okay**, f% **four**, nQ **now**, It **it**, D* **they're**, hiz **he's**, In **in**, lUk **look**, wIT **with**, yuwant **you want**, hu **who**, hi **he**, D&t **that**, Ol **all**, y) **your**, h(**here**, 9TINk **i think**, pUt **put**, D&ts6 **that's a**, WAts **what's**, yuk&n **you can**, hIz **his**, m9 **my**, si **see**, yuwan6 **you wanna**, no **no**, IzD&t **is that**, h9 **high**, huz **whose**, DIs **this**, gUd **good**, D*z **there's**, v*i **very**, siD6 **see the**, Its6 **its a**, IzIt **is it**, Olr9t **alright**, DIsIz **this is**, #yu **are you**, IN **ing**, h&v **have**
BRphono (Middle 20)	siD&t **see that**, nik, lEtmiQt **let me out**, DIsgoz **this goes**, d&diznat **daddy's not**, 9ms%i **i'm sorry**, kIN, lUksl9k6n9s, wITDiz **with these**, hizwe **he's way**, lON **long**, h&p **happen**, lEtssiIf **let's see if**, lEtspUthIm6we **let's put him away**, diIzf%, pR, brEkf6st **breakfast**, h9c* **high chair**, lUk&tD6bUk **look at the book**, W*zD6kIti
BRphono (Tail 20)	Nkyu, T, uyuwant, * **air**, 3, (**ear**, Z, c,), M, InhIzhQs, 6mily6 **amelia**, dOghQs **doghouse**, wITt7z **with toys**, &ndsAmt9mzwi, holdh&ndz **hold hands**, tIkLmi **tickle me**, h9ke **high kay**, tekItQt, k&nyubrAShIzh*
CTB8 (Top 50)	没有 **haven't**, 中国 **China**, 我们 **we**, 经济 **economics**, 已经 **already**, 孩子 **kid**, 但是 **but**, 教育 **education**, 可以 **can**, 目前 **now**, 政府 **government**, 国家 **country**, 一个 **a**, 这些 **these**, 自己 **self**, 不能 **can't**, 如果 **if**, 记者 **journalist**, 今天 **today**, 他们 **they**, 虽然 **although**, 要求 **require**, 技术 **tech**, 进行 **process**, 这个 **this**, 新华社 **Xinhua News Agency**, 希望 **wish**, 问题 **issue**, 就是 **is**, 大陆 **mainland**, 因为 **because**, 一些 **some**, 以及 **and**, 都是 **all are**, 因此 **so**, 现在 **now**, 可能 **may**, 台湾 **Taiwan**, 应该 **should**, 政治 **political**, 发展 **development**, 也是 **also is**, 还是 **also is**, 社会 **society**, 这样 **such**, 通过 **via**, 继续 **continue**, 不是 **isn't**, 上海 **Shanghai**, 的 **'s**
CTB8 (Middle 20)	肝脏 **liver**, 军事政变推翻 **military coup overthrows**, 在其他地方 **in other places**, 在野势力 **opposition force**, 而且这个 **and this**, 泄的, 帮他 **help him**, 宝应县 **Baoying County**, 政治新闻 **political news**, 经济越 **economic more**, 塔肯, 迅速地 **rapidly**, 铅笔 **pencil**, 集体经济 **collective economy**, 起源 **origin**, 邓相扬协助 **Tang Xiangyang assisted**, 建制 **establishment**, 写完 **after writing**, 说的那样 **as said**, 后顾 **look back**
CTB8 (Tail 20)	存在主权 **there is sovereignty**, 确权 **confirm rights**, 草案还 **the draft also**, 桌会议, 第一首相 **the first prime minister**, 迪奥 **dior**, 长大了 **grown up**, 爱他 **love him**, 说他 **say him**, 子虚乌, 有没有参与 **did you participate**, 严谨的 **rigorous**, 仍然是 **is still**, 站上车, 运输署 **Transport Department**, 杀机 **murderous**, 决 **decided**, 建成通车 **completed and opened to traffic**, 主要嫌疑人赖昌星 **the main suspect Lai Changxing**, 已经向加拿大 **has to Canada**

Table .3.: **The top 50 entries, the middle 20 entries and the tail 20 entries in the lexicons.** The original results of BRphono are in phonemic characters; we transcribed the entries containing complete words into English words (in bold font) for ease of presentation. The original results of CTB8 are the Chinese characters; we added the English translations (in bold font) with the entries containing complete words.

