Disfluency Prediction

in Natural Spoken Language

Jiří Zámečník

NIHIN studies
New Ideas in Human Interaction

Disfluency Prediction

in Natural Spoken Language

Inaugural-Dissertation

zur

Erlangung der Doktorwürde

der Philologischen Fakultät

der Albert-Ludwigs-Universität

Freiburg i.Br.

vorgelegt von

Jiří Zámečník

aus Uherské Hradiště

SS 2019

Erstgutachter: Prof. Dr. Dr.h.c. Christian Mair

Zweitgutachter: Prof. Dr. John Nerbonne

Drittgutachter: Prof. Dr. Lars Konieczny

Vorsitzender des Promotionsausschusses
der Gemeinsamen Kommission
der Philologischen und
der Philosophischen Fakultät: Prof. Dr. Dietmar Neutatz

Datum der Disputation: 09.09.2020

To Надя, for giving me hope.

Acknowledgments

I would like to thank my first supervisor, Prof. Dr. Dr.h.c. Christian Mair for all the guidance and expertise that he shared with me during my work on this thesis. I am extremely grateful to him for not pulling the reins in at the moment when I started talking about my thesis as being not only corpus linguistic but computational linguistic as well and I value that he reminded me to keep the big picture in mind every time I focused too much on a narrow topic. Furthermore, I appreciate his understanding of the practical needs of a doctoral student: when I approached him with my intention to write a doctoral thesis, he promised that I would not go starving during my work and he kept his promise most generously.

My sincere gratitude also belongs to my second supervisor, Prof. Dr. John Nerbonne. I am most grateful for his openness in telling me that the work is not finished yet and pointing out the necessary changes in the structure and theoretical framing of this thesis. This suggestion improved the thesis more than any number of hours of my proofreading could achieve. Also, if it was not for his comments, the equations in this thesis would be much harder to read, if not incomprehensible. I also value his patience in trying to understand my e-mails which were often more confusing than explanatory.

Furthermore, I would like to thank all those who provided me with precious feedback, comments and encouragement, both with regard to the work presented here and my other academic projects. This includes the audience of conferences that I attended and most notably the audience of the linguistic research seminar at the English

Department of the University of Freiburg as well as the research colloquium of the DFG GRK Frequency Effects in Language. Their questions and suggestions allowed me to spot some of my errors and improve the work beyond what I would be capable of achieving on my own. Some of the people I am particularly grateful to are Prof. Dr. Dr.h.c. Bernd Kortmann, Prof. Dr. David Denison, Mirka Honkanen, Kyla McConnell, Julia Müller, David Lorenz and Klára Jágrová. A special thank you goes to Jeffrey Mitchell for his patience in explaining his language model to me.

Furthermore, I would like to thank my parents, grandparents and my brother: you supported me, encouraged me and reminded me that research should have a practical purpose. I appreciate your support throughout my studies and my work on this thesis. Similarly, I am grateful to Анна and Павло: you gave me many helpful suggestions, inspired me to pursue concrete goals and drew my attention to the important aspects of academic work. Thank you for your support.

Finally, my most personal gratitude belongs to Надя: thank you for keeping me sane and keeping my spirits up. Thank you for listening to my complaints and finding the positive side of any issue. And thank you for being there for me when I needed you most.

Despite all the help and support received, all errors remain my own.

Zusammenfassung in der deutschen Sprache

Diese Arbeit beschäftigt sich mit Zögerungssignale im gesprochen Englischen. Mit einer Kombination von korpus- und computerlinguistischen Methoden wird versucht, drei Typen von Zögerungssignale (Wiederholungen, stille Pausen und *uh/um*) in einem Korpus des Englischen vorherzusagen. Dies könnte eine praktische Anwendung in der Sprachsynthese finden, da es eine konversationelle und natürlichere synthetisierte Sprache ermöglicht.

Zusätzlich zu den qualitativen und quantitativen Prädiktoren aus der bisherigen Forschung – wie zum Beispiel Mutual-Information-Score, Lexical-Gravity-Score oder Wortart – wird ein neuer Prädiktor vorgeschlagen: Surprisal, also die informationstheoretische Einschätzung der Vorhersagbarkeit eines Wortes. Es dient als eine numerische Einheit mit der die Vorhersagbarkeit des Wortes und vor allem die Last an die kognitive Systeme des/der Sprechers/in und des/der Hörers/in, die durch dieses Wort verursacht wird, gemessen werden kann. Es wird argumentiert, dass Wörter mit einem hohen Surprisalwert öfter nach einem Zögerungssignal erscheinen sollten: in diesem Fall würden die Zögerungssignale dafür dienen, einen rasanten Anstieg im Surprisal zu glätten.

Da Surprisal bisher nie dafür angewandt wurde, Zögerungssignale vorherzusagen, stellt diese Arbeit zuerst die Pilotstudie I vor. In dieser Studie wird die Verbindung zwischen Surprisal und Zögerungssignalen an dem John Swales Conference Corpus getestet. Nach dem erfolgreichen Test wird dann Surprisal zu anderen Prädiktoren aus der bisherigen Forschung hinzugefügt um

Zögerungssignale in dem Michigan Corpus of Academic Spoken English vorherzusagen.

Die Ergebnisse werden in Studien IIa und IIb präsentiert. Sie zeigen, dass Zögerungssignale tatsächlich nicht zufällig verteilt werden, sondern dass sie zu einem bestimmten Grad durch die gewählten Prädiktoren erklärt werden können. Die wichtigste Rolle spielt dabei die Position im Satz oder im Turn: die ersten Wörter im Satz/Turn sind besonders oft die Auslöser von Zögerungssignalen. Auch hier wird das Surprisal als ein relevanter Prädiktor identifiziert; es wird gezeigt, dass sich das Surprisal-Profil von Zögerungssignalen vom restlichen Text unterscheidet.

Studie III präsentiert dann eine rein computerlinguistische Perspektive. Zögerungssignale werden mithilfe einer Encoder-Decoder-Architektur für maschinelle Übersetzung vorhergesagt. Die Ergebnisse sind denen von Studien IIa und IIb ähnlich. Für praktische Anwendungen scheint daher diese Methode besonders geeignet zu sein, da sie wesentlich weniger Datenvorverarbeitung benötigt. Außerdem kann sie auch für die Entfernung von Zögerungssignalen in einem automatisch transkribierten Text angepasst werden, indem die Übersetzungsrichtung geändert wird.

Diese Arbeit bietet einen neuen Ansatz zu dem komplexen Thema der Vorhersage von Zögerungssignalen. Die neuentdeckte Verbindung zwischen Surprisal und Zögerungssignale bietet die Möglichkeit, einzelne Prädiktoren aus bisheriger Forschung durch ein unterliegendes Prinzip zu erklären: die Verarbeitungskomplexität. Außerdem wird gezeigt, dass die Position innerhalb syntaktischer

Struktur von besonderer Wichtigkeit ist: Wörter mit dem gleichen Surprisalwert unterscheiden sich in der Wahrscheinlichkeit mit der sie ein Zögerungssignal auslösen, wenn sie sich in unterschiedlichen Positionen in der syntaktischen Struktur befinden. Letztlich wird gezeigt, dass das neuronale Übersetzungsmodell ähnlich wie das theoretisch motivierte psycholinguistische Modell abschneidet, obwohl es die Regeln der Sprache selbstständig anhand von nicht annotierten Daten erlernen muss.

Contents

Abbreviations used

ASD	Autism spectrum disorder
CART	Classification and regression trees
COCA	Corpus of Contemporary American English
ERP	Event-related potential
G	Lexical gravity score
GRU	Gated recurrent unit
IOB	Inside-outside-beginning chunking
JSCC	John Swales Conference Corpus
LDA	Latent Dirichlet Allocation
LSA	Latent semantic analysis
LSTM	Long short-term memory
MASC	Manually Annotated Subcorpus of the American National Corpus
MI	Mutual information score
MICASE	Michigan Corpus of Academic Spoken English
MLP	Multi-layer perceptron
PCFG	Probabilistic context-free grammar
POS	Part of speech
RNN	Recurrent neural network
TP-B	Backward transitional probability
TP-D	Direct transitional probability
UID	Uniform Information Density hypothesis

Chapter 1
Introduction

Until recently, disfluencies in the human language were outside of the focus of linguistic research. Viewed as noise in the otherwise systematic production of language, they were neglected in theories of language production with the exception of narrowly defined research areas focusing on phenomena such as stuttering or aphasia. This was perhaps a consequence of the fact that linguistic research was dominated by the work on written language. Spoken communication was often viewed as parallel to writing, differing only in the mode of production (Chafe 1992; Linell 1982). Even though some studies on disfluency phenomena did exist (Good & Butterworth 1980; Goldman-Eisler 1968; Beattie & Butterworth 1979; Maclay & Osgood 1959), their distribution in natural speech of healthy speakers only came into prominence after Shriberg's (1994) seminal work. Since then, a number of empirical studies from both psycholinguistic and corpus-linguistic perspective have been published (among other Engelhardt et al. 2017; Eklund et al. 2015; Schneider 2014; Barr & Seyfeddinipur 2010; Arnold et al. 2007; Corley et al. 2007; Ferreira & Bailey 2004; Clark & Fox Tree 2002; Brennan & Schober 2001; Clark & Wasow 1998), striving to describe their distribution and explain the mechanism through which they are placed. Even though no definitive explanation has been found yet, these studies (and others reviewed in Chapters 3 and 4) have shed light on some of the intricate details of their use.

Similarly to more traditional approaches to language, computational linguistics has long viewed disfluencies simply as problems. When discussing disfluencies, much of the debate focused on detecting and removing them (Jamshid Lou & Johnson 2017; Honnibal & Johnson 2014; Johnson & Charniak 2004; Shriberg et al. 1997) in order to reconstruct an underlying written-like message from a spoken utterance. However, with recent advances in speech synthesis and the demand for inclusion of emotions and expressiveness in synthesized speech, the issue of predicting disfluencies became relevant, too. Synthesized speech with disfluencies is reported to be perceived as more natural and conversational (Dall et al. 2014a; Dall et al. 2014b; Adell et al. 2007) than synthesized speech produced in the style of a news anchor reading a scripted text. Disfluency prediction is thus one avenue of research through which speech synthesis may advance.

This thesis combines a corpus linguistic approach with a computational linguistic one. On the one hand, it aims to explain the placement of disfluencies from the quantitative perspective, drawing inspiration from psycholinguistic and information-theoretic observations. On the other hand, it also attempts to predict disfluencies in a cleaned text stream. The disfluency prediction uses a set of previously suggested predictors, as well as a novel approach based on the results of the empirical study presented in Study I. This study employs a psycholinguistically inspired language model to assess the surprisal (Levy 2008, Hale 2001, Attneave 1959, Shannon 1948) of an item and test whether it can provide information about the occurrence of disfluencies. It shows that disfluencies do not occur uniformly, but rather cluster in areas of high surprisal.

To my knowledge, the link between disfluencies and surprisal has not been explored in previous work. Thus, in order to show that surprisal is a meaningful predictor of disfluency placement, I first present the pilot Study I, which explores the placement of disfluencies in a corpus of spoken language. This study tests the following hypothesis:

Hypothesis 1:

> The occurrence of disfluencies may be predicted by the local surprisal.

The confirmation of this hypothesis motivates the work presented in Studies IIa and IIb. These studies attempt to predict disfluencies in a text from which they were previously removed. For this purpose, the surprisal estimate is added to previously observed predictors of disfluencies. The set of predictors is then evaluated through standard frequentist statistical methods and employed in computational methods that aim to predict disfluencies in previously unseen data. The results show that frequency- and probability-based measures are capable of predicting disfluencies to some degree, though they cannot explain the full variation in their use.

The hypothesis in Study IIa is motivated by the observation made in Study I, extending Hypothesis 1 and the observations of previous research.

Hypothesis 2.1:

> The surprisal estimate produced by the model defined in Chapter 2.3 is a predictor of disfluency occurrence in the MICASE corpus.

Hypothesis 2.2:

> The occurrence of a disfluency depends not only on the overall predictability of the word it precedes, but also on the difference in the local information transmission rate – that is the difference in the predictability of the word preceding the position in which the disfluency was inserted and the word following it.

After critically evaluating the performance of disfluency prediction on a corpus of spoken language that includes more data from a broader domain than the one used in Study I, Study IIb extends the set of predictors. It appends the numeric scores of Studies I and IIa with structural properties of individual items in the form of part of speech as well as the position within an intermediate syntactic unit, the chunk. Thus, the structural perspective is explored as well.

Finally, Study III provides a computational outlook by framing the disfluency prediction task as a translation task from a fluent language to a disfluent one. It shows that modern machine translation models can perform similarly to theoretically-motivated disfluency prediction models in spite of not having access to the preprocessed language statistics or predefined structural information.

The disfluencies discussed in this thesis belong to the group of hesitations, i.e. “temporary suspension[s] of flowing speech” (Lickley 2015: 456). These consist of filled pauses (*uh, um*), unfilled pauses and repetitions. Thus, this work may be viewed as a preliminary step to a more complex disfluency prediction model that would also include repairs and deletions. The purpose of such a model is twofold. On the one hand, naturalistic disfluency placement allows the creation of more natural text-to-speech synthesizers and conversation systems. On the other hand, a better understanding of the process that governs disfluency placement will also improve our understanding of the speech production process.

Importantly, the term *disfluency* will be used throughout this thesis as referring to phenomena interrupting the flow of speech without adding any propositional content (Fox Tree 1995), even if these are deliberate hesitations used for rhetoric purposes (O'Connell & Kowal 2004). Thus, it is not used as marking an error or incorrect use of language, but simply as a reference to the location in the speech stream where a temporary suspension of the production occurs (pauses), or where an element is repeated. For a more detailed discussion of the multiple definitions of the term and their connotations, the reader may refer to the article of Gilquin & Cock (2011) or Chapter 2.3.3 of Shriberg’s thesis (1994).

Finally, before the individual studies are presented, an introductory chapter discussing the term surprisal/information as used throughout this thesis is included. This chapter also describes the language model employed for the surprisal estimation.

Chapter 2
Preliminaries

This chapter introduces concepts which are common to the studies presented in this thesis. It first introduces the terms surprisal/information and information density and discusses their use in this thesis as contrasted to the one found in general linguistics. Then, the individual elements of the language model employed in this thesis are defined and explained. Since the language model combines these individual elements into one joint model, the approach used for their integration is presented in Chapter 2.3.

2.1 Surprisal

In addition to other measures suggested by previous studies to explain disfluency placement, this thesis included the surprisal of individual items. This notion is derived from the Mathematical Theory of Communication by Shannon (1948), later extended to Information Theory. In his work, Shannon attempted to define the characteristics of an optimal communication system. While his work was primarily directed at electronically encoded human language, it was later successfully applied as a general theory in a range of scenarios, including human communication in its natural form. It views any communication as an attempt to transmit a message from its source to its destination over a noisy channel as sketched in Figure 1. At the beginning, a message is formulated at the source and translated into some encoding capable of being sent over the chosen channel. Then the encoded signal is sent to the receiver, who decodes it in order to obtain the original message (or at least its close approximation, absolute fidelity is unattainable according to Shannon). Each element of this transmission (whether it is an electronic impulse or a phone/letter/word/etc.) carries with itself some information. This information is defined as the amount by which the uncertainty about the outcome (the message) is reduced. Informative elements reduce the uncertainty greatly, whereas elements that do not reduce the uncertainty at all are considered informationally empty.

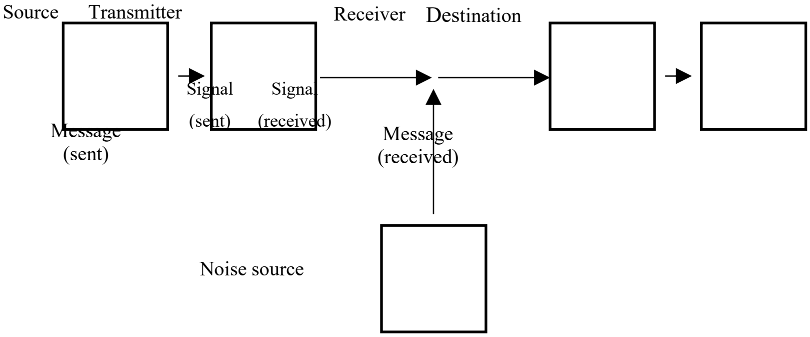


Figure 1: Visualization of a general communication system by Shannon (adapted from Shannon 1948: 380).

Since Shannon was interested in comparing various encodings so as to find the ideal one, he needed to find a measure to quantify this notion. This measure had to adhere to the intuitive features of information, such as the fact that it can be encoded, transmitted and stored. Similarly, it should be additive and non-negative since no transmission should leave the receiver with less information than they had at the beginning (Floridi 2009). Further, it should take into account that the amount of uncertainty about the outcome is dependent on the type of communication. In the simplest scenario, where the transmitter is only capable of sending one message, the receiver knows what the message is going to be even prior to decoding it. Thus, there is no uncertainty to start with and the successful reception cannot and does not reduce it. As a consequence, the amount of information transmitted by each element should equal to 0. On the other hand, the larger the range of possible messages, the larger the amount of information transmitted by each element.

Shannon observed that information defined in such terms is inversely related to probability – the more outcomes exist, the lower the average probability that a particular one will occur. Thus, subtracting probability from 1 would yield a measure conforming to the requirements mentioned above. Additionally, in order to remove the bounding of the measure by the interval $\langle 0,1 \rangle$ and to express that extremely unlikely events carry a substantially larger amount of information than somewhat unlikely ones, the probability is additionally log-transformed. As a result, the informativeness of an item (also called its surprisal, cf. Levy, 2008; Hale, 2001; Attneave, 1959) is expressed as:

$$I = -\log(P) \qquad (1)$$

To achieve optimal communication, the encoding of messages should be devised in such a way that the information carried by each element is as close to the channel capacity as possible. In such a way, the resources are used efficiently. Moreover, the transmission should be smooth, without peaks or troughs in the transmission rate (Jaeger 2006). This suggestion stems from Shannon's acknowledgement of the inherent noisiness of transmission channels. Given that any message is likely to be distorted by noise to some degree (and cannot be reconstructed fully with absolute certainty by any function, hence the aforementioned unattainability of absolute fidelity), smoothing information transmission minimizes the risk that a sudden peak in the noisiness will cause a disproportionate information loss.

In the context of natural language communication, evidence for the optimization of information transmission actually predates the Mathematical Theory of Communication. As observed by Zipf (1932), the frequency of a word is a good indicator of its length. Since relative frequency yields the simplest estimate of probability, this relationship corresponds to a simple transmission optimization strategy: in order to transmit messages efficiently, "messages of high probability are represented by short codes and those of low probability by long codes" (Shannon 1948: 395). This correlation of length and frequency is robust across languages and definitions of length and frequency (Grzybek 2006; Strauss et al. 2006). Importantly, in a more recent study, Piantadosi et al. (2011) were able to verify and generalize the suggestion of Manin (2006) that better estimates of the average surprisal of words will yield a better predictor of their length. By using n-gram based probabilities they were able to show that average predictability in context outperforms relative word frequency in predicting a word's length. Such a relationship results in an encoding that is better optimized in terms of transmission smoothness and efficiency.

The effect of surprisal was identified in other domains of language, too, both from diachronic (e.g. Degaetano-Ortlieb & Teich 2019) and synchronic perspective, the latter prominently represented by the Smooth Signal Redundancy hypothesis (Aylett & Turk 2004) and the Uniform Information Density hypothesis (UID, Jaeger 2010; Jaeger & Levy 2006; Jaeger 2006). Both of these hypotheses suggest that speakers optimize their output constantly, so as to produce smoothened transmissions. In the Smooth Signal Redundancy hypothesis, based on studies of articulatory detail (Gahl 2008; Bell et al. 2009; Aylett & Turk 2004, 2006; Bell et al. 2002), the element-wise surprisal is assessed on the phonetic level. The hypothesis suggests that predictable phones should be produced with less articulatory detail in comparison to those carrying a high information load. The UID, on the other hand, makes predictions about syntactic decisions. Its basic tenet is that "speakers optimize successful transfer and minimize effort if they aim at transmitting information at a uniform rate" (Jaeger 2006: 195). This is achieved by inserting/removing optional elements (e.g. the complementizer *that*) at appropriate choice points so as to produce smoothened output.

An important feature of surprisal is that it can serve as a proxy to the cognitive load caused by the processing of an item (Levy 2008; Hale 2001). High-surprisal contexts should thus be more complex in terms of comprehension. This is supported by studies linking surprisal to other measures of cognitive load, such as reading times (Frank 2013, 2017; Smith & Levy 2013; Mitchell et al. 2010) or event-related potential (ERP) responses (Frank et al. 2013, 2015). Thus, lowering the surprisal of an item by some smoothing method should facilitate its processing. Conversely, increasing the surprisal of an item may increase the efficiency of the transmission, but only at the cost of the processing load. Finally, the amount of information carried by a given element of the transmission, i.e. its surprisal, expresses the information density of the transmission around that element. From the information-theoretic perspective, informationally dense contexts contain elements which have low probability of occurrence.

The crucial task of any study utilizing surprisal is finding a function that will assign the probability of each element as required by Equation 1. As mentioned above, the simplest approach would be taking the relative frequency of an element as its probability. This is clearly suboptimal for human languages, where the probability of each item is partially determined by the context: though *infallibility* is not a particularly frequent lexeme in English, it is more likely to occur after *papal* than one of the most frequent items, *the*.[1] Similarly, while the velar nasal /ŋ/ does not occur more often than the schwa /ə/ in English, their probability after the chain /sɪtɪ/ is not the same: the nasal is much more likely. Thus, the probability of an item in its context must be determined. In previous research, the methods adopted to achieve this goal often differed, depending on the particular focus of each study. In Aylett & Turk (2004), the definition included syllabic trigram probability, givenness and log-transformed word frequency. Frank & Jaeger (2008), on the other hand, employ simple 3-gram probabilities without back-off or smoothing. As a third example, Frank et al. (2015) use three various independent models: an n-gram ($n \in \langle 2,4 \rangle$), a recurrent neural network and a probabilistic phrase-structure grammar model.

The reason for the variability of models used for the estimation of probability of a certain item is simple: as there is no universal language model available, probability (and ultimately information) has to be estimated by means of proxy measures. Aylett & Turk (2004: 39) summarize this point:

> Without understanding all the dependencies between semantics, syntax, pragmatics and the

[1] While *papal infallibility* occurs 51 times in the Corpus of Contemporary American English Davies (2008-), *papal the* is not attested.

> structure of language any measure of redundancy is an approximation.

Here, redundancy is understood as the opposite of information density – highly redundant encodings have low surprisal. The method chosen for redundancy approximation largely depends on the application: studies attempting to predict phonetic effects include a phonetic element, while studies in syntax utilize a syntax-based redundancy measure. Ultimately, however, all the approximations contain an inevitable amount of error. In order to counteract the inherent imprecision, multiple estimators are often used together, such as the aforementioned combination of phonetics, lexical effects and pragmatics (Aylett & Turk 2004; Aylett 2000), joint probability and conditional probability (Bell et al. 2002), or the grouping of n-gram, recurrent neural network and probabilistic phrase-structure grammar (Frank et al. 2015).

However, these models are usually used separately, rather than combined into one joint model. On the one hand, this allows their effects and interactions to be evaluated independently. On the other hand, it limits their ability to balance out each other's weaknesses. In order to respond to this issue, Mitchell (2011) suggested and tested (Mitchell et al. 2010) a compositional model which combines three different language models into one, yielding a robust measure that combines a semantic element with syntactic/lexical models. The following chapters explain each of the individual components in more detail and are followed by an explanation of the procedure used to combine their output, as proposed by Mitchell (2011).

Lastly, a brief comment on the difference between the information-theoretic and general linguistic definition of information is needed. Unlike the information-theoretic approaches to language, general linguistics does not define information as a measurable quantitative variable. Rather, it perceives it as the knowledge that is either stored in the brain of one of the interlocutors or transmitted via the message. This view allows the discussion of information as being old/new to the conversation (Chafe 1976) or the pragmatic implications of information packaging (Vallduví 1993; Chafe 1976), which would not be possible under the information-theoretic definition. From this perspective, informationally dense contexts are those in which many knowledge elements are transmitted through a limited number of surface forms, irrespective of their probability in context.

2.2 Language model elements

2.2.1 Semantic model (LDA)

When attempting to quantify the semantic relationship between a word and its textual history, computational models of language employ models of semantic similarity. The underlying mechanism relies on translating the meaning of a word into a vector in multidimensional space, allowing quantitative measurement of between-word similarity. The practical application of these models includes a range of tasks from modelling semantic priming (Landauer & Dumais 1997) to word sense discrimination (Schütze 1998) and disambiguation in untagged text (McCarthy et al. 2004).

The way in which the semantic vector is constructed may differ, however. Early studies (e.g. Osgood et al. 1957) elicited ratings from human subjects, asking them to rate each word on a number of scales, assuming that these ratings will allow researchers to uncover the latent semantic structure of these words. Such an approach was expected to yield results matching an average speaker's understanding of the test words as well as offer an insight into interspeaker variation. However, it was also limited by its very nature – the number of words to evaluate as well as the size of the semantic space had to remain comparably small as both were constrained by the number of participants from whom the ratings could be elicited given the study budget.

In an attempt to overcome the limitations imposed by the design, text-based distributional models of semantics were suggested, stemming from the slogan by Firth (1957: 11): "You shall know a word by the company it keeps!" These approaches assume that words that appear together should be semantically related (Erk 2012). For example, the relationship between the words *tea* and *water* should be much stronger than between *tea* and *football* given how often the words co-occur.

In order to quantify this relationship, a definition of the context used as "the company" in the co-occurrence calculation is necessary. This may be a passage, sentence or a whole document, including even words uttered/written after a given keyword (Dumais 2004). Once the context is defined, it remains to operationalize its translation into vectors. In the simplest form, each context could be taken as one dimension with the frequency of a given keyword in that context (potentially transformed by some function, such as logarithm) determining its value on that axis. However, given that such semantic spaces would be extremely multidimensional in most cases, some dimensionality reduction method is usually used in order to transform them.

As an example of this technique, the implementation of Latent Sematic Analysis (LSA, Landauer & Dumais 1997; Deerwester et al. 1990) used by Coccaro & Jurafsky (1998) may be taken. In their case, based on 80,000 articles from the Wall Street Journal, the context was defined as the full article in which a word occurred. Upon collecting the frequencies with which each of the 20,000 words of their vocabulary appeared in each of the articles, they transformed these frequencies to reflect the proportion of the total occurrences of this word that were found in a given article. Thus, the values of individual dimensions corresponded to the frequency in the article divided by the overall frequency. The semantic space with 80,000 dimensions (one per article) was then simplified through singular value decomposition to 300 dimensions used to express the meaning of individual words. In order to determine the semantic relation between two words, the cosine of the angle between their vectors was taken – the smaller the value, the stronger the relationship.

Figure 2 gives an example visualization of a simple 2D space, created by Coccaro & Jurafsky (1998). It shows that smaller angles between two vectors – and by extension smaller cosine values – correspond to words which are semantically related, such as *fishing* and *boat* as compared to *fishing* and *Hitler*. This relationship may not necessarily be obvious. The model may capture latent semantic relationships such as those between *fishing* and *Maine/Arizona.* There, *Maine* is shown as standing in a closer relationship to *fishing* than *Arizona*. At first sight, it might seem that these words are completely unrelated to the word *fishing* and should be portrayed as equally different from its vector. However, Coccaro & Jurafsky point out that Maine is more important to the fishing industry than Arizona, a relationship which is reflected in the smaller angle separating their vectors.

Importantly, distributional semantics can also provide insight into language processing by predicting the first pass durations (both word-level and sentence-level LSA) and first fixations in reading-time experiments (Mitchell et al. 2010; Pynte et al. 2008) and the N400 effects as well as areas of neural activity in neuroimaging studies (Frank & Willems 2017). Thus, the similarity ratings may serve as a proxy of the relationship between individual words in the memory, with more similar items being more likely to be activated together.

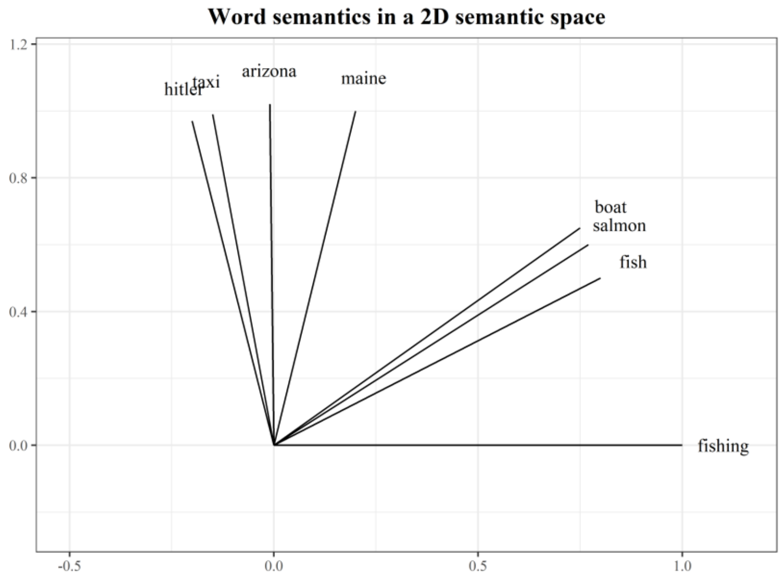


Figure 2: Visualization of word similarity as expressed in distributional semantics by a simple 2D model (adapted from Coccaro & Jurafsky 1998).

The approach for distributional semantics used in this thesis is the Latent Dirichlet Allocation (LDA, Blei et al. 2003). Unlike the LSA, this approach does not construct the semantic vectors on the base of word co-occurrence frequencies, but rather by extracting latent topics which are then used to describe individual words. Concretely, each document in a corpus is modeled as a distribution over *K* topics, each of which is defined as a distribution over words. Rather than representing the relationships between individual words, the meaning vectors represent the probability of a given word to occur given that a certain topic is present in the document. Thus, each of the components of a vector v representing a word w_i equals to:

$$v_k = \frac{p(c_k|w_i)}{p(c_k)} \qquad (2)$$

where c_k stands for a topic in the LDA model.

As these are latent topics, they may not necessarily be easily interpreted or correspond to a set of predefined labels. However, as shown by the example sets of topics extracted from the English Wikipedia (Řehůřek 2018), often a general label is conceivable (e.g. “geography” for the topic in which high probability of occurrence is assigned to *river, lake, island, mountain, area*, or “sports” for the topic typically represented by the words *relay, athletics, metres, freestyle, hurdles*). This factor is stressed by Steyvers & Griffiths (2007), who note that this individual interpretability of topics is a distinct advantage of representing the content of words and documents with probabilistic topics rather than purely spatial representations.

Following Mitchell (2011), the semantic coherence between a word and its lexical history was estimated for content words only. It was measured using cosine similarity ($1 - cosine\ distance$) of the vector representing a given word as compared to the vector of the text history before that word. To construct the history vector for word w_i the vectors of words $w_1 \dots w_{(i-3)}$ (to exclude the range covered by n-grams, described in Chapter 2.2.3) were constructed, merged using vector addition and renormalized so as to yield valid probabilities, summing to 1. As an alternative method of history formation, vector multiplication was tested by Mitchell (2011); in his study, it performed worse than addition.

Thus, the semantic similarity of w_i to the rest of the text is expressed by its cosine similarity to the history vector h_{i-3}. This history vector uses the semantic vectors of words $w_1 \dots w_{i-3}$. For $i < 4$ the history vector is undefined as there is no text history outside the n-gram scope to build the semantic similarity. Finally, except for $i = 4$ where

$$h_1 = w_1 \tag{3}$$

h_{i-3} equals:

$$h_{i-3} = \frac{1}{2}h_{i-4} + \frac{1}{2}w_{i-3} \tag{4}$$

This means that h_{i-3} is calculated by taking the history h_{i-4} and modifying it by the semantic vector of w_{i-3}. As a consequence, the history carries some elements even of words far before a given item, though recent words play a larger role.

Once the history is established, the aforementioned cosine similarity may be used in order to measure semantic coherence between w_i and its history. However, the coherence score cannot be directly used for expressing probability or surprisal as pointed out by Mitchell (2011: 109), who criticized Coccaro & Jurafsky (1998) for "[resorting] to a number of ad-hoc mechanisms to turn the cosine similarities into useful probabilities." Additionally, even if a proper probability was calculated by normalizing the semantic coherence values over the full vocabulary so as to add up to 1, the language model would not take into account the frequency of w_i. Thus, though *brogues* and *shoes* are certainly not as likely to appear after *I walked in my new brown*, a purely semantic model would assign them an almost identical probability, based on their semantic similarity to each other.

In order to remedy this issue, Mitchell (2011: 109) suggested modifying the dot product used in the cosine similarity measure. Thus, rather than calculating:

$$w_i \cdot h_{i-3} = \sum_k \frac{p(c_k|w_i)}{p(c_k)} \frac{p(c_k|h_{i-3})}{p(c_k)} \qquad (5)$$

where the individual vector elements from Equation 2 are multiplied with the corresponding history elements, he suggested to include the underlying probability of the word as expressed by a different model. This can be done in the following manner (to validate that this indeed yields a valid conditional probability, cf. Mitchell 2011):

$$\begin{aligned} & p(w_i|h_{i-3}) \qquad (6) \\ & = p(w_i) \sum_k \frac{p(c_k|w_i)}{p(c_k)} \frac{p(c_k|h_{i-3})}{p(c_k)} p(c_k) \\ & = p(w_i) \sum_k \frac{p(c_k|w_i)p(c_k|h_{i-3})}{p(c_k)} \end{aligned}$$

In this way, the semantic model is used to scale up the probabilities of words that are coherent to their history and scale down those whose semantic profile does not match it. The exact way this rescaling was done will be discussed in further detail in Chapter 2.3.

For the purposes of this thesis, the semantic model was trained using the 400,000,000-word Corpus of Contemporary American English (COCA, Davies 2008-), containing speech transcripts and texts composed in American English between the years 1990-2012[2] with varying degree of formality. Though this is not the largest dataset of English texts available, it provides a good balanced between data size and quality: larger corpora inevitably suffer from increased noise levels which may influence the representativeness as pointed out e.g. by Mair (2015).

The LDA model implementation used was provided by the *Gensim* package (Řehůřek & Sojka 2010). The training employed a pseudo-online learning approach as suggested by Hoffmann et al. (2010), training on chunks of data, rather than the whole dataset at once. In order to avoid potential detrimental influence of topic shift in the corpus, the file order was randomized prior to training. The model training setup is printed in Table 1.

The model fit of the semantic model was verified against the WordSim 353 dataset (Agirre et al. 2009; Finkelstein et al. 2002) where it correlated with the averaged human similarity ratings at r = 0.475. Such a correlation is far from perfect (current state-of-the-art results

[2] This information is related to the version used; COCA is being extended regularly.

achieve a Spearman correlation of 0.828, cf. Speer et al. 2017). Still, it should be sufficient for the integration in the combined surprisal measure as it is only used in the rescaling of the lexical probability which is afterwards interpolated with the syntactic probability. In the interpolation, the semantically modified lexical probability is given a lower weight than the syntactic probability (cf. Chapter 2.3). Moreover, the mean difference between the highest and lowest similarity rating (which could range from 0 to 10) assigned to one word pair of the WordSim 353 dataset by the human raters is 6.29 (SD = 1.749, median = 6.5). Thus, even the scores assigned by two speakers will never be perfectly correlated, suggesting that even a model that provides ratings perfectly correlated with the averages in the WordSim dataset may actually not be modelling the semantic representations of any concrete speaker.

Parameter	**Value**
Latent topics	100
Minimal values of $p(c_k\|w_i)$ that are retained	0.001
Training chunk size (number of documents)	15.000
Passes through the corpus	1
Alpha (a priori belief for $p(c_k)$)	Auto
Eta (a priori belief for $p(c_k\|w_i)$)	0.1
Maximum iterations of a chunk	150

Table 1: Non-default training settings used in *Gensim*. While the whole corpus was only passed through once, the weights were updated on chunks of 15000 documents, resulting in a total of 25 chunks. The algorithm repeatedly adjusted the parameters to each chunk until convergence, but not more than 150 times. The eta prior was selected to overcome the issues of topic sparsity observed during training with different settings. The selection of 100 topics is based on Mitchell’s (2011: 117) observation that the perplexity reduction slows down once the number of topics surpasses 50 and virtually disappears with more than 100 topics.

2.2.2 Syntactic model (parser)

In addition to the semantic model, this work used a syntactic model to provide a deeper insight into the processing complexity of the natural language data. The application of syntactic models used in parsing as psycholinguistic models was suggested by Hale (2001) and further developed by Levy (2008). In this approach, the surprisal of an item is used as a proxy for the processing cost associated with a given linguistic input. It assumes that the processing costs incurred to the listener/reader by the predictability of the input can be approximated by the probabilities assigned to the input by a language model. By employing a parser, the structural predictability can be taken into account as well. This correlation has been successfully tested against the surface realization of processing complexity, reading times (Mitchell et al. 2010; Demberg & Keller 2008), and to a degree the magnitude of the N400 effect (Frank et al. 2013). The processing complexity is believed to be caused by the necessity to update prior expectations about the input every time a new input segment arrives and is incorporated (Kuperberg & Jaeger 2015).

The observed relationship between predictability and processing costs allows us to express the expectations that the human parser has in the form of a distribution over all possible continuations of the input up to a given point. This in turn allows us to measure the magnitude of the change after the update in terms of Kullback-Leibler divergence of the old distribution to the new one (Mitchell et al. 2010). Since the processing cost of a word (surprisal) is defined as the negative logarithm of its probability, it increases as its probability decreases. The highest processing cost would be incurred by items which are not allowed to appear in a given context by the syntactic rules of a language; their probability of occurrence would be arbitrarily small, making their surprisal exceedingly large.

In their survey, Roark et al. (2009) mention a number of methods used to derive the probabilities that serve as a base for the calculation of surprisal. They all share their incremental approach to parsing, mimicking the way humans perceive language. The methods listed include an Earley parser (Earley 1970), Roark incremental top-down parser (Roark 2001) and an n-best version of Nivre et al.'s (2007) incremental dependency parser. Individual studies using surprisal also differ in the level of lexicalization with which the parser operates, occasionally using lexicalized and de-lexicalized parsers alongside (Demberg & Keller 2008; Ferrara Boston et al. 2008), or bleaching the lexical information in the corpus altogether by replacing it by part-of-speech tags (Demberg & Keller 2008). A more recent methodological addition is using recurrent neural networks to estimate the probability of the observed input and approximate the surprisal without actually overtly identifying the underlying syntactic structure (Frank & Willems 2017; Frank et al. 2013, 2015).

The addition of a parser to the semantic model allows to take into account text factors which are not related to semantics and text coherence. While the semantic model can tell that the word *ocean* is more coherent with the string *There is water in the* than with *There is cocoa in the*, it is incapable of distinguishing syntactically well-formed sentences like *There is water in the ocean* from random sequences like *There water is the in ocean*. This information, on the other hand, together with the probability associated with each syntactic element, may be captured by the parser. It takes into account even those effects on the processing which are not related to the meaning but rather to the structure of the input.

The surprisal estimation in this thesis uses the Roark parser (Roark et al. 2009; Roark 2001), a broad-coverage incremental top-down probabilistic parser. This parser operates on the basis of a probabilistic context-free grammar (PCFG), analyzing a text string incrementally in a left-to right manner (for left-to-right writing systems). The tree construction employs a context-free grammar G, a representation of the syntactic rules of a language in the form $G = (V, T, P, S^{\dagger})$. There, V is a set of nonterminal symbols, T a set of terminal symbols, $S^{\dagger}$ is the start symbol (a member of the set V) and P is a set of rule productions in the form $A \rightarrow \alpha$ where $\alpha \in (V \cup T)^*$ (Roark et al. 2009). This is a rule which expands a nonterminal symbol $A \in V$ into one or more terminal or nonterminal symbols. The syntactic tree is a representation of a sequence of such rule expansions, with each individual expansion from a parent to a child being a rule in the grammar. If all leaves (that is nodes without children) of a tree consist of terminal symbols, this tree is considered complete (Roark 2001). For example, an artificial context-free grammar with the rules:

(1) $S \rightarrow NP\ VP \mid VP$

$NP \rightarrow Det\ N$

$VP \rightarrow V$

$Det \rightarrow the$

$N \rightarrow women \mid men$

$V \rightarrow walk \mid speak$

would be capable of generating sentences such as:

(2) Women walk.
Men speak.
Walk!

but sentences such as *Do women walk?* or *Women walk quickly.* would violate the grammar rules.

Probabilistic context-free grammars are distinguished from standard context-free grammars by the fact that each rule is additionally assigned a probability. Usually, these are derived from a training corpus. Thus, a PCFG G can be defined as $\mathrm{G} = (\mathrm{V}, \mathrm{T}, \mathrm{P}, \mathrm{S}^{\dagger}, \rho)$ where ρ is the function linking each rule to its probability. Concretely, the probability expresses the likelihood of the right-hand side of the rule given the left-hand side. If no probability mass is reserved for infinite trees, the PCFG is considered consistent ("tight") and the probabilities it yields belong to a proper probability distribution over completed trees (Roark 2001).

The probabilities associated with the individual rules can be used in order to calculate the prefix probability of each word sequence $w_1 \dots w_i$. This prefix probability equals the sum of probabilities of all partial leftmost derivations[3] that have word w_i as the right-hand side

[3] A leftmost derivation "begins with $S^{\dagger}$ and each derivation step replaces the leftmost non-terminal A in the yield with some α such that $\mathrm{A} \rightarrow \alpha \in \mathrm{P}$", cf. Roark et al. (2009: 326)

product of the last rule. Subsuming all leftmost derivations D that are compatible with PCFG G and string $w_1 \dots w_i$ under the set $\mathcal{D}(G, w_1 \dots w_i)$, the prefix probability can be expressed as (Roark et al. 2009: 326, Equation 2):

$$PrefixProb_G(w_1 \dots w_i) = \sum_{D \,\in\, \mathcal{D}(G, w_1 \dots w_i)} \rho(D) \qquad (7)$$

where the probability of a given derivation $\rho(D)$ is defined as the product of the probabilities of the individual rules applied at each step of the derivation. Thus, for a tree derived in m steps, we calculate its probability as:

$$\rho(D) = \prod_{i=1}^{m} \rho(D_i) \qquad (8)$$

where D_i is a derivation step, i.e. an application of one of the rules from the set V of the PCFG.

This allows us to calculate the conditional probability of any $w_i \in T$ based on the previous sequence of words $w_1 \dots w_{i-1}$ as (Roark et al. 2009: 326, Equation 3):

$$p_G(w_i | w_1 \dots w_{i-1}) = \frac{PrefixProb_G(w_1 \dots w_i)}{\sum_{w_i' \in T} PrefixProb_G(w_1 \dots w_{i-1} w_i')} \qquad (9)$$

In tight grammars, the sum of probabilities for all possible rules that contain a non-terminal $X \in V$ as their left-hand side equals $\sum_{\alpha} \rho(X \rightarrow \alpha) = 1$. Adjusting Equation 9 accordingly, the conditional probability could also be expressed as the ratio of the prefix probability of string $w_1 \dots w_i$ to the prefix probability of the preceding string $w_1 \dots w_{i-1}$, i.e. (Jelinek & Lafferty 1991):

$$p_G(w_i|w_1 \dots w_{i-1}) = \frac{PrefixProb_G(w_1 \dots w_i)}{PrefixProb_G(w_1 \dots w_{i-1})} \qquad (10)$$

Finally, knowing the conditional probability, we can calculate the surprisal as:

$$\begin{aligned} &S_G(w_i|w_1 \dots w_{i-1}) \\ &= -\log \frac{PrefixProb_G(w_1 \dots w_i)}{PrefixProb_G(w_1 \dots w_{i-1})} \end{aligned} \qquad (11)$$

In their 2009 update of the original parser, Roark et al. also introduced the option to tease apart the syntactic segment of the surprisal from the lexical one. To do this, they exploit the fact that the last derivation move for every derivation is from the POS-tag to the lexical item itself. Thus, calculating the conditional probability of the penultimate derivation (identifying the POS-tag) offers insight into the surprisal of the structure before word w_i is integrated, which will be referred to as “syntactic surprisal” from here onwards. Importantly, the mechanism here is different from the one used by Demberg & Keller (2008) when calculating their “unlexicalized surprisal.” While Demberg & Keller replaced the individual strings in the text with their POS-tags, completely excluding the lexical expansions, the syntactic surprisal only excludes the last lexical derivation. Thus, it expresses the processing load associated with the structural properties of w_i once all preceding items have been integrated. While the unlexicalized surprisal yielded mixed results when empirically tested (especially once syntactic surprisal was included as well, cf. Demberg & Keller 2008 vs. Roark et al. 2009), the syntactic surprisal was confirmed to predict the reading times (Demberg et al. 2012; Roark et al. 2009) and outperformed the unlexicalized surprisal in direct comparison (Fossum & Levy 2012).

One of the main advantages of the Roark parser for the application in this thesis is the fact that it processes the data incrementally, from left to right, producing surprisal values after each word in the sentence. Thus, it mimics the predictive cognitive mechanisms in that it does not take into account what comes after a word in determining its probability. As a result, it can simulate garden path effects in humans, assigning low probability to items which were unexpected at the moment of their occurrence.[4]

[4] The Roark parser uses beam parsing, discarding extremely unlikely parses from its list of options as it progresses through the string. Thus, the relevant parse may no longer be kept at the moment the garden path effect is revealed. In order to avoid failure, the parser uses smoothed probabilities, reserving some probability mass to assign in such a case (Roark 2001).

In this thesis, the default syntactic model distributed with the Roark parser (Roark et al. 2009; Roark 2001) was replaced by a custom one. The default model was trained on the Penn Parsed Wall Street Journal (Marcus et al. 1993). However, this corpus is not a good representation of the range of texts to which speakers are exposed in their everyday experience. Thus, another model was built with different training data. It was trained on the Penn Parsed Manually Annotated Subcorpus of the Open American National Corpus (MASC, Ide et al. 2010). MASC covers American English from 1990 to the present day and was built as a mixture of 19 genres across the formality range, containing approximately 25% of spoken data. Thus, it is closer to the input most speakers receive than the written-only, genre-restricted Penn Parsed Wall Street Journal.

From the trained models, only the syntactic surprisal element was taken into consideration: the lexical effects were captured by an n-gram model with a much better accuracy given its substantially larger training corpus as discussed in the next chapter.

2.2.3 N-gram model

The third element of the compositional language model used in this thesis is an n-gram model. N-gram models are comparably simple, yet surprisingly powerful. They are argued to be Markov models, i.e. models capable of employing a limited history of a process to give predictions about its future. In true Markov models, the quality of predictions made from a limited history should match those made with the knowledge of the full history. In this case, it means that by knowing words $w_{i-(n-1)} \dots w_{i-1}$, Markov models can predict word w_i with the same success rate as if they had access to the whole history $w_1 \dots w_{i-1}$. This claim is somewhat exaggerated for models with low n; n-gram models with high n, on the other hand, suffer from data sparsity problems, as vast training datasets are needed for a good estimation of the probabilities of rare items.

The simplest n-gram model calculates the probability through the maximum likelihood estimate:

$$p_{ML}\left(w_i \middle| w_{i-(n-1)} \dots w_{i-1}\right) = \frac{c(w_{i-(n-1)} \dots w_i)}{c(w_{i-(n-1)} \dots w_{i-1})} \qquad (12)$$

where $c(w_{i-(n-1)} \dots w_i)$ and $c\left(w_{i-(n-1)} \dots w_{i-1}\right)$ are the counts of the n-gram $w_{i-(n-1)} \dots w_i$ and its context $w_{i-(n-1)} \dots w_{i-1}$ in a training corpus. This approach has one important weakness: its value is 0 for n-grams which have not been observed yet, without taking into account the underlying frequency of the individual items. Thus, if a trigram such as *Ernest Hemingway is* does not appear in the data, the probability of the word *is* to occur will be zero irrespective of its overall frequency in the corpus.

In order to alleviate this issue, a number of smoothing approaches have been proposed. They adjust the maximum likelihood estimate by reducing very high probabilities and distributing the reserved probability mass over unseen n-grams. As a result, the overall distribution becomes more uniform. The practical implementations of this approach do not only seek to resolve the issue with zero probabilities, but usually attempt to improve the overall accuracy of the language model as well (Chen & Goodman 1999).

Possibly the simplest smoothing approach is the additive smoothing (Laplace 1814), in which a fixed amount is added to the frequency of each n-gram. This fixed amount δ is usually $0 < \delta \leq 1$; Laplace suggests $\delta = 1$ (Laplace 1814). Thus, the n-gram probability is calculated as (Chen & Goodman 1999: 363):

$$p_{ADD}\left(w_i \middle| w_{i-(n-1)} \dots w_{i-1}\right) = \frac{\delta + \mathrm{c}(w_{i-(n-1)} \dots w_i)}{\sum_{w_i}(\delta + c\left(w_{i-(n-1)} \dots w_i\right))} = \frac{\delta + \mathrm{c}(w_{i-(n-1)} \dots w_i)}{\delta|V| + \sum_{w_i}(c\left(w_{i-(n-1)} \dots w_i\right))} \qquad (13)$$

where V is the bigram vocabulary. In the example case of the non-attested trigram *Ernest Hemingway is* the numerator will no longer be 0, but rather equal to δ. The simplicity of this smoothing approach is, however, reflected in its performance. Gale & Church (1994: 189) report that it provides correct probabilities “only by happenstance, if at all.” Most current applications thus use more complex smoothing approaches which usually consist of a combination of discounting – a method to reserve some probability mass to distribute over unseen events – and back-off or interpolation – a method used to estimate the probability of unseen events by lower-order n-grams (Ney et al. 1997).

In the case of interpolation, the probability of both seen and unseen events is calculated by linear interpolation of probabilities assigned by lower-order n-gram models. It is expressed as:

$$\begin{aligned} & p_{SMOOTH}\left(w_i \middle| w_{i-(n-1)} \dots w_{i-1}\right) \\ & = \tau\left(w_i \middle| w_{i-(n-1)} \dots w_{i-1}\right) \\ & + \gamma(w_{i-(n-1)} \dots w_{i-1}) p_{SMOOTH}\left(w_i \middle| w_{i-(n-2)} \dots w_{i-1}\right) \end{aligned} \qquad (14)$$

where $\tau\left(w_i \middle| w_{i-(n-1)} \dots w_{i-1}\right)$ is the distribution of probabilities assigned by the higher-order n-gram, $\gamma(w_{i-(n-1)} \dots w_{i-1})$ a scaling factor chosen so that the conditional probabilities sum to one and $p_{SMOOTH}\left(w_i \middle| w_{i-(n-2)} \dots w_{i-1}\right)$ is the probability assigned by a lower-order n-gram model (Chen & Goodman 1999). The definition may contain recursion, smoothing $p_{SMOOTH}\left(w_i \middle| w_{i-(n-2)} \dots w_{i-1}\right)$ by $p_{SMOOTH}\left(w_i \middle| w_{i-(n-3)} \dots w_{i-1}\right)$ etc. until the unigram is reached. Given how the smoothed probability is defined, unseen n-grams have their probability assigned by the lower-order n-gram model only, since their probability in $\tau\left(w_i \middle| w_{i-(n-1)} \dots w_{i-1}\right)$ equals zero. Items with non-zero frequency have their probability smoothed by the lower-order model.

This is a crucial difference to back-off-based smoothing algorithms, which employ information from lower-order models only in cases where the higher-order model does not contain the n-gram in question and returns a probability of zero. Thus, back-off models could be described as ($w_{i-(n-1)}^{i-1}$ is equivalent to $w_{i-(n-1)} \dots w_{i-1}$ and used here for compactness):

$$p_{SMOOTH}\left(w_i|w_{i-(n-1)}^{i-1}\right) = \begin{cases} \tau\left(w_i|w_{i-(n-1)}^{i-1}\right) & \text{if } c(w_{i-(n-1)}^{i-1}) > 0 \\ \gamma(w_{i-(n-1)}^{i-1})p_{SMOOTH}\left(w_i|w_{i-(n-2)}^{i-1}\right) & \text{if } c(w_{i-(n-1)}^{i-1}) = 0 \end{cases} \quad (15)$$

In the last decades, multiple smoothing approaches have been developed, including those suggested by Ney et al. (1994), Witten & Bell (1991), Katz (1987) or Jelinek & Mercer (1980). The present thesis uses the smoothing approach developed by Kneser & Ney (1995) and modified by Chen & Goodman (1999). This model, unlike the original Kneser-Ney smoothing (which was a back-off algorithm), uses an interpolated smoothing approach, in which the probability of word w_i given $n-1$ of its predecessors is:

$$p_{SMOOTH}\left(w_i\middle|w_{i-(n-1)}^{i-1}\right) = \frac{c\left(w_{i-(n-1)}^{i-1}\right) - D\left(c\left(w_{i-(n-1)}^{i-1}\right)\right)}{\sum_{w_i} c(w_{i-(n-1)}^{i-1})} + \gamma(w_{i-(n-1)}^{i-1})p_{SMOOTH}(w_i|w_{i-(n-2)}^{i-1}) \quad (16)$$

where D is the discounting modifier that shifts some of the probability mass to be assigned by the lower-order models. The Chen-Goodman modification of Kneser-Ney smoothing uses several discounting modifiers D, depending on the frequency with which the context used for prediction occurs in the data. These are defined as

$$D(c) = \begin{cases} 0 & \text{if } c = 0 \\ D_1 = 1 - 2Y\dfrac{n_2}{n_1} & \text{if } c = 1 \\ D_2 = 2 - 3Y\dfrac{n_3}{n_2} & \text{if } c = 2 \\ D_{3+} = 3 - 4Y\dfrac{n_4}{n_3} & \text{if } c \geq 3 \end{cases} \qquad (17)$$

where $Y = \frac{n_1}{n_1+2n_2}$ and n_x is the number of distinct n-grams with count x, i.e. n_1 is the number of n-grams occurring once (Chen & Goodman 1999). Thus, for example, the discounting modifier used for unigrams would be:

$$D_1 = 1 - 2\frac{n_1}{n_1 + 2n_2}\frac{n_2}{n_1} = 1 - \frac{2n_1n_2}{n_1^2 + 2n_1n_2} \qquad (18)$$

Given that the influence of n_1 grows exponentially, the larger the number of unique n-grams, the larger the discounting modifier.

Chen & Goodman (1999) showed that such a discounting procedure performs better than additive smoothing, back-off models or interpolation models with one discounting parameter only. Even though their model has been since superseded in terms of perplexity over test set and accuracy of word prediction by more recent developments in language modelling (Tang & Lin 2018; Mikolov et al. 2010), it was selected for this thesis as there is clear evidence that its probability estimates are representative of processing complexity (Balling & Kizach 2017; Frank 2013, 2017). There are even suggestions (Frank et al. 2015) that the modified Kneser-Ney smoothed n-gram model is a better predictor of predictability effects on human brain activity and outperforms newer models in this domain; the evidence is not conclusive, though (cf. the opposite result in Frank et al. 2013).

For this thesis, a 3-gram model with Chen-Goodman modified Kneser-Ney smoothing and back-off (Chen & Goodman 1999) was trained using the SRILM modelling toolkit (Stoelcke 2002). The training employed the data from COCA. Given its 400 million words, data sparsity should not be a large issue; still, smoothing and back-off were used in order to improve the performance.

2.3 Language model

Ultimately, the individual elements need to be combined in some way in order to yield a single numeric expression of surprisal. The simplest way of combining individual probabilities would be through linear interpolation, that is:

$$p(w_i|h_{i-1}) = \lambda p_1(w_i|h_{i-1}) + (1-\lambda)p_2(w_i|h_{i-1}) \quad (19)$$

where h_{i-1} is the text history from which the probability of word w_i is estimated, e.g. $w_{i-(n-1)} \dots w_{i-1}$ for the n-gram model. This approach has the advantage of providing valid probabilities without any further transformation. However, it is only suitable for combining models that are equally strong and have complementing strengths and weaknesses. In cases where one of the models is much stronger than the other, however, the result will be a model of intermediate strength as the weak model will deteriorate the predictions of the strong one (Mitchell 2011).

Thus, linear interpolation is possible in the case of the syntactic model and the n-gram model, but might deteriorate the performance in the case of the semantic model (Mitchell 2011). Coccaro & Jurafsky (1998: 2405) seem to be aware of the issue, noting:

> We found simple linear combination to be inadequate […], partly because the LSA estimator often predicts words that are syntactically disallowed. We need a non-linear combination function that gives a much higher probability when the two models agree — that is, when the predicted word is both syntactically and semantically likely — and gives a low probability if either estimator believes a word unlikely.

In their work they decided to resolve this problem by using the geometric mean as the combining function. Such a method ensures that the final probability is high only in cases where both models agree that the word is likely to occur. If only one of them assigns a high probability, the final probability remains low. However, because of the non-linear transformation, this approach fails to generate valid probabilities: the probability distribution over the full vocabulary will no longer sum to 1.

The method used in this thesis and based on Mitchell (2011) is slightly different. It uses a rescaling approach (Gildea & Hofmann 1999; Kneser et al. 1997) based on the cosine similarity measure. It draws on the expression of semantic probability as the product of unigram probability $p(w_i)$ and a semantic modifier Δ expressing the similarity of the item to the context in which it appears. Starting from the equation:

$$p(w_i|h_{i-1}) = p(w_i) \cdot \Delta(w_i, h_{i-1}) \tag{20}$$

where

$$\begin{aligned}\Delta(w_i, h_{i-1}) &= \sum_k \frac{p(c_k|w_i)}{p(c_k)} \frac{p(c_k|h_{i-1})}{p(c_k)} p(c_k) \\ &= \sum_k \frac{p(c_k|w_i)\, p(c_k|h_{i-1})}{p(c_k)}\end{aligned} \tag{21}$$

derived from the LDA-based language model, Mitchell assumes that the unigram probability may be replaced by an n-gram version in order to combine the n-gram model with its semantic counterpart. Thus,

$$\hat{p}(w_i) = p(w_i|w_{i-(\mathrm{n}-1)}^{i-1}) \cdot \Delta(w_i, h_{i-n}) \tag{22}$$

Two notes must be made on Equation 22. First, it assumes that the history is conditionally independent of $w_{i-(n-1)}^{i-1}$ (Mitchell 2011: 111). In order to ensure that as far as possible, the semantic modifier is only constructed from elements outside of the n-gram scope. To achieve this, history h_{i-n} is taken to evaluate w_i.[5] Secondly, $\hat{p}(w_i)$ needs to be

[5] A complete independence of h on the direct context of w_i is unattainable, however excluding the overlap should at least minimize the relationship.

normalized over the content words in order to yield valid probabilities (Mitchell 2011). This is done in the following manner:

$$p(w_i|w_{i-(\mathrm{n}-1)}^{i-1}, h_{i-n}) = \hat{p}(w_i)\frac{\sum_{w_c} p(w_c|w_{i-(n-1)}^{i-1})}{\sum_{w_c} \hat{p}(w_c)} \qquad (23)$$

where the sum over w_c is the sum over content words only, as function words are not affected by the rescaling. The rescaling allows the n-gram model to express short-range dependencies while the long-range dependencies are encoded in the semantic modifier. Thus,

$$p_{3-GRAM}(w_i|h_{\mathrm{i}-1}) = \begin{cases} p_{3-GRAM}\left(w_i\middle|w_{i-2}^{i-1}\right) \cdot \Delta(w_i, h_{i-3}), & \text{if } w_i \in W_c \\ p_{3-GRAM}\left(w_i\middle|w_{i-2}^{i-1}\right), & \text{if } w_i \in W_f \end{cases} \qquad (24)$$

where W_c and W_f are the sets of content and function words respectively.

As the last step in creating the combined model, this semantically-rescaled n-gram probability is combined with the syntactic probability estimated by the Roark parser. This is done through simple linear interpolation, as used both by Roark and Mitchell. The linear interpolation is made possible and necessary in this case by two properties of the models, noted by Mitchell. First, the predictive strengths of n-grams and the parser do not differ substantially. Second, renormalizing across the whole vocabulary would be necessary in the case of other combining functions, which would render the approach impractical (Mitchell 2011: 112).

The mixing of probabilities was done using the linear interpolation (based on Equation 19):

$$p(w_i|h) = \lambda \times p(w_i|h)_l + (1 - \lambda) \times p(w_i|h)_s \qquad (25)$$

where $p(w)_l$ is the probability assigned by the n-gram model and $p(w)_s$ is the probability extracted from the parser. The weighting factor λ was set at 0.36 (following Roark 2001), assigning stronger influence to the syntactic model. Afterwards, following Equation 1, the calculated probability was log-transformed (base *e*) and multiplied by -1 to yield the per-word surprisal in nats.[6]

This combined model has been suggested to perform well in predicting reading times (Mitchell 2011; Mitchell et al. 2010) and was successfully used in the prediction of pronunciation durations (Sayeed et al. 2015). Nevertheless, in spite of the intuitive appeal of having one measure which includes and combines the individual models, the scores combined in the model were kept and evaluated separately as well. This was done with the motivation of gaining additional knowledge about their importance to the task at hand.

[6] Nats are units of information yielded by transforming the probability of an event by the natural logarithm. Other common bases for the log-transform are 2 (producing bits) and 10 (producing hartleys).

Chapter 3
Disfluencies

Over the past few decades, the status of speech disfluencies changed from non-linguistic phenomena that were usually omitted in the analysis to a phenomenon tightly connected to the way we produce and perceive speech. If previously any disfluency in speech was argued to be filtered out before the input reached the human parser, this view became unfeasible in the light of evidence. An example by Ferreira & Bailey (2004: 232) illustrates this fact:

> [C]onsider again the […] example: "That Vermeer – uh where is 'The Love Letter' um what museum is it is it in". Notice that the pronoun *it* in the repair must find its antecedent *The Love Letter* in the reparandum, the part of the utterance that was spoken in error. If filtering were the correct solution, the pronoun would not have an antecedent, but clearly people interpret the utterance as if it does.

Similar argumentation is presented by Core & Schubert (1999: 413) drawing on the example "have the engine take the oranges to Elmira, um, I mean, take them to Corning" where the referent of *them* would be removed by filtering, too. Such examples demonstrate that even disfluent passages are perceived as parts of the input by the listeners.

Moreover, since language comprehension operates incrementally, it does not wait for an utterance to be finished. Due to this, there is no chance that the input is cleaned of all disfluencies before being presented to the parser in an ideal print-like form. Not every disfluency can be recognized as such at the moment of its utterance. This includes most repairs where the reparandum fits into the preceding input both by its form and meaning. Consider again the example (Core & Schubert 1999: 413):

(3) …have the engine take the oranges to Elmira, um, I mean, take them to Corning…

At the moment of its utterance – as long as both Elmira and Corning are existing train stations and neither of them is disqualified by the previous context – there is no way of recognizing that the word *Elmira* is not the intended item. This only becomes obvious after it has been naïvely incorporated into the representation of the input.

In contrast to the examples above, hesitations are substantially easier to recognize as disfluencies rather than fluent input. Silent pauses, *um* and *uh* are not produced as a mistake that is afterwards corrected. Even repetitions can be identified fairly easily and rapidly (50 ms after the utterance, MacGregor et al. 2009). Thus, they are substantially easier for the cognitive system to identify and could be completely filtered out before the input is passed to the parser. This led researchers to propose several hypotheses about their origin and purpose. The following paragraphs will summarize the prominent ones.

3.1 Placement and purpose

In his model of speech production, Levelt (1983, 1989) argues that hesitations are a symptom of production difficulties. From this perspective, they appear due to some error in the process of translating the message to be sent into a set of commands to the articulatory organs. Because speakers constantly monitor their planned production prior to its realization by the articulators, they are capable of catching some of these errors before they are uttered. In Levelt's terms, the error occurs in the macroplanning stage in which the information is retrieved and the contents of the utterance-to-come are created and ordered before the microplanning stage converts them into the phones to be pronounced. As a consequence, there is no surface realization of the original error. However, since the correction of unspoken (or covert) errors is similar to the correction of overt errors in that it requires additional processing time, a hesitation is produced because the speakers are incapable of proceeding.

This hypothesis had some support from previous work by Dell (1980), showing that speakers are capable of monitoring their inner speech for errors. Similar suggestions, viewing disfluencies as a consequence of the unavailability of fluent output were made by researchers suggesting that disfluencies occur when speakers are still searching their memory for a word (Goodwin 1987; Harness Goodwin & Goodwin 1986) perhaps due to its low frequency or contextual probability (Beattie & Butterworth 1979). Alternatively, they could reflect speaker's uncertainty about the truthfulness of the message

(Smith & Clark 1993). Overall, these perspectives view disfluencies as a “symptom” (Clark & Fox Tree 2002) of speaker’s inability to proceed due to still being engaged in the “speech-productive labor” (Goffman 1981).

These observations suggest that hesitations may be a consequence of speaker’s need for more time before proceeding with the utterance. This is corroborated by studies observing a positive correlation between the effort needed for the macroplanning of the upcoming utterance and the relative frequency of disfluencies (Oviatt 1995). Evidence for such claims is found in empirical research showing that tasks which are more cognitively demanding elicit more disfluencies. Thus, for example, the description of a cartoon – a task comparably simpler than its interpretation – contains fewer silent pauses (Goldman-Eisler 1968). Similarly, description of a familiar route requires less planning than description of an unfamiliar one and results in fewer unfilled pauses (Good & Butterworth 1980).

The view of disfluencies as symptoms was criticized as oversimplifying by researchers who argued that disfluencies have a purpose and thus function as signals rather than symptoms. The proposed functions of disfluencies included holding the floor (Rochester 1973; Maclay & Osgood 1959), expressing the speaker’s mental state (Brennan & Williams 1995), or forewarning the listeners that they should expect an unfamiliar referent (Arnold et al. 2007), prepare their own utterance (Jefferson 1974) or aid the speaker in finishing the current utterance (Harness Goodwin & Goodwin 1986).

Furthermore, it was previously argued that the use of hesitations provides various hints as to what is coming in the conversation: an unpredictable/uncommon word (Schnadt & Corley 2006; Beattie & Butterworth 1979), a long/short delay (Clark & Wasow 1998), something complex (Watanabe et al. 2008; Arnold et al. 2007) or something new to the conversation (Barr & Seyfeddinipur 2010). More generally, disfluencies have been claimed to announce "the immediate initiation of what is expected to be a minor, or major, delay in speaking" (Clark & Fox Tree 2002: 92).

The arguments for the communicative motivation of disfluencies have found support in studies showing their beneficial impact on processing. This has been demonstrated through faster response times and gaze/mouse movement towards the unusual/new/unpredictable (Owens et al. 2018; Bosker et al. 2014; Barr & Seyfeddinipur 2010). Similar suggestions were made by neuroimaging studies. Corley et al. (2007) observed an attenuation of the N400 effect that is usually associated with the processing of unpredictable items, if these items were preceded by a disfluency. The influence went even beyond the immediate processing: words preceded by a filled pause (in the case of Corley et al. the transcription *er* is used) performed better in a subsequent memory test (similar results were obtained by Fraundorf & Watson 2011). This not only suggests that words after filled pauses are processed differently from other words, but provides evidence that the "ERP differences are not due to contamination of the N400 waveform by spillover effects from the processing of the *er* [itself]” (Corley et al. 2007: 666).

Importantly, although disfluencies may perform a communicative function, this is not fully dependent on their surface form. Rather, at least a part of the effect is likely to be related to the temporal disruption of the production. Such an argument would follow from the observation of Bailey & Ferreira's (2003), who were able to replicate some of the effects even with pauses consisting of non-speech sounds. Similarly, Eklund et al. (2015) demonstrated potential partial equivalence of filled and unfilled pauses by observing the brain during these phenomena. They conclude that filled and unfilled pauses are similar in that they equally affect listener’s attention. However, this result should not be taken as a proof for their complete identity. Rather, it was observed (ibid.) that while filled pauses modulated motor areas of the brain, unfilled pauses did not produce a matching effect. They modulated the syntax processing areas and when compared to filled pauses and fluent speech, they were much closer to the latter. To summarize, while some distinction between hesitations with and without articulatory realization seems to exist, there are effects which are shared by both types. Intuitively, these effects should be related to the fact that disfluencies provide additional processing time without busying the listener with resolving whether a valid lexical input has been received, i.e. whether this input is not disqualified by the context.

In this respect, one of the examined classes of disfluencies, disfluent repetitions, is somewhat different from the rest. First, they are not as easily discernible from the genuine input by their phonology. Thus, even if filled pauses were removed by a simple phonetic filter, repetitions would still affect the parsing procedure. Secondly, there is a growing body of evidence suggesting that they are speaker-related and symptomatic of production problems rather than providing cues to the listener. This argument requires discussion; after all, repetitions do introduce additional time into the processing of input in a way similar to pauses. During this time no new input is incoming and the listeners only need to incorporate what was said before. Thus, the processing of repetitions could be similar to that of filled/unfilled pauses and yield positive effects on comprehension. Such a view is even more appealing given that repetitions are recognized much faster than other lexically legal, but contextually problematic material. While other non-fitting lexemes trigger the P600 effect with an onset at 200 ms (Kutas et al. 2006), repetitions elicit a relative positivity starting as early as 50 ms after the utterance (MacGregor et al. 2009: 42, 44). With such a quick recognition, one might expect the processing to be improved by a temporary workload decrease: for a short span of time, the cognitive system should only be "catching up" on the previous input, not having to process and incorporate new items. Such a scenario should render repetitions beneficial to the listener just like filled/unfilled pauses.

This is likely not the case. In an experiment measuring event-related potentials, MacGregor et al. (2009) assessed the effect of repetitions on the human brain. They report an early positivity (between 100 and 400 ms after the stimulus) indicating that repetitions are not filtered out and that their presence introduces additional complexity for the listeners. While this shows that repetitions are perceived as more complex than fluent items, it would not necessarily disqualify them from helping the subsequent processing. However, a subsequent N400 effect, caused by an unpredictable word uttered after the repetition, is not mitigated by the additional time given to the listener for processing the input up to that point. Thus, there is likely no benefit for the listener. Rather, the subsequent positivity (600-900ms) points in the opposite direction: the listeners might be resolving additional difficulties in parsing continuation when faced with a disfluency (MacGregor et al. 2009).

Such an observation supports the analysis of Lake et al. (2011) suggesting that repetitions are not used in order to create an anchoring point to the point before the disfluency occurred. Rather, they may be an automatic outcome of correcting problems in own speech as argued by Clark & Wasow (1998) in their continuity hypothesis: it is easier for a speaker to return to a continuous output by producing a full constituent rather than only its fragment.

The speaker-orientation of repetitions is further supported by studies on individuals with autism-spectrum disorders (ASD) summarized in Table 2. The studies reviewed by Engelhardt and colleagues (2017) as well as their own paper show that participants with ASD are uniformly distinguished from their typically-developing counterparts by a substantial overuse of repetitions. Since "individuals with high-functioning forms of autism spectrum disorders (HFA) tend to have a self-centric approach to dialogue and poor pragmatic skills" (Engelhardt et al. 2017: 2885) and "often do not have language impairments per se but do have impairments in pragmatic aspects of language use, as well as atypical prosody" (ibid.), it is assumed that their overuse of certain language features is a sign that these features are not helpful (or are disturbing) to the listeners. This is clearly suggested for repetitions. On the contrary, the overuse of pauses either lacks conclusive evidence (unfilled pauses), or is completely unattested (filled pauses).

Study	***N*** **ASD:TD**	**Filled**	**Unfilled**	**Repetitions**	**Repairs**	**Task**
Irvine et al. (2016)	24:16	TD	NA	NA	NA	Monologue, painting descriptions
Lake et al. (2011)	13:13	TD	ASD	ASD	TD	Dialogue, question answering
Shriberg et al. (2001)	30:53	NA	ASD	ASD	ASD	Dialogue, ADOS interview
Suh et al. (2014)	15:15	NS	NA	ASD	ASD	Monologue, story telling
Thurber & Tager-Flusberg (1993)	10:10	NA	TD	NS	NS	Monologue, story re-telling

NA: Not analyzed, NS: not significant, ASD: Significantly more common in ASD-individuals, TD: significantly more common in typically developing individuals

Table 2: Disfluencies in individuals with ASD compared to controls (adapted from Engelhardt et al. 2017: 2887).

To conclude, there is both evidence suggesting that disfluencies may help listeners with processing linguistic input and that they are disruptive. They may originate on the side of the speaker, as a symptom of retrieval problems with infrequent words or planning a complex syntactic structure to be produced shortly. Alternatively, they may be directed as helpful cues for the listener, preparing them for an unpredictable continuation (on the basis of the context or of its general rarity). Unanimous consensus has so far not been reached. Importantly, many of the suggested triggers should be reflected in the local surprisal value to some degree. On the one hand, it is an expression of the difficulty with which an item is processed and incorporated into the mental representation. On the other hand, it may also offer insights into potential retrieval and planning issues, as suggested by Cole & Reitter (2017), especially if production and comprehension are tightly interwoven, as advanced by Pickering & Garrod (2013). Furthermore, it reflects the predictability of an item in context, its overall frequency and – to the degree the semantic model can capture it – its newness. Thus, if all these features are claimed to predict disfluencies, the combined estimate of surprisal should predict them, too.

Additionally, by extension of the UID hypothesis, the magnitude of the local change in information density should aid the prediction as well. Such an expectation is motivated by the following logic: an information-theoretically optimal communication transmits a constantly dense stream of information from the speaker to the listener, arbitrarily close to the channel capacity. A sudden positive spike in the information transmission rate might cross the boundary of what the speaker can plan/retrieve/encode online prior to articulation, or the listener's capacity to process such an input. If this spike goes beyond the level that can be corrected by modulations of articulatory redundancy, it may require the insertion of additional time into the transmission. This additional time may be provided by optional syntactic elements, as argued in the original UID (Jaeger 2006, 2010), or by inserting hesitations, if such an optional element is not available/sufficient, as argued in this thesis.

3.2 Disfluencies from the computational perspective

In the last few decades, computational linguists developed algorithms for the production of synthesized speech that do not simply concatenate the pronunciations of individual words in the text. Suprasegmental phenomena, such as intonation, pitch or assimilation across lexeme boundary have found their way into the individual speech models, creating a more natural output. Nevertheless, up until recently, the main focus of this research area was on the production of fluent, read-aloud-like speech, not too different from the speech of news anchors reading from a teleprompter. The creation of conversationalist synthesized speech was outside the mainstream, largely due to the fact that most applications of speech synthesis did indeed consist of reading aloud written texts – articles, messages, bus stop names. For such purposes any noise in the signal was not desirable. This obviously included disfluencies.

With the advent of technologies such as digital personal assistants, this approach changed. These assistants are not supposed to simply read prefabricated phrases. Rather, they are trained to mimic natural conversation, receiving commands in natural human language and responding in a way that simulates a human counterpart. For such applications, it is necessary to create a natural sounding output, including phenomena deemed undesirable in an idealized view of the language, such as disfluencies. Moreover, disfluency processing must be included both in the signal receiver as well as in the signal transmitter. Despite the fact that the disfluency handling logic would treat the same phenomenon on both ends, research in this area has mostly focused on identifying – and filtering out – disfluencies in the input, rather than predicting them in the output. The motivation of this priority is clear – humans are still superior to machines in handling imperfect natural language data and have more robust repair mechanisms available. The following chapters will discuss the approaches suggested both for disfluency detection and prediction in order to show their conceptual similarities as well as the different challenges faced in each of these tasks.

3.2.1 Disfluency detection

A broad variety of approaches was adopted for the detection and removal of identified disfluencies. These include, but are not limited to:

- Conditional random field (Cho et al. 2013; Fitzgerald et al. 2009b)

- Modified statistical machine translation techniques (Cho et al. 2014a; Honal 2003)
- Neural networks (Cho et al. 2015)

The underlying mechanism in disfluency detection and correction often leans on the Noisy Channel Approach, stemming from Shannon's Mathematical Theory of Communication. This approach, visualized in Figure 3, assumes that each unit in speech is created as a fluent one and then transmitted through a noisy channel which adds the disfluencies before passing the unit to the output. Such an assumption may not be a psycholinguistically realistic representation of the speech producing processes: it would disqualify disfluencies caused by retrieval errors or delays. Similarly, on utterance level, the aforementioned research (Good & Butterworth 1980; Goldman-Eisler 1968) shows that not every utterance is fully planned before the onset of speaking. Instead, the planning may be incremental, too. However, the Noisy Channel approach allows a simple and efficient description of the process of disfluency removal.

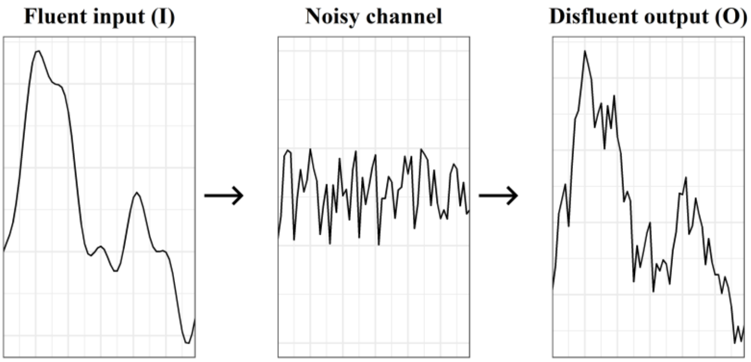


Figure 3: Visualization of the Noisy Channel Approach. The fluent string I is transmitted through a noisy channel yielding the disfluent string O.

If namely the speaker chooses to utter a fluent string I which is transformed by a noisy channel into a disfluent string O via a pass-through filter, then the noisy channel can be described as mapping the inputs to the outputs via a matrix of $P(O|I)$. In such a case, in order to reconstruct the original fluent string, we only need to find and train a probability measure to score candidate strings $\hat{I}_{1\ldots k}$ selecting the most likely fluent counterpart $\hat{I}$ of the disfluent string O in following manner:

$$\hat{I} = \underset{I}{\operatorname{argmax}}\ P(I|O) \tag{26}$$

The additional advantage of such a view is that it is consistent with Ferreira & Bailey's (2004) critique of simple filtering approaches claiming that $\hat{I}$ can be produced via a function which will simply strip anything deemed disfluent. If that was indeed the case, disfluency removal would not need to deal with missing references or left out information. However, as the human parser does operate with the contents of disfluencies, an efficient disfluency removal setup must be capable to extract the information contained in them, too. Thus, rather than performing a simple clean-up, speech reconstruction is usually more desirable as "more complex and less deterministic changes are often required for generating fluent and grammatical speech text" (Fitzgerald et al. 2009a: 256). One of the main challenges for speech reconstruction is the efficient training of the estimator. Unlike simple clean-up, it cannot be trained on a corpus of well-formed texts and then consider anything that violates the rules extracted from this corpus a disfluency to be removed. In addition to a disfluency detector, a disfluency repair mechanism needs to be implemented and taught how to remove a given disfluency. This requires either a database of rules for various types of disfluencies or training on a parallel corpus containing a fluent track aligned with the disfluent one.

The availability and limited size of such aligned corpora is one of the limiting variables in the research of disfluency reconstruction. While disfluency detection has reached high efficiency, even on the harder-to-detect disfluencies, such as repairs, discourse markers and interruptions (summarized in Table 3), reconstruction is still in an earlier stage of development.

One of the added problems that disfluency reconstruction needs to face is the low interrater agreement between human subjects cleaning transcripts of disfluencies. In this respect, Fitzgerald & Jelinek (2008) showed that two annotators produce exactly the same reconstructed fluent strings only in 57% of the compared cases. Thus, it is hard to find a gold standard according to which a disfluency should be repaired. Despite this challenge, mechanisms for speech reconstruction are one the key elements for the future development of various other systems which currently depend on receiving well-formed input: parsers, taggers, machine translation systems.

Model	**F_1-score**
Yoshikawa et al. (2016)	62.5
Johnson & Charniak (2004)	79.7
Johnson et al. (2004)	81.0
Rasooli & Tetreault (2013)	81.4
Qian & Liu (2013)	82.1
Honnibal & Johnson (2014)	84.1
Ferguson et al. (2015)	85.4
Zwarts & Johnson (2011)	85.7
Zayats et al. (2016)	85.9
Jamshid Lou & Johnson (2017)	86.8

Table 3: F-scores for disfluency detection on the Switchboard corpus. Adapted from Jamshid Lou & Johnson (2017: 551).

3.2.2 Disfluency prediction

The importance of disfluency detection and correction in computational linguistics pushed the attempts of disfluency prediction somewhat out of the spotlight. After all, perfectly fluent synthesized speech can be understood by human listeners without major issues. Thus, the need to predict the occurrence of disfluencies only came with the need to increase the naturalness of the produced speech while simultaneously attempting to minimize the demands that listening to synthesized speech poses on the comprehenders.

Additionally, by creating disfluent synthesis, researchers attempt to exploit the psycholinguistic benefits of hesitations, described in the previous chapter. The improvement of reaction times after hesitations described by Fox Tree (2001, 1995) is commonly cited as the inspiring stimulus. Moreover, the last years have also seen an increase in attempts to create synthesis systems capable of manipulating the expressivity and emotions of the synthesized speech (Andersson et al. 2012; Andersson et al. 2010). These efforts ultimately aim to create speech synthesis systems capable of simulating various personality types by conveying emotional and psychological states. For such efforts, disfluency prediction is also required.

The current attempts at disfluency synthesis can be divided into two branches according to their methodology. One branch inserts the disfluencies in concatenative speech synthesis on the basis of the underlying fluent sentence (Adell et al. 2007). The other approach (Andersson et al. 2012) uses Hidden Markov Model synthesis and treats disfluencies as regular words in the speech stream (Dall et al. 2014a) which are only synthesized if they are included in the original sequence. As a consequence, it does not predict the occurrence of disfluencies, it only synthesizes their phonetic representation. Both of these approaches match state-of-the art synthesis systems without disfluency insertion in their naturalness rating. In addition, they are shown to outperform these systems in terms of perceived conversationality (Dall et al. 2014b) and are claimed to be preferred by the comprehenders over fluent-only systems (Adell et al. 2007). On the other hand, in a psycholinguistically motivated study, Dall et al. (2014b) were not able to replicate the effects that natural disfluencies have on language processing with the help of synthetic disfluencies. Though the explanation for this discrepancy is not clear, it illustrates that the ability with which we are able to synthesize disfluencies is far from perfect.

Being among the pioneers of disfluency prediction, the work of Adell et al. (2007) achieved substantial success early. In their combination of probabilistic language modelling with a decision tree algorithm, they used the following set of items to train a decision tree classifier to identify words after which a hesitation should occur:

- Word w_i
- POS-tag of w_i, as well as its close context (w_{i-1}, w_{i+1})
- Probability of w_i given the preceding context h_i
- Probability of the word w_{i+1} given the context h_{i+1}
- Probability of a filled pause to occur after w_i
- Candidate (explained below)

The fourth variable can be viewed as a way of capturing Schneider's (2014) observation that filled pauses tend to appear outside of chunks[7] – items which are likely to occur due to their preceding context are also more likely to be parts of a chunk. As a consequence, they should be less likely to be preceded by a disfluency. On the other hand, if there is a chunk boundary between words w_i and w_{i+1}, the likelihood of disfluency occurrence in that location should increase as its retrieval should require more effort.

Two elements of the approach used by Adell et al. are worth further discussion. First, while most of the psycholinguistic evidence hints on the connection between a disfluency and the output following it, their predicting mechanism focuses on the words before the location where the disfluency may be inserted. Second, they operate with the concept of a candidate: a word which allows a filled pause to follow it. In Adell et al.'s approach, only those tokens (between 30 and 40, depending on the setup) which are most frequently followed by disfluencies are considered candidates and further processed by the algorithm. If a word is not a candidate, it will not be processed any further. On the one hand, this step decreases the computational complexity by removing most words from the processing. On the other hand, it means that evaluation is done only for those items for which

[7] Schneider's definition of chunks is based on a usage-based approach to language. Thus, for her, a chunk is "a mentally represented multi-word unit" Schneider (2014: 2).

the system is fairly certain that they can be followed by a filled pause. This might have helped the precision of the prediction (96.7% in the best model). The trade-off however, is the recall (57.7%), as many words followed by a disfluency were not considered at all. Additionally, the use of such an item increases the demands on the training data, since a large corpus of transcribed speech data needs to be used to calculate the probabilities of individual types to be followed by a disfluency.

Finally, the classification attempted by Adell et al. is a two-way distinction: filled pause/no filled pause. As a consequence, their results might be outperforming other models because this approach is less sensitive to the ability to predict individual realizations. Such an attempt was made e.g. by Ohta et al. (2008). In their work on filler prediction, they presented a pipeline system consisting of a filler inserting model and a filler selector, predicting 51 different filler types. Similarly to Adell et al. they worked with fluent input strings into which the fillers were inserted. The filler inserting model was based on a language model built using a conditional random-field model trained on a corpus of transcribed speeches. From this corpus, the model calculated the probability of a filler to occur after a certain context (in Ohta et al's case 2-word context). It substantially outperformed their own implementation of uni- and trigram-based hidden Markov models in predicting locations of fillers (F_1 of 0.26 compared to 0.05 and 0.14).

Items selected by the filler inserting model were passed further in the pipeline to the filler selecting model (a unigram or a trigram model) which selected the most likely type of filler to follow. This part showed a considerably worse performance with the highest F_1 achieved being 0.06. One of the possible reasons for such a result may be the fact that Ohta et al. trained their model on a different language domain than the one from which the test data came.

Similarly to Adell et al. (2007), a more recent study by Dall et al. (2014a) attempts to predict the occurrence of disfluencies per se, specifically filled pauses. Training multiple models (4-gram, recurrent neural network, support vector machine, decision trees) on a range of corpora, they were able to correctly identify the position of a filled pause in more than 50% of the test sentences which were specifically selected to contain exactly 3 filled pauses. Unlike Adell et al. they did not employ a limited range of insertion points (the aforementioned candidates), but rather attempted a general prediction.

A recent addition was made by Qader (2017) who attempted to predict pauses and repetitions in the Buckeye Corpus (Pitt et al. 2005) by a model using conditional random field. There, the prediction of each disfluency type was done by a separate function. The fluent utterance was first processed by the repetition-predicting function and then by the pause-predicting one. The performance of the functions was not identical: while the F_1-score of repetition prediction over the test set was 9.2%, pause generation was much more successful, with an F_1-score of 25.1%.

To conclude, a substantial improvement in disfluency prediction is still required. So far, the best results were obtained by studies that sought to predict a limited number of disfluency types at a limited number of insertion points. A complete model of disfluency insertion thus remains as a challenge for future development, to which this thesis aims to contribute. To do so, it first presents a pilot study (Study I), motivating the selection of surprisal as a predictor for the task at hand. Afterwards, three studies are presented, attempting disfluency prediction in a text which was previously cleaned of disfluencies.

Chapter 4

Study I: Linking surprisal and disfluencies

The pilot study presented in this chapter provides the foundation for the use of the local surprisal estimate in disfluency prediction. Moreover, it suggests that the individual disfluency subtypes also have different profile in terms of surprisal at the location where they are inserted.