Research Data Management

Chapter 2:
The dataset was downloaded from the Internet; its license allowed unrestricted use for research. The URL of the dataset and the analysis code were described in the main text and appendix.
Chapter 3:
This chapter involves the analyses of behavioral (reaction time) and EEG data. The data were collected by the author. The participants were anonymized by their experimental ID in order to protect the participants' privacy. The reaction time and EEG data of any participant cannot be used to identify the participant. Participants provided their ages and genders, but the chapter reports only the summary of these information.
In the context of scientific integrity, I followed the policy of my institute and archived the research data associated with my publication (including raw data and analysis scripts) in a folder in my Radboud University work group folder (i.e., in my "werkgroepmap") for a minimum of 10 years.
The data sharing is possible since the data can be regarded as completely anonimised. With the purpose of data reuse, I made my data public by publishing it on Open Science Framework (`https://osf.io/ecqrd/`). This includes raw data and analysis scripts, and is under the Open Data Commons Attribution License.
Chapter 4:
This chapter involves the analyses of behavioral (reaction time) and EEG data. The data were collected by the author. The participants were anonymized by their experimental ID in order to protect the participants' privacy. The reaction time and EEG data of any participant cannot be used to identify the participant. Participants provided their ages and genders, but the chapter reports only the summary of these information.
In the context of scientific integrity, I followed the policy of my institute and archived the research data associated with my publication (including raw data and analysis scripts) in a folder in my Radboud University work group folder (i.e., in my "werkgroepmap") for a minimum of 10 years.
The data sharing is possible since the data can be regarded as completely anonimised and participants agreed to this via informed consent. With the purpose of data reuse, I made my data public by publishing it on Open Science Framework (`https://osf.io/bvpz2/`). This includes raw data and analysis scripts, and is under the Open Data Commons Attribution License.
Chapter 5:
The dataset was downloaded from the Internet; its license allowed unrestricted use for research. The URL of the dataset were described in the main text. The analysis script can be found at `https://github.com/ray306/LiB`.
Chapter 6:

The dataset was downloaded from the Internet; its license allowed unrestricted use for research. The URL of the dataset were described in the main text. The analysis script can be found at `https://github.com/ray306/LiB-predicts-eye-fixations`.

Nederlandse samenvatting

Taal wordt vaak gezien als een opeenvolging van discrete eenheden, zodat de meeste analyses van taalmateriaal en studies van taalcognitie beginnen met het bepalen van de eenheden van taal. De meeste mensen bepalen intuïtief taalkundige eenheden (b.v. karakters, lettergrepen, woorden en zinnen) in termen van discrete grenzen. De eerste taalwetenschappers die de structuur van taal als een formeel systeem zagen, introduceerden nieuwe eenheden (b.v. morfemen, zinnen en bijzinnen) om de formele structuur van taal in meer detail weer te geven. Met de vooruitgang van de cognitiewetenschap, zijn geleerden onlangs begonnen met het benadrukken van cognitieve factoren en het feitelijke taalgebruik, wat heeft geleid tot meer flexibele taaleenheden. Om de echte eenheden van onze taalcognitie (d.w.z. cognitieve eenheden) te achterhalen, gaat deze dissertatie uit van een op gebruik gebaseerde ('usage based') visie en probeert zo dichter bij de cognitieve werkelijkheid te komen.