Even though some of the studies discussed in the previous chapter suggested that disfluencies may be related to phenomena that are captured by surprisal, such as the rarity of words or their unpredictability in a given context (which may or may not be caused by their low frequency), so far there has been no study verifying the exact relationship between surprisal and disfluency occurrence. Thus, in order to validate the empirical grounding for the disfluency prediction approach used in Studies IIa and IIb, the study described in this chapter was carried out.

This study used a corpus of transcribed speech and evaluated it using the language model described in 2.2 and 2.3. Afterwards, the profile of disfluencies was assessed in order to verify the following hypothesis:

Hypothesis 1:

The occurrence of disfluencies may be predicted by the local surprisal.

Concretely, it was predicted that disfluencies should occur in locations of high surprisal. In such a case, depending on their cause, they could either be symptoms of the increased cognitive load associated with the production of high-surprisal items, or they could be listener-directed items, inserted into the speech stream in order to provide the listener with additional time to resolve high-surprisal input or warn them about an upcoming unpredictable item. Thus, in a way, disfluencies could serve as smoothing agents following the predictions of the Uniform Information Density hypothesis (Jaeger 2006) for locations where no optional syntactic element is available, or where the production of a properly smoothed utterance is not possible due to time pressures. By lowering the average information transmission rate, they could increase the likelihood of a successful transmission.

The presence of time pressure is one of the prominent features of spoken language production. It is also one of the factors linking the speaker and listener-oriented views of disfluencies. Next pages will briefly discuss this link. Additionally, other specifics of the spoken language production will be mentioned in order to show their connection to disfluency use, both from the speaker- and listener-related perspective.

4.1 Speaker-related constraints of spoken language production

Even though both the planning of speech and text use identical resources on multiple levels, sharing e.g. their representations of syntax and lexicon (Cleland & Pickering 2006; Allen & Badecker 2002; Swinney 1979), the two modes still differ substantially, even beyond the ultimate motor execution commands being sent to the hand or the articulating organs. One of the main differences is the time pressure under which speakers work. Where the writers use virtually as much time as desired to draft, re-draft, encode and re-encode a sentence, speaking requires practically online processing, as prolonged pauses may be understood as signals of ceding the floor. Even in scenarios like lectures or public speeches, where the speaker's right to the floor is almost undisputed, it is impossible to take as much time to formulate a sentence as is available in writing. Thus, speakers constantly find themselves under time pressure and have little opportunity to draft a written-like utterance before producing it.

Stress factors, among which time pressure no doubt belongs, have been shown to project themselves into the output produced. Saslow et al. (2014) observed that subjects under stress lower the linguistic complexity of their speech. This may be a consequence of the fact that working memory is impaired under stress conditions (Luethi et al. 2008; Robinson et al. 2008; Schoofs et al. 2008; Oei et al. 2006). Thus, when speakers are pressured to produce an output, their processing resources are limited, urging them to construct phrases and sentences along the well-known paths, resource-efficiently, rather than risk major disfluencies and eventual floor loss.

This may be one of the reasons behind the increased use of prefabricated chunks (or multi-word sequences) in speech. These are units consisting of several items capable of standing on their own, yet bound together by frequent co-occurrence. Even if their definition is still far from being unanimously agreed upon, as is the way in which they are stored, prefabs are more common in the spoken language than in the written one. Depending on the definition, we find 28 (spoken)/20% (written) (Biber et al. 1999) or even 58.6/52.3% (Erman & Warren 2000) of the language output to consist of prefabs/multi-word sequences.

The mechanism is a straightforward one: when the mind fights a battle against the clock, it should reach for a prebuilt item, represented in memory and ready to be used, rather than compose a new one. Experimental research has yielded evidence for such a view, even though the mechanism seems to be more complex than posited in Sinclair's Idiom Principle (Sinclair 1991: 110). Even though there is ample evidence that frequently co-occurring units are processed differently from novel phrases (e.g. Grimm et al. 2017; Janssen & Barber 2012; Arnon & Snider 2010; Bybee & Scheibman 1999), there is no consensus that they constitute a single choice (Siyanova-Chanturia & Martinez 2014). Still, the presence of an entrenched sequence of words which co-occur above chance level has been shown to result in a processing advantage for both the speaker and the listener (Siyanova-Chanturia & Martinez 2014; Siyanova-Chanturia et al. 2011).

And yet, the increased use of prefabricated chunks is not sufficient to warrant production that would be free of disfluencies. However, the speaker's choice to employ pre-built multi-word items should be reflected in the disfluency placement. Given that highly entrenched items should be retrieved and planed much more efficiently than novel formations, they should also lead to fewer disfluencies occurring inside them. As a consequence, speaker-related disfluencies should be pushed into chunk boundaries, where the cognitive load is comparably higher, as hypothesized by Schneider (2014). In terms of the surprisal measure, this would mean that locations with higher surprisal are more likely to contain disfluencies, as items inside of mental chunks should be predictable by the preceding history.

4.2 Comprehender-related constraints

Similarly to the production of speech, its comprehension is also not identical to that of written communication. In the case of reading, comprehenders can vary the input arrival rate dynamically and individually as a function of its complexity and the cognitive load incurred. This is manifested by observations of a correlation between complexity of a text and the probability of regressions, fixation length and second-pass reading time (Shaoul & Westbury 2011). Such variables do not exist in speech processing. Even though partial repetitions of utterances are common, it cannot be claimed that they are solely motivated by collateral signals from the listeners. Similarly, the amount of time the listener is perceiving the input is often beyond their control. Even though they can ask for the input to be presented more/less rapidly, the degree to which such an instruction can be followed is inversely related to the number of listeners participating at the conversation. The speaker is obviously not able to match the ideal input rate of multiple listeners simultaneously. Similarly, the equivalent of second-pass reading time, the speed at which repetitions are uttered, is outside the domain controlled by the listeners.

Given the lack of control that listeners have over the speech input, they cannot flexibly adjust the input rate to the ease with which they process the speech, especially in strongly monologic scenarios like lectures or public speeches. Even the number of “second passes” (repetitions) is usually limited in the course of a conversation: asking for repetition of every utterance is clearly a dispreferred option. Thus, listeners are most of the time limited to processing the information stored in their memory. This poses them with a problem – if the input stream has a higher transmission rate than what they can process online, they are more likely to experience a processing lag since predictability is closely related to processing difficulty (see Kuperberg & Jaeger 2015 for an overview). This lag might lead to parts of the input not being processed at all since new words arrive before the old ones have been encoded into sparse categories and cleared from the working memory to make space for incoming input.

An ideal communication should prevent such cases. This could be achieved either through macromanagement, i.e. by encoding messages specifically in such a way that surprisal is kept low. The use of frequent multi-word units/chunks is one of the methods of such macromanagement: comprehension is faster for prefabs/chunks/multi-word sequences as their frequency of occurrence increases (Tremblay et al. 2011; Arnon & Snider 2010). However, beyond using prefabricated chunks, speakers rarely have the time to plan an ideal utterance to be produced, taking into account the exact probabilities of possible encoding options. Still, given that they are aware of the probabilities of the upcoming production prior to its phonetic encoding (Gomez Gallo et al. 2008), they have the option of micromanaging the surprisal in their output. One option to do this is by smoothing the surprisal on the phonetic level. Studies such as those by Aylett & Turk (2004, 2006) have shown that speakers modify the level of acoustic detail according to the surprisal value, with more informationally dense structures being encoded more redundantly, i.e with more articulatory detail.

Finally, smoothing by disfluency seems to offer itself in cases where the speaker identifies the planned production as potentially problematic for the listener, yet phonetic smoothing is not sufficient to resolve this problem. If there is no optional syntactic element to be inserted (the claim of the UID hypothesis), they must search for another smoothing option. By inserting a disfluency into a high-surprisal location, additional time is provided to the listeners. In this time they do not need to resolve new input. This in turn may help them catch up on the input that remains unprocessed.

The assumption that speakers employ (some) disfluencies as a mechanism that aims to prevent the listeners from experiencing processing lag is based on the view of working memory as a finite resource or a system with limited capacity to be assigned to currently processed tasks (Baddeley 1986; Daneman & Carpenter 1980). Despite some critique expressed on the notion of limited-capacity (Allport 1989; Navon 1984) and the lack of a universally accepted measure of the capacity itself, this construct serves well in explaining the observed phenomena of human cognition (Schneider et al. 2007). In the case of comprehension, working memory and processing system of a listener present a bottleneck. Should they be occupied at the moment new input is received, they may not be able to process it at all: the message is ephemeral, any unprocessed traces of it disappear from our memory in 50 ms (Remez et al. 2010) to 100 ms (Elliott 1962). The listeners are thus faced with a "Now-or-Never bottleneck" (Christiansen & Chater 2016). Whatever they don't process immediately, may be lost forever. This is especially likely if asking the speaker for a repetition is not an option. In order to avoid such cases, speakers should strive to communicate in a way that is robust against such disruptions – potentially by providing the listeners with additional time to resolve the previous input.

This need is even more pronounced in dialogic scenarios. Considering that the average separation of two turns is approximately 500 ms (Dąbrowska 2014) and that turn overlapping is an exceedingly common phenomenon, we might assume that the processing of the input as well as planning and encoding of the response occur during the comprehension. In such a case, fewer resources are available for the processing of the received input. In order to communicate efficiently, the average information density should be lowered. In this manner, the likelihood of a processing lag is minimized. In particular, it should be ascertained that there are no sudden peaks in the information transmission rate which would exceed the capacity of the listener. Disfluencies are one approach to do so in cases where other approaches do not suffice.

4.3 Testing methodology

In order to verify the hypothesis stated in the introduction and explore the characteristics of disfluencies in terms of surprisal, a corpus of speech transcripts was evaluated. Concretely, the John Swales Conference Corpus (JSCC, Swales et al. 2009) was used. The JSCC is a corpus of speech transcripts from a conference in the honor of John Swales, held at the University of Michigan in June 2006. Each text contains one presentation delivered during the conference; in total, there are 23 texts, amassing to approximately 80,000 words of fairly formal, monologic spoken English. Even though the speech transcripts contained in this corpus do not necessarily represent the prototypical scenario of spoken dialogic communication, they should offer an insight into disfluency placement without the influence of the interlocutor affecting the results. Since all of the speeches were delivered to a larger audience, the disfluency placement there should not be tailored to the needs of one specific listener but rather represent an averaged use. An additional advantage from the perspective of listener-oriented disfluency placement is the low availability of repetition requests – speakers should be aware of the fact that audience is unlikely to ask for repetition of a sentence they did not understand.

Moreover, external noise which could also induce repetition should be kept to a minimum.

On the other hand, the distance over which the communication is executed is longer than in the case of most dialogues. Speakers are aware of this issue as shown in the example below:

(4) Can everybody hear me.

Because I'm not gonna speak in here I'll use
my wire the microphone.

Can everybody hear me at the back.

JSCC: 20

Here the speaker repeatedly reassures at the beginning of their speech that all of the intended recipients of the message are within the reach of their voice. The verification occurs not only at the beginning of the talk, but also in the process as seen in Example 5:

(5) […] a bit of work has been done on the uh,
pictures, you can't hear me.

JSCC: 11

A final caveat with respect to the data should be mentioned. The JSCC markup does not contain any encoding of pauses. Thus, the purpose of this study was mainly to observe whether disfluencies in general differ from the fluent material in terms of surprisal. The actual attempt to predict them was made in the subsequent studies.

Each of the texts in the corpus was tokenized (matching the tokenization used by the parser and 3-gram model). The surprisal at each word was measured with the compositional model as described in Chapter 2.3. An example output of the model is shown below in Table 4. Afterwards, the profile of disfluencies was evaluated and a simple logistic regression model was fitted in order to assess whether the surprisal measure may provide any insight into disfluency use.

Lexeme	***Issues***	***already***	***arise***	***from***	***this***
Δ	1.127	*NA*	1.56	*NA*	*NA*
p_{3-GRAM}	0.001	0.0001	0.000008	0.20	0.02
p_{PARSER}	0.043	0.018	0.13	0.16	0.39
$LexSurprisal$	6.71	8.84	11.28	1.61	4.02
$SynSurprisal$	3.13	4.02	2.01	1.81	0.95
$Surprisal$	3.57	4.46	2.46	1.73	1.37

Table 4: Sample output of the language model trained on COCA+MASC, defined in Chapter 2.3. The fact that *issue* and *arise* have associated Δ-values is due to the fact that this sample is not the first sentence of the text. Thus, the semantic history can be constructed. Values displayed are rounded to two digits after decimal point or first non-null digit.

4.4 Results

This chapter will briefly report the results obtained by the methodology discussed in the previous chapter. Prior to discussing the results as related to the current hypothesis, a brief overview of the surprisal scores found within the data will be presented.

4.4.1 Overall distribution of surprisal

In the JSCC, the median surprisal was 2.08 nats, with substantial variation around the mean of 2.37 nats, as implied by the standard deviation of 1.44. Partially due to the presence of a logarithmic transform in the course of surprisal calculation, the surprisal values were not normally distributed in the data but rather had a long-tailed distribution, as visualized in Figure 4.

The individual surprisal values were completely uncorrelated with their neighbors – suggesting that high surprisal locations are not immediately followed/preceded by low surprisal items counterbalancing the temporary trough/peak in information transmission rate (Spearman correlation coefficients of surprisal at w_i with the surprisal at w_{i-1} and w_{i+1} were both ρ = 0.02).

The n-gram-based surprisal and the syntactic surprisal were distributed similarly, with most items being fairly predictable given the preceding history (cf. the density plots in Appendix A.1). Here, too, the surprisal of the previous/following word was unrelated to the surprisal of a given word w_i.

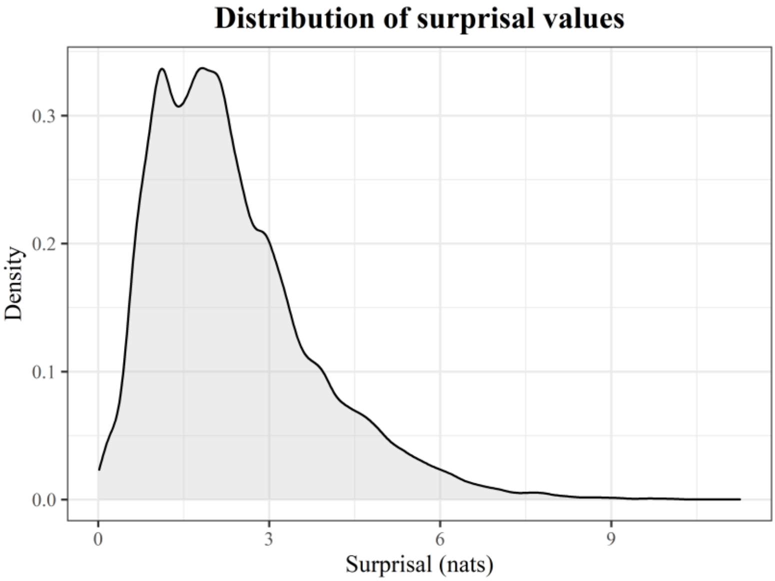


Figure 4: The distribution of the individual values of surprisal in the data. The density plot can be read in a way similar to a histogram, i.e. peaks represent values which are frequent and areas approaching 0 on the y-axis represent non-attested values.

4.4.2 Surprisal profile of disfluencies

If disfluencies should serve as surprisal smoothing agents, they should occur more often in contexts with high surprisal values. In order to explore this assumption, the locations of high surprisal were further analyzed. The results are shown in Table 5, visualizing that approximately a third of the words with high surprisal (defined here as a value above 7.5 nats, i.e. probability below 0.0005, which roughly covers the most extreme 1% of cases) were cases of speech-specific phenomena which belong to broadly defined disfluencies: repetitions, restarts and filled pauses.

Phenomenon	Share	Example
Repetition	6.5%	So **what what** does it mean?
Restart/self-correction	11.1%	What do **I we** do?
Filler/hesitation	15.1%	Look at the example on page **um** seven.
Parse failure	9.1%	…long **stretches** of road… (*stretches* identified as a verb)
Other	58.3%	

Table 5: High surprisal elements (based on a sample of 199 occurrences)

The fact that disfluencies often have a high surprisal value themselves is somewhat unexpected from the information-theoretic point of view.[8] In such a case, they should substantially reduce the listeners uncertainty about the received message. This seems hard to achieve e.g. by repetitions, which are prime examples of redundant encoding. An alternative explanation is that the presence of high

[8] It is, however, less surprising from the computational-linguistic point of view from which high surprisal corresponds only to low probability of occurrence.

surprisal estimates at disfluency location could be an artefact of the surprisal estimation procedure; this will be addressed later on.

Nevertheless, should disfluencies act as information smoothing agents, then it is in particular the surprisal of the surrounding items that is of interest. Disfluencies should be inserted near peaks in surprisal in order to stretch the information transmitted at those peaks over a longer period of time. In order to test whether such a hypothesis might have some grounding in the data, further analysis was performed.

First, the surprisal at the following word was compared individually for the following two phenomena:

- Filled pause (the two most common fillers in the data were taken into account: *uh, um*)
- Repetition

The occurrences were extracted automatically by a regular expression search looking for all 1-word repetitions and filled pauses. Then, the surprisal at the word following the disfluency location (for repetitions, this meant the last occurrence of the repeated word/words) was extracted. Finally, the surprisal values were compared with respect to the status of the word w_i.

The comparison showed that there is a substantial difference between the surprisal of word w_{i+1} depending on whether w_i was a case of disfluency or not. This difference, based on 640 individual observations of filled pauses and 441 repetitions, translates into a difference of means of 1.56 (um, n = 207, $\Delta median = 1.92$, Cohen's d = 1.15), 1.39 (*uh*, n=433, $\Delta median = 1.62$, Cohen's d = 1.02) and 1.55 nats (repetitions, $\Delta median = 1.26$, Cohen's d = 1.14). This is also visible in the violin plot in Figure 5: there is a substantial difference in the distributions, which translates into a pronounced difference in the medians, too.

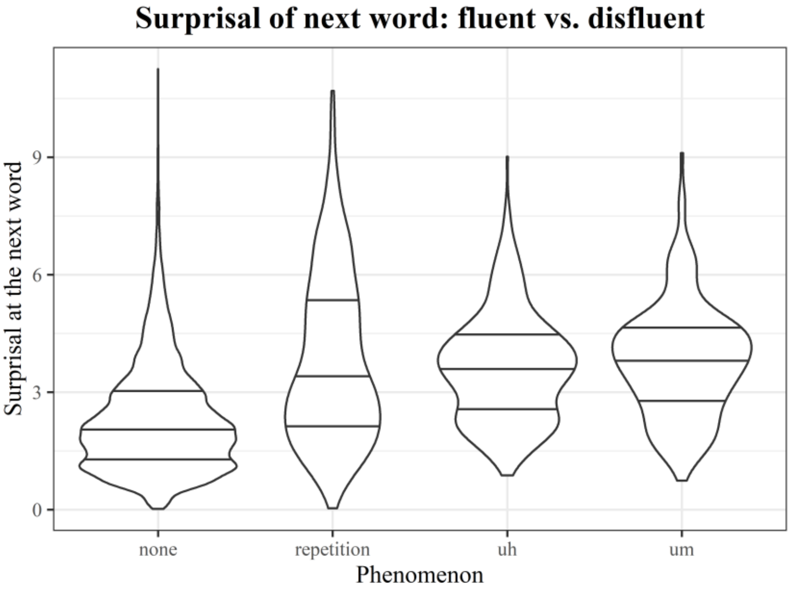


Figure 5: Surprisal of w_{i+1} as related to the status of w_i. The horizontal line in the violin shows the median, while the width of the violin corresponds to a density plot rotated 90° anticlockwise and mirrored along the y axis.

From this analysis, the first element w_i of repetitions thus seems to be stretching the information which would otherwise be conveyed by the second element w_{i+1} only. Such an explanation is not contested by the alternate analysis (proposed by Adell et al. 2007): there is no substantial difference between the information value of the first element w_i and the general distribution in the fluent data.

Such a simplified analysis, however, cannot be presented as a proof that repetitions and filled pauses are tools for transforming the information transmission rate into a more uniform one in constructions where no optional choice point (in the sense of the UID hypothesis) is available. Nor should it be seen as strong evidence for the identifiability of disfluencies by the surrounding surprisal values. First and foremost, it must be excluded that the heightened surprisal of w_{i+1} is caused by the presence of w_i. For this purpose, the files were cleaned of these phenomena (i.e. the filled pauses and one of the repeated elements were removed) and reprocessed through the surprisal estimating script. Afterwards, they were semi-automatically aligned with the original files to identify the cleaned locations. Though five types of filled pauses were observed (*eh*, *ehm*, *uh*, *uhm* and some occurrences of *yeah*), only two of them were frequent enough to justify an analysis (*uh* with 424 observations and *um* with 200 cases).

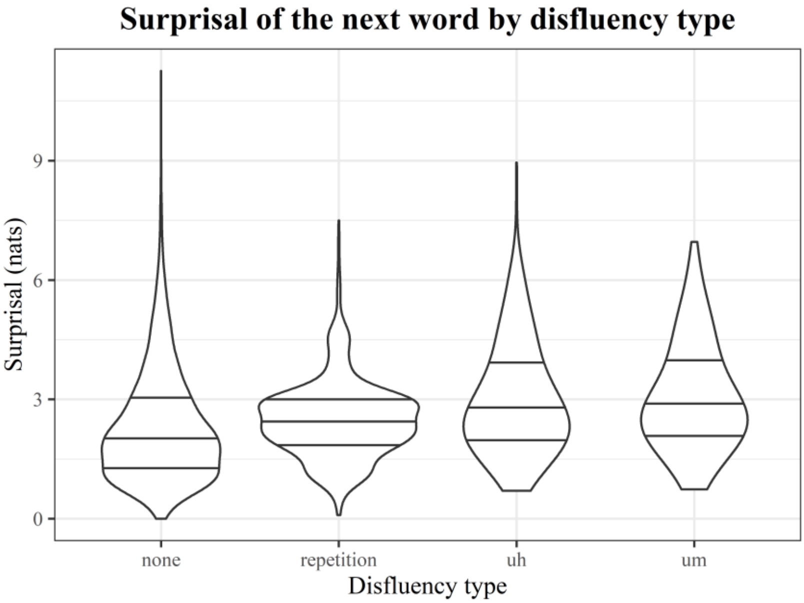
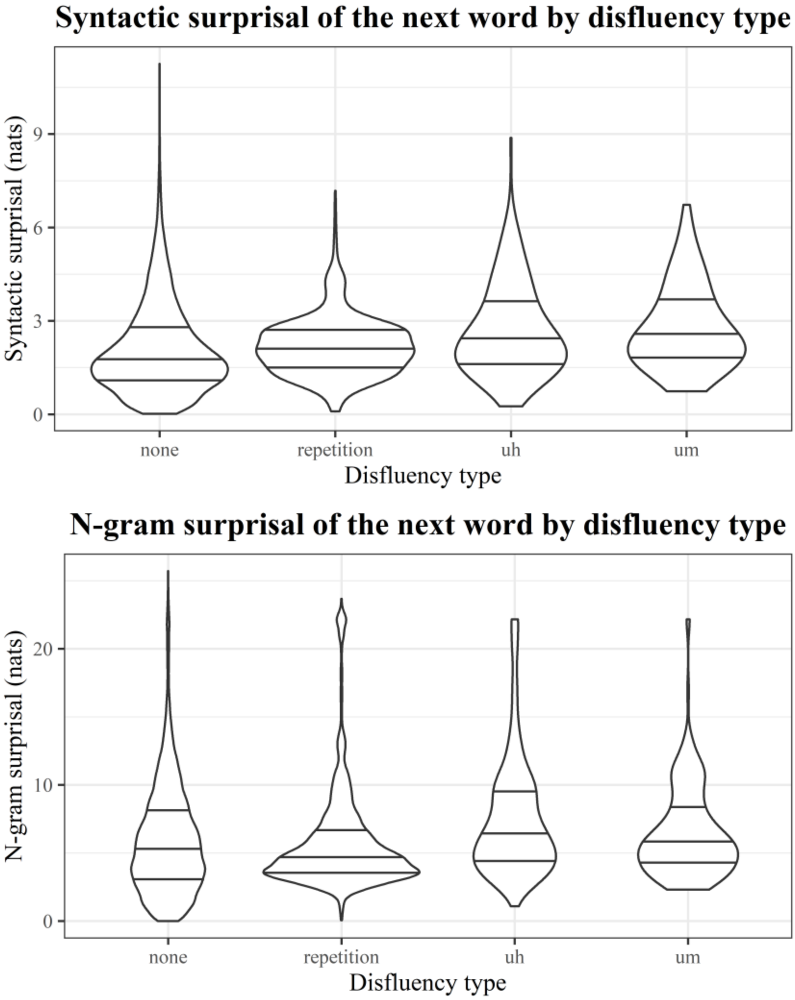


Figure 6: Surprisal of w_{i+1} as related to the status of w_i. Surprisal values were obtained after cleaning the disfluencies mentioned from the data.

The effect, though attenuated, persisted (Figure 6). More prominently for filled pauses (*um:* difference of means of 0.73 nats, $\Delta median = 0.64$, Cohen's d = 0.52, *uh:* difference of means of 0.66 nats, $\Delta median = 0.61$, Cohen's d = 0.46) than for repetitions (difference of means 0.17 nats, $\Delta median = 0.43$, Cohen's d = 0.12), the median surprisal was higher for items after the position where the disfluency was inserted.



Syntactic surprisal of the next word by disfluency type

N-gram surprisal of the next word by disfluency type

Figure 7: Syntactic/N-gram surprisal of word w_{i+1} as related to the type of disfluency inserted at w_i. Surprisal measured after disfluencies had been removed.

For repetitions, a large portion of this trend can be tracked to the syntactic surprisal (which is by definition responsible for 64% of the overall surprisal). As shown in Figure 7, the n-gram surprisal of words following a location of repetition (i.e. the first item of the repeated element itself) is actually lower than in the case of the fluent text. On the other hand, the trend direction is maintained for both syntactic and n-gram surprisal for the filled pauses. Finally, the values of the semantic cohesion coefficient Δ did not differ between the individual disfluency groups.

The observations are congruent with the view of disfluencies as symptoms of processing effort (Clark & Fox Tree 2002): if the upcoming output is highly unlikely, it might be impossible to plan online. Then, the insertion of a disfluency (Clark & Fox Tree discuss filled pauses) can be a mere symptom of the inability to proceed. Similarly, the data supports the alternative explanation from the information-theoretic perspective, in which disfluencies could serve the purpose of a “smoothing particle” and which follows the claims of those researchers, who suggest that hesitations do not impact comprehension above and beyond the extra processing time they offer to the listener (Bailey & Ferreira 2003; Brennan & Schober 2001). This extra processing time can also be viewed as an additional amount of time over which information is spread. The tendency to insert this additional time prior (rather than after) the locations of high information density becomes visible when visualizing the medians of surprisal around the position in which the filler/repetition occurs (Figure 8). This shows a clear peak in the per-word information following the position of the disfluency, contrasting to the baseline value.

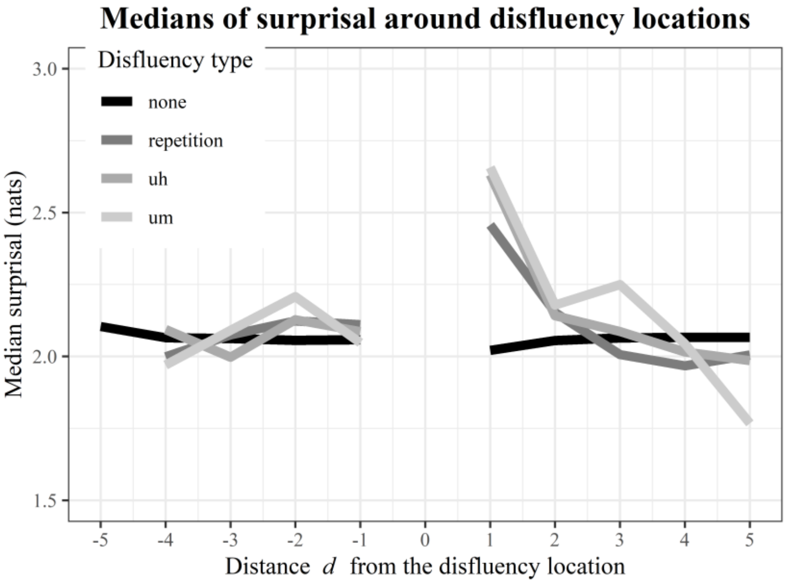


Figure 8: Surprisal of w_{i+d} as related to the status of w_i for $d \in [-5, 5]$. Surprisal values were obtained after cleaning the disfluencies mentioned from the data. w_i is the location at which the disfluency occurred in the uncleaned data and thus has no value in the cleaned data.

Such an observation is suggestive of the information smoothing hypothesis. Interestingly, the median surprisal drops below that of fluent text after the disfluency location. This pattern could be interpreted in several ways: it could suggest that the first uncommon/unpredictable item specifies the continuation to such a degree that all subsequent items become easier to predict. For example, though *Albert* will be comparably hard to predict, it will simplify the prediction of *Einstein* substantially. Alternatively, from the UID perspective, this could mean that the peak in surprisal is smoothed both by disfluency insertion and by the lowering of the surprisal of the following items. In such a way, the average surprisal of the extended context would still be within the bounds of what comprehender can manage. Finally, focusing on the speaker, the rise-fall pattern could suggest that the preparation of this unlikely item is so difficult that the following context must be constructed along the well-known path in order to maintain fluency. Obviously, the simple plot does not allow the decision which of these explanations – if any – reflects the reality. Further research in this area is required.

In spite of the observed peak in surprisal after the disfluency location, it is likely not the case that the difference in the surprisal estimate of two neighboring words alone predicts the occurrence of a filled pause/repetition. Even though the proportion of disfluencies correlates with the increase in this difference (see Figure 9): the relationship is not monotonic. While the ratio of disfluencies increases with small discrepancies in the information transmission rate, it decreases again once the difference between two neighboring items is larger than 3 nats (cf. Appendix A.2 to see the proportion to which the different disfluency types are represented in individual bands of surprisal difference).

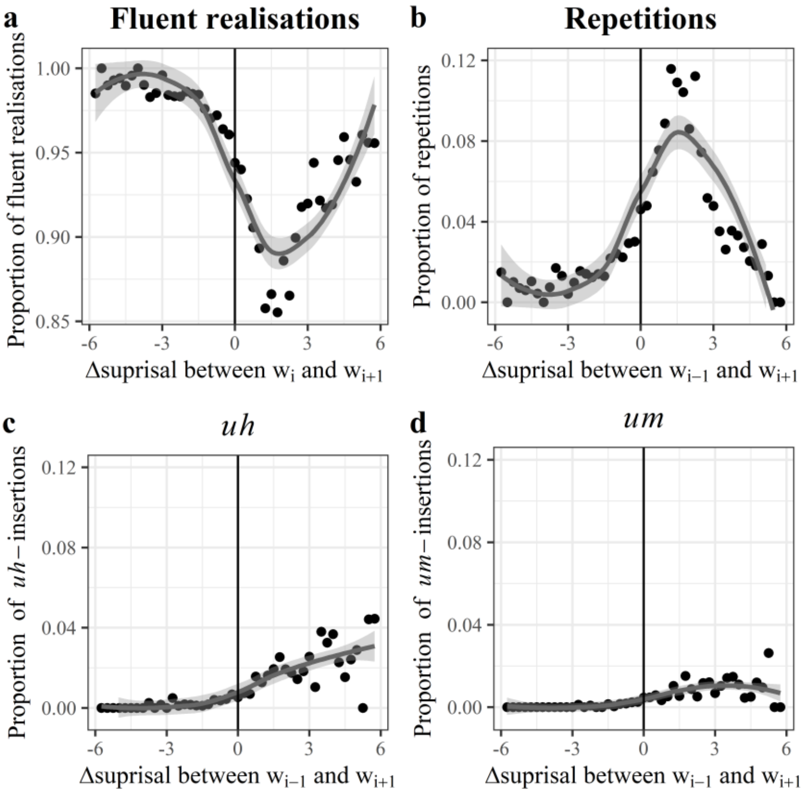


Figure 9: Surprisal difference of w_{i-1} (the item preceding the disfluency) and w_{i+1} as related to the proportion of individual disfluency phenomena being inserted. For the fluent realizations in plot a), the surprisal of two neighboring items is compared. Vertical bars visualize the surprisal difference of 0.

It might be intuitive to expect a surprisal smoothing technique such as a disfluency to be used *after* the high surprisal location, allowing the receiver longer time to process the information given. However, this seems not to be the case, as the surprisal before the filled pause/repetition tends to be the same as in the rest of the data. This preference to place disfluencies prior to high-surprisal locations may be related to the other factors, suggested to explain disfluency occurrence as some of them are captured by the surprisal measure as well. As summarized in Chapter 3, most previous research on disfluencies argues that they are either the result of a speaker's inability to proceed with the production, or a signal to the listener about the status of the upcoming input. In the first case, the increased surprisal may reflect the complexity of the produced item. In such a case, the disfluency should indeed occur before the high-surprisal item. Similar observation is expected from the disfluency-as-a-signal perspective: if disfluencies signal that the upcoming input is unexpected/rare/unpredictable, they must precede it. The unexpected/rareness/unpredictability should then be reflected in the surprisal of the item after the disfluency.

The data used in this study does not permit an exploration of the concrete mechanism through which disfluencies interact with surprisal – most importantly, how do they influence the processing of the high-surprisal locations. It thus remains a task for further studies to disentangle whether disfluencies indeed serve as surprisal smoothing agents and how exactly do they operate: whether they work as signals of upcoming complexity, or whether their purpose is providing time for the comprehender to free their working memory before a complex item. The present study, however, has shown that disfluencies do differ in their profile from the fluent text. The next chapter will explore whether this difference is useful in disfluency prediction.

4.5 Disfluency prediction by surprisal

This chapter shows that the difference between disfluencies and fluent text in terms of surprisal of the item following the location of a disfluency may be used as a predictor of disfluency occurrence. It achieves this by verifying the trends reported in the previous chapter through a logistic regression model.

The surprisal smoothing hypothesis explains likely only a proportion of the data. Other than taking into account the inevitable presence of noise (such as the fact that some repetitions may be caused by background noise overpowering the speech and motivating re-utterance), previous research suggested factors that are not captured by the language model used in this thesis. Such factors include e.g. the failure in retrieval that is caught by the speaker prior to production. In order to explore the influence of the measurements collected in the present study on the use of disfluencies, a simple logistic regression model was built, using the following predictors:

- Sentence initial (true/false)
- Surprisal of the next word
- N-gram based probability of the next word
- Probability of the next word as assigned by the parser
- The difference in surprisal between words w_{i-1} and w_i (here, the word indexes are based on the data from which disfluencies were cleaned)

The outcome variable was a binary decision task: will w_i be disfluent (preceded by filled pause/repetition)? The individual surprisal elements (i.e. the syntactic surprisal and the n-gram surprisal) were not included in the model due to their collinearity with the overall surprisal score. The n-gram/syntactic probability was kept as a predictor as it has comparable performance in comparison to the surprisal elements in terms of explanatory power, but lower correlation coefficient with the other predictors. The data also contained information about the position of the word within the sentence (sentence-initial or not) as this was shown to form a clear pattern in the data: almost all sentence beginnings (defined by a full stop in the transcribed text) were disfluent; strikingly,

	Coefficient	**Std. error**	**Wald's z**	**p-value**
Intercept	-3.78892	0.20286	-18.678	***
Sentence initial	8.36555	0.24341	34.369	***
N-gram probability	-4.03457	0.78059	-5.169	***
Syntactic probability	-0.41557	0.34618	-1.200	
Surprisal	-0.09750	0.05707	-1.708	.
Difference in surprisal $w_i - w_{i-1}$	0.22246	0.03623	6.139	***

Table 6: Resulting model of disfluency prediction, p-values key: . significant at 0.1, * significant at 0.05, ** significant at 0.01, * significant at 0.001**

approximately 74% of disfluencies occurred sentence-initially. The model was trained using 80% of the data for training, its performance was evaluated on the remaining 20%. Table 6 summarizes the fitted model parameters.

The deviance of the null model (i.e. one that would assign all cases to the majority class) was 15182, the model presented above had a deviance of 5408, a statistically significant improvement according to the chi-squared test ($p < 0.001$). Tjur's pseudo-$R^2$of the model was 0.72 suggesting that the model was reasonably capable of explaining the variation in the training data. Leaving out any of the individual predictors led to deterioration of the model fit as assessed by the AIC. The relative importance of the individual predictors was evaluated using the *caret* package (Kuhn 2017) for R and assigned the highest relative importance to the fact whether an item was sentence initial or not. Indeed, as suggested by the related coefficient; disfluencies are more than 4000 times more likely ($\Delta odds$) to occur in a sentence-initial position than elsewhere. The surprisal itself has a small non-significant negative coefficient assigned, suggesting that it is of little importance to disfluency prediction, unlike the difference in surprisal between two neighboring items. Here, the coefficient is larger (coeff. 0.22, $p < 0.001$) and positive, suggesting that disfluencies are more likely to occur between two items which have a large positive difference in the amount of information they transmit. N-gram probability is also a significant predictor, with a negative coefficient, suggesting that items which have a high probability of co-occurrence (expressed by probabilities assigned by the n-gram model) are unlikely to be divided by a disfluency.

The training data was used to optimize the cutoff threshold to decide whether a case should be assigned as disfluent or not as later used for the calculation of performance measures. The performance over the test data was additionally visualized using an ROC curve (Figure 10).

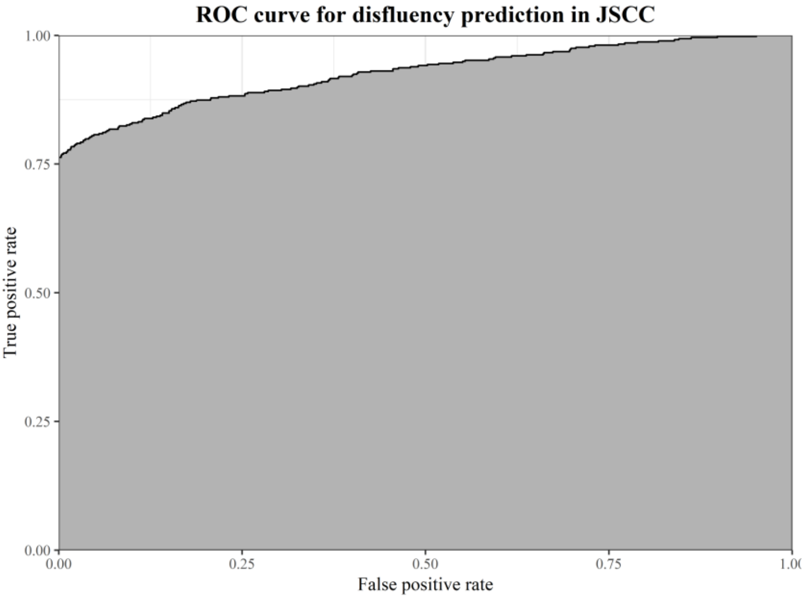


Figure 10: ROC curve of the disfluency prediction model over the test data. The shape of the curve suggests a non-random performance: the rate of true positives grows faster than that of false positives.

This shows that the model is not assigning randomly, but rather its true positive rate is higher than the false positive rate. The area under the curve is 0.93, showing a high probability that a randomly selected positive example will be rated higher than a randomly selected negative example. Using the optimized cutoff threshold, the model predicted correctly 367 out of 477 disfluencies (recall of 76.9%) in the test set. Given that it wrongly expected 37 cases to be disfluent, the precision was 90.8%. The overall accuracy was 98.4%.

The strong influence of the position of a word in sentence on the prediction is reflected in the performance if this parameter is removed. In such a case, the deviance of the model rises to 14303, Tjur's R^2 decreases to 0.03 and the area under the ROC curve becomes 0.75. Still, this model is significantly better than the null model.

To conclude, while some of the surprisal-based and probability-based measures were identified as significant predictors of disfluencies, the difference between the location of a word at the beginning of a sentence and elsewhere is a much stronger determiner of disfluency in the data analyzed. Besides being a real pattern in the data, this could also be an artefact of the JSCC compilation method. If the transcribers tended to include a particular sort of disfluencies – e.g. only those occurring sentence initially – this would inevitably be reflected in the performance. Next chapters will thus seek to address this issue by using a different dataset. Additionally, they will employ models capable of fitting more complex functions and thus accounting for e.g. interactions between the individual predictors.

4.6 Conclusion

The previous chapter has shown that surprisal may serve as a predictor of disfluency occurrence in spoken language. When analyzing the relationship between the surprisal of an item and its fluency status, a clear trend was observed: disfluencies tend to occur before items with higher surprisal values. However, further analysis has suggested that the surprisal value itself may not be as strong a predictor of disfluency as the magnitude of the local change in surprisal. Thus, items which are much less predictable than their immediate context are more likely to be disfluent than items which have the same high surprisal value, yet do not differ from their context in this respect.

The pronounced influence of sentence-initial position on the disfluency placement is somewhat surprising. Still, it is not contrary to the observations made in previous work: Shriberg (1994) reports that sentence-initial locations are substantially more likely to be disfluent in comparison to sentence-medial positions in all three corpora that she analyzed. Additionally, it is intuitive to expect disfluencies caused by processing effort linked to language production to occur sentence-initially, as the planning at sentence/utterance beginnings is not restricted to the next element only; rather, the message to be expressed by that sentence needs to be formulated as well.

The issue of time was not taken up in this study. However, speakers are known to manipulate the speed and phonetic detail with which they pronounce words in order to keep the information transmission rate smooth (Aylett & Turk 2006, 2004). It is thus possible that some of the locations in which disfluencies should be inserted on the basis of the local change in surprisal employed an alternative approach – that of smoothing by articulatory detail and/or speech rate. Still, the present data suggests that such a smoothing approach is not always available or feasible. This could happen in cases where the upcoming output is not ready for production yet, urging the speaker to issue a floor-holding signal.

Given the range of functions assigned to disfluencies by previous research, it is unlikely that every occurrence of a filled pause or repetition can be explained through the surprisal estimate, even though many of the causes should be reflected by the scores assigned by the language model, e.g. the rareness of the upcoming input or its unexpectedness. This was confirmed by the observations made in this study: some disfluencies remained unexplained. Similarly, given the range of information flow smoothing techniques, not every change that is too abrupt to be smoothed by phonetic detail will be smoothed using a disfluency. However, the results obtained here suggest that the surprisal estimate could be used to further improve the results of studies aiming to predict disfluencies, such as those mentioned in Chapter 3.2. The attempt to do so will be presented on the following pages.

Chapter 5
Study IIa: Predicting disfluencies by context

The results obtained in Study I suggested that the use (or lack of) a disfluency is connected to the local surprisal, and its local change in particular. In order to validate the results, a subsequent study was carried out, described in this chapter.

5.1 Introduction

As mentioned in Chapter 3.2, most previous work in the computational linguistic area has been directed at recognizing (and removing) disfluencies in human-generated texts in order to yield cleaned transcripts similar to written texts. However, the prediction of disfluencies has so far remained somewhat aside from the mainstream.

Despite this fact, some good results have been obtained. Most notable being the work of Adell et al. (2007), summarized above. They report achieving precision of 96% and recall of 58% in predicting filled pauses in a Spanish corpus with their combination of a probabilistic language model and a decision tree algorithm.[9] Ohta et al. (2008), who also included discourse markers in their prediction of Japanese fillers,

[9] However, they did not develop this approach further and their latest publication in this area does not incorporate the disfluency predicting module at all (Adell et al. (2012), which is somewhat unexpected considering the early success.

state that the performance of their conditional random field model ranged from $F_1 = 0.23$ (precision: 0.26, recall: 0.21) when merging all fillers under one label to $F_1 = 0.06$ (precision: 0.08, recall: 0.05) when combining the filler insertion model with a trigram-based filler selection model choosing the correct filler type.

In a more recent study on disfluencies in English, Schneider (2014) explored the impact of chunking on hesitation placement. In a set of specific environments (such as sentence initial subject-verb cluster), her model achieves misclassification rates varying between 51.8% and 1.6% (i.e. accuracy between 48.2% and 98.4%) using random forests and single decision trees as estimators and a set of co-occurrence-based predictors. Unfortunately, the cases in which the model has the lowest misclassification rate are the cases where the classifier does not perform significantly better than a baseline model, predicting all cases to belong to the most common class. The low misclassification rate thus mirrors the overwhelming dominance of one class in the data and does not correspond to a high macro-averaged F_1 score.

The independent variables used for training and prediction in these studies describe both the context before and after the position in which the disfluency was inserted and can be roughly divided into two major classes – quantitative and qualitative. The quantitative predictors include various probability and co-occurrence measures, such as:

- Direct transitional probability

- Backward transitional probability
- Mutual Information Score
- Lexical Gravity Score
- Bi-/trigram probability
- Probability of a disfluency to occur after word w_{i-1}

The qualitative predictors include the part-of-speech tags assigned to the items around the position in which a disfluency occurred in the training data (used as an approximation of the syntactic pattern surrounding the disfluency), individual lexemes and sentence boundary tags. In this way, the authors hope to give the classifier sufficient information to learn the distribution of disfluencies in order to apply it later to unseen data. The main limitation of such approaches is their reliance on word-based statistics, using only approximations of the underlying syntactic structure, or operating within precisely specified syntactic environments.

Additionally, none of the studies explored the issue from the perspective that this thesis is employing: as interacting with the local surprisal. This chapter seeks to explore and describe how well can the combined measure of surprisal (described in Chapter 2.3) predict the use of disfluencies. In the course of this chapter, the following hypotheses will be explored:

Hypothesis 2.1:

The surprisal estimate produced by the model defined in Chapter 2.3 is a predictor of disfluency occurrence in the MICASE corpus.

Hypothesis 2.2:

The occurrence of a disfluency depends not only on the overall predictability of the word it precedes, but also on the difference in the local information transmission rate – that is the difference in the predictability of the word preceding the position in which the disfluency was inserted and the word following it.

Both of these hypotheses are based on the observations made in the previous Study I. This chapter also seeks to validate the observations on a different dataset in order to verify that they are not artefacts of the data collection procedure. Concretely, the hypotheses are tested on the MICASE corpus, described in more detail in 5.2.7.

This explanatory analysis was combined with an active attempt to predict the disfluencies. Thus, rather than only observing the proportion of disfluency occurrence that can be explained post-hoc, a model was fitted on a part of the data, attempting to predict the disfluencies in a held-out test set. The prediction used a combination of quantitative and qualitative predictors, described below.

5.2 Present study

This study attempted to verify the strength of relationship between surprisal and the disfluency occurrence, as observed in Chapter 4.5. The hypotheses listed in Chapter 5.1 were explored on transcribed speech data from the MICASE as the JSCC contains only a limited number of disfluencies.

In development, the disfluency prediction was tested on the basis of two different algorithms: the classification tree algorithm and the multi-layer perceptron. The predictors used are largely based on the previously listed research with the combined measure of surprisal and the local difference in the surprisal of two neighboring words added. The full list of predictors used thus includes:

- Sentence position (initial vs. medial/final)
- Bigram frequency
- Direct transitional probability
- Backward transitional probability
- Mutual Information Score
- Lexical Gravity Score
- Lexical surprisal
- Syntactic surprisal
- Surprisal
- Surprisal compared to the previous word

The last four measures were taken from the output of the combined measure or its components and will not be described here further. For detailed description, please refer to Chapter 2. The sentence-initial position was a simple binary variable, indicating whether a given word is the first one in a sentence or turn. Thus, sentence starts were not inserted as special pseudo-words or meta-tags, but were expressed by one of the values associated with a word.

The remaining scores: direct transitional probability, backward transitional probability, mutual information score and lexical gravity are all different measures aimed at expressing the connection between two words. Their implementation was done mostly in agreement with Schneider (2014) and will be briefly described in the following chapters. The fact that multiple scores for the same underlying notion were used is due to the lack of a universally acknowledged measure of collocation strength: all measures devised so far suffer from some bias or sensitivity to data imperfections (Gries 2013), in spite of the considerable progress in this area. Using multiple measures together may alleviate some of their weaknesses. Furthermore, the decision to implement several collocation scores was motivated by the effort to establish continuity with previous research.

Some of these scores are raw probabilities, rather than log-transformed surprisal estimates. These measures are included in order to establish continuity with previous research, without being strongly tied to any underlying theoretical assumptions. Naturally, the probabilities could be simply transformed into surprisal estimates. As a matter of fact, there are good reasons to use surprisal only: first, it would be more in line with the information-theoretical view of disfluencies as information smoothing agents. Secondly, raw probabilities tend to be poor numeric variables. Yet, given the choice of algorithms used for the prediction, the influence of such a transformation would be negligible – if not non-existent. Thus, to maintain connection to previous research, probabilities were used.

On the next pages, the non-trivial scores used for disfluency prediction will be described in more detail. Similarly, the two algorithms employed will be briefly presented.

5.2.1 Direct transitional probability

The direct transitional probability (TP-D) is the first measure of association strength used. It expresses the probability $p(w_i|w_{i-1})$ of word w_i to follow a given history, in this case represented by word w_{i-1}. The probability is calculated by dividing the count $c(w_{i-1}w_i)$ of a given bigram $w_{i-1}w_i$ by the overall frequency $c(w_{i-1})$ of its first word w_{i-1} (Kapatsinski 2004). Thus,

$$p(w_i|w_{i-1}) = \frac{c(w_{i-1}w_i)}{c(w_{i-1})} \qquad (27)$$

For words which are closely associated such as *peanut butter*, its value should be approaching 1, as most occurrences of *peanut* will be followed by *butter.*[10] By definition, direct transitional probability can only express the likelihood of w_i to occur after w_{i-1}, but it does not contain any information about the likelihood of w_{i-1} to occur given w_i. This leads to the loss of the information that while *Hansel* will be followed very commonly by *and*, *and* itself is rarely preceded by *Hansel*. It is thus a directional measure.

Importantly, the direct transitional probability as defined here is not equivalent to the reciprocal of the lexical surprisal score raised to the power of e. The direct transitional probability is equivalent to the n-gram calculation by maximum likelihood estimation as defined by Equation 12. It uses $n = 2$. The n-gram probability used for the lexical surprisal is calculated with $n = 3$ by Equations 16 and 17 and includes smoothing and interpolation.

5.2.2 Backward transitional probability

Backward transitional probability (TP-B) expresses association of words in the opposite direction when compared to the TP-D. It captures the probability that word w_{i-1} will precede w_i. The formula used to calculate it differs only minimally from the one used to calculate TP-D (Kapatsinski 2004):

[10] In COCA, the actual frequency of *peanut butter* is 2893 out of 5064 occurrences of *peanut*, thus the conditional probability is approx. 0.57

$$p(w_{i-1}|w_i) = \frac{c(w_{i-1}w_i)}{c(w_i)} \qquad (28)$$

Given the close similarity of the definitions between TP-D and TP-B, they share their weakness in expressing the word association unidirectionally. As a consequence, while TP-B can express well that *York* is often preceded by *New*, it says nothing about the likelihood of *York* to appear after *New*. To mitigate this issue, both of these two measures tend to be used, unless there is a theoretical motivation to focus on one direction only.

5.2.3 Mutual information score

The mutual information score (or MI score) is an attempt to rectify the unidirectional limitations of TP-B and TP-D statistically. It assesses the association strength of the two words by comparing the observed frequency of the bigram with its expected frequency. The resulting score is then an expression of how much more likely are the words to occur together in comparison to what would be expected if the words in the corpus were distributed randomly, maintaining the token frequencies observed in the corpus.

Several distinct equations are used to calculate the MI score. Some of them are capable of calculating the MI score for words (possibly) separated by intervening material (e.g. Davies 2018). The current application uses the formula adapted by Schneider (2014) from the one proposed by Wiechmann (2008) and based on Church & Hanks (1990). It does not contain any window size parameter as it is used exclusively to measure the association strength between two direct neighbors. It calculates the MI score as the log-transformed ratio of the observed and expected frequencies:

$$MI = \log \frac{observed}{expected} \tag{29}$$

To calculate the expected bigram frequency, the product of the observed word frequencies of the individual elements is divided by the overall word count (*N*) of the corpus. The formula used for calculation is thus the following one:

$$MI = \log\left(\frac{c(w_{i-1}; w_i)}{\frac{c(w_{i-1}) \times c(w_i)}{N}}\right) \tag{30}$$

after simplifying the fraction:

$$MI = \log\left(\frac{c(w_{i-1}; w_i)}{c(w_{i-1}) \times c(w_i)} N\right) \tag{31}$$

Given how the MI score is calculated it should favor co-occurring words and penalize words that co-occur less than expected. The highest score of $\log \frac{N}{c(w_{i-1}; w_i)}$ will be reached by items which co-occur every time they appear in the corpus. This also points to the pitfall of the measurement: two co-occurring hapax legomena will have unrealistically inflated scores. The developers of the measure were aware of the fact that “the association ratio becomes unstable when the counts are very small" (Church & Hanks 1990: 24) and did not use it for bigrams of frequency lower or equal to 5. This practice was not followed in the present study as the CART algorithm should be capable of finding the optimal threshold of bigram frequency at which the MI score becomes an unreliable predictor – if it is used as a predictor at all.

5.2.4 Lexical gravity score

One of the main omissions in the make-up of TP-D, TP-B or MI score is the fact that it does not recognize the dependence between the co-occurrence of forms and the syntax of a language. In other words, all the previously presented measures assume that a given word (string) can occur after/before any other word or even itself. This is, however, not the case in human languages. There are strong preferences for the order in which words appear and strong limitations as to which words can appear together. Thus, for example, even though the word *of* is one of the most frequent words in the English language, it virtually never appears after another of the most frequent words, *the*. In order to capture this information in the measure in some way, the lexical gravity score (G) was suggested (Daudaravičius & Petrauskaitė 2004), taking into account the count $c_{type}(w_{i-1})$ of possible continuations after a given word and the count $c'_{type}(w_i)$ of types observed to precede w_i. Due to this fact, G may and usually will not correlate with the MI score of a bigram, even though Schneider (2014) suggests that it correlates strongly with log-transformed bigram frequency. It is calculated as follows:

$$G = \log\left(\frac{c(w_{i-1}w_i) \times c_{type}(w_{i-1})}{c(w_{i-1})}\right) + \log\left(\frac{c(w_{i-1}w_i) \times c'_{type}(w_i)}{c(w_i)}\right) \quad (32)$$

Given the formula, G tends to be high for bigrams which consist of words that have many possible continuations, yet co-occur more than would be expected by their sheer numbers.

5.2.5 Classification and regression trees

For the prediction of disfluency occurrence, the classification and regression tree (CART, commonly referred to as decision trees) algorithm was used (Breiman et al. 1984). This algorithm's main strength lies in its ability to divide the data into multiple decision paths, where every path may employ different predictors or different thresholds. This allows for example the non-discriminative inclusion of the MI score: in most other algorithms, a top-down decision is needed to select the conditions under which a predictor becomes unreliable. Given the mode of operation of CART, the algorithm is able to discover these conditions on its own and only apply the predictor if it actually explains the trends in the data. The following paragraphs will sketch the CART algorithm as used in the present thesis.

It is a recursive algorithm operating by repeatedly splitting the data into two groups (branches) at a time. For this purpose, the data consisting of training vectors $x_i \in \mathbb{R}^n$, where $i = 1 \dots l$, and a label vector $y \in \mathbb{R}^l$ is split at each node m according to the splitting criterion $\theta_m = (j, t_m)$. This splitting criterion consists of the feature j and a threshold t_m and is determined through the following procedure: for the data Q at a given node m, split the data into two groups Q_{left} and Q_{right} according to each candidate parameter θ. Thus,

$$Q_{left}(\theta) = (x, y) | x_j \leq t_m \qquad (33)$$

$$Q_{right}(\theta) = Q \setminus Q_{left}(\theta)$$

Afterwards, the impurity $G(Q, \theta)$ is calculated. This is the weighted average of the impurities of $Q_{left}(\theta)$ and $Q_{right}(\theta)$ as determined by a function $H()$ and weighted by the proportion of Q contained in the given group:

$$G(Q, \theta) = \frac{n_{left}}{N_m} H\left(Q_{left}(\theta)\right) + \frac{n_{right}}{N_m} H(Q_{right}(\theta)) \qquad (34)$$

Finally, θ_m is chosen such that

$$\theta_m = \underset{\theta}{\operatorname{argmin}}\, G(Q, \theta) \qquad (35)$$

Afterwards, the algorithm proceeds recursively through $Q_{left}(\theta_m)$ and $Q_{right}(\theta_m)$, until a stopping criterion is met (Pedregosa et al. 2019).

The impurity score of a group expresses how varied are the values of the dependent variable in that group. A group containing only examples of one of the categories would be considered absolutely pure. Otherwise, the impurity is calculated through a function determined by the objective (classification or regression). In the current study, the objective is classification and the concrete measure of impurity was the Gini measure.

This is defined as follows: for each outcome k from a range of values $0{,}1 \dots K-1$, calculate the ratio of cases with label k at node m, representing a region R_m (subset of the original data) with $N_m$observations, according to this equation:

$$p_{mk} = \frac{1}{N_m} \sum_{x_i \epsilon R_m} I(y_i = k) \qquad (36)$$

Then, the Gini impurity equals:

$$H(Q_m) = \sum_{k=0}^{K-1} p_{mk}(1 - p_{mk}) \qquad (37)$$

Thus, it is minimal in those cases, where one of the proportions equals to 1 (the others then inevitably equal 0), i.e. where all cases in the region R_m have the same label k.

Given that the splitting criterion is newly calculated at each node, both the feature and the threshold will differ, allowing the algorithm to exploit those predictors which are most powerful in a given scenario.

As a recursive algorithm, it requires some mechanism to decide when the splitting should be stopped, otherwise it will proceed until all of the groups contain only one example. To prevent overfitting, the algorithm may be forced to stop after a certain number of splits, or when the branches only contain a given number of examples or when no split would improve the purity of the individual branches substantially.

In addition to its ability to discriminate between helpful predictors in varied decision paths, the CART algorithm is also particularly suited to the problem explored here since it can handle unbalanced predictors and complex interactions as noted by Tagliamonte & Baayen (2012) for random forests, an extension of the algorithm. Additionally, it can be used even when more than one output is to be predicted. Such an ability was a prerequisite of the current study: unlike in Adell et al. (2007), the classifier was not asked to answer a yes-no question ("Is there a disfluency?") but a *wh*-question ("Which type of disfluency, if any, should occur before this word?").

5.2.6 Multi-layer perceptron

The second algorithm employed for disfluency prediction was the multi-layer perceptron (MLP). It is a machine learning approach that is inspired by the makeup of neural networks found in the nature. It consists of units – artificial neurons – that process the input signal and pass it forward towards the output. Unlike in logistic regression, it does not only combine the individual inputs after multiplying them by their weights, but allows for interactions as well. This is achieved by adding one or more hidden layers of neurons. These intermediate layers between the model input and output receive their input from multiple neurons, transform it through some activation function and pass it further towards the output. The transformation depends on the activation function chosen and a weight matrix, giving weight to each connection in the model. The training of the perceptron thus consists of searching for a matrix of optimal weights by minimizing the result of a loss function comparing the predicted output to the real one. This is achieved by the backpropagation of error (Rumelhart et al. 1986) – the errors are calculated at the output and then propagate backwards through the model. At each step an update of the weights is performed, modified by the learning rate of the model in order to avoid large changes in the parameters. The learning rate could be fixed as is the case in the traditional stochastic gradient descent: in such a case, its choice can determine both the quality of the model and the speed with which it is fitted (LeCun et al. 1998). In order to improve the speed with which the model converges to the optimal weights, multiple methods have been devised over the past years (Kingma & Ba 2015; Schaul et al. 2013; Sutskever et al. 2013), expanding the original approach.

Figure 11 represents a simple example of an MLP neural network illustrating its operation. Each of the neurons in the input layer is connected to the neurons in the hidden layer and each of those is connected to both of the output neurons. Thus, interactions in the form any-to-any can be taken into account. This architecture allows the MLP model to approximate any smooth measurable function between the input and output vectors, provided there is a sufficient number of hidden layers and neurons (Hornik et al. 1989).

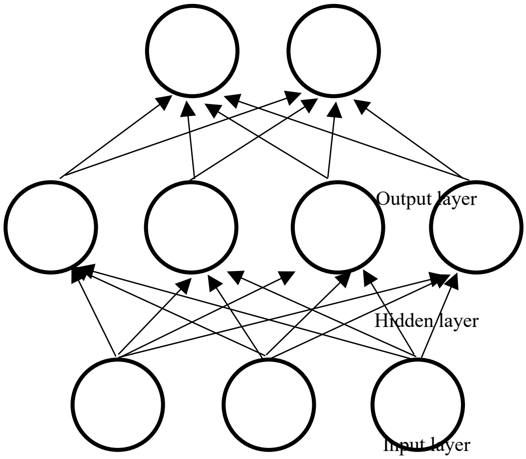


Figure 11: A simple multilayer-perceptron with three input neurons, one hidden layer of four neurons and two output neurons. The input is passed from the input neurons to each of the hidden neurons which transform it and pass it further to the output.

The operation of the neural network is as follows: in the input layer $j = 1$, the number of cells corresponds to the number of features in the training vector x (with one additional feature equal to 1 if the bias unit is used). Each of the cells takes the value of feature x_i as its activation. Then, for all layers of the network, each of the units $i = 1,2, \dots |j|$ at layer j is connected to each unit $i = 1,2, \dots |j + 1|$ in the following layer $j + 1$. The activation of each unit $a_i^{(j+1)}$ is then calculated according to $\Theta^{(j)}$, a matrix of parameters defining the activation function g(z) mapping the inputs from layer j to layer $j + 1$. This procedure is repeated until the output layer is reached. At the output layer, the activation of the output neuron corresponds to the prediction made by the multilayer perceptron. If multiple outputs are needed (i.e. for multiclass classification), the output layer will contain a corresponding number of neurons. Importantly, while the activation spreads forward through the network, the network's parameters are updated backwards, through the backpropagation of error. This means, that once the prediction on the basis of the training vector x_i is made, it is compared to the real value y_i in order to calculate the cost $C(\Theta)$ given the parameters Θ and a cost function C. Then, this cost is minimized by calculating the necessary changes in Θ. To do that, first the error of the output layer is calculated:

$$\delta_i^J = \frac{\partial C}{\partial a_i^J} g'(z_i^J) \qquad (38)$$

expressing how fast the cost changes as a function of the output neuron's activation (the partial derivative of the cost function) and how fast the activation function g changes given the input from the previous layer and its parameters. This is used to calculate the "error" δ^j for each layer $j = J - 1, J - 2$ ... 2 on the basis of δ^{j+1} as:

$$\delta^j = ((\Theta^j)^T \delta^{j+1}) \odot g'(z^j) \qquad (39)$$

Finally, the $\delta^j$is used to calculate the partial derivative of the cost function. This is done for each weight w_{ik}^j connecting unit *k* in layer *j* with unit *i* in layer *j+1*:

$$\frac{\partial C}{\partial w_{ik}^j} = a_k^{j-1}\delta_i^j \qquad (40)$$

The obtained partial derivatives are then used to update the weights (usually, the updates are averaged over a set of training examples in order to reduce the influence of noise). Importantly, the algorithm uses a parameter α (the learning rate) to decrease the size of the change in weights. This ascertains that the model will converge to a minimum of the cost function (though not necessarily a global one, the algorithm may converge to a local minimum as the cost function may not be convex). Should the learning rate be too high, the model may diverge; thus, its value is usually $\alpha < 0.1$.

Importantly, in the process of fitting a neural network model, weights are initialized randomly from a distribution. This procedure is required in order to break the symmetry of the network – if all weights were initialized e.g. as 0, the neurons in individual layers would be identical. By breaking the symmetry, each neuron fits a slightly different function.

Similarly to the trees produced by the CART algorithm, multi-layer perceptrons are prone to overfit, i.e. to yield weights that can recreate the training data well, but generalize poorly to unseen data. In order to avoid such scenarios, several approaches may be used – regularization approaches (preventing any of the weights to be too high), limiting the number of iterations of the backpropagation algorithm or early stopping. Early stopping approaches are usually implemented by reserving a portion of the data as a validation set and verifying the model's performance on this set regularly without using it for the model's training. Once the model's performance on the held-out portion of the data stops improving, the training is interrupted, even if the performance on the training dataset continues improving.

Finally, in comparison to the classification and regression trees, neural networks do not lend themselves equally well to decision path analysis, as their decision logic is not equally transparent.

5.2.7 Data (MICASE)

In order to verify the results obtained in the JSCC dataset (presented in Chapter 4), it was replaced by a different data source: the Michigan Corpus of Academic Spoken English (MICASE, Simpson-Vlach et al. 2002). which was also used to collect the probabilistic/frequency-based measures defined in 5.1 and 5.2.1-5.2.4 apart from those contained in the combined measure of surprisal.

The MICASE is a corpus collected at the University of Michigan at Ann Arbor between 1997 and 2001. It contains 1.8 million words of transcribed speech, corresponding to more than 190 hours of recorded speech in 152 files. The texts cover a range of situations, from lectures, over classroom discussions to one-on-one advising sessions. The recordings were made in a university environment. Thus, they represent a rather specific register; still, the range of situations covered is broader than in other comparably-sized spoken corpora.

Importantly, not every speaker in the corpus is a native speaker. However, since the speaker ID was not included in the predictors, the current study bases on the assumption that non-native's placement of disfluencies is largely identical to the native use. This assumption may not reflect the reality, as not even all native speakers behave identically with respect to disfluency placement: Shriberg (1994) distinguishes two main types: deleters and repeaters; Fruehwald (2016) argues that different types of filled pauses operate as a sociolinguistic variable. However, two reasons drove the decision not to include the speaker variable. First, there are only approximately 7.5% of non-native speakers in the corpus. Thus, they should not skew the distribution substantially. More importantly, though, there are 1571 speakers in the MICASE overall. As a consequence, each speaker would on average be represented by approx. 1200 words, containing roughly 30 disfluencies (based on the frequency of disfluencies, see Chapter 5.3.1). Thus, should the model fit individual intercepts for each speaker, it would often base the estimate on extremely sparse data, leading to poor model. On the other hand, a model averaging the trend over 1500 speakers is less likely to be influenced by individual idiosyncrasies and should generalize better.

5.2.8 Method

To collect the individual statistics and train the model, a five-step procedure was adopted:

1. Data preprocessing
2. Measure collection
3. Alignment and dataset splitting
4. Classifier training
5. Classifier validation

The following pages will describe each step in detail.

5.2.8.1 Data preprocessing

Before any statistics could be collected, the data had to be preprocessed. The preprocessing removed the metadata contained in the files, reformatted the text (e.g. by splitting *that's* into *that 's* to match the tokenization used by the parser and n-gram model) and identified and removed the disfluencies. In such a way, the scores were calculated from a less disfluent input, bringing the procedure conceptually closer to the Noisy Channel approach to disfluency prediction. Additionally, assigning the scores to a fluent input guaranteed that there would be no information leak about the location of disfluencies through the scores themselves, e.g. through a specific value of one of the scores after a disfluency.

The disfluency removal was a fairly simple script consisting of three separate functions, one to identify/remove pauses, one removing *uh*s and *um*s, and one for repetitions. Pauses were identified on the basis of the transcripts; the various pause lengths were conflated. *Um*s and *uh*s were searched for by a regular expression that collected their occurrences as individual strings, distinguishing between *uh uh* (which was considered a case of two *uh*s, separated by a short break) and *uhuh* (which was not considered a case of disfluency at all). The rare cases of *uhm* were treated as cases of *um.*

Repetitions were identified by a function searching for literal repetitions of one-word or two-word strings. If a repetition was found, it was repaired by deleting all but the last item. Thus, single, double and multiple repetitions (single and double repetition are shown in Example 6 were all handled as examples the category “repetition.”

```
(6)  …the the man…
     …the the the house…
```

Partial string repetitions were not collected, as there would need to be a mechanism deciding whether the unfinished string is a case of a repetition, repair or another word which only happens to start similarly to the preceding/following one. Such a decision is not always clear even with manual processing and automated decision making would likely introduce a substantial number of errors. Consider Example 7: even for a human scorer, it is impossible to tell with certainty, whether *p-* is the beginning of a repetition or an unfinished realization of e.g. *particle* that was repaired.

(7) `…high energy, p- physics prize for nineteen ninety-three by the European Physical Society…`

MICASE: col485mx069

Lastly, most files also contained several cases of mixed disfluencies, e.g. repetitions separated by *uh* or with a pause inserted between them. As such cases were rare in the MICASE, they were handled separately in the data preprocessing, but omitted from further analysis, as there would not be enough data for training and testing. In other corpora, however, their counts may be higher, allowing their inclusion in the training: Osborne (2011) reports that they constitute a large portion of longer disfluencies.

5.2.8.2 Statistic measures

The statistic measures described before were all collected from the preprocessed data, i.e. the model did not know whether a repetition of a certain word occurred at all, or how often it was preceded by *uh*. This is a substantially different approach from Adell et al. (2007) who used their concept of candidates, leveraging the fact that most disfluencies occur after a limited set of words in their corpus. The present classifier did not have this information and had to rely on frequential/probabilistic information only. This, on the one hand, is closer to the Noisy Channel view of disfluency placement, where disfluencies are added into an originally fluent output. On the other hand, it is also more psycholinguistically realistic, as it is unlikely that humans learn to place disfluencies after/before certain words. Rather, they may place them on the basis of contextual probability, as suggested in this work, smoothing the transmission and providing the listeners with an input that is easier to process. Furthermore, this view is compatible with the view of disfluencies as symptoms of errors in retrieval: speakers should be more likely to retrieve an incorrect item if they are attempting to retrieve a word that is unlikely or uncommon. On the other hand, enforcing the deterministic concept of candidates would mean that most words should be always retrieved correctly; errors would be limited to a small set of items only. This assumption is far from the psycholinguistic reality.

Most of the individual measures (bigram frequency, TP-D, TP-B, MI, G) were – following Schneider's (2014) methodology – extracted directly from the MICASE. Thus, they were more representative of the in-domain probabilities than the combined measure of surprisal which was based on overall probabilities. On the negative side, this meant that data sparsity was more of an issue given the smaller size of the training corpus. Because they were calculated from a pre-processed version, these measures did not require the splitting of the corpus into a training/validation/testing dataset: no information about the placement of disfluencies or their probability of occurrence could be leaked into the finished model.

The combined measure of surprisal was calculated in accordance with Chapter 2.3, i.e. trained using data from COCA and MASC and also applied to the pre-processed data (i.e. with disfluencies removed). The data associated with each word contained not only the final score, but also all the individual components (trigram probability, syntactic surprisal, semantic modifier). In addition, following the observation made in Chapter 4, the difference in the surprisal value of two neighboring words was calculated as:

$$D = I_w - I_{w+1} \qquad (41)$$

Items at the beginning of a turn were marked as sentence-initial. Additionally, the items following a dot in the transcript were labeled sentence-initial, too. This decision requires brief explanation since dots in MICASE denote short breaks (Simpson-Vlach et al. 2003). However, they are often placed at locations that would be sentence boundaries if the transcript was transformed into a written text, as shown in Example 8. For disfluency prediction, these items were viewed as fluent unless preceded by another disfluency.

(8) no i didn't. i've had one year of Honors Chemistry

and then i i would, then i would talk about the two-eight_ two-ninety-five. if you want more discussion about two-ninety-five, then tomorrow morning between nine and eleven

so that, works you know really pretty well for some people. and then you said you took the uh English too?

MICASE: adv700ju023

5.2.8.3 Alignment

After the files were processed, a semi-automated alignment was performed, aligning the pre-processed files with the raw ones, noting whether a disfluency preceded a given word or not. In the case of repetitions, the disfluency was marked on the first word that was not repeated, i.e. in the case shown in Example 9 the word *would* was marked as preceded by a disfluency.

(9) `and then i i would`

MICASE: adv700ju023

If a given word was preceded by a chain of several disfluencies, the automatic alignment was suspended to allow for manual coding, unless the chain consisted simply of several repetitions of the disfluent word.

After the alignment, the disfluency mark-up was conflated (i.e. one-word repetitions and two-word repetitions were moved to a single category “repetition”), the data was cleaned (removing the complex disfluency interactions) and separated into three datasets: 70% training data, 15% validation, 15% test data. The larger proportion of validation/test data in comparison to the more conventional 80:10:10 was chosen so that these datasets contain a sufficient number of disfluencies to be predicted thus increasing the representativeness of the results.

5.2.8.4 Classifier training and validation

The classification and regression tree algorithm was trained using its *scikit-learn* implementation (Pedregosa et al. 2011) with the Gini measure of impurity. The hyperparameters were optimized using the

GridSearch() function according to their performance over the development set as measured by a macro-averaged F_1 score.[11] The *GridSearch()* function implements exhaustive hyperparameter optimization, trying out all possible combinations of the optimized hyperparameters to find the best performing one. Such an approach is comparably computationally expensive, allowing only a limited range of parameters to be optimized. In this thesis, two parameters were optimized using *GridSearch()*: the minimum leaf size (following values were tested: 2,4,6,8,10,12,14) and maximum depth (the values tested were 3,6,9,12).

After the classifier was fitted using its performance on the validation set as an estimate of its generalizability, its actual predictive power was tested on the test set. As it did not have access to this data before in the training/validation phase, the performance measured there could be considered a good reflection of its performance over unseen data.

[11] Throughout this thesis, the micro-averaged F_1 is calculated by counting the totals of true positives, false negatives and false positives and using those to calculate the overall precision and recall. Macro-averaged F_1, on the other hand, is obtained by calculating the F_1 score from the average by-class precision and average by-class recall.

The multi-layer perceptron was also implemented using the *scikit-learn* package, with categorical cross-entropy as its cost function. A number of parameters was optimized and the best model was selected by the macro-averaged F_1 score. Table 7 presents the most important parameters of the setup.

Parameter	**Values**
Hidden layer configurations	(10,6,3), (10,8,6,4), (10,4), (10,2)
Alpha (α)	0.01, 0.001, 0.0001, 0.00001, 0.000001
Maximum iterations	5000
Early stopping tolerance	0.001, 0.0001, 0.00001, 0.000001

Table 7: Non-default parameters used in the training of the MLP classifier. These parameters include the number of neurons in each hidden layer of the model (parentheses surround individual models tested), the learning rate α defining the step size of each weight update, the maximum number of iterations of the backpropagation algorithm and the minimum improvement in the loss over the validation set needed to continue the training.

5.2.8.5 Baseline estimation

To assess whether the used predictors give the model an advantage in predicting disfluency occurrence and disfluency type, a baseline was estimated. This baseline corresponded to a model that would not gain any advantage from the predictors, though it would observe the frequency distribution of the individual disfluency types as well as the frequency of disfluency occurrence in the fluent text. Thus, if the model trained on the predictors would not outperform this baseline, it would suggest that disfluencies are independent of the predictors used.

5.3 Results

5.3.1 Data statistics

In the whole MICASE, approximately 2.8% of words were preceded by the disfluency types that were to be predicted. Among the disfluency types, repetitions, *um*s and *uh*s are represented more or less evenly, pauses are substantially less common. After cleaning the data of rows with missing values, the number of disfluencies shown in Table 8 remained for the classifier training, validation and testing.

Disfluency type	**Frequency**
Repetition (1 or more words fully repeated)	13481
Uh	11797
Um	13815
Pause	1899

Table 8: Disfluency counts in MICASE by type.

Because of the low frequency of the phenomenon to predict, oversampling of the disfluencies was performed in order to prevent the algorithms from defaulting to classifying everything as belonging to the dominant class. Such a behavior may occur in spite of a model's robustness to unbalanced samples. If the proportion of individual categories is too skewed towards one class, the classifier may overuse the dominant category as this will result in a better performance overall as measured e.g. by the misclassification error or the micro-averaged F_1.

As the current study also used a macro-averaged F_1 as its performance measure, it was observed that the models indeed tended to revert to classifying all cases as fluent. As a corrective measure, the minority categories were oversampled by repetition – i.e. their examples were included repeatedly in the dataset. Since the probabilistic statistics describing each item were noted independently of the other items and the actual order of the items was not used for the model training at this stage, the oversampling function also involved shuffling of the training examples in order to prevent the MLP from being influenced by the occurrence of an oversampled case in a batch.

The most frequent bigrams in the MICASE corpus are summarized in Table 9. They can be divided into three groups: contractions (*it's, that's, do n't*), syntactically bound units (*of the*) and potentially lexified multiword expressions (*you know*). Neither of these can be viewed as belonging exclusively to the genre represented in the corpus and only one is an example of a speech-specific expression. Contractions may appear in writing, too – even in formal styles, though they are often avoided there (Hyland & Swales 1999). On the other hand, the expression *you know* in the sense of a semantically largely bleached particle is limited to spoken communication. All in all, the five most common bigrams largely fulfil the expectations for any corpus of speech.

Bigram	Frequency
It 's	12898
That 's	8273
Of the	7559
You know	6854
Do n't	6602

Table 9: Most frequent bigrams in MICASE.

At the same time however, large majority (254553 out of 371311) of the bigrams in the MICASE only appear a single time in the corpus, which contains 10717 hapax legomena (out of 30337 word types observed). Given that, it does not come as a surprise that many bigrams (11620) have a forward probability score of 1, usually due to the fact that their first element is a hapax. The examples include expected pairs such as *Woody Allen* or *umbilical cord*, but also clear examples of items bound together only due to the limited size of the corpus, such as *loudness sharpness* or *multidimensional answer*.

Similarly, there are 12051 bigrams that have the backward transitional probability of 1. These, too, include logical pairs, such as *necrotizing myotis* or *in memoriam*, and random combinations brought by the structure of the corpus, e.g. *have benches* or *general riskiness*.

The MI score seems to be more robust to the influence of the corpus composition. Nevertheless, the five highest-ranking pairs (Table 10) do not necessarily seem like logical collocations. Moreover, the bigram with the highest score was likely a partial repetition and not a true bigram. Nevertheless, the mean MI score is 3.001 (n = 371311, SD = 2.792) with majority (89.35%) of the scores being larger than zero. This confirms that in spite of the noise in the highest-ranking bigrams, the data is not randomly distributed.

Bigram	**MI Score**
Lau Laurel	14.43
Ethan Ebner	13.74
Greedily expecting	13.33
Parodied juxtaposed	13.04
Plea bargaining	12.92

Table 10: Bigrams with the highest MI score in MICASE.

Despite being designed in order to eliminate the influence of syntax on the association metric, the G-score (mean = -1.488, SD = 1.539) did not seem to be capable of this in the given dataset. All five of the top scoring items, listed in Table 11, are clearly syntactically bound to each other. On the other hand, the G-score seems to perform better than the other measures in terms of assigning the highest scores to actually commonly co-occurring items rather than being confused by the hapaxes.

Bigram	G-Score
Of the	13.15
This is	12.63
It 's	12.46
In the	12.44
That 's	12.1

Table 11: Bigrams with the highest G score in MICASE.

In terms of surprisal measured by the methodology used throughout this thesis, the average value of information transmitted by a word was 2.732 (SD = 1.382, median = 2.477, not normally distributed).

The baseline was established at $F_1 = 2.8\%$ for disfluency occurrence prediction, macro-averaged $F_1 = 20\%$, (micro-averaged $F_1 = 94.7\%$) for combined disfluency occurrence and type prediction and macro-averaged $F_1 = 25\%$, (micro-averaged $F_1 = 30.7\%$) for the disfluency type selection only. In the disfluency selection task, the baseline F_1 scores for the individual disfluency types are:

- Pauses: 4.6%
- Repetition: 32.9%
- *Uh:* 28.8%
- *Um:* 33.7%

5.3.2 Disfluency statistics

The statistics observed in Chapter 5.3.1 were evaluated in order to explore the patterning of the individual predictors in the vicinity of the disfluencies and to verify whether the patterns observed in Chapter 4 hold true for the larger and more varied dataset as well.

Figure 12 shows that words which were preceded by a disfluency tended to have higher surprisal estimates than words which were not. Thus, as observed by the previous studies reviewed in Chapter 3.1, disfluencies tend to occur before less predictable items. Fluent text, on the other hand, is made of more predictable units. The observed difference in the average predictability of the fluent items as compared to those preceded by disfluencies is unlikely to be caused by chance (Wilcoxon-Mann-Whitney test of fluent vs. disfluent items with a one-sided hypothesis returned $W = 6.68 \times 10^{10}$, $p < 0.001$).

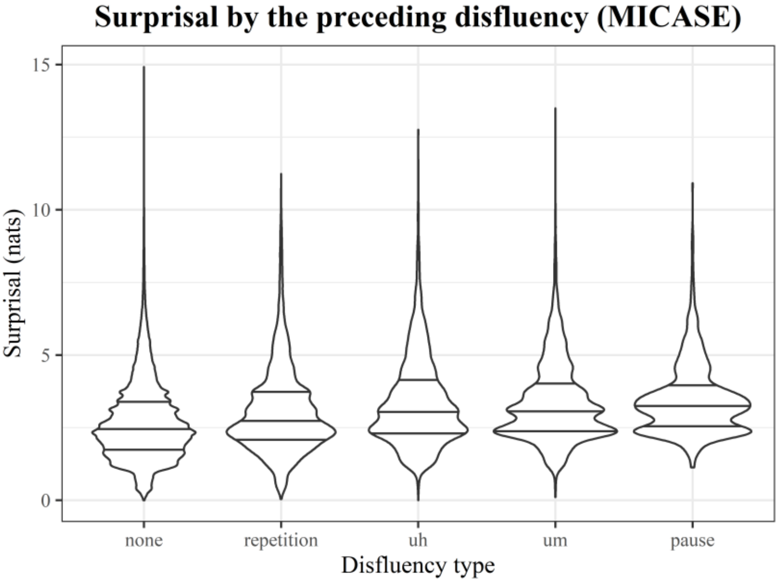


Figure 12: Surprisal of word w_i as related to the type of disfluency inserted before it. Surprisal measured after disfluencies had been removed.

Similarly, the difference in the local information transmission rate as linked to both various disfluency types and the contrast of fluent and disfluent speech persisted, as visualized by Figure 13. The mean change in the information transmission rate of two neighboring words is close to zero (-0.001). Considering only the fluent items, their average change in the amount of information transmitted by word w_i as compared to w_{i-1} is fairly similar to the overall average (0.015); disfluent items, on the other hand, differ substantially.

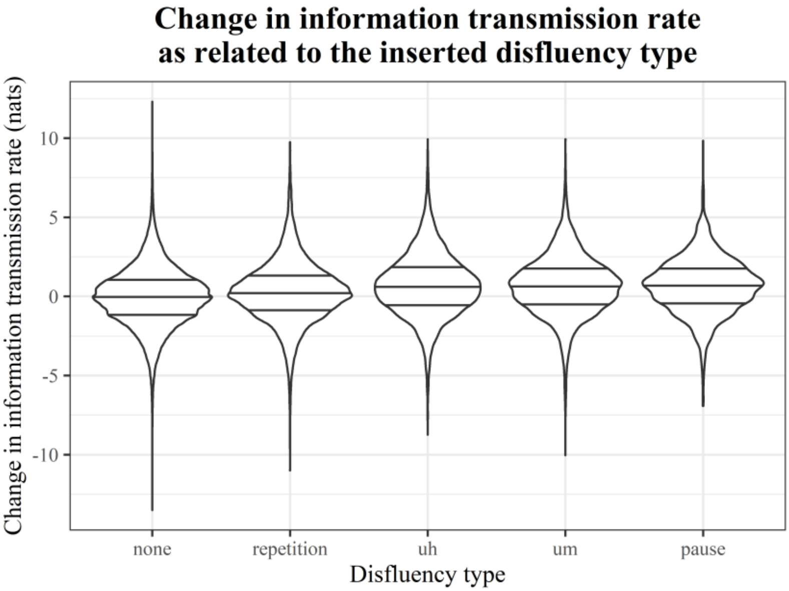


Figure 13: Difference in the surprisal of words w_{i-1} and w_i ($w_{i-1} - w_i$) as related to the type of disfluency inserted between them. Surprisal measured after disfluencies had been removed.

Items which were preceded by a disfluency were on average less predictable then the items before them, with a mean difference in the surprisal estimate of w_i and w_{i-1} being -0.548 nats. Thus, items which carry more information (in the information-theoretic sense) in comparison to their preceding context are more likely to trigger a disfluency. The observed difference is again unlikely to be due to sampling noise (Wilcoxon-Mann-Whitney test of fluent vs. disfluent with a one-sided hypothesis returned W = 6.3×10^{10}, $p < 0.001$).

This patterning continues across the scores, with the potential triggers of disfluency occurrence being less likely in the given context.[12] When comparing the observations reported here to those made within the JSCC corpus, a shared pattern can be identified: in both cases, repetitions are less distinct from the fluent speech than filled and unfilled pauses. In contrast to this, the magnitude of the difference between fluent and disfluent speech is smaller in the MICASE corpus when compared to the JSCC.

Lastly, it should be mentioned that in spite of the fact that fluent speech differs from the disfluent one with respect to the various scores assessed, there is still a substantial overlap between them. This is in agreement with the expectation that probabilistic measures are not capable of predicting disfluencies perfectly. However, the small difference between the individual disfluency types also raises the question whether they are distinct in terms of the measures used.

5.3.3 Disfluency prediction

After the statistics were collected and the individual files evaluated, a logistic regression model was first fitted in order to estimate the ability

[12] This is expectable given the inevitable collinearity of some of the individual measures. The correlation coefficients range from -0.02 (closest to zero: difference in surprisal $D_{I_w - I_{w+1}}$ as correlated to TP-D) to 0.66 (strongest correlation, surprisal of w_i as correlating with $D_{I_w - I_{w+1}}$).

of the predictors to predict disfluencies as such and distinguish them from fluent speech. To fit the model, the numeric predictors were centered and the model was fitted and evaluated on the whole dataset, rather than on the training-validation-test triplet, as the purpose of this model was only a preliminary evaluation. The fitted model (an overview of the full list of coefficients and associated statistics is in the Appendix A.3) performed better than null model (assigning all cases to the majority class) in disfluency prediction (deviance decrease of 23184 with a decrease in degrees of freedom of 50, the associated p-value being below 0.001). Still, its Nagelkerke's R^2 was 0.09, indicating that the relationship is only limited and that the predictors used should not be expected to provide a complete explanation of the use of disfluencies, certainly not under the assumption of linearity. Additionally, given that some of the predictors were identified as collinear, the individual coefficients may not always be reliable. Especially problematic is the collinearity of syntactic surprisal and the overall surprisal, which is inevitable given that these two measures are correlated by definition; second potentially problematic correlation may be that of the g-score and n-gram surprisal ($r = 0.55$). However, in spite of these issues, it is obvious that the position of an item is very likely to play a role in disfluency prediction: sentence-initial items were more often disfluent, with a coefficient of 1.786 (std. error 0.04, $p < 0.001$). This corresponds to $\Delta odds = 5.96$, i.e. sentence-initial items are 6 times more likely to be disfluent than sentence-medial or final elements.