Deze dissertatie bestudeert cognitieve eenheden door drie belangrijke vragen te onderzoeken. Ten eerste, welke stukken taal worden geleerd als cognitieve eenheden? Vervolgens, hoe kunnen mensen taalinput segmenteren in cognitieve eenheden zonder zichtbare patronen? Tenslotte, hoe verwerken mensen de eenheden in hun cognitief systeem? Deze vragen, samen met het belang van onderzoek naar cognitieve eenheden, worden in Hoofdstuk 1 geïntroduceerd.

In Hoofdstuk 2 introduceer ik een toolbox, *EasyEEG*, die is ontwikkeld voor onze daaropvolgende multivariate patroonanalyse (MVPA) van EEG-gegevens. De MVPA methoden vereisen geen voorafgaande kennis van de timing of locatie van neurale activiteit. Dit voordeel is van cruciaal belang voor de studies in de volgende twee hoofdstukken, omdat eerdere studies niet voldoende kennis hadden van de timing of locatie van de neurale activiteit voor verwerking van cognitieve eenheden.

In de Hoofdstukken 3 en 4, presenteer ik de empirische (gedrags- en neurowetenschappelijke) bevindingen over cognitieve eenheden. Hoofdstuk 3 stelt een conceptueel model voor dat twee stadia van unitisering tijdens het lezen beschrijft: In de eerste fase (detectiefase) kan ons brein gelijktijdig alle herkenbare eenheden in de buurt van een kijkpunt detecteren, ongeacht hun grootte; In de latere fase (herkenningsfase) geeft ons brein voorrang aan grotere gedetecteerde eenheden boven kleinere. Hoofdstuk 4 herbevestigt de detectiefase bij het lezen van complexere reeksen, maar de herkenningsfase wordt niet bevestigd of gefalsifieerd.

De bovenstaande empirische bevindingen suggereren dat ons brein de voorkeur geeft aan de kleinste inspanning en grotere eenheden, wat me inspireerde tot het ontwikkelen van een niet-gesuperviseerd computationeel model, *Less-is-Better* (*LiB*), gepresenteerd in Hoofdstuk 5. Gebaseerd op de hypothese dat de cognitieve eenheden de cognitieve inspanning van taalgebruikers kunnen minimaliseren, probeert het LiB model de eenheden te vinden die zowel het aantal tokens (inspanning van het werkgeheugen) als het aantal types (inspanning van het lange termijn geheugen) minimaliseren. Het resultaat is dat het model een gegeven tekst kan segmenteren in opeenvolgingen van eenheden die betere computationele prestaties laten zien dan andere veelgebruikte eenheden.

De aannemelijkheid van de model-afgeleide eenheden als cognitieve eenheden moet worden getest in realistische cognitieve taken, d.w.z. taken die betrekking hebben op empirisch menselijk gedrag. Daarom probeer ik in Hoofdstuk 6 de Less-is-Better eenheden te gebruiken om oogfixaties tijdens het lezen te voorspellen onder de hypothese dat de oogfixaties tijdens het lezen zich rond de centra van cognitieve eenheden bevinden. De correcte voorspellingen ondersteunen niet alleen de hypothese, maar ook de cognitieve realiteit van de LiB eenheden.

In het laatste hoofdstuk vat ik de bevindingen van de vorige hoofdstukken samen, beantwoord ik de drie vragen uit Hoofdstuk 1, en bespreek ik de implicaties van de bevindingen en de theoretische verbanden met andere domeinen. De uitkomsten van dit proefschrift vormen een verre van volledig begrip van de cognitieve eenheden van taal, maar openen in ieder geval een nieuw pad voor het bestuderen van taaleenheden. Toekomstig werk kan gericht zijn op het verzamelen van meer empirische informatie over cognitieve eenheden, het verbeteren van het computationele model van cognitieve eenheden, en het gebruik van cognitieve eenheden in andere taaltaken om de prestaties op deze taken te verbeteren.