Assessing the importance of the individual variables using the *caret* package for R (Kuhn 2017) additionally identified the MI-score (coeff. -.37, std. error 0.006, $p < 0.001$) as an important predictor. The negative coefficient for the MI-score indicates that tightly bound words are less likely to be separated by a disfluency. However, the coefficient of the G-score points in the opposite direction. This suggests that at least a part of the effect may be traced to the syntactic rules increasing the rate of co-occurrence of these words (though this coefficient might be influenced by the identified collinearity). The MI-score additionally interacts with forward transitional probability: items with high MI-score and high forward probability are substantially less likely to be disfluent (coeff. -5.45, std. error 0.32, $p < 0.001$) than their highly unrelated and unlikely counterparts. The coefficient of surprisal is significantly different from zero, too (coeff. 0.41, std. error 0.04, $p < 0.001$), suggesting that highly unlikely items should be disfluent. However, due to the collinearity mentioned before, this coefficient is not entirely reliable. Finally, the magnitude of the difference in the surprisal between two neighboring items has small influence on the disfluency use if other factors are controlled for, thus leaving hypothesis 2.2 unconfirmed.

The aforementioned Nagelkerke R^2 of the model (0.09) suggests that a large portion of the variation between fluent and disfluent realizations is not to be explained by the predictors used or their first-order interactions. The following chapters will thus report the performance of models which are capable of handling more complex higher-order interactions at the cost of being less readily interpretable in comparison to the logistic regression.

The performance of the CART classifier was better than the baseline. The best iteration of the CART algorithm (which consisted of 7 levels and 127 binary decision nodes) achieved a macro-averaged F_1 = 26.13%, at the cost of the micro-averaged F_1 = 91.60%). Still, it did not match the results of Adell et al. (2007) when jointly predicting the disfluency occurrence and disfluency type. Concretely, the CART algorithm was able to predict 23.3% of the locations where disfluencies occurred. However, only 9.7% of the predicted disfluencies were actual cases of a disfluent speech, the rest being false positives. The classifier was capable of learning that disfluencies constitute only a relatively small proportion of the input (6.5% of the items were predicted to be preceded by a disfluency) and identifying some contexts where disfluencies are more likely to occur. However, it was not able to recognize these contexts of occurrence reliably. It was observed that the classifier performed the worst in predicting repetitions, which may be caused by their similarity to fluent speech in comparison to other disfluency types, as observed in Chapters 4.4.2 and 5.3.2. Even at such a small tree depth, some signs of overfitting were found, with some nodes containing few examples (though still sufficient to produce the minimum leaf size of 4). Concretely, 43 (34%) of the nodes/leaves of the tree contained less than 1% of the data each. These would be potential candidates for pruning, which, however, was not supported by the *scikit-learn* implementation at the time of writing.

The MLP classifier did not perform substantially better. Rather, its performance was similar to that of the CART classifier: it understood the rarity of disfluencies in the data, yet did not find an efficient way of predicting them on the base of the predictors used. In terms of the performance measures used, the best model had a macro-averaged F_1 of 25.44% and micro-averaged F_1 of 91.1%. Thus, it was slightly better than the baseline model, yet it did not outperform the CART classifier. Similarly to CART, MLP was capable of recognizing approximately a quarter of the locations where a disfluency should occur (25.11% exactly). At the same time, it matched the CART's precision, with 9.6% of the predicted disfluency locations matching the actual ones. Table 12 shows the results.

Model	**Macro-averaged F_1**	**Micro-averaged F_1**
CART	26.1%	91.6%
MLP	25.4%	91.1%

Performance on individual disfluency types (F_1)

Model	**Pause**	**Repetition**	***Um***	***Uh***
CART	2.9%	0.2%	9.2%	0.6%
MLP	2.7%	0.1%	9.3%	0%

Table 12: Performance of the classifiers in predicting and selecting disfluency types.

To conclude, the two classifiers clearly could recognize some of the locations of the individual disfluency types. This is reflected by the macro-averaged F_1. Still, repetitions were particularly difficult to predict for both of the set-ups: their F1 score remained below 1%.

The observed performance shows that the occurrence of disfluencies can be predicted by purely probabilistic measures only to a limited degree. In order to gain a deeper insight into this claim – and to explore the assertion that the disfluency type may be predicted by these measures, too, the disfluency prediction was split into two tasks, as done by Ohta et al. (2008): the prediction per se and the selection of the correct disfluency type. In occurrence prediction, the macro-averaged F_1 score of the best model (CART, see Table 13) was 56.4% (F1 of disfluencies 14.6%, overall misclassification rate 7.2%), outperforming the baseline, yet still suggesting that disfluencies are largely independent of the predictors used.

Model	Macro-averaged F_1	Micro-averaged F_1	Disfluency F_1
CART	56.4%	92.9%	14.6%
MLP	55.7%	93%	13.6%

Table 13: Performance of the classifiers in predicting disfluencies.

Decision trees were also more efficient in disfluency type selection (Table 14) and reached a micro-averaged F_1 of 38% (macro-averaged $F_1 = 43\%$). Thus, the algorithm was capable of predicting the type of two out of five disfluencies. The classifier was efficient in identifying repetitions ($F_1 = 56.9\%$); on the other hand, it was overly conservative in predicting *um*. There, it did not outperform the baseline, as it only predicted 59 occurrences of *um* in the sample of 6328 disfluent locations instead of the actual 2144 present.

Model	**Macro-averaged F_1**	**Micro-averaged F_1**
CART	43%	38%
MLP	39.8%	36.1%

Performance on individual disfluency types (F_1)

Model	**Pause**	**Repetition**	***Uh***	***Uh***
CART	22.5%	56.9%	42.2%	3.2%
MLP	20.9%	55.6%	39.5%	14.5%

Table 14: Performance of the classifiers in selecting disfluency types.

Unlike Adell et al. (2007), I did not find predicting disfluencies by the characteristics of the word preceding the location of a potential disfluency more efficient. On the contrary, when tested with the CART classifier, the performance of the disfluency occurrence prediction decreased from $F_1 = 14.6\%$ to $F_1 = 4\%$, the performance of disfluency type selection fell below the baseline.

5.4 Post-hoc models

Evaluating the classifiers brought the representativeness of the statistics used for their training into question. Given the structure of the corpus they were collected from (MICASE) and considering the large proportion of unique bigrams and hapax legomena, it is likely that the statistics collected may contain a large proportion of noise. The limited size also defines the resolution to which two items may be distinguished – if two items share their frequency in a corpus of 1.5 million words, this will not necessarily be so in a larger corpus of e.g. 400 million words.

In order to obtain more representative scores (bigram frequency, TP-D, TP-B, MI-score, G-score), their collection was repeated on the COCA corpus (Davies 2008-) used for the training of the language model from which the surprisal estimates were derived. These newly collected statistics were then used within the aforementioned framework.

Additionally, as it has been observed in the disfluency type prediction task that the classification trees and multi-layer perceptron perform differently with respect to various disfluency types, a compositional model was created, mixing the predictions made by each of the classifier. This compositional model operated by extracting the probabilities assigned by the classifier to each of the predicted disfluency types d given the characteristics of each word w_i. These probabilities were then combined through elementwise multiplication:

$$\hat{p}_{CART}(d|w_i) \times \hat{p}_{MLP}(d|w_i) \qquad (42)$$

Such a mixing favored those cases where both of the classifiers assigned high probability, rather than being overly influenced by a high score given only by one of the models. The values yielded by such a function are obviously not proper probabilities (and would not be even after taking the square root, thus calculating the geometric mean), as they may not sum to 1. After the probabilities were mixed, the disfluency type with the highest score was selected as the one that should precede w_i.

Moreover, the concept of a candidate as used by Adell et al. (2007), was implemented. Concretely, the statistics about which forms tend to be preceded by disfluencies were collected and 50 items that were the most frequent triggers of disfluency use were classified as candidates. The classification was done in two distinct ways: first, the candidate category was only binary, distinguishing only between items that are (or are not) candidates. Second, the candidates were lexified, i.e. in addition to the binary definition, the exact lexical item was stored for the candidates. All lexical items that did not belong to the candidate group were grouped under one label.

As a last change, the extended context was included into the classification task by using a recurrent neural network as a classifier.

5.4.1 Recurrent neural networks

Recurrent neural networks (RNNs) are conceptually similar to multilayer perceptrons in that they, too, employ a network of cells (artificial neurons) to which the input is fed and which process it and pass it further toward the output. The crucial advantage of RNNs over MLP is that they recognize the fact that the input (in this case linguistic) is not a series of independent data points, but rather an event unwrapping in time where an input may depend on the one preceding it. Thus, the states of the neurons in the hidden layers are not reset for each input. Rather they are used to add some elements from the old input to the new one.

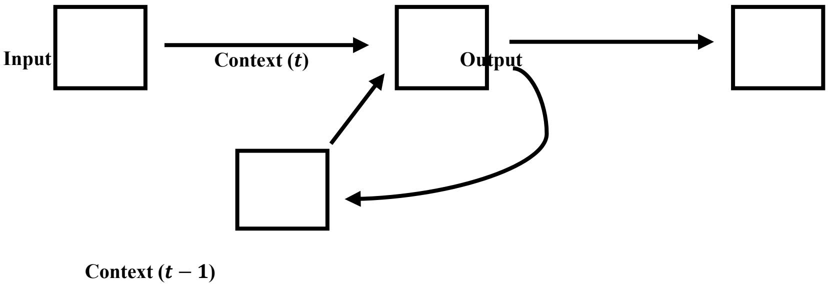


Figure 14: Simple RNN architecture.

A simple RNN architecture, also called Elman network (Elman 1990), is visualized in Figure 14. In such a network, the input at time t is created by concatenating the actual input vector (e.g. the semantic space representation of a word) with the output of the hidden layer at time $t-1$. Thus, each new input also carries some elements of the previous one. The hidden state at time t equals:

$$h_t = g(W_h x_t + U_h h_{t-1} + b_h) \tag{43}$$

where g is the activation function, W_h, b_h the bias unit and U_h the weight matrices applied to the input x_t and history h_{t-1}.

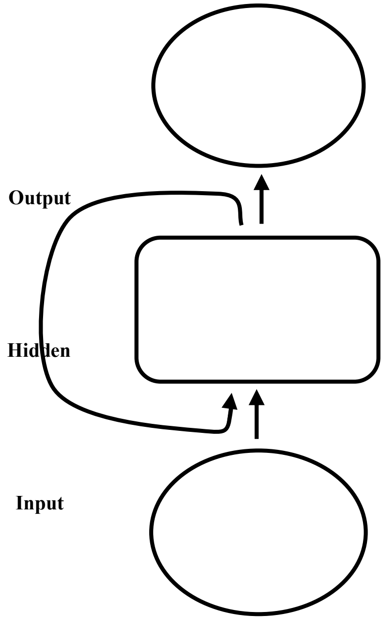


Figure 15: Fully connected recurrent neural network.

Importantly, the Elman network truncates its history to $t - 1$ only, assuming that full history can be approximated in this way. In this respect, it stands in contrast to the fully connected recurrent neural network, presented in Figure 15. In the case of the fully connected network, the whole previous history is included both in the calculation of the output and in the subsequent learning via the backpropagation through time algorithm. This algorithm is conceptually similar to the standard backpropagation, except the error is propagated backwards not only through the layers, but through time as well. Practically, in order to limit the computational complexity and speed up the training, the context that is provided and backpropagated through is often limited even in fully recurrent models.

The consequences of this are obvious – the influence of items outside the limited context is completely removed. Moreover, there is an additional less obvious though no less inherent danger in the architecture of the recurrent neural networks, the issue of vanishing gradients (Pascanu et al. 2013; Bengio et al. 1994). Considering how the previous input is included in the model, words further in the history are inevitably overpowered by more recent ones. Thus, the RNN is often not capable of capturing long-range dependencies, even if these are included in the available history. Further advances in the area were made in order to remedy this issue, such as the long short-term memory (LSTM, Hochreiter & Schmidhuber 1997) network or networks built from the gated recurrent units (GRU, Cho et al. 2014b). Still, the default RNN has been shown to yield very good results on tasks related to computational linguistics (Mikolov et al. 2010), psycholinguistics (Frank et al. 2015) or outside the linguistic domain altogether (Pascanu et al. 2013).

The RNN classifier was implemented through the API *Keras* (Chollet et al. 2015) with *TensorFlow* backend (Abadi et al. 2015). Table 15 summarizes the optimized parameters as well as the crucial elements of the setup, different from the default Keras setup.

Parameter	Value(s)
Optimizer	Adam (Kingma & Ba 2015)
Loss	Binary/Categorical cross-entropy
Layers	RNN: 32, 64
	Dense (2/4)
Max. epochs	5, 10, 15
Batch size	16, 32, 64, 128, 1024

Table 15: Parameters used in the set-up of the RNN classifier. Where values are separated by slash, the left value applies to disfluency occurrence prediction, the right one refers to disfluency type selection. Values separated by commas are listed options that were tried out. The RNN and dense layer were combined, i.e. the output from the RNN layer was propagated to the dense layer which determined the output.

5.5 Results of post-hoc tests and discussion

Using the COCA as the corpus from which the individual statistics were collected proved beneficial for the performance of the classifier in predicting the disfluency type. The CART classifier was more powerful in selecting the correct disfluency than the remaining classifiers, though the difference is marginal only, as shown by Table 16. The combined CART×MLP classifier did not perform better than the CART algorithm on its own.

The prediction of disfluency occurrence per se remained largely unchanged with the best performing model reaching a macro-averaged F_1 of 56.8% (F_1 in disfluency prediction equaling to 15%) and an overall accuracy of 93.2%. These scores were achieved by the MLP classifier trained using oversampling to prevent it from minimizing the cost by classifying all occurrences as fluent, without employing the lexical information or information about membership in the candidate group.

Model	Macro-averaged F_1	Micro-averaged F_1
CART	43.1%	48.9%
MLP	39%	43.5%
RNN	39.4%	48.5%
CART×MLP	42.7%	48.8%

Performance on individual disfluency types (F_1)

Model	Pause	Repetition	*Uh*	*Um*
CART	6.6%	61.1%	33%	47%
MLP	22.5%	61.4%	35%	33.6%
RNN	2.4%	59.9%	40%	43.8%
CART×MLP	24.6%	61.6%	33.9%	46.6%

Disfluency type selection

Table 16: Performance with COCA-based statistics.

This classifier had a recall similar to the one reported by Ohta et al. (2008), that is 23.6%. However, its precision was somewhat lower, 11%. Post-processing the predictions of this classifier by a simple procedure that converted all words which were not candidates to the category of fluent ones (imitating the results which would be obtained by a classifier that would not process the non-candidates at all, as suggested by Adell et al. 2007) did not change the performance: both correctly and incorrectly predicted disfluencies were affected to approximately the same degree.

The partial independence of disfluency placement on probabilistic measures could be due to the fact that disfluencies are governed by a mechanism that employs other features, too: the position of a word in the sentence is a likely candidate. Even though some of the previous research has suggested that disfluencies occur as signals pointing towards phenomena that should be mirrored by probabilistic measures – unpredictable/uncommon words (Schnadt & Corley 2006; Levelt 1983; Beattie & Butterworth 1979) or something complex (Watanabe et al. 2008; Arnold et al. 2007) – it seems that only one part of the story is told by the numeric expressions of that information as used in this thesis. The other part, centered around factors that cannot be measured by the scores used here (or perhaps any numeric scores at all) seems to be more dominant in this case.

On the other hand, the disfluency type indeed is likely to be partially guided by the predictability of the upcoming input. However, the prediction process cannot be summarized into a simple rule assigning a disfluency type to a particular predictability score, allowing choice even in precisely defined contexts. Here, too, a non-numeric factor may play a role which is more important than the scores calculated here.

Though the measures used do not contain enough information to reliably predict the type of every disfluency in the dataset, the classifier was clearly not assigning the disfluency type randomly. Additionally, its performance revealed a particular pattern – the more frequent disfluency types were selected substantially more reliably than the rarest one in the MICASE corpus – silent pauses. This raises the question of whether the results were not influenced by the training data quantity. The MICASE is not a particularly large corpus and only a limited number of disfluencies were identified and available for training. However, even though the present dataset was substantially larger than the one used by Qader (2017), the results did not differ substantially, suggesting that an even larger dataset may be necessary.

Since including the concept of a candidate into the model – that is retaining the lexical information about the 50 types most frequently preceded by a disfluency – did not substantially improve its performance, it seems likely that the model would not benefit substantially from the full lexical information being included either. This is especially likely given to the large proportion of hapaxes in the MICASE corpus.

The difference between the classifier performance with statistics collected in the domain used for testing (in the MICASE corpus itself) and in a general corpus representing a broad range of genres and registers (COCA) is somewhat striking. It has previously been suggested that general corpora are better capable of predicting language effects connected to the probabilistic information stored in human brain

(Sayeed et al. 2015). Thus, statistics collected in the COCA corpus should be a better source of data for predicting probabilistic phenomena. Additionally, as COCA is 200 times larger than MICASE, each of the statistics should also be more representative. The small difference between the models is thus unexpected. A possible explanation could be that a large part of the vocabulary used in MICASE is so specialized (as suggested by the high number of hapaxes) that it appears only in the academic section of COCA. In such a case, little advantage would be gained by including domain-general rather than domain-specific data.

5.6 Conclusion

In this chapter, I have shown that the differences between the surprisal profile of fluent and disfluent speech, observed in Chapter 4, can be observed in a larger dataset, too. The approach used to explain the occurrence of disfluencies had two main outcomes. First, it has been shown that probabilistic (and probability-related, i.e. the information-theoretic) measures are significant though weak predictors of the occurrence of disfluencies in speech. Among other, the influence of surprisal, suggested in Study I, was observed. Importantly, support for the importance of surprisal and its local change is found if considering them in isolation (Chapter 5.3.2). However, when controlling for other factors, the picture is not as clear (5.3.3). The coefficient of surprisal is affected by collinearity, the difference in the surprisal of two neighboring words has only small impact on disfluency placement. Thus, though disfluencies do tend to occur before less predictable items, their function as surprisal smoothers is limited and may only be a side-effect of the true cause of their occurrence. In such a case, the contribution of surprisal to disfluency prediction may be that of a proxy for another predictor, such as the complexity or novelty of the upcoming output. On the other hand, the type of the disfluency used does seem to reflect the probability of the upcoming text to a larger degree.

The degree to which the disfluency prediction succeeded, while being comparable to other attempts made in previous research, raises the question of the importance of other aspects that were not included in the prediction at this point. One likely candidate are the structural properties of the text, as suggested by previous research (Schneider 2014; Bortfeld et al. 2001; Maclay & Osgood 1959). By excluding this information, the classifiers used for disfluency prediction may have lacked important input.

In addition to showing that surprisal may indeed play a role in disfluency placement, this study provided some support to previous observations that mental chunks are less likely to be separated by a disfluency. This was reflected in the negative coefficients of the MI-score and bigram frequency, though other predictors suggested quite the opposite. Most notably, the forward transitional probability suggested that items which are highly probable given their immediate predecessor are more likely to be preceded by a disfluency than unlikely ones. This clash of trends may be related to the directionality of the individual scores – the MI score captures not only the relationship of word w_i to

w_{i-1}, but also the relationship between w_{i-1} and w_i. Thus, it may be a better expression of the degree of relationship between the items than the strictly unidirectional forward transitional probability.

Given the performance of the RNN model – most notably its similarity to simple MLP/CART models, the present data suggested that there is little influence of the extended context on the probability of disfluency use. Importantly, this claim is restricted to the way the context was presented, i.e. in the form of scores expressing the collocation strength, predictability and rareness of the individual items.

To conclude, some of the disfluencies could be predicted by the scores used and the previously reported relationships found some support. However, the performance was substantially different from what was suggested by a model trained on a different dataset (Chapter 4). Further improvement is unlikely to occur as a consequence of increasing model complexity (since even comparably complex models offered only limited gain in terms of the F_1 score). Rather, it seems likely that efficient disfluency prediction requires additional input beyond what was used here. Considering the importance of the only predictor tied to the structural properties of the text – the position of an item at a beginning of a sentence, I expect that models which can employ such structural information will perform better on the task at hand.

Chapter 6
Study IIb: Structural factors in disfluency prediction

The previous chapter built on the observations made in Chapter 4. Viewing disfluencies as potential surprisal smoothing agents, it attempted to predict them in a natural text from which they were first removed. The prediction was based on the properties of an upcoming item, estimated by the model adapted from Mitchell (2011) and described in Chapters 2.2 and 2.3, as well as through a number of measures devised to express the strength of the relationship between two neighboring items. These were inspired by the approach employed by Schneider (2014) in order to explore her usage-based hypothesis of hesitation placement.

The limited extent to which the disfluency prediction succeeded on the basis of these exclusively numeric predictors suggests that disfluencies are governed by rules more complex than the predictability of a string given its context. As a consequence, the information-theoretic hypotheses (notably the UID hypothesis) may be unable to explain a substantial proportion of the disfluencies observed in natural linguistic data and the surprisal estimate may be only a correlate of the underlying function.

The previous chapter simplified the language data to a fairly abstract array of numeric values expressing the surprisal of each item or the association strength between the two elements of a bigram as measured by the formulae presented in 5.2. Thus, some less abstract categories which bear influence on language production were left out or translated into the numeric expressions used. However, given that these numeric expressions necessarily conflated multiple individual factors into one number, it is possible that this simplification was already too radical. A suggestive example is the position within sentence, which was identified as an important factor in disfluency prediction. If the sentence-initial locations were not explicitly labelled, but solely encoded in the surprisal values, this trend would be much more difficult to isolate and employ in prediction.

This chapter will explore the importance of structured higher-level information for disfluency prediction. So far, the predictors contained syntactic surprisal as an expression of the cognitive load caused by the probability of the observed linguistic data from the syntactic point of view. This purely numeric expression conflates the syntactic structure of the input: any given value of the syntactic surprisal may potentially appear both at the boundary of constituents and within them. However, speakers are aware of these boundaries in their language production, as shown e.g. by Clark & Wasow (1998) in the context of disfluencies. Clearly not all disfluencies are positioned at phrase boundaries (Bortfeld et al. 2001: 137): some phrase boundaries are even very unlikely to contain a disfluency. For example, only 1.8% of the 6317 disfluencies observed by Schneider were located at the boundary of a verb phrase (Schneider 2014). Other phrasal boundaries, on the other hand, are particular attractors (Schneider 2014; Bortfeld et al. 2001; Goldman-Eisler 1961; Maclay & Osgood 1959).

It is precisely this kind of information that this chapter attempts to introduce into disfluency prediction. It presents a way of including syntactic information in the form of chunking. Additionally, it introduces an intermediate level of abstraction between the syntactic constituents and lexical information: the part-of-speech categories. These have been identified as predictors of disfluency placement by previous research (Gráf & Huang 2019; Pfeiffer 2014). Before the results are reported, a brief summary explaining the motivation to employ coarser measures of processing complexity (such as those based on POS-bigram frequency) is presented, together with a discussion of the cognitive reality of part-of-speech categories, which is a necessary prerequisite of the proposed approach.

6.1 Structural influences on disfluency placement

Previous chapters observed that disfluencies tend to appear in informationally denser contexts. From the information-theoretic perspective, especially within the UID framework, this could be understood as an information transmission smoothing approach: introducing additional time in order to counterbalance an upcoming spike in surprisal. So far, the surprisal was estimated using the measure combining syntactic, semantic and n-gram language models. However, prior research has shown that the influence of the UID hypothesis can also be observed when less fine-grained estimates of processing complexity are used, such as the proportion of complement clauses after a specific verb (Jaeger 2010). These coarser measures have the advantage of being calculated from larger – and thus more representative – samples.

In addition to employing the POS-tags as categorical predictors, this chapter uses them to draw one such coarser estimate of processing complexity, by tracking POS-tag and POS-tag-bigram frequencies. When observing the relationship between the POS-tag/POS-tag-bigram frequency (as a variant of the association scores defined in Chapter 5.2) and disfluencies, it is expected that the more frequent POS tags and POS-tag bigrams are less likely to be preceded by a disfluency. They should be easier to both retrieve and comprehend and thus less likely to require informational smoothing. The expectation is thus the same as in the case of bigram frequency in Chapter 5, only the observation is carried out on a higher level of abstraction.

Importantly, if speakers should distribute disfluencies on the basis of the parts of speech, they must be sensitive to (though not necessarily aware of) their distribution. Thus, if they should not store the information about the word class category of an item, they would not be able to use disfluencies as smoothing agents governed by the probability/frequency of that word class. This makes the psycholinguistic reality of parts of speech a necessary prerequisite.

Even though some approaches, such as the Radical Construction Grammar (Croft 2001) would not consider parts of speech an inherent category of language, they would still view them as emergent categories of linguistic structure acquired by the analysis of constructions (Diessel

2015).[13] The structured representation of lexemes is supported by the evidence from the research on the slip-of-the-tongue phenomena. It has been shown that in cases where one word is replaced by another, the word class is almost unanimously retained (Harley 2006; Bock & Levelt 1994; Fromkin 1971). Although there does not seem to be a single neural marker of word class (Federmeier et al. 2000), some distinction seems to be present: different cortical regions are activated in response to the concepts of objects and actions which would usually be expressed through nouns and verbs, respectively (Błaszczak & Klimek-Jankowska 2016).

Whether an item is assigned to a pre-existing part of speech in the language acquisition process, or whether these categories emerge on the basis of language experience, speakers should store information about the frequency/probability of each of these categories, for example about the likelihood that a verb will occur after another verb. This could be viewed as a manifestation of awareness about the distribution of syntactic units at a low level of abstraction. This awareness may be used in the placement of disfluencies – and by extension in their prediction. Additionally, the higher level of abstraction, accessible through the

[13] These emergent categories may not be identical to those defined by non-constructionist linguistics for English.

sequence of part-of-speech labels allows for a more representative quantification of rare lexemes.

Assuming the existence of traditional parts of speech, it is still necessary to select the set to be used. It “has been recognized for centuries that word classes […] are fundamental to the grammatical systems of human languages” (Kemmerer 2014: 28) and as a consequence, they are featured in some way in most linguistic theories and occupy a prominent position in other branches of research. Still, the number of labels and the rules for their assignment differ substantially even within the individual disciplines. The differences are especially pronounced in terms of the non-basic categories, that is those beyond the noun-verb-adjective triad. There, a major divergence exists between approaches arguing for language-specific categories and typologically based approaches searching for the universally present parts of speech.

The present study uses language-specific categories as devised for English. Two of the most popular tagsets used include the more concise list of part-of-speech categories defined and used by the Penn Treebank Project and a substantially more detailed CLAWS7 tagset. Although these two tagsets overlap partially in their definition of the main categories, a one-to-one or many-to-one mapping between them is not possible as the additional labels of the CLAWS7 tagset are not sublabels of the Penn Treebank tagset. In order to provide more fine-grained information about the data, the CLAWS7 tagset was used in this study. Naturally, as the number of categories increases, the number of examples belonging to each category decreases. With a large number of categories, it is highly likely that some of them will be represented by only a limited number of cases, or not at all. If the dataset cannot be further extended, it may be necessary to collapse some categories to minimize the influence of random noise on the performance.

There are several levels of abstraction between the parts of speech and the complete syntactic tree. While a full representation of the syntactic structure may be useful for an analysis of individual dependencies, it is also comparably more complex. Additionally, given the corpus size, the number of representations of the individual subtrees is likely to be limited. In order to avoid this issue, an intermediate approach was adopted, the so-called IOB-chunking. For each word, it determines whether it occurs i(nside), o(utside) or at the b(eginning) of a larger chunk. Here, chunks are not defined as mentally represented collocations (as in the usage-based approaches, e.g. Bybee 2010), but rather as larger – though not recursive – syntactic units at the phrase level. Thus, in addition to the part-of-speech label of each item, the classifier also received information about the position of that item with respect to the predefined syntactic chunks. An example of an IOB-tagged text is given below in Example 10.

The influence of structural factors was suggested already by Maclay & Osgood (1959): they reported that repetitions were more likely to occur with function words while unfilled pauses tend to cluster before lexical words. Filled pauses, finally, occur with function words and lexical words at approximately the same rate. Similarly, some syntactic boundaries are more likely to attract disfluencies than others:

noun phrase boundaries attract disfluencies (Schneider 2014; Bortfeld et al. 2001; Goldman-Eisler 1968; Maclay & Osgood 1959) while verb phrase boundaries are argued to repel them (Schneider 2014; Maclay & Osgood 1959). These boundaries are identified by the POS-chunking; its inclusion should thus improve the performance of disfluency prediction by providing the classifier with awareness of the position of an item in the syntactic structure.

6.2 Method

The data used for disfluency prediction was the same as in Study IIa. It consisted of the MICASE corpus data processed by the script estimating the surprisal of each word. This was combined with the additional predictors derived from the previous research and presented in Chapter 5.2, based on the COCA. Additionally, the part-of-speech tag for each of the items was assigned by the CLAWS tagger (Garside & Smith 1997).

The chunking was done by a maximum entropy tagger, based on the implementation in the *NLTK* package (Bird et al. 2009). The default tagger was further trained using the CoNLL-2000 dataset (Tjong Kim Sang & Buchholz 2000) and achieved an accuracy of 95.5% over the test set. The distinguished tags are listed in Table 17.

Tag	Meaning	
B-NP	Beginning of a	noun phrase
B-VP		verb phrase
B-PP		prepositional phrase
I-NP	Inside a	noun phrase
I-VP		verb phrase
I-PP		prepositional phrase
O	Outside the distinguished chunks	

Table 17: The distinguished IOB-chunk tags.

Each word in the MICASE corpus was thus labelled with its POS- and IOB-tags in addition to the scores collected in Study IIa. Example 10 below visualizes the outcome of this procedure. The labels and scores were then used as an input to both the disfluency prediction and disfluency type selection tasks. Similarly to Study IIa, a (comparably) simple explanatory model was initially created. Then a more complex classifier (in this case the RNN-based model) was trained to be used in prediction.

(9)

Word	you	‘re	from	Hartland	Michigan
POS-tag	PPY	VBR	II	NP1	NP1
IOB-tag	B-NP	B-VP	B-PP	B-NP	I-NP

MICASE: adv700ju023

6.3 Results

6.3.1 POS-tags

The CLAWS tagger assigned 237 different tags to the individual items in the data, including the “ditto tags” which mark highly co-occurring multi-word sequences as one unit. Out of these, 67 tags occurred fewer than 35 times which would be the expected number of occurrences needed to observe one disfluency if these were randomly distributed. Collapsing the ditto tags reduced the number of tags observed in the data to 112. Given that the frequency distribution of individual tags is clearly long-tailed (cf. the density plot in Figure 16), the frequencies were log-transformed before further processing.

Still, a simple plot of the frequencies of individual tags against the ratio with which they were preceded by a disfluency shows that there is likely little relationship between the frequency of a word’s part of speech and the likelihood that it will be fluent/disfluent. Figure 17 shows the ratio with which individual POS-tags were preceded by disfluency. Neither the distribution of the individual points nor the LOESS curve suggest a strong relationship. Pearson’s *r* does not suggest any relationship at all, either. It points to a clear lack of a linear trend with its coefficient of $r = 0.03$ (t = 0.58, df = 229, p = 0.56). On the other hand, the Spearman correlation (which is more sensitive to non-linear trends and tolerant to the heteroscedasticity observed in the data) would suggest that there is in fact a negative relationship between the two variables: (rho = -0.48, S = 3054000, p < 0.0001). This would

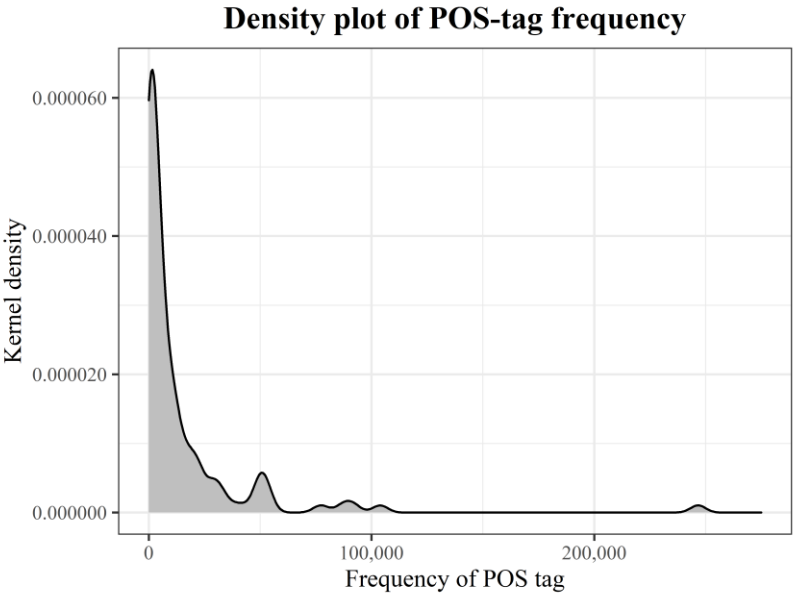


Figure 16: The distribution of the POS-tag frequencies in the MICASE corpus. Similarly to word frequency distributions, most tags are rare, few are highly frequent.

mean that more frequent tags are actually more prone to be preceded by a disfluency. Closer inspection revealed that this is linked to the absence of categorically fluent tags with frequency above 1300 (roughly 7 on the logarithmic scale, with the notable exception of the tag GE[14]) and

[14] This tag is used for the Germanic genitive marker *'s*. It obviously does not allow a disfluency to precede it.

the slight decrease in the ratio of fluent realizations in this area of the chart.

On the other hand, the less frequent items (with frequency below 12, corresponding to 2.5 after the log transform) are the only ones with a fluency rate below 0.9, though this is partially related to their frequency itself as a single disfluent occurrence causes a large change in the fluency ratio.

Similar trend is observed when shifting the perspective to the preceding POS-tag. The Spearman correlation is even more pronounced

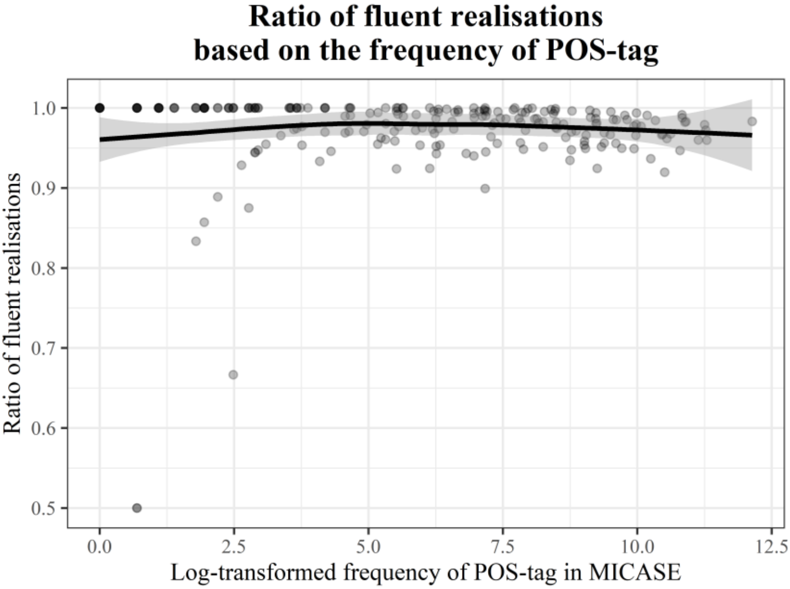


Figure 17: Log-transformed frequency of POS-tags as related to the ratio of fluent realizations of words labeled with that tag.

at rho = -0.53 (S = 3193400, p < 0.0001), though a linear trend is still not present (Pearson's r = -0.05, t = -0.81, df = 230, p-value = 0.42). The trend identified by the LOESS curve (Appendix A.4) is very similar, though it does not exhibit the same rise in fluency rates in the low frequency area (log-transformed frequencies between 0 and 5).

In order to decrease the amount of noise in the data caused by the presence of rare tags, the tagset was simplified. To achieve this, the ditto tags were collapsed by removing the numbers showing the position of an item in the identified multiword sequence (thus, a ditto tag II31 would become simply II. In such a way, the number of tags was reduced to 112.

The impact of this step is visible in the plot in Appendix A.5. As ditto tags were rare compared to the general tags, the most affected part of the chart was the one with log frequency below 2.5 (frequency below 12). There were three exceptions to the otherwise exclusively fluent realizations of the rare items. Two of them overlapped with log frequency 0.69 (actual frequency 2) and fluency ratio 0.5, third has a frequency of 12 and fluency ratio of 0.67. The rarity of these items prevents any strong conclusions, however.

Excluding the rare items and keeping only items with frequency above 35 (which should have at least one of their occurrences preceded by a disfluency if disfluency placement was a random process) yielded the plot in Figure 18.

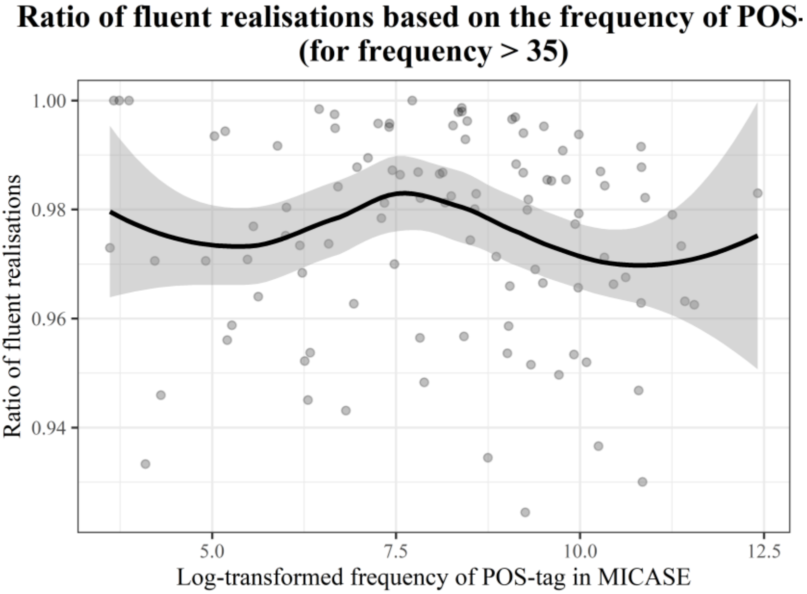


Figure 18: Log-transformed frequency of POS-tags as related to their fluency ratio. Only tags with n>35 which should be disfluent at least once if disfluencies were randomly distributed.

This figure shows that even among the more frequent items, there seems to be little relationship between their frequency and their fluency rate. The Pearson and Spearman correlations support this view, both of them suggesting a weak relationship (r = -0.08, ρ = -0.12, p-values of both are above 0.05). Additionally, it shows that the disfluency rate of all the tags which are represented sufficiently in the data lies between 0% and 7% percent. With such a small difference, it is unlikely that the part of speech on its own will be a strong predictor of the disfluency occurrence, though it may still prove to play a role if all else is equal. The narrow range of fluency ratios also suggests that though individual parts of speech may differ with respect to the frequency with which they precede disfluencies (Pfeiffer 2014), this information alone may not be sufficient to predict them efficiently.

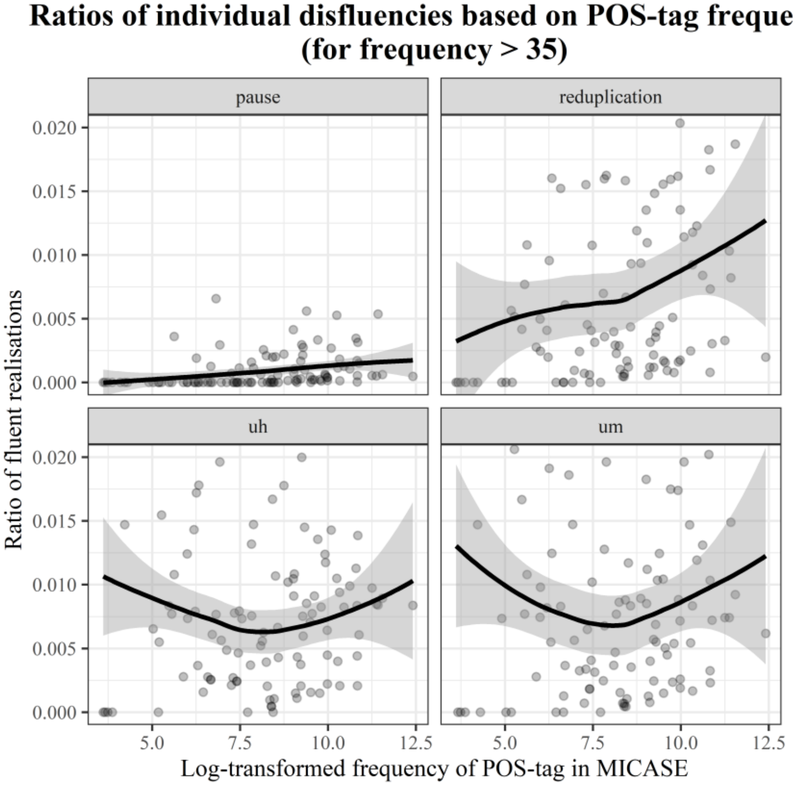


Figure 19: Frequency of a POS tag as related to the ratio with which it is preceded by individual disfluency types.

Visualizing the individual disfluency types as in Figure 19 shows that some types behave contrary to the original expectation and are more common with POS-tags that are more frequent in the data. Only the two filled pauses – *uh* and *um* – suggest at least an initial decrease in the frequency of occurrence with more frequent tags. However, even here, the data is not clustered, but varies substantially instead.

When grouping function words and content words and comparing one group to another, the disfluencies relevant to this work tend to appear before content words (which also tend to be associated with higher surprisal, cf. Kermes & Teich 2017). Thus, the relationship proposed by Maclay & Osgood (1959) did not appear in the present data. The difference was most pronounced in terms of repetitions, but other disfluency types followed the pattern as well (see Figure 20 for a detailed plot).

Finally, in order to explore whether more generally defined parts of speech differ with respect to their likelihood to be preceded/followed by a disfluency, the tagset was further simplified. The simplification consisted of removing all but the first letter of the tag (with the exception of the tags AT/APPGE which would become ambiguous), leaving 19 very general distinctions, e.g. noun/verb. Afterwards, the ratio of occurrences of this tag that were preceded by a disfluency was calculated, shown in Table 18. The proportion of disfluencies following the individual tags was assessed in a similar manner and is shown in Table 19.

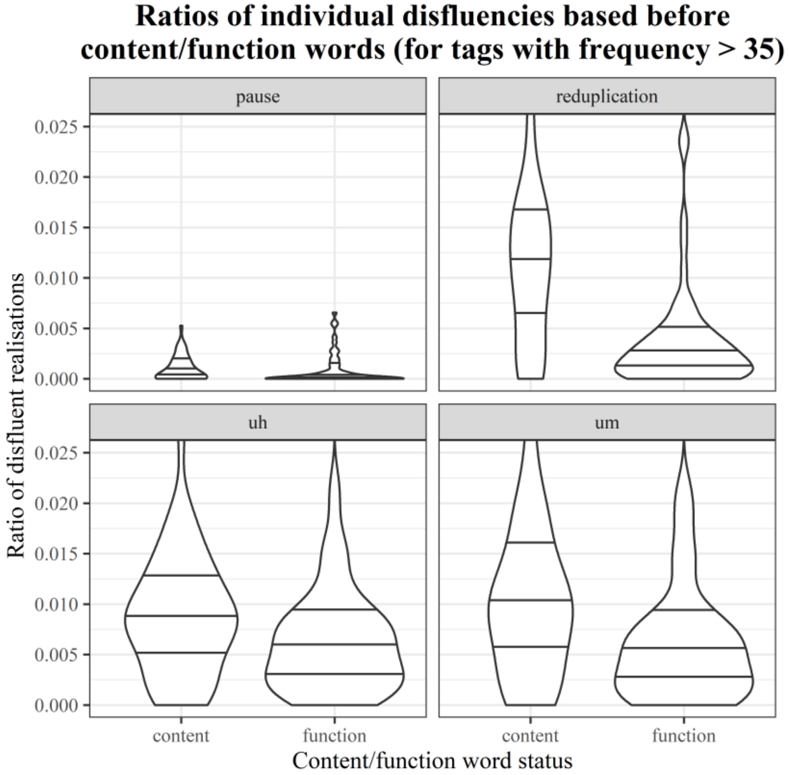


Figure 20: Distribution of individual disfluency types with respect to the function/content word parts of speech. The plots were constructed by calculating the disfluency rate for individual POS-tags (after simplifying the ditto tags) and then visualizing the distribution of the per-tag disfluency rates. The horizontal bar represents the median.

These two tables show that even though there is some variation between the individual tags, their disfluency rates all lie between 0% and 6.5%. They are thus virtually identical to the disfluency rates observed with the extended tagset earlier in this chapter. It would thus seem that the general distinction of parts-of-speech provides only limited information about disfluency placement in the MICASE corpus, differing from Pfeiffer’s (2014) observations. Similarly, there is little relationship between the frequency of disfluencies before and after the individual tags (Spearman’s rho = 0.04, S = 1094, p-value = 0.87).

Tag	**Group**	**Disfluency rate**				
		Pause	Repetition	*Uh*	*Um*	Overall
APPGE	Possessive pronoun	0.05	1.55	0.71	1.04	3.35
AT	Article	0.06	1.87	0.89	0.92	3.75
B	Before-clause marker	0	0.25	0.74	0.98	1.96
C	Conjunction	0.25	1.66	1.17	1.88	4.97
D	Determiner	0.18	1.48	0.9	1.11	3.67
E	Existential *there*	0.22	1.19	1.78	3.36	6.55
F	Foreign	0	1.38	0.92	1.11	3.41
G	*'s*	0	0	0	0	0
I	Preposition	0.04	1.06	0.72	0.64	2.47
J	Adjective	0.05	0.32	0.95	0.73	2.05
M	Number	0.18	0.81	0.96	0.88	2.83
N	Noun	0.05	0.22	0.83	0.61	1.7
P	Pronoun	0.17	1.52	1.04	1.59	4.32
R	Adverb	0.42	0.82	0.78	1.24	3.26
T	Infinitive marker *to*	0.01	1.35	0.45	0.26	2.07
U	Interjection	0.53	3.3	1.05	1.47	6.34
V	Verb	0.06	0.29	0.48	0.45	1.27
X	*not*	0.04	0.18	0.22	0.19	0.62
Z	Letter	0.1	2.26	1.45	1.03	4.85

Table 18: Ratios of disfluencies preceding individual tags in the MICASE corpus.

Tag	**Group**	**Disfluency rate**				
		Pause	Repetition	*Uh*	*Um*	Overall
APPGE	Possessive pronoun	0.02	0.20	0.62	0.46	1.29
AT	Article	0.01	0.35	0.54	0.29	1.19
B	Before-clause marker	0.00	0.74	0.00	0.00	0.74
C	Conjunction	0.03	1.14	1.14	1.16	3.47
D	Determiner	0.11	0.52	0.64	0.66	1.93
E	Existential *there*	0.00	0.11	0.02	0.02	0.14
F	Foreign	0.07	1.18	1.18	0.26	2.69
G	*‘s*	0.04	0.18	1.29	1.07	2.58
I	Preposition	0.02	0.80	0.73	0.66	2.22
J	Adjective	0.14	0.68	0.97	1.10	2.90
M	Number	0.29	0.69	0.71	0.75	2.45
N	Noun	0.33	1.22	1.26	1.60	4.41
P	Pronoun	0.05	0.39	0.30	0.39	1.12
R	Adverb	0.26	1.24	0.95	1.51	3.96
T	Infinitive marker *to*	0.00	0.12	0.59	0.54	1.26
U	Interjection	0.32	2.92	0.96	2.13	6.33
V	Verb	0.07	0.97	0.66	0.63	2.32
X	*not*	0.06	0.48	0.35	0.30	1.18
Z	Letter	0.23	1.46	1.29	0.58	3.56

Table 19: Ratios of disfluencies following individual tags in the MICASE corpus.

6.3.2 IOB-tags

The individual IOB-tags could be divided into three groups according to their frequency:

- B-NP (about $\frac{1}{3}$ of the data)
- O, B-VP and I-NP (each constituting about $\frac{1}{6}$ of the data)
- B-PP and I-VP (each approximately $\frac{1}{12}$ of the data).

The tag I-PP was rare (400 occurrences overall, less than 1% of the data). Figure 21 visualizes the relationship between the individual IOB-tags and the distribution of disfluencies. It shows that first words of noun phrases and words categorized as outside of the recognized phrases were the most likely to be preceded by a disfluency. In contrast to this, items inside the phrase were comparably less likely to be disfluent, similar to the beginnings of verb phrases. The beginnings of prepositional phrases lie between these two groups.

This plot also shows that the distribution of individual disfluency types is not uniform when comparing the IOB-chunks. Locations preceding the chunks I-VP and I-NP are dominated by filled pauses containing *uh*, while silent pauses are almost exclusive to phrase beginnings and items outside of IOB-chunks (only 5% of them appearing inside of phrases).

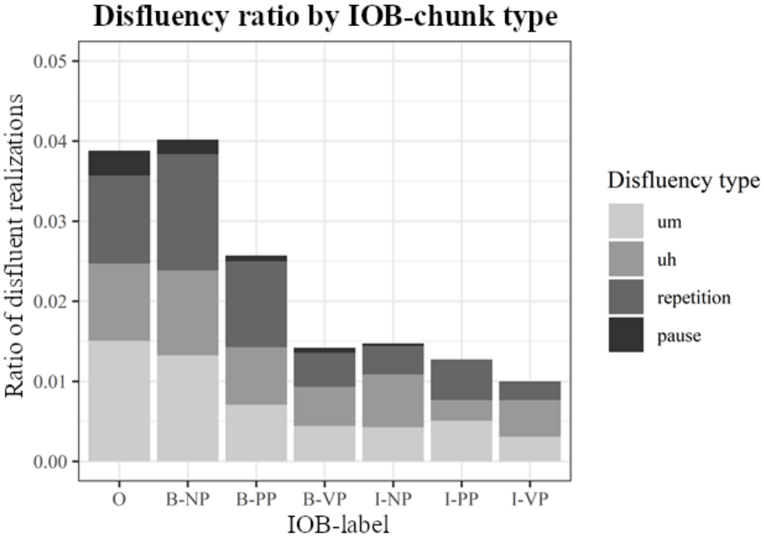


Figure 21: Disfluency rates for the individual IOB-labels.

Additionally, the interplay between the proportion with which the IOB-labels are preceded by a disfluency and the frequency of them introducing sentences should be mentioned (Figure 22). It seems that a part of the variation in the fluency rate of the individual IOB-tags can be explained by their tendency (or in the case of the I-tags, the lack of possibility) to occur sentence-initially. This is further suggested by the strength of the correlation between these two measures (r = 0.96, t = -8.13, df = 5, p-value < 0.001).

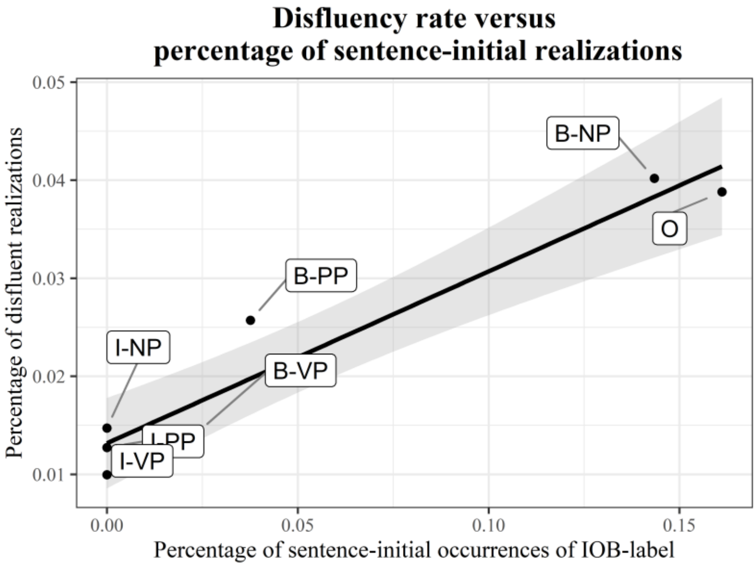


Figure 22: The relationship between the disfluency rate of individual IOB-chunks and their percentage of occurrence as sentence-initial.

6.3.3 POS & IOB-tags for disfluency prediction

In order to assess the degree to which the newly added information about the POS and IOB-tag of each item may improve the prediction of disfluencies, two models were built. First, a simple logistic regression model, used to evaluate the prediction of disfluencies as such. Second, a classification tree, assessing the quality of disfluency selection on the basis of the information. In order to allow interpretation of the results, the POS-tags were simplified substantially for the logistic regression model: only a binary distinction between content words and function words was kept. Additionally, the frequency of each tag was included in the model. Finally, an RNN-based classifier was trained with access to the fine-grained POS-tags and IOB-tags.

The logistic regression model (full model summary including all non-significant predictors and their interactions is in Appendix A.6) exhibited some expected properties. The base odds ratio expressed by the intercept (odds ratio 0.02, coeff.-3.86, std.error 0.05, p-value < 0.0001) increased 4 times (coeff. 1.45, std.error 0.04, p-value < 0.0001) if an item appeared sentence-initially. This increase was even more pronounced if the sentence-initial item was a content word (interaction of sentence-initial position with content-word status has coeff. 0.4, std. error 0.05, p-value < 0.0001). The surprisal score played an important role, too – items with the highest surprisal of 14.75 were 4.8 times more likely to be disfluent than items which had to occur in a given context (and had a theoretical surprisal value of 0). Concretely, per 1 nat increase in surprisal, the odds ratio increased 1.12 times (coeff. 0.11, std. error 0.02, p-value < 0.0001). This contradicts the effect assigned to the forward probability; here, items that were very likely to appear were also more likely to be preceded by a disfluency (coeff. 6, std. error 4.3, p-value 0.16). However, since the coefficient was not significant, no strong conclusions should be based on it.

In contrast to the variables listed in the previous paragraph which promote disfluencies, some of the variables were shown to repel them. Content words were less likely to be preceded by disfluencies than function words (coeff.-0.64, std. error 0.05, p-value < 0.0001); thus, the relationship observed in Figure 20 was reversed. Words with a high MI-score tended to be realized fluently, too (coeff. -0.25, std. error 0.01, p-value < 0.0001). In comparison to the intercept, which was estimated for items outside of the defined chunk categories, all other chunk labels were less likely to be disfluent, except for the B-NP label (coeff. 0.26, std. error 0.05, p-value < 0.0001).

To provide more detail, Table 20 displays the coefficients of a simple logistic regression model predicting disfluencies by individual IOB-labels. It shows that while items at the beginning of chunks (or outside of them) tend to be disfluent more often than items inside chunks, the difference is fairly small. The size of the difference between the beginnings of noun phrases and the remaining categories is rather striking, the trend is, however, fully in accordance with previous research claiming that noun phrase boundaries attract disfluencies (cf. Chapter 6.1).

IOB-label	Coeff.	Std.error	p-value	$p_{disfluency}$	Rank
O (intercept)	-3.86	0.05	2×10^{-16}	0.021	2
B-NP	2.53	0.05	2×10^{-16}	0.209	1
B-VP	-0.09	0.07	0.25	0.019	3
B-PP	-0.49	0.08	1×10^{-10}	0.013	5
I-NP	-0.4	0.07	8×10^{-08}	0.014	4
I-VP	-0.54	0.1	3×10^{-08}	0.012	6
I-PP	-107.8	522	0.84	$<1\times10^{-48}$	7

Table 20: IOB-labels by their likelihood of representing disfluent items.

Some interactions between the individual predictors are worth reporting here as well. Sentence-initial items were less likely to be disfluent if they had a high surprisal value (coeff. -0.19, std. error 0.02, p-value<0.0001). Importantly, the forward and backward probability interacted (coeff. -33.7, std. error 17.3, p = 0.05). This would suggests that words which are tightly linked to both their preceding and following context are extremely unlikely (probability of disfluency being $p_{disf} = 1\times10^{-15}$) to be disfluent. Thus, items in the middle of set expressions or commonly co-occurring items (e.g. *respect* in *with respect to, Fitzgerald* in *John Fitzgerald Kennedy*) should be very likely to be fluent.

Lastly, the effect of POS-tag frequency lies beyond the conventional threshold of statistical significance and is very weak (coeff. 1×10^{-06}, std. error 6×10^{-07}, p-value 0.05), suggesting that if all the remaining parameters are held constant, the sole frequency of the POS in the corpus has little influence on the occurrence of disfluencies.

Overall, the model does not seem to promise a large improvement in disfluency prediction over a model that did not have the IOB-chunk labels, the POS frequency information or the information about the content/function word status of the individual items. The model's deviance decreases by approximately 10% in comparison to the base model, with Nagelkerke's R^2 being 0.1.

To assess the performance in disfluency type selection, a classification tree model was fitted using the *rpart* package (Therneau et al. 2018) with the same predictors as the binary logistic regression: sentence-initial position, IOB-label, MI-score, G-score, forward/backward probability, bigram frequency, POS-tag frequency, surprisal, surprisal change in comparison to previous word and the content/function word status; in this case, the simplified POS-tag itself was included as well. Additionally, the following parameters were used in order to prevent overfitting:

- Smallest leaf size = 20 (do not split if this creates groups with fewer than 20 cases)
- Smallest non-terminal node size = 100 (do not split groups with fewer than 100 cases)
- Minimal reduction in complexity parameter cp = 0.002 (optimized through pruning)

The resulting tree with 19 nodes including the root yielded the results below, which will be discussed by disfluency type.

For *um*, the decisive factor was the sentence position. In the data, 49% of sentence-initial items were *um*s, the remaining categories being substantially less represented. In sentence-internal and sentence-final positions, *um* tended to occur in specific contexts only: before chunk-initial adverbs, verbs and existential *there* if they were weakly related to the previous word (MI-score < 0.5) but not completely improbable (TP-F > 27×10^{-06}).

With respect to *uh,* the situation is more complex. Outside of sentence-initial contexts, its likelihood of occurrence increased in front of content words (adjectives and nouns specifically) and inside VPs. Chunk-initial, chunk-external and NP-internal observations were more likely to yield *uh* if they were unrelated to the previous word and very unlikely to occur (MI-score < 0.5 and TP-F < 27×10^{-06}). This observation may suggest a trend opposite to Clark & Fox Tree (2002), who observed *uh* to occur before minor and *um* before major delays: major delays would be expected before words with lower probability, as they should require more time to be prepared for production.

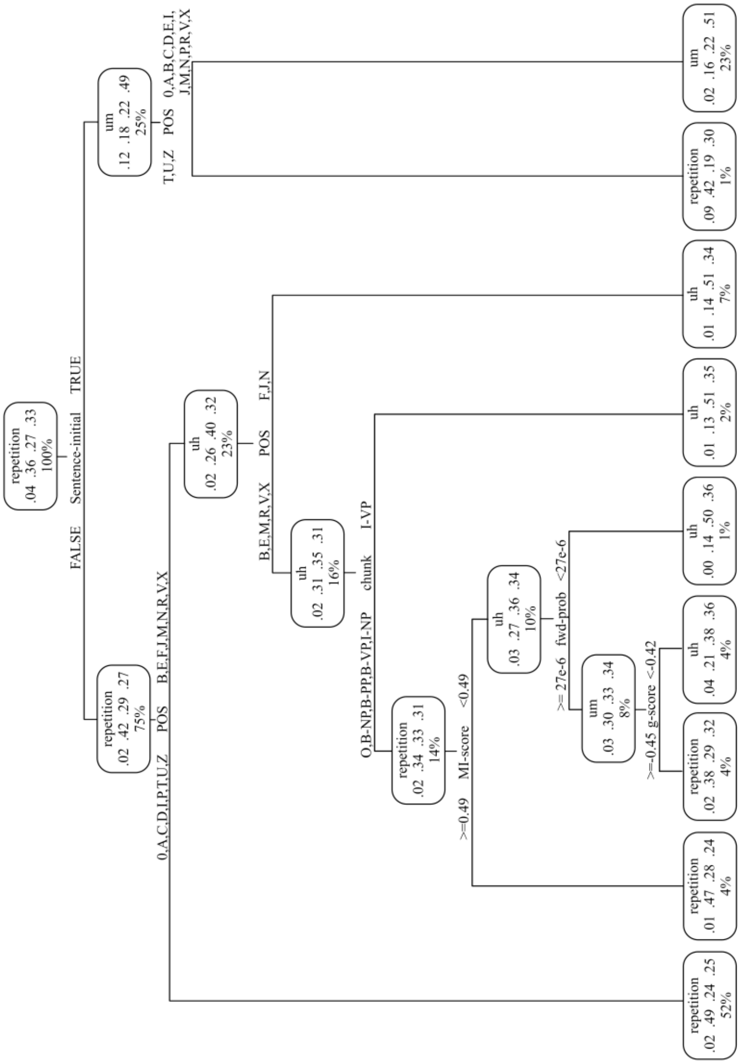


Figure 23: Classification tree deciding the disfluency type after pruning (at cp = 0.002). Percentages at the bottom of each node represent the share of data that a given node describes. The ratios of the individual disfluency types are shown in the second row of each box in the following order: pause, repetition, *uh*, *um*.

All the remaining contexts were most likely to choose repetition as the appropriate disfluency type. Pauses were never identified to be the most common type of disfluency in a given context. This is likely due to the fact that their overall frequency is substantially lower. The highest proportion of pauses in a given context was observed sentence-initially (12%) and in the very specific case of chunk-initial/chunk-external function words (and adverbs) with low MI-score, high forward probability and a low G-score (4%).

Task	F_1**– macro**	F_1**– micro**	F_1**- disfluencies**
Disfluency prediction	57%	93.5%	16.2%
Type selection	38.5%	47.2%	-

Table 21: Performance of POS/chunk-aware RNN classifier at disfluency prediction/selection.

With respect to the disfluency prediction by an RNN model, including the information about the part of speech and IOB-chunk status improved the prediction of disfluency occurrence. The improvement (in comparison to the best model presented in 5.5) was achieved in all three performance measures, though it was not large. The disfluency type selection, on the other hand, was not improved at all. Thus, the additional information does not seem to give any substantial advantage to the classifier in disfluency selection, as also suggested in Chapters 6.3.1 and 6.3.2. The performance of the best

RNN model in disfluency prediction and disfluency type selection is noted in Table 21.

6.4 Discussion

In order to interpret the results yielded by the logistic regression and the CART algorithm, an information-theoretic perspective alone is not sufficient. This is shown by assessing the importance of the individual variables in the logistic regression model using the *caret* package for R (Kuhn 2017). There, the importance of surprisal is lower than that of sentence initial position, MI-score or content/function-word status (yet still higher than of the other individual predictors). It would thus seem that the first criterion deciding about the placement of disfluency is the fact whether an item is sentence-initial or medial/final.

A readily available interpretation of this observation is the disfluency as a symptom hypothesis mentioned earlier in Chapter 3: if a large proportion of the planning has to be done before the sentence is started, the speaker may have issues producing the sentence immediately due to still being busy drafting the concept to be expressed, extracting the necessary items, assigning them to correct positions and translating at least the first items into phonemes. This may lead to the insertion of a hesitation. The extent to which this effect is pronounced depends on the scope of processes that are to be carried out before the utterance may start: the larger the scope, the more likely is it that the amount of work to be done will be too large for an immediate onset of speech.

Experimental evidence suggests that the planning at the beginning of a sentence does not extend over the whole sentence. It may incorporate the first verb phrase and the subject phrase; in the case of lexical planning even less may be prepared prior to production (Zhao & Yang 2016; Zhao et al. 2015; Allum & Wheeldon 2007; Smith & Wheeldon 1999, 2001). In languages which are not head-initial (such as Japanese) Allum & Wheeldon (2007) additionally observe that the syntactic planning which precedes the onset of an utterance is likely to include the initial phrase as well. Thus, in total, up to three phrases may be planned before anything has been uttered.

In this case, the term phrase does not necessarily correspond to syntactically defined phrases. Rather than referring to subject noun phrases or initial noun phrases, Allum & Wheeldon (2007) and Zhao et al. (2015) advocate the concept of functionally defined phrases. Such phrases represent units in the thematic representation of an utterance but may not necessarily correspond to syntactic phrases. Rather, they represent functions such as modifier or agent. *The flower above the dog* in *the flower above the dog is red* is not a single planning unit in spite of being the subject noun phrase and the theme of the sentence. It consists of two functionally defined units: the theme and the modifier. In Allum & Wheeldon's view, such smallest functional roles constitute units of the planning scope as each of them corresponds to a function in the process of functional assignment (Allum & Wheeldon 2007).

In terms of lexical retrieval, utterance planning may be even simpler, as suggested by studies inspired by the radically incremental approaches to lexical planning (represented by Zhao & Yang 2016). Since the adherents of these approaches claim that lexical retrieval operates on a strict word-by-word basis, the effect of utterance planning before the onset of pronunciation should be only due to the complexity associated with the retrieval of the next lemma. In such scenarios, the sentence-initial positions would not differ from sentence-medial or sentence-final with respect to the lexical retrieval. They might, however, still differ in the remaining elements, notably in terms of the processing of the syntactic structure.

If, however, the planning is as radically incremental as suggested by Zhao & Yang (2016) and the proponents of lexically-based theories of sentence-production (Griffin 2003), the amount of processing necessary before the beginning of the utterance is reduced. In such a case, the strong effect that the position of an item in the sentence has on its fluency would be less likely under the disfluency as a symptom view. On the other hand, it would be more compatible with the other views including that of disfluencies as surprisal smoothing agents: the beginnings of sentences have on average a greater surprisal value, the mean surprisal of sentence-initial items being 0.28 nats higher ($\Delta median = 0.51$) in comparison to the overall average/median. This is intuitive as there are more options in which a sentence could begin than continuations/endings compatible with a given beginning.

The coefficients of IOB-chunks provide support to the observation of Boomer (1965) and more recently Schneider (2014) that disfluencies should predominantly occur on the boundaries of phrases rather than inside them. In the individual chunk pairs, the coefficients associated with chunk-initial positions were higher than those associated with chunk-internal ones. The fact that disfluencies still do occur inside the phrases distinguished here is the easiest to explain by the lexical theories of sentence-production: if language production is radically incremental, each word may present a potential issue. However, chunk-internal disfluencies are permitted even by theories arguing for the existence of multi-word units, such as Schneider's chunking hypothesis. Even these theories permit disfluencies to occur chunk-internally – provided that the disfluent location lies within a grammatical chunk, not a deeply entrenched mental one.

The placement of disfluencies at the beginning of chunks is further predicted by syntax-based accounts of language production combined with the disfluency as a symptom hypothesis. The fact that beginnings of phrases are more likely to be disfluent follows from the principle that each phrase needs to be planned prior to the onset of its production. If the planning cannot be done online in the course of articulating the previous phrase, it is inevitable that the speech stream is interrupted until the planning is finished and the phrase to be uttered is translated into commands for the articulators. Only then can the speaker begin with the actual articulation.

The range of contexts in which they appear makes repetitions seem like the default option unless a different choice is specified by the context. An example of such a context is the sentence-initial position where *um* was substantially more likely to appear in the data. The use of *um* rather than *uh* sentence-initially agrees with Clark & Fox Tree's (2002) observation that it signals a major delay, since silent pauses are longer between sentences than within them (Krivokapić 2007; Sanderman & Collier 1995). Thus, if inter-sentence pauses should be longer, they should be more prone to contain *um* (in contrast to *uh*). On the other hand, the preference of *um* to appear in contexts with higher direct transitional probability in sentence-medial/final positions would seem to contradict this view. Yet, as this preference only appears in very specific contexts (described in Chapter 6.3.3), representing only 1% of the overall data, it certainly should not be viewed as a strong argument.

6.5 Conclusion

The present study presented an attempt to account for the variation in disfluency occurrence which was not explained by Study IIa. In order to do so, it drew upon additional structural factors: the POS-tags and IOB-chunk labels. However, though some trends with respect to these factors were identified (the tendency of disfluencies to appear at the beginning of noun-phrases, identified by previous research as well), this did not lead to a major improvement in the prediction of disfluencies, or the selection of the disfluency type to be inserted.

Though the approach to disfluency prediction assumed here was diametrically different from that suggested by Qader (2017) and similar to that proposed by Adell et al. (2007), it yielded results closer to the first study rather than the latter and disagreeing with the results obtained in Chapter 4. Unfortunately, Adell et al. do not provide information about the sentence-initial/medial/final distribution of the disfluencies in their data, which largely influenced the discrepancy in performance between Chapter 4 and Chapters 5 and 6. The results obtained highlight the fact that disfluency prediction (and selection) is still in its early phases and vitally linked to data quality. Further development requires additional qualitative/quantitative empirical research into the origin of disfluencies as well. This is a complex task, as repairs in the speech production such as:

(10) `when you do sear- uh when you're searching`

MICASE: svc999mx104

are likely hard to predict: little seems to suggest that the speaker will decide to rephrase *when you do sear[ches]* to *when you're searching*. Though deciding whether a disfluency should be inserted in the interregnum may be comparably easier, true disfluency prediction under the Noisy Channel hypothesis should be capable of transforming fluent input into a disfluent output in a natural way. Thus, an ideal system should be also able to predict repairs, especially considering that they are the second most common type of disfluencies after filled pauses (Shriberg 1994: 137).

The next chapter presents a different approach to disfluency prediction. Rather than employing a set of theoretically motivated predictors extracted by models trained on large corpora, it proposes a purely computational linguistic model, inspired by machine translation.

Chapter 7

Study III: Encoder-decoder architecture for disfluency prediction

For the major part of this thesis, the disfluency prediction/disfluency type selection were attempted with a cognitively-motivated approach. The achieved performance suggested that this approach still leaves a substantial proportion of the variation unexplained. Further improvement may be possible through refinement of the predictors used, addition of new predictors[15] or setting up a more complex model. However, the contribution of these steps, carried out in Chapters 5.4-5.5 and Chapter 6, led only to marginal changes in the performance.

Thus, rather than seeking to improve the theoretically motivated, yet weak classifiers, this chapter views the attempt at disfluency prediction from a purely computational-linguistic perspective. It approaches the task at hand by adapting a successful machine translation architecture to disfluency placement. In other words, disfluency placement is viewed as a translation task from a fluent

[15] An obvious example of a promising predictor that has not been used in the previous attempts is that of the speaker ID as there is a clear evidence of individual preferences of speakers (Shriberg 1994). Nevertheless, as mentioned in Chapter 5.2.8, the data is not suitable for fitting models that include the speaker ID as a predictor.

language to a disfluent one. This means that the Noisy Channel approach is maintained.

The architecture used is the RNN encoder-decoder model, as proposed by Cho et al. (2014b). In this architecture, the lengths of input and output sequences may differ, allowing one-to-many and many-to-one correspondences. Additionally, interactions between the elements of each of the sequences are possible in both directions, i.e. individual elements may influence the "translation" of those after them as well as those before. The next pages will briefly introduce the architecture as used in machine translation; afterwards, the adaptation to disfluency prediction will be presented.

7.1 Encoder-decoder

Early applications of neural networks to machine translation targeted the improvement of existing statistical machine translation systems (Schwenk 2012; Zamora-Martínez et al. 2010; Schwenk et al. 2006). They aimed to improve the estimation of the probability of a given phrase in the source language to be translated into another phrase in the target language. For this purpose, they usually refined the traditional statistical machine translation approach in which sentences from the source language s were translated to the target language t and based on the equation (Schwenk 2012):

$$\hat{t} = \underset{t}{\operatorname{argmax}} P(t|s) = \underset{t}{\operatorname{argmax}} \frac{P(s|t)P(t)}{P(s)} \\ = \underset{t}{\operatorname{argmax}} P(s|t)P(t) \tag{44}$$

The neural network was implemented in order to improve the estimate of the language model assigning the probability $P(t)$, which was usually estimated using an n-gram model (Schwenk 2012). It realized the suggestion of (Bengio et al. 2003) that the joint distribution should be modeled in a continuous rather than a discrete space. In a discrete space, where each word is viewed as an independent variable, a single change in one of the discrete variables may change the outcome substantially. On the other hand, if the modelling is done in continuous space, the probability function to be learned can be smooth. Thus, while the machine translation was still based on pairs of multi-word sequences extracted from the training data, the probability with which one sequence was translated into another depended to a degree on the probability of a given sequence in the target language as estimated by a model operating in a continuous space.

A further extension of this approach allowed to use the continuous space model not only for modelling the target language, but to estimate the probability $p(t|s)$ as well (Cho et al. 2014b; Sutskever et al. 2014). In this approach, two neural networks are trained at the same time: the encoder, translating the source string into a fixed-length vector in continuous space, and the decoder, translating the fixed-length vector into the target language. Importantly, the continuous space is shared between the two models, allowing simultaneous training of the encoder and decoder, as visualized in Figure 24. While the approach of Cho et al. (2014b) only employed the encoder-decoder architecture in order to score phrase pairs from a traditional phrase table, Sutskever et al. (2014) suggested using the model for direct translation.

In their approach, the encoder-decoder model selects individual words from the vocabulary. Specifically, the encoder translates the source sequence $x = (x_1 \dots x_{T_x})$ into a vector c, e.g. by means of a recurrent neural network with state h_t (after the item x_t of the source sequence) equal to:

$$h_t = f(x_t, h_{t-1}) \tag{45}$$

Thus, the shared vector c is produced by:

$$c = g(h_1 \dots h_{T_x}) \tag{46}$$

where $g()$ can be any function capable of handling the range of inputs. For Sutskever et al. and Cho et al. the final state is taken as the representation of the whole sequence. Thus,

$$g(h_1 \dots h_{T_x}) = h_{T_x} \tag{47}$$

As a consequence, while the encoder generates a range of hidden states (one for each element of the input sequence), only the last one is used. Importantly, this state does contain elements of all items in the input sequence; it is not the case that the last hidden state would only reflect the last element.

The decoder is then trained to produce the target sequence one word at a time. Thus, instead of searching for a sequence with the highest joint probability, it decomposes the joint probability of a target sequence y into the ordered conditionals (Bahdanau et al. 2015):

$$p(y) = \prod_{t=1}^{T} p(y_t | y_1 \dots y_{t-1}, c) \tag{48}$$

Where $p(y_t | y_1 \dots y_{t-1}, c)$ is estimated by the (recurrent) decoder on the basis of the previously produced item y_{t-1}, its hidden state h_t and the encoded vector c. Hence,

$$p(y_t | y_1 \dots y_{t-1}, c) = f_{decoder}(y_{t-1}, h_t, c) \tag{49}$$

Practically, the shared vector c is only used directly in the prediction of the first character. For the remaining characters, its effect is present in the form of the hidden state, yet the vector itself is no longer included in the input.

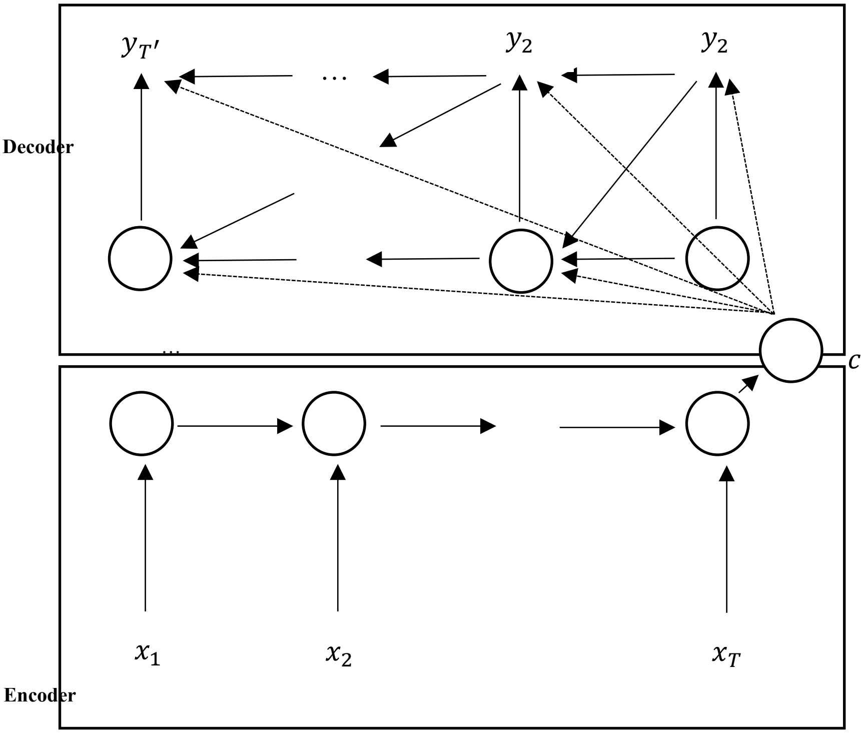


Figure 24: Schematic representation of the encoder-decoder architecture. The nodes $x_1 \dots x_T$ represent the individual inputs, while $y_1 \dots y_{T'}$ represent the outputs. Note the shared representation c in continuous space. The dotted arrows represent the indirect influence of the shared vector on the outputs. Adapted from Cho et al. (2014b).

Given that encoder-decoder models often have to deal with long sequences, the recurrent neural network setup as described in

Chapter 5.4.1 is not suitable. This is due to the fact that the errors flowing through simple recurrent units have a tendency to either blow up or vanish (Pascanu et al. 2013; Hochreiter & Schmidhuber 1997; Bengio et al. 1994) which makes learning long-range dependencies very difficult. Striving to remedy this weakness, Hochreiter & Schmidhuber (1997) proposed the long short-term memory (LSTM) unit which is capable of keeping track of items over long time lags. For this purpose, each unit is provided with input and output gates protecting its contents from being influenced by irrelevant inputs and preventing the contents from negatively influencing other units. Moreover, the addition of a forget gate (Gers et al. 2000) allowed the unit to reset its own state without being specifically told to do so.

Both the original LSTM used by Sutskever et al. (2014) as well as its modified version (gated recurrent unit, GRU) developed by Cho et al. (2014b) were capable of building translation models that matched the performance of the state-of-the-art statistical machine translation systems. Further extension of the architecture was proposed by Bahdanau et al. (2015), who suggested that an attention mechanism should be added, allowing the model to learn which parts of source sentences are particularly important for individual elements of the target sequences. The crucial contribution of the attention model is that it replaces the single vector in continuous space by a distinct vector for each target word, created by an alignment model from the sequence of hidden states. Each of these distinct vectors contains the information about the whole source sequence, with particular focus on the surroundings of a given word (Bahdanau et al. 2015).

With respect to the input format, Sutskever et al. (2014) proposed to present the source sequence backwards in order to create short dependencies. However, such an approach also lengthens the short dependencies of the traditional forward RNN. To remedy this, the bidirectional recurrent neural network (Schuster & Paliwal 1997) may be employed (Wu et al. 2016; Cho et al. 2014b). In the bidirectional RNN, the outputs of forward and backward RNNs are concatenated, thus providing the model with access to the whole input sequence at any stage.

Furthermore, the early models segmented their input into individual words in order to simplify the modelling of long-range dependencies. However, such an approach has an inherent weakness for the application in machine translation – it makes the approach only easily applicable in analytic languages. In any language with rich morphology, word-based translation is inherently linked to data sparsity which complicates the learning of language rules and their application to rare words. An application to agglutinating languages becomes virtually impossible. Thus, it has been suggested that the models should work with smaller units such as “wordpieces” (Wu et al. 2016) or individual characters (Lee et al. 2017). On the one hand, this allows the model to learn the application of morphology and improves the processing of rare words (Wu et al. 2016). On the other hand, it makes the computation of the attention mechanism substantially more challenging since "computational complexity of the attention mechanism grows quadratically with respect to the sentence length, as it needs to attend to every source token for every target token" (Lee et al. 2017: 368).

7.2 Present study

In this study, the encoder-decoder architecture was employed in order to evaluate the suitability of the machine translation approach for disfluency prediction, exploring a possible avenue of further research. Thus, the disfluency prediction was viewed as a translation task from a fluent language into a disfluent one.

For this purpose, the data from MICASE was used again as the input. The disfluencies were identified and removed in a manner closely similar to the procedure described in Chapter 5.2.8.1. In order to allow training, locations in which disfluencies were present in the original text were marked with a special character ("~"). The training data consisted of a series of sentence pairs, similar to Example 12.

(11) `what 's the name for blood in spanish.` **Input**

`~ what 's the name for blood in spanish.` **Output**

In order to simplify the training process, the tag "<unk>" was used as a replacement for extremely rare words: the original vocabulary of 19124 words was limited to the 5964 most frequent words by only keeping those with frequency ≥ 5). To further improve the training efficiency, extremely long and extremely short sentences were removed (keeping the central 90%, that is 79884 sentences). Finally, the input and output were translated into a form suitable for the model, i.e. the individual characters were mapped to integer values which were then one-hot encoded.

Then, the encoder model was built with three layers each consisting of 1024 GRUs (Cho et al. 2014b), the first layer consisting of 512 units reading the input sequence forwards and 512 units processing the input backwards. The encoder was then connected to the decoder with three layers of 1024 GRUs each, too. The hidden states at the end of the input sequence processing were used as the initial states for the decoder (cf. Equation 47).

During the training, the decoder was tasked to predict the next character given the encoded sequence and the previously produced characters. The previous characters were not taken from the output of the decoder; rather they were drawn from the actual output sequence one character at a time. Thus, the training implemented teacher forcing (Williams & Zipser 1989). Additionally, the training used dropout rate (Srivastava et al. 2014) of 0.5 in order to reduce the probability of overfitting and improve the generalizability of the model. This means that during training, 50% of randomly selected units were dropped along with their connections as a means of preventing the individual units to co-adapt too much, i.e. to depend too strongly on the output of other units. Figure 25 visualizes the model.

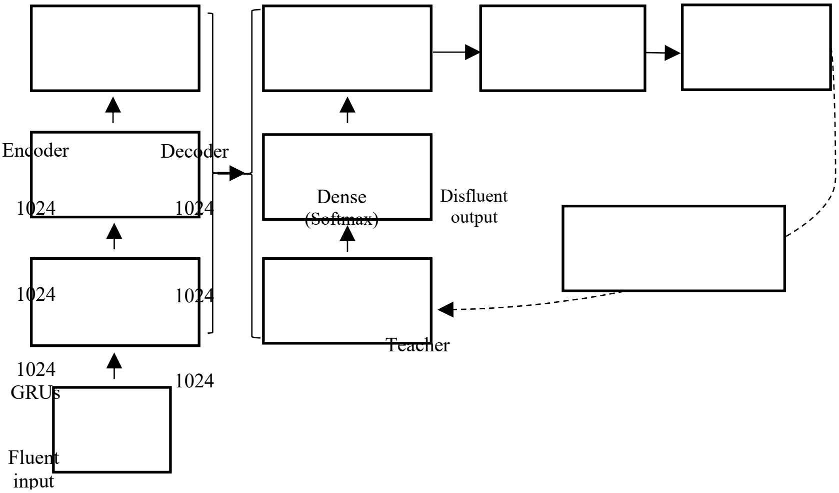


Figure 25: The architecture used in this chapter. Note that the output at time t is fed back into the decoder (in training, teacher forcing replaces the actual prediction with the correct one) to be used for prediction at $t + 1$. The final dense layer maps the decoder output into probabilities of the individual characters.

The training minimized the categorical cross-entropy function through the RMSProp algorithm (Tieleman & Hinton 2012) with initial learning rate of 0.001 and learning rate decay set to 0.00001.[16] The improvement was monitored on a held-out validation set (containing approx. 10% of the data) and interrupted once the performance on the

[16] The learning rate α_i at iteration i with decay rate $d = 0.00001$ and initial learning rate α_0 thus equals $\alpha_i = \frac{1}{1+di}\alpha_0$

validation set stopped improving. The performance of the model was evaluated on a separate test set (another 10% of the data) without teacher forcing. Two models were fitted, one distinguishing between the individual disfluency types and one with a single tag for all types of disfluencies.

Importantly, an exact parallel of the disfluency type selector proposed in the previous chapters is not possible within the architecture used in this chapter.[17] Similarly, the F_1 score is not directly comparable: first, in the present study, the F_1 score is calculated on a per-character base. As a consequence, there are many more cases where the model can be correct/incorrect. Secondly, given that the F_1 in this case verifies the exact correspondence of each character, one additional/missing character (or disfluency tag) can influence further scoring. For example, even though the disfluency before *you know* is predicted correctly in Example 13, it is not counted as such in the evaluation, as it did not occur at the same index [6]. While it would be possible to align the

[17] A close approximation of the disfluency type selector could replace the disfluencies with a pair of special characters, one signaling a disfluency, the other denoting its type. Then, teacher forcing could be used in the prediction for all characters except the disfluency type tag. In this case, the model would likely quickly learn that a disfluency tag can only be followed by the disfluency type tag and the prediction at this location would mostly consist of the selection of the correct tag to predict.

predicted sequences with the correct ones to some degree, this would be unfeasible for cases where the network's output is vastly different from the actual one, such as in the case produced at an early iteration, where the sequence *actually he's just teaching the honors intro for this fall* was reproduced as *actually he's just teaching the honors into froit harily.*

(13)

b	u	t		~		y	o	u		k	n	o	w			**Predicted**
~		b	u	t		~		y	o	u		k	n	o	w	**Original**
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	**Index**

In addition to the F_1 score, the performance was evaluated with two additional metrics: the exact string accuracy and normalized Levenshtein distance. The exact string accuracy counts the proportion of character matches between the predicted/original sequence. First, the shorter of the sequences is padded to the length of the longer one with a non-character symbol Ø, then the characters at each index are compared. The final score equals the number of character matches divided by the length of the longer of the sequences.

This makes the metric particularly sensitive to misalignments that occur early in the sequence. This is visible in Example 13 used previously for the illustration of the F_1 score. Because the decoder did not predict the initial disfluency, all subsequent items are misaligned and thus labeled as incorrect. As a consequence, the accuracy metric would be very low, in spite of most of the sequence being predicted correctly.

This issue becomes particularly pronounced once teacher forcing is disabled. With teacher forcing enabled, the decoder can recover from some of the errors which would otherwise propagate. I will illustrate this on Example 13: the processing starts with the encoder reading the cleaned sequence *but you know* forwards and backwards and projecting it into the multidimensional continuous space. This projection is passed to the decoder. The decoder works incrementally, one character at a time; each character depends on the encoded input and the characters produced up to that point. In an ideal case, the decoder should produce the sequence *~ but ~ you know*. However, it may happen that the decoder fails to identify the first disfluency. In such a case, the first character produced will be *b*. With teacher forcing disabled (as was the case in the testing scenario), *b* would be fed back into the decoder which would then likely continue to produce *ut ~ you know*. However, as Example 14 shows, this sequence would not be identified as matching with the true sentence, yielding very low exact string accuracy in spite of the considerable overlap between the predicted sequence *but ~ you know* and the true sequence *~ but ~ you know*.