English Summary

Language is often viewed as a sequence of discrete units, so most analyses of language material and studies of language cognition begin by determining the units of language. Most people intuitively determine linguistic units (e.g., characters, syllables, words, and sentences) in terms of distinct boundaries. Early scholars in linguistics who viewed the structure of language as a formal system introduced new units (e.g., morphemes, phrases, and clauses) to represent the formal structure of language in more detail. With the advancement of cognitive science, some scholars recently emphasized the influence of cognitive factors and actual language usage on language structure, which has led to more flexible language units. To know the language units that are genuinely used in our cognitive processes (i.e., cognitive units), this thesis adopts a usage-based view and attempts to get closer to the cognitive reality.

This thesis studies cognitive units by investigating three main research questions. First, what chunks of language are learned as cognitive units? Next, how can people segment language input into cognitive units without overt patterns? Last, how do people process the cognitive units in their minds? These questions, together with the importance of investigating cognitive units, are introduced in Chapter 1.

In Chapter 2, I introduce a toolbox, *EasyEEG*, which was developed for our subsequent multivariate pattern analysis (MVPA) of EEG data. The MVPA methods require no prior knowledge of the timing or location about neural activity. This advantage is critical to the studies in the next two chapters since previous studies had not provided sufficient timing or location knowledge of the neural activity of cognitive unit processing.

In Chapters 3 and 4, I present the empirical (behavioral and neuroscience) findings of cognitive units. Chapter 3 suggests a conceptual model describing two stages of unitization during reading: In the early stage (detection stage), our mind can simultaneously detect all recognizable units nearby a point of gaze, regardless of their sizes; In the later stage (recognition stage), our minds prioritize larger detected units over smaller ones. Chapter 4 again confirms the detection stage when reading more complex strings, but fails to confirm or falsify the recognition stage.

The above empirical findings suggest that our mind favors least effort and larger units, which inspired me to construct an unsupervised computational model, *Less-is-Better* (*LiB*), presented in Chapter 5. Based on the hypothesis that the cognitive units can minimize the cognitive effort of language users, the LiB model tries to find the units that minimize both the number of unit tokens (effort of working memory) and the num-

ber of unit types (effort of long-term memory). As a result, the model can segment any given text into sequences of units that show better computational performance over other commonly used units.

The plausibility of the model-derived units as cognitive units should be tested in realistic cognitive tasks, i.e., tasks related to empirical human behavior. Therefore, in Chapter 6, I attempt to use the Less-is-Better units to predict eye fixations during reading under the hypothesis that the eye fixations during reading locate around the centers of cognitive units. The successful predictions support not only the hypothesis, but also the cognitive reality of the LiB units.

In the final chapter, I summarize the findings of previous chapters, answer the three questions listed in Chapter 1, and discuss the findings' implications and the theoretical connections to other domains. The current outcomes of this thesis are far from a complete understanding of the cognitive units of language, but at least it opens up a new path for studying language units. Future work can be centered around gathering more empirical information about cognitive units, improving the computational model of cognitive units, and using cognitive units in other language tasks to improve performance on these tasks.

Chinese Summary

我们通常把语言视为一串离散单元的序列组合，因此在分析语言材料和研究语言认知之前，往往要先确定语言中的单元是什么。多数人会凭直觉，使用语言中的明显边界确定语言单元（如音节、字、词和句子）。早期的语言研究者相信语言的结构是形式化的，他们为了让语言的结构更能体现形式化而引入了新的单元（如语素、短语和句子）。随着认知科学的发展，最近有些研究者们开始强调认知因素和现实中灵活的语言用法对语言结构的影响，这使得语言单元更加灵活多样。为了探索我们的认知过程中实际在采用的语言单元（即认知单元），本论文延续了基于用法研究语言的思路，并更加关注真实的人类认知。

本论文通过三个主要研究问题来研究认知单元。首先，哪些语言组块能够成为人们学到的认知单元？其次，既然认知单元没有明显可被识别的特征，人们又是怎样将语言输入分割成认知单元的？最后，人们的头脑中是如何加工这些分割出的认知单元的？第1章解释了研究认知单元的重要性，并详述了上面三个问题。