(12)

Index	**Original**	**Teacher forcing**	
		Yes	**No**
0	~	$\underline{\text{b}}$	$\underline{\text{b}}$
1			$\underline{\text{u}}$
2	b	b	$\underline{\text{t}}$
3	u	u	_
4	t	t	$\underline{\sim}$
5			
6	~	~	$\underline{\text{y}}$
7			$\underline{\text{o}}$
8	y	y	$\underline{\text{u}}$
9	o	o	_
10	u	u	$\underline{\text{k}}$
11			$\underline{\text{n}}$
12	k	k	$\underline{\text{o}}$
13	n	n	$\underline{\text{w}}$
14	o	o	_
15	w	w	_
Accuracy		$\frac{15}{16}$	$\frac{1}{16}$
Disfluency F_1		**0.7**	**0**

Underlined characters do not match the original sequence

On the other hand, with teacher forcing enabled, the first produced character *b* would be reported on the output, but the correct prediction ~ would be sent back to the decoder. Then, the decoder may produce the rest of the sequence correctly as *but ~ you know.* The accuracy metric and the F_1 score would still report the error in the first character; however, the rest would not be affected. Example 14 illustrates this in more detail.

It was thus crucial that the influence of teacher forcing be taken into account both during training and evaluation. Exact string accuracy is a good representation of the ability to predict the next character; however, it is too strict to be used with models without teacher forcing. As a consequence, it was used in order to observe the improvement of the model during training. For the evaluation of the model performance on the test data, a different score had to be found, capable of dealing with the misalignment caused e.g. by the incorrect prediction of disfluencies.

For this purpose, the Levenshtein distance was selected (Levenshtein 1965). This metric calculates the minimum cost at which one string can be converted into another by editing single characters. The available edits are substitution, insertion and deletion and may be assigned various costs (in this case, all operations had the cost of 1). Importantly, the metric is not inherently normalized, i.e. the string length is not taken into account. The possible approaches to normalization include the alignment length (used e.g. by Heeringa et al. 2006) or the length of the longer string (used e.g. by Inkpen et al. 2005).

In the present thesis, the latter method was adopted. To illustrate, the raw Levenshtein distance between the true sequence in Example 14 and the output of the decoder would be:

- 1 with teacher forcing (replacing *b* by ~ in *b but ~ you know*)
- 2 without teacher forcing (inserting ~ and a space prior to *but ~ you know*)

After normalization (i.e. division by 16), the final scores would by $\frac{1}{16}$ and $\frac{2}{16}$ respectively (smaller is better), reflecting the similarity of both sequences to the correct one better than the exact string accuracy scores of $\frac{15}{16}$ and $\frac{1}{16}$ (larger is better).

Finally, beam search strategy (with beam width 10) was used during inference. Thus, rather than keeping only the most likely output at time t and using it to predict the output at $t + 1$, ten outputs with the highest joint probability were kept at any given moment and used for the inference of the next character. This should allow the network to recover from some of the errors caused by assigning an incorrectly high probability to a character. To reduce the computational complexity, the outputs were pruned after each iteration. The inference was stopped once the end-of-sequence character was produced by the decoder or once the maximum observed length was reached. In this manner, the model never had to work with sequences longer than those it has seen in the training set.

The models were implemented using the *keras* API (Chollet & others 2015) with *TensorFlow* backend (Abadi et al. 2015) and trained using a dedicated GPU. The loss function minimized was the categorical cross-entropy.

7.3 Results

The first model (distinguishing between individual disfluency types) was trained for approx. 180000 iterations (or 36 hours) with batches of 256 sentences (each batch was a randomly drawn sample of sentence pairs, padded to the length of the longest sentence in the sample). At the moment the training was interrupted, the training loss fell below 0.001, with validation loss 0.053 and validation accuracy 99.3%, i.e. the model correctly predicted 993 characters out of 1000 in the validation set (validation used teacher forcing). Its mean normalized Levenshtein distance was 0.02 (SD = 0.03, median = 0). On the test set, the architecture achieved 67.6% accuracy. This deterioration is likely to be linked to the lack of teacher forcing in the processing of the test set – erroneous predictions may propagate further through the sentence. This is supported by the mean normalized Levenshtein distance on the test set which was 0.09 (SD = 0.16, median = 0).

Table 22 shows the performance of the model over individual disfluency types. The achieved performance is clearly not balanced across the distinguished types: repetitions seem easier to predict than filled pauses (*um*) in particular. This may be related to their frequency – as mentioned in Chapter 5.3.1, repetitions are the most frequent type of disfluencies in the data. However, this does not explain the observed trend completely: the model was more successful in predicting silent pauses than filled ones, in spite of their relative rareness. Thus, it would seem that filled pauses, even though they are more different from the fluent text in terms of the surprisal at their location (cf. Chapter 4.4.2), are harder to predict solely from the text. This suggests that they may be related to background processes not having any realization in the surface form, such as the silent repairs suggested by Levelt (1983).

Disfluency	**Precision**	**Recall**	F_1
Pause	5.48%	15.38%	8.08%
Repetition	9.01%	20.13%	12.45%
Uh	3.28%	9.32%	4.86%
Um	1.82%	5.06%	2.68%
Overall	8.71%	22%	12.48%

Table 22: The performance of disfluency prediction using encoder-decoder architecture. The overall scores were calculated after individual disfluency types were conflated in both the predicted and real test data.

The second model (only working with one tag for all disfluency types) required approx. 150000 weight updates (corresponding to 30 hours of training) on batches of 256 sentences. At the moment the training stopped, it achieved loss on the training data below 0.001. The performance over validation data was even slightly better than in the first model (lower loss of 0.045, same accuracy of 99.3%), with mean normalized Levenshtein distance of 0.01 (SD = 0.03, median = 0).

Similarly, this model achieved better results over the test set with an overall accuracy of 68.8% and mean normalized Levenshtein distance of 0.08 (SD = 0.15, median = 0). Table 23 shows that this model outperformed the one trained with distinct tags for each disfluency type both in precision and in recall. Such an observation suggests that the training data was not sufficiently large for the network to learn the underlying function regulating the placement of the disfluency types distinguished in this data. Though conflating the types may have increased the noisiness of the dataset, it also provided the network with many more examples of a single category to train on. This improvement hints that while disfluency type is not selected randomly (as shown in Study IIa by comparison to the baseline), the individual types do share some of the characteristics of their placement.

Model	**Precision**	**Recall**	$\boldsymbol{F_1}$
Occurrence + type	8.71%	22%	12.48%
Occurrence	9.86%	24.31%	14.03%

Table 23: The performance of disfluency prediction using encoder-decoder architecture. Comparison of the model which was trained using distinct disfluency types (thus predicting both occurrence and type of disfluency) with a model trained with only one tag for all disfluencies (thus predicting occurrence only).

This shared characteristic allows the model to improve its performance. Conceptually, it is similar to replacing all nouns in a corpus with a single tag. While it is clearly not true that all nouns are mutually interchangeable in any context, a model learning on such data may acquire better understanding of the syntactic constraints of noun placement than a model that is trained on an identical dataset with the actual noun strings in place.

7.4 Conclusion

The study presented in this chapter shows that approaching disfluency prediction as a translation task from fluent language to a disfluent one may achieve results similar to predicting disfluencies through a set of theoretically motivated predictors. Even though the F_1 scores are not directly comparable (due to the difference in the resolution: character vs. word), the encoder-decoder model was clearly capable of achieving some understanding of disfluency placement. Furthermore, the superiority of the model that did not distinguish individual disfluency types provides two suggestions that further research should address. First, the tandem of disfluency prediction and disfluency selection models, as presented by Ohta et al. (2008) may present a potential avenue of improvement. Second, to achieve better results, the training data should be even larger. In the context of machine translation, the MICASE is a rather small dataset, compared to, e.g. the 5 million sentences used by Wu et al. (2016).

Additionally, the implementation of attention mechanism, as suggested by Bahdanau et al. (2015) may further improve the performance, especially in the case of long sentences, as shown by Vaswani et al. (2017). The present study did not employ attention due to the limitations of the hardware used: a character-based model with attention did not fit into the memory of the GPU used for training. This issue could be alleviated by using subword-units as the basic units of the model, reducing the computational complexity. Such an approach would, however, be negatively reflected in the increased sparsity of the training data.

To conclude, advanced models used in machine translation may yield good results on the task of disfluency prediction, similar in performance to models employing theoretically motivated predictors. However, they suffer from the limited size of the training data, too. This negative influence is perhaps even more pronounced in the case of the architecture used in this chapter, as it only had access to the MICASE dataset. The approach developed in Studies I and II, on the other hand, employed statistics drawn from a much larger corpus. Further research should thus explore whether pre-training the encoder-decoder model on a large general dataset (dominated by written language) with subsequent fine-tuning on the disfluent corpus improves the performance of the system by yielding better representations in the shared space of the encoder and the decoder.

Chapter 8
Final discussion

This thesis attempted to predict three types of disfluencies in the human language: filled and unfilled pauses (further distinguishing *uh* and *um*) and repetitions. The prediction included a number of predictors proposed by previous research, such as sentence/syntactic unit boundary, part of speech, bigram association metrics. In addition to these measures, the estimate of the surprisal of each word in the text was added. This was motivated by the claim of the Uniform Information Density (Jaeger 2006, 2010) hypothesis, as extending the Information Theory (Shannon 1948). Concretely, the assumption was that disfluencies may be used as surprisal smoothing agents where other smoothing options are unavailable (e.g. the insertion of an optional syntactic element, cf. Jaeger 2006, 2010) or insufficient (e.g. the smoothing by phonetic detail, cf. Sayeed et al. 2015; Aylett & Turk 2004, 2006). In such cases, the interruption of the speech stream by a pause or by repeating an element allows additional time to pass between the pronunciation of two neighboring items, thus lowering the average rate at which information is transmitted.

Naturally, this hypothesis was unlikely to explain the variation in disfluency placement completely. First of all, the surprisal estimate employed in this thesis does not contain the information whether an optional syntactic element is available or to what extent was the phonetic smoothing employed. Furthermore, interspeaker variation

(Shriberg 1994), influence of the topic complexity (Goldman-Eisler 1968) or familiarity (Good & Butterworth 1980), or the truthfulness of the conveyed message (Smith & Clark 1993) were not coded for, even though all of these are claimed to be predictors of disfluencies.

On the other hand, the surprisal measure does capture many other hypothesized triggering events to some degree. Among other, it gives information whether the upcoming input is unpredictable/uncommon (Schnadt & Corley 2006; Beattie & Butterworth 1979), new to the conversation (Barr & Seyfeddinipur 2010) or complex (Watanabe et al. 2008; Arnold et al. 2007).

Given the design of the studies presented and the results, the causal link to high surprisal or – as observed in Study I – a sudden surprisal peak in comparison to the previous item, will require further verification through experimental studies. Importantly, the decomposition of the surprisal elements in Figure 7 (Chapter 4.4.2) has shown that the difference in surprisal at disfluency locations is partially limited to syntactically complex environments, with limited differences in the case of the n-gram surprisal and the semantic coherence measure. This suggests that models with some awareness of the underlying syntactic structure should perform better in explaining disfluency placement than those without.

This matches the observations of previous research pointing out that disfluencies tend to occur at syntactic boundaries (Schneider 2014; Bortfeld et al. 2001; Maclay & Osgood 1959). These are compatible with the function of disfluencies as time-inserting surprisal-smoothing agents: at syntactic boundaries, additional time would benefit both the listener (as would be the traditional motivation for surprisal lowering in the UID) and the speaker. This perspective is also interesting in the light of the observations reported by Engelhardt et al. (2017) and summarized in Chapter 3.1: there are both studies arguing for the listener- and speaker-orientation of unfilled pauses on the basis of patterns observed in individuals affected by autism spectrum disorder. A perspective, in which disfluencies (or at least some of their types) are beneficial for both the listener and the speaker would allow one to unite these studies. The conflicting results would then not be consequences of study design imperfections, but merely due to different study designs putting more focus on the listener- or speaker-related sources of disfluencies.

Ultimately, however, even syntactically informed models were not capable of predicting disfluencies flawlessly. This applies both to the model used in Study IIb, which was specifically provided with syntactic information, and to the encoder-decoder architecture presented in Study III, which should be capable of inferring the syntactic structure of a language automatically through training on examples. Both of these studies show that even the addition of syntactic rules does not make disfluency prediction a deterministic process. Rather, at least some of the disfluencies seem to be placed by a stochastic process. This is particularly likely for those disfluencies which are symptoms of silent repairs in the output formulation, at least from the perspective of the current state of research. Even though past studies have uncovered some regularities in the uncaught retrieval errors (Harley 2006; Bock & Levelt 1994; Fromkin 1971), the process causing the wrong retrieval remains largely unexplained. If we cannot predict the cause (the retrieval error caught by the self-monitoring system prior to pronunciation), predicting its consequence (the disfluency as a symptom of repair in the background) is equally difficult.

In this light, the results of Adell et al. (2007), who achieved a striking success with a precision score of 96.7% and recall of 57.7%, seem particularly interesting. Outperforming other studies in the area (Qader 2017; Schneider 2014; Ohta et al. 2008) as well as the present study, it is worth pointing out the differences in their setup. On the one hand, there are the algorithmic differences: while Adell et al. used classification trees (as did this thesis and – as an element of the random forest algorithm – Schneider), Ohta et al. and Qader opted for the conditional random field model. On the other hand, in terms of predictors, Adell et al. employed the concept of candidate (i.e. a word particularly likely to be followed by a disfluency) which was to my knowledge not included in any other work except in the present thesis. Their other predictors (part-of-speech tags of surrounding words, bigram probability, word string and probability of a disfluency to occur given the preceding word) were in some way included in the other models. In summary, the concept of candidate is the most pronounced distinguishing factor separating their approach from the other studies.

Considering the similarity of the method proposed by Adell et al. and other previously used approaches in terms of algorithms and predictors, it would seem that precisely this concept of candidate (though not having a particular theoretical motivation) is the decisive factor. However, Study IIa in this thesis did not confirm this claim; rather, the inclusion of the candidate predictor did not lead to any change in the performance at all (see Chapter 5.2). A potential answer to this discrepancy is provided by this thesis, concretely by the comparison of Study I with Studies IIa, IIb and III. While Study I achieved very good results (recall of 76.9%, precision of 90.8% on a held out test set), Studies IIa and IIb did not match this performance, in spite of including the independent variables of Study I in their predictor sets and using more advanced models capable of handling more complex interactions as well as imbalanced data. Even the implementation of a state-of-the-art translation model in Study III did not bring the results any closer to those of Study I. The reason for this discrepancy is likely to lie in the largest difference between the studies: the data. While Study I was very successful in disfluency prediction in the JSCC, Studies IIa, IIb and III attempted to find disfluent locations in the MICASE dataset. It is possible that the results of Adell et al. are influenced by the structure of the dataset used, too. Unfortunately, to my knowledge, there has been no replication study attempting to use the model of Adell et al. on a different Spanish dataset or test a much simpler model on the dataset used by Adell et al., i.e. the data collected in the LC-STAR project (Bisani et al. 2003). However, the note of Adell et al. (2007: 361) that 10 candidates account for 53% of all disfluencies in the corpus may hint towards similarity to the JSCC, in which 74% of disfluencies were sentence-initial. The similarity would in this case lie in the systematic skew of the transcription conventions. In the JSCC, it seems likely that sentence-initial disfluencies had a higher probability of being transcribed than those located sentence-medially or sentence-finally, given that the overwhelming domination of sentence-initial disfluencies has not been reported by previous research. If a similar transcription policy was adopted in the compilation of the LC-STAR dataset, Adell et al. would inevitably be training their model on unrealistic data.

This observation raises the inevitable concern about the availability of high quality data and a standardized benchmarking dataset. Since each of the studies reviewed worked with different data, their results cannot be compared directly. While a model may perform well on the LC-STAR corpus, the same architecture may not perform well on the corpus of Japanese used by Ohta et al. (2008). Even within the same language, this thesis has shown that large differences may exist between models trained and tested on two different datasets, in spite of both datasets being compiled by trained transcribers. Nevertheless, since disfluencies belong to the items most commonly neglected in transcription (Lindsay & O'Connell 1995), they may have been transcribed selectively in some of the datasets used for their prediction. This is especially likely if the transcribers were not specifically instructed to listen for them and transcribe them, as listeners tend to not perceive disfluencies unless they focus on them (Rieger 2003). To conclude, future advances in the area of disfluency prediction would likely benefit from a standardized benchmark dataset similar to those used for the development of machine translation or speech recognition systems.

From the corpus linguistic perspective, this thesis validated some of the previously observed trends. It found evidence for the claim that noun phrase beginnings attract disfluencies (Schneider 2014; Bortfeld et al. 2001; Goldman-Eisler 1968; Maclay & Osgood 1959) while verb phrase beginnings repel them (Schneider 2014; Maclay & Osgood 1959). Furthermore, it provided further perspective to Schneider's analysis of association scores, generalizing her observations outside of specific contexts and showing that complex interplay between the individual scores exists and that individual collocation metrics may even suggest opposite explanations. For example, the mutual information score (MI, defined in Chapter 5.2.3) of a bigram was found to be negatively correlated with its probability of being disfluent (all else being equal) while the lexical gravity score (G, defined in Chapter 5.2.4) was positively correlated. Thus, while one metric would hint that strongly mentally associated items should not be separated by a disfluency, the other provides conflicting evidence. In this respect, the comment of Gries (2013) that there is no unanimously accepted measure of collocation strength becomes particularly relevant: the uncertainty as to which of the measures actually correlates with the mental representation strength complicates the explanation of such an observation. One helpful perspective is provided by a recent paper by McConnell & Blumenthal-Dramé (2019). In their self-paced reading study, they bring the cognitive reality of many of the metrics used for association strength measurement into question, showing that they are outperformed by transitional probability and bigram frequency. They argue that while these metrics may be useful for the corpus-linguistic definition of collocations, they are not capable of predicting reading times well. As a consequence, they may not correlate with the association strength of individual items in the brain. If this was the case, they would also not be capable of capturing the degree to which a bigram is stored as a fused chunk in the memory, inseparable by a disfluency. Rather, the correlation of these scores with the probability of disfluency insertion may be due to another underlying variable.[18] Thus, further inquiries into the cognitive reality of the individual association metrics are needed. In particular, it should be validated which of the scores, if any, truly capture the strength of association of two elements of a bigram in the memory. Similarly, careful analysis of the correlation of association metrics with other effects is needed in order to explore their strengths and weaknesses.

[18] As an example of the effect of the MI as independent on the G score, this variable could be related to the syntactically-dependent association that is contained in the MI score, but should not be present in the G score

For the sake of compactness, this thesis did not address some of the previously described facets of disfluency use, as it did not expect that these could be employed as powerful predictors of disfluency location with the algorithms and training data used. However, a brief discussion of these is necessary in order to show potential avenues of further improvement.

First, the sociolinguistic perspective was suppressed due to the limited variety of speakers in the JSCC and (to a smaller degree) the MICASE. However, previous research has shown that there is both considerable variation in terms of individual speakers, e.g. the ‘deleters’ and ‘repeaters’ described by Shriberg (1994), and speaker groups, such as males versus females or younger versus older speakers (Rousier-Vercruyssen et al. 2019; Fruehwald 2016; Laserna et al. 2014; Tottie 2014, 2011; Acton 2011). Most of the differences are restricted to the choice of disfluency to be used, particularly the *um/uh*-variation. Concretely, Fruehwald (2016) observes an almost complete transition from *uh* to *um* in the course of the 20th century: speakers born around 1900 used almost exclusively *um*, while speakers born in late 1990s were much more likely to choose *uh*. As a result, including this information as a predictor could improve the precision of disfluency type selection. In terms of practical application to disfluency prediction as an element of a speech synthesis pipeline, systems aware of this variation could simulate various speaker profiles, leading to a more natural output. However, it would also require that other parts of the speech synthesis system be matched with the speaker profile used in the disfluency insertion/selection mechanism. Thus, the practical applicability is most relevant for systems that use identical training datasets for both the speech synthesis and disfluency prediction. Yet, in such systems, the disfluency prediction may not necessarily be a separate element in a pipeline; rather it may be a joint task performed together with the synthesis.

Secondly, this thesis did not evaluate the impact of disfluency placement on the message itself. Operating within the Noisy Channel framework, it assumed that the message is drafted prior to disfluencies being inserted. This is not an uncontroversial assumption, even though it is useful as a helpful simplification in disfluency prediction. On the contrary, there are studies pointing out that disfluencies may be used on purpose as an inherent part of the message (O'Connell & Kowal 2004). In such a case, disfluencies may be used to express information (in the traditional sense) or to structure it. From the functional perspective, disfluencies may operate e.g. as elements of junctures[19] segmenting an utterance into utterance parts (Daneš 1960). This may lead to emphasis on a particular segment as in the example (Daneš 1960: 51) below (the pipe character is used to indicate the juncture):

[19] Daneš (1960: 44) defines junctures as the combination of a pause and the preceding intonation contour.

(13) It is the country | that suits my wife best. **Normal complex sentence**

It is the country that suits my wife best. **Contrastive emphasis**

Alternately, the juncture location may signal different grammatical structure (and by extension a different message) as in the difference between *мост деревянный* ('a wooden bridge') and *мост* | *деревянный* ('the bridge is made of wood'), as also pointed out by Daneš (1960: 53).[20] Thus, the inclusion/exclusion of a disfluency may serve as a pointer that a specific sentence perspective should be employed. Such an interplay between the factors influencing the functional sentence perspective and disfluency placement was not fully included in the present model of disfluency placement due to both limitations in terms of data and of the language models used. First of all, many of the juncture-marking pauses are likely not transcribed given that MICASE transcription includes pauses of 1 second or more (Simpson-Vlach et al. 2003). Second, the language model defined in Chapters 2.2 and 2.3 is not capable of distinguishing individual sentence perspectives unless they are marked by the change in the textual surface form. In order to be capable of making fine-grained

[20] In this respect, it should be noted that *мост деревянный,* while attested in the Russian National Corpus (2019), is a marked version (6 attestations in approx. 300 million words) of the default *деревянный мост* (77 attestations).

distinctions of the perspective applying to a given sentence, the model would also need to include phonetic information such as the intonation pattern. Then, it would also need a large enough training dataset in which each message is encoded including the intended perspective, i.e. the description of a sentence as a distributional field of degrees of communicative dynamism (Firbas 1985). Only then could the model learn both the unmarked and marked intonation patterns and align them with the corresponding distributions of degrees of communicative dynamism. Finally, in the disfluency prediction phase, the model would need to be provided with both the fluent message and its description from the viewpoint of the functional sentence perspective: this would be the only way to distinguish such cases as in Example 15. Including this factor in the disfluency prediction – and in the speech synthesis in particular – may lead to even more natural synthesized speech. However, to my knowledge, there is currently no large enough dataset with annotation of the degrees of communicative dynamism, nor an automated tool devised for such a purpose. Thus, an attempt to include such a feature in the disfluency prediction would be beyond the scope of the present thesis.

Finally, the function of disfluencies is not entirely resolved. While multiple perspectives exist – the aforementioned functional sentence perspective approach being one of them – neither has been able to reliably account for all disfluencies. The origin of disfluencies is likely not a single one of the three basic motivations defined by Clark & Fox Tree (2002) for filled pauses in the “filler-as-symptom”, “filler-as-nonlinguistic-signal" and "filler-as-word" views. Rather, each of the views may account for a share of disfluency observations. Thus, while some disfluencies may carry a meaning (thus being representatives of the disfluency-as-word view), other may be symptoms of the stochastic process of retrieval errors or floor-holding signals issued while the next output is being prepared.

Appending the list of disfluency triggers, this thesis explored an additional information-theoretically inspired perspective, framing disfluency use as an example of information transmission smoothing. It has shown that this perspective, too, can explain some of the disfluencies used and that disfluent locations do have a surprisal profile different from the fluent ones. Though there is an overlap between the values of the surprisal estimate at fluent and disfluent locations, disfluent locations tend to be less predictable, i.e. have higher surprisal. In other words, there is a correlation between the processing complexity of an item and its probability of being disfluent.

Chapter 9
Concluding remarks and outlook

This thesis set out to present a system for disfluency prediction that could be incorporated into a speech synthesis pipeline, improving the naturalness of synthesized speech. At the same time, observations of previous research were validated against new data and outside of precisely specified experimental contexts.

This thesis proposed an additional predictor of disfluency use: the surprisal, an estimate of the processing complexity associated with each item, assessed by a psycholinguistically motivated language model. Because locations with high surprisal should be harder to process, they should also be less likely to be available and prepared in time for production. Thus, it was hypothesized that high-surprisal locations should be more likely to be disfluent. This hypothesis drew on and extended the Uniform Information Density hypothesis. It was argued that disfluencies fulfil the role of surprisal smoothing agents: items employed in order to smoothen sudden peaks in surprisal. These may be employed where the smoothing methods described by previous research (optional syntactic elements or phonetic smoothing) are unavailable or insufficient. Indeed, Studies I and IIa showed that surprisal correlates with disfluency use. They have also revealed that the position of an item within a sentence and a turn plays the decisive role in disfluency placement.

The importance of the position of an item also pointed to the influence of structural factors. This was further explored in Study IIb through POS-tagging and IOB-chunking. This study also confirmed some observations of previous research, such as the tendency of noun phrase boundaries to attract disfluencies and verb phrase boundaries to repel them. On the other hand, it also showed that some of the previously described structural relationships only hold when explored in isolation: in particular the relationship between function/content words and disfluency use was completely reversed once other factors (such as surprisal, association strength of two neighboring words or bigram frequency) were taken into account.

Finally, Study III offered a purely computational perspective on disfluency prediction. Framing it as a translation task from a fluent language into a disfluent one, it employed state-of-the-art neural translation model (the encoder-decoder architecture) in order to evaluate the relevance of this approach. The results showed that this model can perform on par with theoretically motivated models employing a range of complex predictors. In spite of training on a relatively small dataset, the encoder-decoder model achieved performance similar to models employing statistics drawn from a corpus of more than 400 million words.

All in all, this thesis offered a new approach to the complex matter of disfluency prediction. The discovered link between local surprisal and disfluency placement points to a framework that can unite some of the previously suggested triggers of disfluencies under one notion, that of processing complexity. Additionally, it has been shown that structural factors must not be neglected in disfluency prediction: items with equal surprisal values have distinct probabilities of being disfluent, dependent on their structural embedding. Thus, models with structural awareness should perform better in disfluency prediction. Finally, this thesis has presented two approaches to disfluency prediction: the theoretically motivated prediction by surprisal and the computational neural translation model. Since they performed similarly, both of them should be developed by further research.

The encoder-decoder architecture presented in Study III is particularly promising for practical applications in speech synthesis. On the one hand, it allows efficient training without the need to find or create extremely large corpora of the desired language or preprocess the data heavily; on the other hand, it can be adapted to serve for disfluency clean-up in automated speech recognition. This versatility allows efficient simultaneous training of disfluency prediction and detection systems for speech synthesis and recognition pipelines. The main limitation of this approach is the size of the training data. Given that the encoder-decoder architecture as implemented here must learn the complete rules of language from the training corpus, the MICASE is not sufficient. Future attempts at disfluency prediction should either employ a larger training dataset of transcribed natural speech or experiment with pre-training the model on a large collection of written texts, so as to improve the quality of the representations in continuous space.

Additionally, the application of the psycholinguistically motivated measure of surprisal provided a new perspective on the origin of disfluencies: the processing load caused by an item plays a role in determining its likelihood of being disfluent. Future research should investigate this observation further. Among other, experimental studies should verify the relationship while controlling for other factors not captured by the model used here. At the same time, large scale corpus studies should extend the model by other potential predictors, such as the profile of the speaker and the interlocutor.

References

Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Łukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu & Xiaoqiang Zheng. 2015. *TensorFlow*: *Large-Scale Machine Learning on Heterogeneous Systems*. https://www.tensorflow.org/.

Acton, Eric K. 2011. On Gender Differences in the Distribution of um and uh. *University of Pennsylvania Working Papers in Linguistics* 17(2).

Adell, Jordi, Antonio Bonafonte & David Escudero. 2007. Filled Pauses in Speech Synthesis: Towards Conversational Speech. In Václav Matoušek & Pavel Mautner (eds.), *Proceedings of the 10th International Conference on Text, Speech and Dialogue*, 358–365. Berlin: Springer.

Adell, Jordi, David Escudero & Antonio Bonafonte. 2012. Production of Filled Pauses in Concatenative Speech Synthesis Based on the Underlying Fluent Sentence. *Speech Communication* 54(3). 459–476.

Agirre, Eneko, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Pasca & Soroa Aitor. 2009. A Study on Similarity and Relatedness Using Distributional and WordNet-based Approaches. In *Conference of the North American Chapter of the Association for Computational*

Linguistics: Human Language Technologies. Boulder: Association for Computational Linguistics.

Allen, Mark & William Badecker. 2002. Inflectional Regularity: Probing the Nature of Lexical Representation in a Cross-Modal Priming Task. *Journal of Memory and Language* 46(4). 705–722.

Allport, Alan. 1989. Visual Attention. In Michael I. Posner (ed.), *Foundations of Cognitive Science*, 631–688. Cambridge: Massachusetts Institute of Technology Press.

Allum, Paul H. & Linda R. Wheeldon. 2007. Planning Scope in Spoken Sentence Production: The Role of Grammatical Units. *Journal of Experimental Psychology. Learning, Memory, and Cognition* 33(4). 791–810.

Andersson, Sebastian, Kallirroi Georgila, Robert A.J. Clark, David Traum & Matthew P. Aylett. 2010. Prediction and Realisation of Conversational Characteristics by Utilising Spontaneous Speech for Unit Selection. In *Speech Prosody Conference*. Chicago.

Andersson, Sebastian, Junichi Yamagishi & Robert A.J. Clark. 2012. Synthesis and Evaluation of Conversational Characteristics in HMM-based Speech Synthesis. *Speech Communication* 54(2). 175–188.

Arnold, Jennifer E., Carla L. H. Kam & Michael K. Tanenhaus. 2007. If You Say Thee Uh You Are Describing Something Hard: The On-Line Attribution of Disfluency During Reference Comprehension. *Journal of Experimental Psychology. Learning, Memory, and Cognition* 33(5). 914–930.

Arnon, Inbal & Neal Snider. 2010. More Than Words: Frequency Effects for Multi-word Phrases. *Journal of Memory and Language* 62(1). 67–82.

Attneave, F. (1959). *Applications of information theory to psychology: A summary of basic concepts, methods, and results.* Oxford: Henry Holt.

Aylett, Matthew P. 2000. *Stochastic Suprasegmentals: Relationships Between Redundancy, Prosodic Structure and Care of Articulation in Spontaneous Speech.* Edinburgh: University of Edinburgh PhD thesis.

Aylett, Matthew P. & Alice Turk. 2004. The Smooth Signal Redundancy Hypothesis: A Functional Explanation for Relationships Between Redundancy, Prosodic Prominence, and Duration in Spontaneous Speech. *Language and Speech* 47(1). 31–56.

Aylett, Matthew P. & Alice Turk. 2006. Language Redundancy Predicts Syllabic Duration and the Spectral Characteristics of Vocalic Syllable Nuclei. *Journal of the Acoustical Society of America* 119(5). 3048-3058.

Baddeley, Alan D. 1986. *Working Memory*. Oxford: Clarendon.

Bahdanau, Dzmitry, Kyunghyun Cho & Yoshua Bengio. 2015. Neural Machine Translation by Jointly Learing to Align and Translate. In *Proceedings of the International Conference on Learning Representations*.

Bailey, Karl G.D. & Fernanda Ferreira. 2003. Disfluencies Affect the Parsing of Garden-path Sentences. *Journal of Memory and Language* 49(2). 183–200.

Balling, Laura W. & Johannes Kizach. 2017. Effects of Surprisal and Locality on Danish Sentence Processing: An Eye-Tracking Investigation. *Journal of Psycholinguistic Research* 46(5). 1119–1136.

Barr, Dale J. & Mandana Seyfeddinipur. 2010. The Role of Fillers in Listener Attributions for Speaker Disfluency. *Language and Cognitive Processes* 25(4). 441–455.

Beattie, Geoffrey W. & Brian L. Butterworth. 1979. Contextual Probability and Word Frequency as Determinants of Pauses and Errors in Spontaneous Speech. *Language and Speech* 22(3). 201–211.

Bell, Alan, Jason M. Brenier, Michelle Gregory, Cynthia Girand & Daniel Jurafsky. 2009. Predictability Effects on Durations of Content and Function Words in Conversational English. *Journal of Memory and Language* 60(1). 92–111.

Bell, Alan, Michelle L. Gregory, Jason M. Brenier, Daniel Jurafsky, Ayako Ikeno & Cynthia Girand. 2002. Which Predictability Measures Affect Content Word Durations? In William Byrne, Eric Fosler-Lussier & Daniel Jurafsky (eds.), *Pronunciation Modeling and Lexicon Adaptation for Spoken Language Technology,* 65-70. Estes Park.

Bengio, Yoshua, Réjean Ducharme, Pascal Vincent & Christian Jauvin. 2003. A Neural Probabilistic Language Model. *Journal of Machine Learning Research* 3. 1137–1155.

Bengio, Yoshua, Patrice Simard & Paolo Frasconi. 1994. Learning Long-Term Dependencies with Gradient Descent Is Difficult. *IEEE transactions on Neural Networks* 5(2). 157–166.

Biber, Douglas, Stig Johansson, Edward Finegan, Geoffrey Leech & Susan Conrad. 1999. *Longman Grammar of Spoken and Written English*. Harlow: Longman.

Bird, Steven, Ewan Klein & Edward Loper. 2009. *Natural Language Processing with Python*. Beijing, Farnham: O'Reilly.

Bisani, Maximilian, Antonio Bonafonte, Nuria Castell, Elviira Hartikainen, Giulio Maltese, Asunción Moreno, Shaunie Shammas & Ute Ziegenhain. 2003. Lexicon and Corpora for Speech to Speech Translation (LC-STAR). *Procesamiento del lenguaje natural* 31. 317–318.

Błaszczak, Joanna & Dorota Klimek-Jankowska. 2016. What Can Psycholinguistic Research on Word Class Ambiguities Tell Us About Categories? *Questions and Answers in Linguistics* 3(2). 485.

Blei, David M., Andrew Y. Ng & Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3. 993–1022.

Bock, Kathryn & Willem J. M. Levelt. 1994. Language Production: Grammatical Encoding. In Morton A. Gernsbacher (ed.), *Handbook of Psycholinguistics*, 945–984. San Diego, London: Academic Press.

Boomer, Donald S. 1965. Hesitation and Grammatical Encoding. *Language and Speech* 8(3). 148–158.

Bortfeld, Heather, Silvia D. Leon, Jonathan E. Bloom, Michael F. Schober & Susan E. Brennan. 2001. Disfluency Rates in Conversation: Effects of Age, Relationship, Topic, Role, and Gender. *Language and Speech* 44(Pt 2). 123–147.

Bosker, Hans R., Hugo Quené, Ted Sanders & Nivja H. de Jong. 2014. Native 'Um's Elicit Prediction of Low-Frequency Referents, but Non-Native 'Um's Do Not. *Journal of Memory and Language* 75. 104–116.

Breiman, Leo, Jerome H. Friedman, Richard A. Olshen & Charles J. Stone. 1984. *Classification and Regression Trees*. Belmont: Wadsworth.

Brennan, Susan E. & Michael F. Schober. 2001. How Listeners Compensate for Disfluencies in Spontaneous Speech. *Journal of Memory and Language* 44(2). 274–296.

Brennan, Susan E. & Maurice Williams. 1995. The Feeling of Another's Knowing: Prosody and Filled Pauses as Cues to Listeners About the Metacognitive States of Speakers. *Journal of Memory and Language* 34(3). 383–398.

Bybee, Joan L. 2010. *Language, Usage and Cognition*. Cambridge: Cambridge University Press.

Bybee, Joan L. & Joanne Scheibman. 1999. The Effect of Usage on Degrees of Constituency: The Reduction of Don't in English. *Linguistics* 37(4). 575–596.

Chafe, Wallace. 1976. Givenness, Contrastiveness, Definiteness, Subjects, Topics and Point of View. In Charles N. Li (ed.), *Subject and Topic*, 25–55. New York: Academic Press.

Chafe, Wallace. 1992. Information Flow in Speaking and Writing. In Pamela A. Downing (ed.), *The Linguistics of Literacy*, 17–30. Amsterdam: John Benjamins.

Chen, Stanley F. & Joshua Goodman. 1999. An Empirical Study of Smoothing Techniques for Language Modeling. *Computer Speech & Language* 13(4). 359–393.

Cho, Eunah, Kevin Kilgour, Jan Niehues & Alex Waibel. 2015. Combination of NN and CRF Models for Joint Detection of Punctuation and Disfluencies. In Sebastian Möller, Hermann Ney, Stefan Steidl, Bernd Möbius & Elmar Nöth (eds.), *Interspeech*: *16th Annual Conference of the International Speech Communication Association*, 3650–3654. Dresden: International Speech Communication Association.

Cho, Eunah, Jan Niehues & Alex Waibel. 2014a. Machine Translation of Multi-party Meetings: Segmentation and Disfluency Removal Strategies. In Marcello Federico, Sebastian Stücker & François Yvon (eds.), *11th International Workshop on Spoken Language Translation*, 176–183. Lake Tahoe.

Cho, Eunah, Thanh-Le Ha & Alex Waibel. 2013. CRF-based Disfluency Detection using Semantic Features for German to English Spoken Language Translation. In *International Workshop for Spoken Language Translation*.

Cho, Kyunghyun, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk & Yoshua Bengio. 2014b. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. *arXiv preprint*.

Chollet, François & others. 2015. *Keras*: https://www.keras.io.

Christiansen, Morten H. & Nick Chater. 2016. The Now-Or-Never Bottleneck: A Fundamental Constraint on Language. *Behavioral and Brain Sciences* 39. 1-72.

Church, Kenneth W. & Patrick Hanks. 1990. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics* 16(1). 22–29.

Clark, Herbert H. & Jean E. Fox Tree. 2002. Using Uh and Um in Spontaneous Speaking. *Cognition* 84(1). 73–111.

Clark, Herbert H. & Thomas Wasow. 1998. Repeating Words in Spontaneous Speech. *Cognitive Psychology* 37(3). 201–242.

Cleland, Alexandra A. & Martin J. Pickering. 2006. Do Writing and Speaking Employ the Same Syntactic Representations? *Journal of Memory and Language* 54(2). 185–198.

Coccaro, Noah & Daniel Jurafsky. 1998. Towards Better Integration Of Semantic Predictors In Statistical Language Modeling. In *5th*

International Conference on Spoken Language Processing, 2403–2406. Sydney: International Speech Communication Association.

Cole, Jeremy & David Reitter. 2017. The Timing of Lexical Memory Retrievals in Language Production. Preprint (17 December, 2018).

Core, Mark G. & Lenhart K. Schubert. 1999. A Syntactic Framework for Speech Repairs and Other Disruptions. In Robert Dale & Kenneth W. Church (eds.), *37th Annual meeting of the Association for Computational Linguistics*, 413–420. Morristown: Association for Computational Linguistics.

Corley, Martin, Lucy J. MacGregor & David I. Donaldson. 2007. It's the Way That You, Er, Say It: Hesitations in Speech Affect Language Comprehension. *Cognition* 105(3). 658–668.

Croft, William. 2001. *Radical Construction Grammar*: *Syntactic Theory in Typological Perspective.* Oxford: Oxford University Press.

Dąbrowska, Ewa. 2014. Recycling Utterances: A Speaker's Guide to Sentence Processing. *Cognitive Linguistics* 25(4).

Dall, Rasmus, Marcus Tomalin, Mirjam Wester, William Byrne & Simon King. 2014a. Investigating Automatic & Human Filled Pause Insertion for Speech Synthesis. In International Speech Communication Association (ed.), *15th Annual Conference of the International Speech Communication Association*, 51–55. Singapore: International Speech Communication Association.

Dall, Rasmus, Mirjam Wester & Martin Corley. 2014b. The Effect of Filled Pauses and Speaking Rate on Speech Comprehension in Natural, Vocoded and Synthetic Speech. In International Speech Communication Association (ed.), *15th Annual Conference of the International Speech Communication Association*, 56–60. Singapore: International Speech Communication Association.

Daneman, Meredyth & Patricia A. Carpenter. 1980. Individual Differences in Working Memory and Reading. *Journal of Verbal Learning and Verbal Behavior* 19(4). 450–466.

Daneš, František. 1960. Sentence Intonation from a Functional Point of View. *Word* 16(1). 34–54.

Daudaravičius, Vidas & Rūta Petrauskaitė. 2004. Gravity Counts for the Boundaries of Collocations. *International Journal of Corpus Linguistics* 9(2). 321–348.

Davies, Mark. 2008-. The Corpus of Contemporary American English: 520 million words, 1990-present. http://corpus.byu.edu/coca/ (11 November, 2016).

Davies, Mark. 2018. Mutual Information. https://corpus.byu.edu/mutualinformation.asp (20 December, 2018).

Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer & Richard Harshman. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science* 41(6). 391–407.

Degaetano-Ortlieb, Stefania & Elke Teich. 2019. Toward an optimal code for communication: The case of scientific English. *Corpus Linguistics and Linguistic Theory*.

Dell, Gary S. 1980. *Phonological and Lexical Encoding in Speech Production: An Analysis of Naturally Occurring and Experimentally Elicited Speech Errors.* Toronto: University of Toronto PhD thesis.

Demberg, Vera & Frank Keller. 2008. Data from Eye-Tracking Corpora as Evidence for Theories of Syntactic Processing Complexity. *Cognition* 109(2). 193–210.

Demberg, Vera, Asad Sayeed, Philip Gorinski & Nikolaos Engonopoulos. 2012. Syntactic Surprisal Affects Spoken Word Duration in Conversational Contexts. In *Joint Conference on*

Empirical Methods in Natural Language Processing and Computational Natural Language Learning. Jeju Island.

Diessel, Holger. 2015. Usage-based Construction Grammar. In Ewa Dąbrowska & Dagmar Divjak (eds.), *Handbook of Cognitive Linguistics* (Handbücher zur Sprach-und Kommunikationswissenschaft 39), 296–322. Berlin, Boston: De Gruyter Mouton.

Dumais, Susan T. 2004. Latent Semantic Analysis. *Annual Review of Information Science and Technology* 38(1). 188–230.

Earley, Jay. 1970. An Efficient Context-Free Parsing Algorithm. *Communications of the ACM* 13(2). 94–102.

Eklund, Robert, Peter Fransson & Martin Ingvar. 2015. Neural Correlates of the Processing of Unfilled and Filled Pauses. In *7th Workshop on Disfluencies in Spontaneous Speech*.

Elliott, Lois L. 1962. Backward and Forward Masking of Probe Tones of Different Frequencies. *Journal of the Acoustical Society of America* 34(8). 1116–1117.

Elman, Jeffrey L. 1990. Finding Structure in Time. *Cognitive Science* 14(2). 179–211.

Engelhardt, Paul E., Oliver Alfridijanta, Mhairi E. G. McMullon & Martin Corley. 2017. Speaker-Versus Listener-Oriented Disfluency: A Re-examination of Arguments and Assumptions from Autism Spectrum Disorder. *Journal of Autism and Developmental Disorders*.

Erk, Katrin. 2012. Vector Space Models of Word Meaning and Phrase Meaning: A Survey. *Language and Linguistics Compass* 6(10). 635–653.

Erman, Britt & Beatrice Warren. 2000. The Idiom Principle and the Open Choice Principle. *Text* 20(1).

Federmeier, Kara D., Jessica B. Segal, Tania Lombrozo & Marta Kutas. 2000. Brain Responses to Nouns, Verbs and Class-Ambiguous Words in Context. *Brain* 123(12). 2552–2566.

Ferguson, James, Greg Durrett & Dan Klein. 2015. Disfluency Detection with a Semi-Markov Model and Prosodic Features. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 257–262. Denver: Association for Computational Linguistics.