在第2章中，我介绍了一个编程工具箱，即EasyEEG。它是为了对EEG数据进行多变量模式分析（MVPA）而开发的。MVPA方法不需要事先了解关于神经活动的时间或位置。因为对于认知单元加工的神经活动，前人的研究没有提供足够的关于时间或位置的背景知识，MVPA的这一优势对后面两章的研究至关重要。

在第3章和第4章中，我介绍了认知单元的经验性（行为学和神经科学的）发现。第3章提出了一个概念模型，描述了阅读过程中单元化的两个阶段。在早期阶段（探测阶段），我们的大脑可以同时探测到一个注视点附近的所有的可识别的单元（不管这些单元是大是小）；在后期阶段（识别阶段），我们的大脑会优先识别探测到的较大的单元。第4章再次证实了在阅读更复杂的字符串时探测阶段仍然存在，但这里未能证实或证伪识别阶段的存在。

上述经验性发现表明，我们的头脑倾向于更少的认知负担和较大的认知单元，这也启发我构建了一个无监督的计算模型，即*Less-is-Better* (意思是“越少越好”；简称*LiB*)模型。我在第5章中介绍了这个模型。我做了这样一个假设：认知单元可以最小化语言使用者的认知负担。基于此假设，LiB模型试图把语料库分割为一串语言单元，并使语料库中单元(tokens)的数量（工作记忆的负担）和同时建立的词库中单元(types)的数量（长时记忆的负担）都最少。该模型可以将任何输入文本分割成一连串单元（LiB单元）。而且，在多个计算语言学任务的性能评测中，把语料分割为LiB单元比起分割为其它常用的语言单元（例如，词）都更好。

LiB单元是否就是认知单元，这个问题需要在真实的认知任务（即与人类经验行为有关的任务）中被检验。因此，在第6章中，我使用了LiB单元来预测阅读中眼睛注视点的位置。这种预测背后的假设是，阅读的单元就是认知单元，阅读的注视点

会落在每个认知单元大约正中的位置。最终的成功预测不仅支持了这个假设，也支持了LiB单元作为认知单元的真实性。

在最后一章中，我总结了前几章的发现，回答了第1章中列出的三个研究问题，并讨论了这些发现的意义以及与其他领域的理论性相关。虽然本论文目前的成果远未做到完整理解语言认知单元，但至少它为研究语言单元开辟了一条新的道路。未来的工作可以围绕收集更多关于认知单元的经验性信息，改进认知单元的计算模型，以及在其他语言任务中使用认知单元来提高这些任务的表现。

Acknowledgements

My PhD journey started five years ago when I got the fellowship from the Max Planck Institute for Psycholinguistics. There are so many people I would like to thank. The first two are, of course, my supervisors, Antal and Stefan. I am grateful to have met such patient supervisors! Every time I finished a draft of my paper, they double-checked every word. Their revisions greatly improved the quality of my papers, and I could gradually understand the subtler ideas of writing from their comments. Moreover, I must admit that I am a hopeless scheduler - I am always lazy about the necessary but curious about all kinds of unnecessary things. I couldn't have finished my thesis without Stefan and Antal pulling me back on track many times. I also appreciate the feedback from my thesis committee members, Mirjam Ernestus, Fermín Moscoso del Prado Martín, Inbal Arnon, Tianlin Wang, and Lars Meyer. They showed me what I had missed.

I have also received much support from staff and colleagues throughout my PhD career. Whenever I have a workflow problem, Kevin is the first person I consider asking. Margret helped me with my experimental schedule. Bob helped me when I had problems with the lab equipment. Henk organized seminars and shared his Overleaf accounts with me. Angela guided me on my contracts and institute accounts. Edith and Luise handled my conference travel arrangements and reimbursements. Johan built the trigger box I requested. Wessel helped me with problems on the server. I have three great office-mates, Marten, Afrooz, and Danny. I am not used to join the Dutch-style party to make friends, but because of the close distance and familiarity, I can enjoy the time with them.

Life abroad can be difficult without friends from the same country. Fortunately, I met a lot of Chinese friends in my life in Nijmegen. Chen Shen was my first Chinese colleague, and she was the first new friend I met in Nijmegen. We used to live closely with each other, so we used to leave work together and discussed about everything. I heard she just had a baby; congratulations! Zhen Zhang invited me into a circle where there are also Ting Mei, Weitao Zhang, Chao Tang, and Mingqian Guo. We traveled around, had parties, or enjoyed delicious food made by Zhen and Ting almost weekly. I have a lot of great neighbors, Chao Guo, Tao He, Mingmei Sun, Chunan Guo, Qi Song, Liu Shi, Fugang Gao, and Juanyuan Zhao. We often had parties or visited each other's houses. I often had dinner with my CLS friends Yu Bai, Wei Xue (and her husband, Zhengyu Zhao), Xiaoru Yu, Hongling Xiao, Lei Wang, my Donders friends Xiangzhen Kong, Xiaochen Cheng, Bohan Dai, Xu Gong, Wei Liu, and Huan Wang. Lei Li, Junfei Hu, Yiguang Liu, Wanxiao Tang, Shan Jiang, Jiahao Xu, Qiming Su, Jiaqi Wang, and

Tongyu Wu, we played TRPG or chatted online frequently. We had a great time, and I miss them all. Some of them are far away. But the world is such a small place. Who knows?

The most important support came from my family. My grandfather, father, mother, aunts, sister, and niece stayed in China, but I still had close contact with them from frequent video calls. Their concern relieved me of homesickness, and their understanding encouraged me in my career. Although we went through a hard time, we got through it well. I hope our family will always be happy, safe, and healthy. About three years ago, Sizhu Han came to Europe as a visiting scholar. Fate brought us together, and she became my girlfriend, my first lover. Her presence in my life changed me. I believe she will become the other half of my life.

Curriculum Vitae

Jinbiao Yang was born in Fuyang, China, on December 31st 1990. He obtained his bachelor's degree in Computer Science and Technology from Anhui Agricultural University, China, in 2013. He considered that human language ability as the key to general artificial intelligence, so he started studying psycholinguistics and cognitive neuroscience at East China Normal University, China where he received his MSc degree in 2016. He then worked as a research associate at New York University Shanghai, China for one year. In 2017, he received the PhD fellowship from the Max Planck Institute for Psycholinguistics (MPI) and became a PhD candidate at both MPI and the Centre for Language Studies, Radboud University, the Netherlands. He finished this thesis in 2022, and joined the Language and Computation in Neural Systems Group at MPI as a postdoctoral researcher.

Publications

Yang, J., van den Bosch, A., & Frank, S. L. (2022). Unsupervised text segmentation predicts eye fixations during reading. *Frontiers in Artificial Intelligence, 5*:731615.

Yang, J., Frank, S. L., & van den Bosch, A. (2020). Less is Better: A cognitively inspired unsupervised model for language segmentation. *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, 33–45.

Yang, J., Cai, Q., & Tian, X. (2020). How do we segment text? Two-stage chunking operation in reading. *eNeuro, 7*(3).

Teng, X., Ma, M., Yang, J., Blohm, S., Cai, Q., & Tian, X. (2020). Constrained Structure of Ancient Chinese Poetry Facilitates Speech Content Grouping. *Current Biology: CB, 0*(0).

Yang, J., Zhu, H., & Tian, X. (2018). Group-Level Multivariate Analysis in EasyEEG Toolbox: Examining the Temporal Dynamics Using Topographic Responses. *Frontiers in Neuroscience, 12*, 468.

Frank, S. L., & Yang, J. (2018). Lexical representation explains cortical entrainment during speech comprehension. *PloS One, 13*(5), e0197304.