Ferrara Boston, Marisa, John Hale, Reinhold Kliegl, Umesh Patil, Cornell University, Marisa F. Boston, University of Potsdam & Shravan Vasishth. 2008. Parsing Costs as Predictors of Reading Difficulty: An Evaluation Using the Potsdam Sentence Corpus. *Journal of Eye Movement Research* 2(1). 1–12.

Ferreira, Fernanda & Karl G.D. Bailey. 2004. Disfluencies and Human Language Comprehension. *Trends in Cognitive Sciences* 8(5). 231–237.

Finkelstein, Lev, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman & Eytan Ruppin. 2002. Placing Search in Context: The Concept Revisited. *ACM Transactions on Information Systems* 20(1). 116–131.

Firbas, Jan. 1985. Thoughts on Functional Sentence Perspective, Intonation and Emotiveness. *Brno Studies in English* 16. 11-48.

Firth, John R. 1957. A Synopsis of Linguistic Theory 1930-55. Oxford: Philological Society.

Fitzgerald, Erin, Keith Hall & Frederick Jelinek. 2009a. Reconstructing False Start Errors in Spontaneous Speech Text. In Alex Lascarides, Claire Gardent & Joakim Nivre (eds.), *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, 255–263. Morristown: Association for Computational Linguistics.

Fitzgerald, Erin & Frederick Jelinek. 2008. Linguistic Resources for Reconstructing Spontaneous Speech Text. In *International Conference on Language Resources and Evaluation*. Marrakech: European Language Resources Association.

Fitzgerald, Erin, Frederick Jelinek & Robert Frank. 2009b. What lies beneath: Semantic and syntactic analysis of manually reconstructed spontaneous speech. In Keh-Yih Su (ed.), *47th Annual Meeting of the Association for Computation Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, 746–754. Barcelona: Association for Computational Linguistics.

Floridi, Luciano. 2009. Philosophical Conceptions of Information. In Giovanni Sommaruga (ed.), *Formal Theories of Information*: *From Shannon to Semantic Information Theory and General Concepts of Information*. Berlin: Springer.

Fossum, Victoria & Roger P. Levy. 2012. Sequential vs. Hierarchical Syntactic Models of Human Incremental Sentence Processing. In *3rd Workshop on Cognitive Modeling and Computational Linguistics*, 61–69. Montreal: Association for Computational Linguistics.

Fox Tree, Jean E. 1995. The Effects of False Starts and Repetitions on the Processing of Subsequent Words in Spontaneous Speech. *Journal of Memory and Language* 34(6). 709–738.

Fox Tree, Jean E. 2001. Listeners' Uses of Um and Uh in Speech Comprehension. *Memory & Cognition* 29(2). 320–326.

Frank, Austin F. & Tim F. Jaeger. 2008. Speaking Rationally: Uniform Information Density as an Optimal Strategy for Language Production. 939–944.

Frank, Stefan L. 2013. Uncertainty Reduction as a Measure of Cognitive Load in Sentence Comprehension. *Topics in Cognitive Science* 5(3). 475–494.

Frank, Stefan L. 2017. Word Embedding Distance Does Not Predict Word Reading Time. In Glenn Gunzelmann, Andrew Howes, Thora Tenbrink & Eddy J. Davelaar (eds.), *39th Annual Conference of the Cognitive Science Society*, 385–390. Austin.

Frank, Stefan L., Leun J. Otten, Giulia Galli & Gabriella Vigliocco. 2013. Word Surprisal Predicts N400 Amplitude During Reading. In Hinrich Schuetze, Pascale Fung & Massimo Poesio (eds.), *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, 878–883. Sofia.

Frank, Stefan L., Leun J. Otten, Giulia Galli & Gabriella Vigliocco. 2015. The ERP Response to the Amount of Information Conveyed by Words in Sentences. *Brain and Language* 140. 1–11.

Frank, Stefan L. & Roel M. Willems. 2017. Word Predictability and Semantic Similarity Show Distinct Patterns of Brain Activity During Language Comprehension. *Language, Cognition and Neuroscience* 32(9). 1192–1203.

Fraundorf, Scott H. & Duane G. Watson. 2011. The Disfluent Discourse: Effects of Filled Pauses on Recall. *Journal of Memory and Language* 65(2). 161–175.

Fromkin, Victoria A. 1971. The Non-Anomalous Nature of Anomalous Utterances. *Language* 47(1). 27–52.

Fruehwald, Josef. 2016. Filled Pause Choice as a Sociolinguistic Variable. *University of Pennsylvania Working Papers in Linguistics* 22(2).

Gahl, Susanne. 2008. Time and Thyme Are not Homophones: The Effect of Lemma Frequency on Word Durations in Spontaneous Speech. *Language* 84(3). 474–496.

Gale, William A. & Kenneth W. Church. 1994. What's Wrong with Adding One. In Jan Aarts, Nelleke Oostdijk & Pieter de Haan (eds.), *Corpus-Based Research into Language*, 189–200. Amsterdam: Rodopi.

Garside, Roger & Nathan D. Smith. 1997. A Hybrid Grammatical Tagger: CLAWS4. In Roger Garside, Geoffrey Leech & Tony McEnery (eds.), *Corpus Annotation: Linguistic Information from Computer Text Corpora.*, 102–121. London: Longman.

Gers, Felix A., Jürgen Schmidhuber & Fred Cummins. 2000. Learning to Forget: Continual Prediction with LSTM. *Neural Computation* 12. 2451–2471.

Gildea, Daniel & Thomas Hofmann. 1999. Topic-Based Language Models Using EM. In Fergus R. McInnes, David Attwater, Mike Edgington, Mark S. Schmidt & Mervyn A. Jack (eds.), *6th European Conference on Speech Communication and Technology*. Bonn: European Speech Communication Association.

Gilquin, Gaëtanelle & Sylvie de Cock. 2011. Errors and Disfluencies in Spoken Corpora: Setting the Scene. *International Journal of Corpus Linguistics* 16(2). 141–172.

Goffman, Erving. 1981. *Forms of Talk*. Philadelphia: University of Pennsylvania Press.

Goldman-Eisler, Frieda. 1961. The Significance of Changes in the Rate of Articulation. *Language and Speech* 4(3). 171–174.

Goldman-Eisler, Frieda. 1968. *Psycholinguistics: Experiments in Spontaneous Speech*. London: Academic Press.

Gomez Gallo, Carlos, Tim F. Jaeger & Ron Smyth. 2008. Incremental Syntactic Planning across Clauses. In Bradley C. Love, Ken McRae & Vladimir M. Sloutsky (eds.), *Proceedings of the 30th Annual Meeting of the Cognitive Science Society*. Austin: Cognitive Science Society.

Good, David A. & Brian L. Butterworth. 1980. Hesitancy as a Conversational Resource: Some Methodological Implications. In Hans W. Dechert & Manfred Raupach (eds.), *Temporal Variables in Speech*: *Studies in Honour of Frieda Goldman-Eisler*, 2010th edn. (Janua Linguarum. Series Maior 86), 145–152. Berlin, Boston: De Gruyter Mouton.

Goodwin, Charles. 1987. Forgetfulness as an Interactive Resource. *Social Psychology Quarterly* 50(2). 115–130.

Gráf, Tomáš & Lan-fen Huang. 2019. Repeats in Native and Learner English. In Liesbeth Degand, Gaëtanelle Gilquin, Laurence Meurant & Anne C. Simon (eds.), *Fluency and Disfluency Across Languages and Language Varieties*, 219–242. Louvain-La-Neuve: Presses Universitaires.

Gries, Stefan T. 2013. 50-Something Years of Work on Collocations: What Is or Should Be Next… *International Journal of Corpus Linguistics* 18(1). 137–166.

Griffin, Zenzi M. 2003. A Reversed Word Length Effect in Coordinating the Preparation and Articulation of Words in Speaking. *Psychonomic Bulletin & Review* 10(3). 603–609.

Grimm, Robert, Giovanni Cassani, Steven Gillis & Walter Daelemans. 2017. Facilitatory Effects of Multi-Word Units in Lexical Processing and Word Learning: A Computational Investigation. *Frontiers in Psychology* 8. 555.

Grzybek, Peter. 2006. History and Methodology of Word Length Studies. In Peter Grzybek (ed.), *Contributions to the Science of Text and Language*: *Word Length Studies and Related Issues*, 15–90. Dordrecht: Springer.

Hale, John. 2001. A Probabilistic Earley Parser as a Psycholinguistic Model. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics on*

Language Technologies. Morristown: Association for Computational Linguistics.

Harley, Trevor A. 2006. Speech Errors: Psycholinguistic Approach. In Keith Brown (ed.), *Encyclopedia of Language & Linguistics*, 2nd edn., 739–745. Amsterdam: Elsevier.

Harness Goodwin, Marjorie & Charles Goodwin. 1986. Gesture and Coparticipation in the Activity of Searching for a Word. *Semiotica* 62(1-2). 51-76.

Heeringa, Wilbert, Peter Kleiweg, Charlotte Gooskens & John Nerbonne. 2006. Evaluation of String Distance Algorithms for Dialectology. In *Proceedings of the COLING/ACL 2006 Workshop on Linguistic Distances*, 51–62. Sydney: Association for Computational Linguistics.

Hochreiter, Sepp & Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation* 9(8). 1735–1780.

Hoffmann, Matthew, David M. Blei & Francis Bach. 2010. Online Learning for Latent Dirichlet Allocation. In John D. Lafferty, Christopher K. I. Williams, John R. Shawe-Taylor, Richard S. Zemel & Aron Culotta (eds.), *23rd International Conference on Neural Information Processing Systems* (1), 856–864. Vancouver: Curran Associates Inc.

Honal, Matthias. 2003. *Correction of Disfluencies in Spontaneous Speech using a Noisy-Channel Approach.* Karlsruhe & Pittsburgh: University of Karlsruhe & Carnegie Mellon University Studienarbeit.

Honnibal, Matthew & Mark Johnson. 2014. Joint Incremental Disfluency Detection and Dependency Parsing. *Transactions of the Association of Computational Linguistics* 2(1). 131–142.

Hornik, Kurt, Maxwell Stinchcombe & Halbert White. 1989. Multilayer Feedforward Networks Are Universal Approximators. *Neural Networks* 2(5). 359–366.

Hyland, Ken & John Swales. 1999. Informal Elements in English Academic Writing: Threats or Opportunities for Advanced Non-Native Speakers. In Christopher Candlin & Ken Hyland (eds.), *Writing*: *Texts, Processes and Practices*, 145–167. London, New York: Routledge, Taylor & Francis Group.

Ide, Nancy, Collin Baker, Christiane Fellbaum & Rebecca J. Passonneau. 2010. The Manually Annotated Sub-Corpus: A Community Resource for and by the People. In Jan Hajič, Sandra Carberry, Stephen Clark & Joakim Nivre (eds.), *Proceedings of the 48th Conference of the Association for Computational Linguistics*, 68–73. Uppsala: Association for Computational Linguistics.

Inkpen, Diana, Oana Frunza & Grzegorz Kondrak. 2005. Automatic Identication of Cognates and False Friends in French and English. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, 251–257. Borovets.

Irvine, Christina A., Inge-Marie Eigsti & Deborah A. Fein. 2016. Uh, Um, and Autism: Filler Disfluencies as Pragmatic Markers in Adolescents with Optimal Outcomes from Autism Spectrum Disorder. *Journal of Autism and Developmental Disorders* 46(3). 1061–1070.

Jaeger, Tim F. 2006. *Redundancy and Syntactic Reduction in Spontaneous Speech:* Stanford University.

Jaeger, Tim F. 2010. Redundancy and Reduction: Speakers Manage Syntactic Information Density. *Cognitive Psychology* 61(1). 23–62.

Jaeger, Tim F. & Roger P. Levy. 2006. Speakers Optimize Information Density Through Syntactic Reduction. *Advances in Neural Information Processing Systems*. 849–856.

Jamshid Lou, Paria & Mark Johnson. 2017. Disfluency Detection Using a Noisy Channel Model and a Deep Neural Language Model. In Regina Barzilay & Min-Yen Kan (eds.), *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, 547–553. Stroudsburg: Association for Computational Linguistics.

Janssen, Niels & Horacio A. Barber. 2012. Phrase Frequency Effects in Language Production. *PloS one* 7(3).

Jefferson, Gail. 1974. Error Correction as an Interactional Resource. *Language in Society* 3(2). 181.

Jelinek, Frederick & John D. Lafferty. 1991. Computation of the Probability of Initial Substring Generation by Stochastic Context-Free Grammars. *Computational Linguistics* 17(3). 315–323.

Jelinek, Frederick & Robert L. Mercer. 1980. Interpolated Estimation of Markov Source Parameters from Sparse Data. In Edzard S. Gelsema & Laveen N. Kanal (eds.), *Proceedings of the Workshop on Pattern Recognition in Practice*, 381–397. Amsterdam: North Holland.

Johnson, Mark & Eugene Charniak. 2004. A TAG-Based Noisy Channel Model of Speech Repairs. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, 33–39. Barcelona: Association for Computational Linguistics.

Johnson, Mark, Eugene Charniak & Matthew Lease. 2004. An Improved Model for Recognizing Disfluencies in Conversational Speech. In *Rich Transcription Workshop*.

Kapatsinski, Vsevolod. 2004. Measuring the Relationship of Structure to Use: Determinants of the Extent of Recycle in Repetition Repair. *Proceedings of the Annual Meeting of the Berkeley Linguistics Society* (30). 481–492.

Katz, Slava M. 1987. Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer. *IEEE Transactions on Acoustics, Speech, and Signal Processing* 35(3). 400–401.

Kemmerer, David. 2014. Word Classes in the Brain: Implications of Linguistic Typology for Cognitive Neuroscience. *Cortex* 58. 27–51.

Kermes, Hannah & Elke Teich. 2017. Average surprisal of parts-of-speech. In *9th International Corpus Linguistics Conference*. Birmingham.

Kingma, Diederik P. & Jimmy Lei Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference for Learning Representations*. San Diego.

Kneser, Reinhard & Hermann Ney. 1995. Improved Backing-Off for M-Gram Language Modeling. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 181–184. Detroit: Institute of Electrical and Electronics Engineers.

Kneser, Reinhard, Jochen Peters & Dietrich Klakow. 1997. Language Model Adaptation Using Dynamic Marginals. In *5th European Conference on Speech Communication and Technology*. Rhodes.

Krivokapić, Jelena. 2007. Prosodic Planning: Effects of Phrasal Length and Complexity on Pause Duration. *Journal of Phonetics* 35(2). 162–179.

Kuhn, Max. 2017. *caret*. https://topepo.github.io/caret/

Kuperberg, Gina R. & Tim F. Jaeger. 2015. What Do We Mean by Prediction in Language Comprehension? *Language, Cognition and Neuroscience* 31(1). 32–59.

Kutas, Marta, Cyma K. van Petten & Robert Kluender. 2006. Psycholinguistics Electrified II (1994–2005). In Matthew J. Traxler

& Morton A. Gernsbacher (eds.), *Handbook of Psycholinguistics*, 2nd edn., 659–724. London: Academic Press.

Lake, Johanna K., Karin R. Humphreys & Shannon Cardy. 2011. Listener vs. Speaker-Oriented Aspects of Speech: Studying the Disfluencies of Individuals with Autism Spectrum Disorders. *Psychonomic Bulletin & Review* 18(1). 135–140.

Landauer, Thomas K. & Susan T. Dumais. 1997. A Solution to Plato's Problem: The Latent Semantic Analysis Theory of Acquisition, Induction, and Representation of Knowledge. *Psychological Review* 104(2). 211-240.

Laplace, Pierre S. 1814. *A Philosophical Essay on Probabilities*. London: John Wiley & sons, 1902.

Laserna, Charlyn M., Yi-Tai Seih & James W. Pennebaker. 2014. Um… Who Like Says You Know: Filler Word Use as a Function of Age, Gender, and Personality. *Journal of Language and Social Psychology* 33(3). 328–338.

LeCun, Yann, Léon Bottou, Genevieve B. Orr & Klaus-Robert Müller. 1998. Efficient BackProp. In Klaus-Robert Müller & Genevieve B. Orr (eds.), *Neural Networks*: *Tricks of the Trade*. Berlin: Springer.

Lee, Jason, Kyunghyun Cho & Thomas Hofmann. 2017. Fully Character-Level Neural Machine Translation without Explicit Segmentation. *Transactions of the Association of Computational Linguistics* 5. 365–378.

Levelt, Willem J. M. 1983. Monitoring and Self-Repair in Speech. *Cognition* 14(1). 41–104.

Levelt, Willem J. M. 1989. *Speaking*: *From Intention to Articulation*. Cambridge: Massachusetts Institute of Technology Press.

Levenshtein, Vladimir I. 1965. Двоичные коды с исправлением выпадений, вставок и замещений символов. *Доклады Академий Наук СССР* 163(4). 845–848.

Levy, Roger P. 2008. Expectation-Based Syntactic Comprehension. *Cognition* 106(3). 1126–1177.

Lickley, Robin J. 2015. Fluency and Disfluency. In Melissa A. Redford (ed.), *The Handbook of Speech Production*, 445–474. Hoboken: John Wiley & sons.

Lindsay, Jean & Daniel C. O'Connell. 1995. How Do Transcribers Deal with Audio Recordings of Spoken Discourse? *Journal of Psycholinguistic Research* 24(2). 101–115.

Linell, Per. 1982. *The Written Language Bias in Linguistics*. Linköping: University of Linköping.

Luethi, Mathias, Beat Meier & Carmen Sandi. 2008. Stress Effects on Working Memory, Explicit Memory, and Implicit Memory for Neutral and Emotional Stimuli in Healthy Men. *Frontiers in behavioral neuroscience* 2. Article 5.

MacGregor, Lucy J., Martin Corley & David I. Donaldson. 2009. Not All Disfluencies Are Are Equal: The Effects of Disfluent Repetitions on Language Comprehension. *Brain and Language* 111(1). 36–45.

Maclay, Howard & Charles E. Osgood. 1959. Hesitation Phenomena in Spontaneous English Speech. *Word* 15(1). 19–44.

Mair, Christian. 2015. Response to Davies and Fuchs. *English World-Wide* 36(1). 29–33.

Manin, Dmitrii Y. 2006. Experiments on Predictability of Word in Context and Information Rate in Natural Language. *Journal of Information Processes* 6(3). 229–236.

Marcus, Mitchell P., Mary A. Marcinkiewicz & Beatrice Santorini. 1993. Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics* 19(2). 313–330.

McCarthy, Diana, Rob Koeling, Julie Weeds & John Carroll. 2004. Finding Predominant Word Senses in Untagged Text. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*. Barcelona: Association for Computational Linguistics.

McConnell, Kyla & Alice Blumenthal-Dramé. 2019. Effects of Task and Corpus-Derived Association Scores on the Online Processing of Collocations. Ahead of print. *Corpus Linguistics and Linguistic Theory*.

Mikolov, Tomáš, Martin Karafiát, Lukáš Burget, Jan Černocký & Sanjeev Khudanpur. 2010. Recurrent Neural Network Based Language Model. In Takao Kobayashi, Keikichi Hirose & Satoshi Nakamura (eds.), *11th Annual Conference of the International Speech Communication Association*, 1045–1048. Makuhari: International Speech Communication Association.

Mitchell, Jeffrey J. 2011. *Composition in distributional models of semantics.* Edinburgh: University Doctoral dissertation.

Mitchell, Jeffrey J., Mirella Lapata, Vera Demberg & Frank Keller. 2010. Syntactic and Semantic Factors in Processing Difficulty: An Integrated Measure. In Jan Hajič, Sandra Carberry, Stephen Clark & Joakim Nivre (eds.), *Proceedings of the 48th Conference of the Association for Computational Linguistics*, 196–206. Uppsala: Association for Computational Linguistics.

Navon, David. 1984. Resources - a Theoretical Soup Stone? *Psychological Review* 91(2). 216–234.

Ney, Hermann, Ute Essen & Reinhard Kneser. 1994. On Structuring Probabilistic Dependences in Stochastic Language Modelling. *Computer Speech & Language* 8(1). 1–38.

Ney, Hermann, Sven Martin & Frank Wessel. 1997. Statistical Language Modeling Using Leaving-One-Out. In Steve Young & Gerrit Bloothooft (eds.), *Corpus-Based Methods in Language and Speech Processing*, 174–207. Dordrecht, London: Springer.

Nivre, Joakim, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov & Erwin Marsi. 2007. MaltParser: A Language-Independent System for Data-Driven Dependency Parsing. *Natural Language Engineering* 13(2). 95-135.

O'Connell, Daniel C. & Sabine Kowal. 2004. The History of Research on the Filled Pause as Evidence of The Written Language Bias in Linguistics (Linell, 1982). *Journal of Psycholinguistic Research* 33(6). 459–474.

Oei, Nicole Y. L., Walter T. A. M. Everaerd, Bernet M. Elzinga, Sonja van Well & Bob Bermond. 2006. Psychosocial Stress Impairs Working Memory at High Loads: An Association with Cortisol Levels and Memory Retrieval. *Stress* 9(3). 133–141.

Ohta, Kengo, Masatoshi Tsuchiya & Seiichi Nakagawa. 2008. Evaluating Spoken Language Model Based on Filler Prediction Model in Speech Recognition. In *9th Annual Conference of the International Speech Communication Association*. Brisbane: International Speech Communication Association.

Osborne, John. 2011. Errors and Disfluencies in Spoken Corpora. *International Journal of Corpus Linguistics* 16(2). 276–298.

Osgood, Charles E., George J. Suchard & Percy H. Tannenbaum. 1957. *The Measurement of Meaning*. Urbana: University of Illinois Press.

Oviatt, Sharon. 1995. Predicting Spoken Disfluencies During Human–Computer Interaction. *Computer Speech & Language* 9(1). 19–35.

Owens, Sarah J., Justine M. Thacker & Susan A. Graham. 2018. Disfluencies Signal Reference to Novel Objects for Adults but Not Children. *Journal of Child Language* 45(3). 581–609.

Pascanu, Razvan, Tomáš Mikolov & Yoshua Bengio. 2013. On the Difficulty of Training Recurrent Neural Networks. In Sanjoy Dasgupta & David McAllester (eds.), *30th International Conference on Machine Learning*, 1310–1318. Atlanta.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Alexandre Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot & Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12. 2825–2830.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Alexandre Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot & Édouard Duchesnay. 2019. Scikit-learn: Decision Trees. https://scikit-learn.org/stable/modules/tree.html#tree-algorithms (1 July, 2019).

Pfeiffer, Martin. 2014. Welche Wortarten werden am häufigsten repariert? In Pia Bergmann & Peter Auer (eds.), *Sprache im Gebrauch*: *Räumlich, zeitlich, interaktional. Festschrift für Peter Auer*, 249–274. Heidelberg: Universitätsverlag Winter.

Piantadosi, Steven T., Harry Tily & Edward Gibson. 2011. Word Lengths Are Optimized for Efficient Communication. *Proceedings of the National Academy of Sciences of the United States of America* 108(9). 3526–3529.

Pickering, Martin J. & Simon Garrod. 2013. An Integrated Theory of Language Production and Comprehension. *Behavioral and Brain Sciences* 36(4). 329–347.

Pitt, Mark A., Keith Johnson, Elizabeth Hume, Scott Kiesling & William D. Raymond. 2005. The Buckeye Corpus of Conversational Speech: Labeling Conventions and a Test of Transcriber Reliability. *Speech Communication* 45(1). 89–95.

Pynte, Joël, Boris New & Alan Kennedy. 2008. On-Line Contextual Influences During Reading Normal Text: A Multiple-Regression Analysis. *Vision Research* 48(21). 2172–2183.

Qader, Raheel. 2017. *Pronunciation and Disfluency Modeling for Expressive Speech Synthesis.* Rennes: Université de Rennes 1.

Qian, Xian & Yang Liu. 2013. Disfluency Detection Using Multi-step Stacked Learning. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 820–825. Atlanta: Association for Computational Linguistics.

Rasooli, Mohammad S. & Joel Tetreault. 2013. Joint Parsing and Disfluency Detection in Linear Time. In *Conference on Empirical Methods in Natural Language Processing*, 124–129. Seattle: Association for Computational Linguistics.

Řehůřek, Radim. 2018. Experiments on the English Wikipedia. https://radimrehurek.com/gensim/wiki.html (22 February, 2018).

Řehůřek, Radim & Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *International Conference on Language Resources and Evaluation*, 45–50. Valletta: European Language Resources Association.

Remez, Robert E., Daria F. Ferro, Kathryn R. Dubowski, Judith Meer, Robin S. Broder & Morgana L. Davids. 2010. Is Desynchrony

Tolerance Adaptable in the Perceptual Organization of Speech? *Attention, Perception & Psychophysics* 72(8). 2054–2058.

Rieger, Caroline L. 2003. Disfluencies and Hesitation Strategies in Oral L2 Tests. In *Proceedings of Disfluency in Spontaneous Speech Workshop*, 41–44.

Roark, Brian. 2001. Probabilistic Top-Down Parsing and Language Modeling. *Computational Linguistics* 27(2). 249–276.

Roark, Brian, Asaf Bachrach, Carlos Cardenas & Christophe Pallier. 2009. Deriving Lexical and Syntactic Expectation-Based Measures for Psycholinguistic Modeling via Incremental Top-Down Parsing. In Philipp Koehn & Rada Mihalcea (eds.), *Conference on Empirical Methods in Natural Language Processing*, 324–333. Singapore: Association for Computational Linguistics.

Robinson, Sarita J., Sandra I. Sünram-Lea, John F. Leach & P. J. Owen-Lynch. 2008. The Effects of Exposure to an Acute Naturalistic Stressor on Working Memory, State Anxiety and Salivary Cortisol Concentrations. *Stress* 11(2). 115–124.

Rochester, Sherry R. 1973. The Significance of Pauses in Spontaneous Speech. *Journal of Psycholinguistic Research* 2(1). 51–81.

Rousier-Vercruyssen, Lucie, Anne Lacheret-Dujour & Marion Fossard. 2019. When and Why Do Old Speakers Use More Fillers Than Young Speakers? In Liesbeth Degand, Gaëtanelle Gilquin, Laurence Meurant & Anne C. Simon (eds.), *Fluency and Disfluency Across Languages and Language Varieties*, 91–108. Louvain-La-Neuve: Presses Universitaires.

Rumelhart, David E., Geoffrey E. Hinton & Ronald J. Williams. 1986. Learning Representations by Back-Propagating Errors. *Nature* 323(6088). 533–536.

Russian National Corpus. 2019. Национальный корпус русского языка. http://www.ruscorpora.ru (20 June, 2019).

Sanderman, Angelien A. & René P.G. Collier. 1995. Prosodic Phrasing at Sentence Level. In Katherine S. Harris, Fredericka Bell-Berti & Lawrence J. Raphael (eds.), *Producing Speech*: *Contemporary Issues: For Katherine Safford Harris*, 321–332. New York: AIP Press.

Saslow, Laura R., Shannon McCoy, Ilmo van der Lowe, Brandon Cosley, Arbi Vartan, Christopher Oveis, Dacher Keltner, Judith T. Moskowitz & Elissa S. Epel. 2014. Speaking Under Pressure: Low Linguistic Complexity Is Linked to High Physiological and Emotional Stress Reactivity. *Psychophysiology* 51(3). 257–266.

Sayeed, Asad, Stefan Fischer & Vera Demberg. 2015. Vector-Space Calculation of Semantic Surprisal for Predicting Word Pronunciation Duration. In Chengqing Zong & Michael Strube (eds.), *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, 763–773. Beijing.

Schaul, Tom, Sixin Zhang & Yann LeCun. 2013. No More Pesky Learning Rates. In *International Conference on Machine Learning*, 343–351. Atlanta.

Schnadt, Michael & Martin Corley. 2006. The Influence of Lexical, Conceptual and Planning Based Factors on Disfluency Production. In Ron Sun & Naomi Miyake (eds.), *28th Meeting of the Cognitive Science Society*. Hillsdale: Lawrence Erlbaum Associates; Cognitive Science Society.

Schneider, Bruce A., Liang Li & Meredyth Daneman. 2007. How Competing Speech Interferes with Speech Comprehension in Everyday Listening Situations. *Journal of the American Academy of Audiology* 18(7). 559–572.

Schneider, Ulrike. 2014. *Frequency, Chunks and Hesitations*: *A Usage-based Analysis of Chunking in English.* Freiburg i. Breisgau: Albert-Ludwigs-Universität PhD thesis.

Schoofs, Daniela, Diana Preuss & Oliver T. Wolf. 2008. Psychosocial Stress Induces Working Memory Impairments in an N-Back Paradigm. *Psychoneuroendocrinology* 33(5). 643–653.

Schuster, Mike & Kuldip K. Paliwal. 1997. Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing* 45(11). 2673–2681.

Schütze, Hinrich. 1998. Automatic Word Sense Discrimination. *Computational Linguistics* 24(1). 97–123.

Schwenk, Holger. 2012. Continuous Space Translation Models for Phrase-Based Statistical Machine Translation. In *24th International Conference on Computational Linguistics*, 1071–1080. Mumbai: Association for Computational Linguistics.

Schwenk, Holger, Daniel Dchelotte & Jean-Luc Gauvain. 2006. Continuous Space Language Models for Statistical Machine Translation. In *Proceedings of the COLING/ACL*, 723–730. Sydney: Association for Computational Linguistics.

Shannon, Claude E. 1948. A Mathematical Theory of Communication. *Bell Systems Technical Journal* 27. 379–423.

Shaoul, Cyrus & Chris Westbury. 2011. Methodological and Analytic Frontiers in Lexical Research (Part II). *The Mental Lexicon* 6(1). 171–196.

Shriberg, Elizabeth E. 1994. *Preliminaries to a Theory of Speech Disfluencies.* Berkeley: University of California PhD thesis.

Shriberg, Elizabeth E., Rebecca Bates & Andreas Stoelcke. 1997. A Prosody-Only Decision-Tree Model for Disfluency Detection. In Kokkinakis, George, Fakotakis, Nikos, Dermatas, Evangelos (ed.), *5th European Conference on Speech Communication and Technology*, 2383–2386.

Shriberg, Lawrence D., Rhea Paul, Jane L. McSweeny, Ami Klin, Donald J. Cohen & Fred R. Volkmar. 2001. Speech and Prosody Characteristics of Adolescents and Adults With High-Functioning Autism and Asperger Syndrome. *Journal of Speech, Language, and Hearing Research* 44(5). 1097.

Simpson-Vlach, Rita, Sarah Briggs, Janine Ovens & John Swales. 2002. *The Michigan Corpus of Academic Spoken English*. Ann Arbor: The Regents of the University of Michigan.

Simpson-Vlach, Rita, David Y.W. Lee & Sheryl Leicher. 2003. *MICASE Manual: The Michigan Corpus of Academic Spoken English*. Ann Arbor: University of Michigan.

Sinclair, John. 1991. *Corpus, Concordance, Collocation*. Oxford: Oxford University Press.

Siyanova-Chanturia, Anna, Kathy Conklin & Norbert Schmitt. 2011. Adding More Fuel to the Fire: An Eye-Tracking Study of Idiom Processing by Native and Non-Native Speakers. *Second Language Research* 27(2). 251–272.

Siyanova-Chanturia, Anna & Ron Martinez. 2014. The Idiom Principle Revisited. *Applied Linguistics* 36(5). 549-569.

Smith, Mark & Linda R. Wheeldon. 1999. High Level Processing Scope in Spoken Sentence Production. *Cognition* 73(3). 205–246.

Smith, Mark & Linda R. Wheeldon. 2001. Syntactic Priming in Spoken Sentence Production – an Online Study. *Cognition* 78(2). 123–164.

Smith, Nathaniel J. & Roger P. Levy. 2013. The Effect of Word Predictability on Reading Time Is Logarithmic. *Cognition* 128(3). 302–319.

Smith, Vicki L. & Herbert H. Clark. 1993. On the Course of Answering Questions. *Journal of Memory and Language* 32(1). 25–38.

Speer, Robert, Joshua Chin & Catherine Havasi. 2017. ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In Satinder Singh & Shaul Markovitch (eds.), *31st AAAI Conference on Artificial Intelligence*, 4444–4451.

Srivastava, Nitish, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever & Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* 15. 1929–1958.

Steyvers, Mark & Tom Griffiths. 2007. Probabilistic Topic Models. In Thomas K. Landauer, Danielle S. McNamara, Simon Dennis & Walter Kintsch (eds.), *Handbook of Latent Semantic Analysis*, 424–440. New York, London: Routledge.

Stoelcke, Andreas. 2002. SRILM – an Extensible Language Modelling Toolkit. In *International Conference on Spoken Language Processing*.

Strauss, Udo, Peter Grzybek & Gabriel Altmann. 2006. Word Length and Word Frequency. In Peter Grzybek (ed.), *Contributions to the Science of Text and Language*: *Word Length Studies and Related Issues* (Text, speech, and language technology v. 31), 277–294. Dordrecht: Springer.

Suh, Joyce, Inge-Marie Eigsti, Letitia Naigles, Marianne Barton, Elizabeth Kelley & Deborah A. Fein. 2014. Narrative Performance of Optimal Outcome Children and Adolescents with a History of an Autism Spectrum Disorder (Asd). *Journal of Autism and Developmental Disorders* 44(7). 1681–1694.

Sutskever, Ilya, James Martens, George Dahl & Geoffrey E. Hinton. 2013. On the Importance of Initialization and Momentum in Deep

Learning. In *International Conference on Machine Learning*, 1139-1147. Atlanta.

Sutskever, Ilya, Oriol Vinyals & Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. *Advances in Neural Information Processing Systems*. 3104–3112.

Swales, John, Begoña Bellés-Fortuño, Inmaculada Fortanet-Gómez, Christine A. Räisänen, Stephen DiDomenico, Caitlin Gdowski, Reese Havlatka, Kristen Keller, Mercedes Querol-Julián, Matthew Brook O'Donnell, Miranda Kozman, Jesse Sielaff & Stefanie Wulff. 2009. *John Swales Conference Corpus*. Ann Arbor: The Regents of the University of Michigan.

Swinney, David A. 1979. Lexical Access during Sentence Comprehension: (Re)Consideration of Context Effects. *Journal of Verbal Learning and Verbal Behavior* 18. 645–659.

Tagliamonte, Sali A. & Rolf Harald Baayen. 2012. Models, Forests, and Trees of York English: Was/Were Variation as a Case Study for Statistical Practice. *Language Variation and Change* 24(02). 135–178.

Tang, Raphael & Jimmy Lin. 2018. *Progress and Tradeoffs in Neural Language Models.* arXiv preprint.

Therneau, Thierry, Beth Atkinson & Brian Ripley. 2018. *rpart.* https://github.com/bethatkinson/rpart

Thurber, Christopher & Helen Tager-Flusberg. 1993. Pauses in the Narratives Produced by Autistic, Mentally Retarded, and Normal Children as an Index of Cognitive Demand. *Journal of Autism and Developmental Disorders* 23(2). 309–322.

Tieleman, Tijmen & Geoffrey E. Hinton. 2012. *Lecture 6.5-Rmsprop, Coursera: Neural Networks for Machine Learning*. Toronto.

Tjong Kim Sang, Erik F. & Sabine Buchholz. 2000. Introduction to the CoNLL-2000 Shared Task: Chunking. In *Conference on Computational Natural Language Learning*. Lisbon: Association for Computational Linguistics.

Tottie, Gunnel. 2011. Errors and Disfluencies in Spoken Corpora. *International Journal of Corpus Linguistics* 16(2). 173–197.

Tottie, Gunnel. 2014. On the Use of Uh and Um in American English. *Functions of Language* 21(1). 6–29.

Tremblay, Antoine, Bruce Derwing, Gary Libben & Chris Westbury. 2011. Processing Advantages of Lexical Bundles: Evidence From Self-Paced Reading and Sentence Recall Tasks. *Language Learning* 61(2). 569–613.

Vallduví, Enric. 1993. *Information Packaging: A Survey*. Edinburgh: HCRC Publications.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser & Illia Polosukhin. 2017. Attention Is All You Need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna Wallach, Rob Fergus, S.V.N. Vishwanathan & Roman Garnett (eds.), *30th Conference on Neural Information Processing System*.

Watanabe, Michiko, Keikichi Hirose, Yasuharu Den & Nobuaki Minematsu. 2008. Filled Pauses as Cues to the Complexity of Upcoming Phrases for Native and Non-Native Listeners. *Speech Communication* 50(2). 81–94.

Wiechmann, Daniel. 2008. On the Computation of Collostruction Strength: Testing Measures of Association as Expressions of Lexical Bias. *Corpus Linguistics and Linguistic Theory* 4(2). 265.

Williams, Ronald J. & David Zipser. 1989. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation* 1(2). 270–280.

Witten, Ian H. & Timothy C. Bell. 1991. The Zero-Frequency Problem: Estimating the Probabilities of Novel Events in Adaptive Text Compression. *IEEE Transactions on Information Theory* 37(4). 1085–1094.

Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg S. Corrado, Macduff Hughes & Jeffrey Dean. 2016. *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. arXiv preprint.

Yoshikawa, Masashi, Hiroyuki Shindo & Yuji Matsumoto. 2016. Joint Transition-based Dependency Parsing and Disfluency Detection for Automatic Speech Recognition Texts. In *Conference on Empirical Methods in Natural Language Processing*, 1036–1041.

Zamora-Martínez, Franciso, María J. Castro-Bleda & Holger Schwenk. 2010. N-Gram-Based Machine Translation Enhanced with Neural Networks for the French-English BTEC-IWSLT'10 Task. In Marcello Federico, Ian Lane, Michael Paul, Francois Yvon & Joseph Mariani (eds.), *Proceedings of the 7th International Workshop on Spoken Language Translation*, 45–52.

Zayats, Victoria, Mari Ostendorf & Hannaneh Hajishirzi. 2016. Disfluency Detection Using a Bidirectional Lstm. In *16th Annual Conference of the International Speech Communication Association*, 2323–2327. San Francisco: International Speech Communication Association.

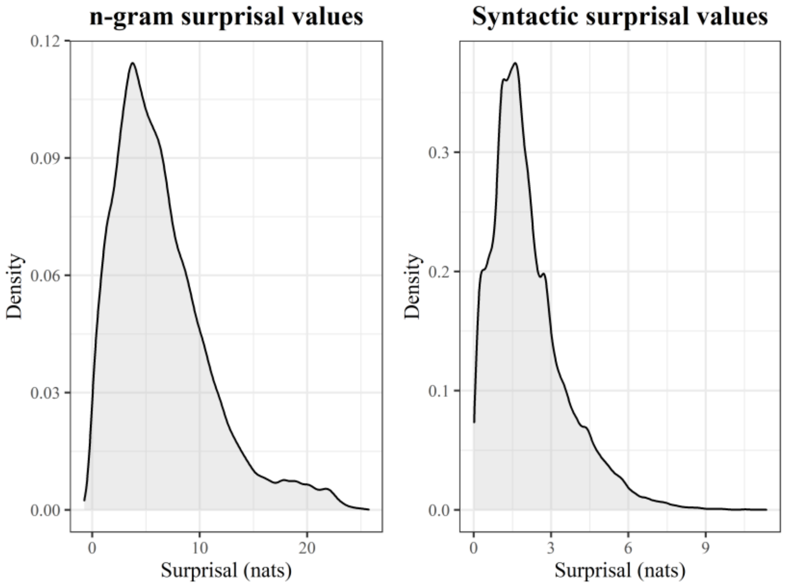
Zhao, Li-Ming, F.-Xavier Alario & Yu-Fang Yang. 2015. Grammatical Planning Scope in Sentence Production: Further Evidence for the Functional Phrase Hypothesis. *Applied Psycholinguistics* 36(05). 1059–1075.

Zhao, Li-Ming & Yu-Fang Yang. 2016. Lexical Planning in Sentence Production Is Highly Incremental: Evidence from ERPs. *PloS one* 11(1).

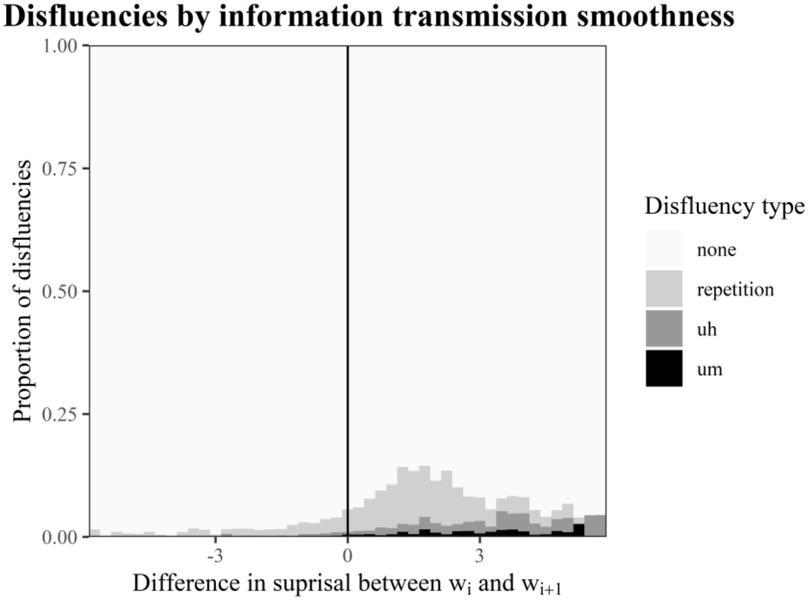
Zipf, George K. 1932. *Selected Studies of the Principle of Relative Frequency in Language*. Cambridge: Harvard University Press.

Zwarts, Simon & Mark Johnson. 2011. The Impact of Language Models and Loss Functions on Repair Disfluency Detection. In Dekang Lin (ed.), *49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 703–711.

Appendix



A.1: The distributions of syntactic and n-gram surprisal estimates in the JSCC corpus.



A.2: Effect of the magnitude of local change in surprisal on disfluency placement in the JSCC.

	β	**Std.Error**	**z**	**p-value**	
(Intercept)	-4.18	2.80×10^{-02}	-149.35	<0.001	***
Sentence-initial	1.79	3.79×10^{-02}	47.16	<0.001	***
MI	-3.67×10^{-01}	6.04×10^{-03}	-60.70	<0.001	***
G	7.83×10^{-02}	4.39×10^{-03}	17.81	<0.001	***
TP-D	$3.37\times10^{+01}$	4.19	8.05	<0.001	***
TP-B	9.78×10^{-01}	1.90×10^{-01}	5.14	<0.001	***

Bigram frequency	-6.56×10^{-06}	5.78×10^{-07}	-11.36	<0.001	***
Surprisal difference	2.01×10^{-02}	5.74×10^{-03}	3.50	<0.001	***
Surprisal	4.11×10^{-01}	3.55×10^{-02}	11.58	<0.001	***
n-gram surprisal	3.23×10^{-02}	5.51×10^{-03}	5.87	<0.001	***
Syntactic surprisal	-2.33×10^{-01}	3.18×10^{-02}	-7.33	<0.001	***
Sentence-initial : MI	1.81×10^{-01}	1.12×10^{-02}	16.21	<0.001	***
Sentence-initial : G	-4.31×10^{-02}	4.37×10^{-03}	-9.86	<0.001	***
Sentence-initial : TP-D	-9.55	9.56×10^{-01}	-9.99	<0.001	***
Sentence-initial : TP-B	-1.06	4.94×10^{-01}	-2.15	0.031	*
Sentence-initial : bigram frequency	1.16×10^{-06}	2.71×10^{-07}	4.28	<0.001	***
Sentence-initial : surprisal difference	-3.38×10^{-04}	1.08×10^{-02}	-0.03	0.975	
Sentence-initial : surprisal	-1.19	2.10×10^{-01}	-5.69	<0.001	***

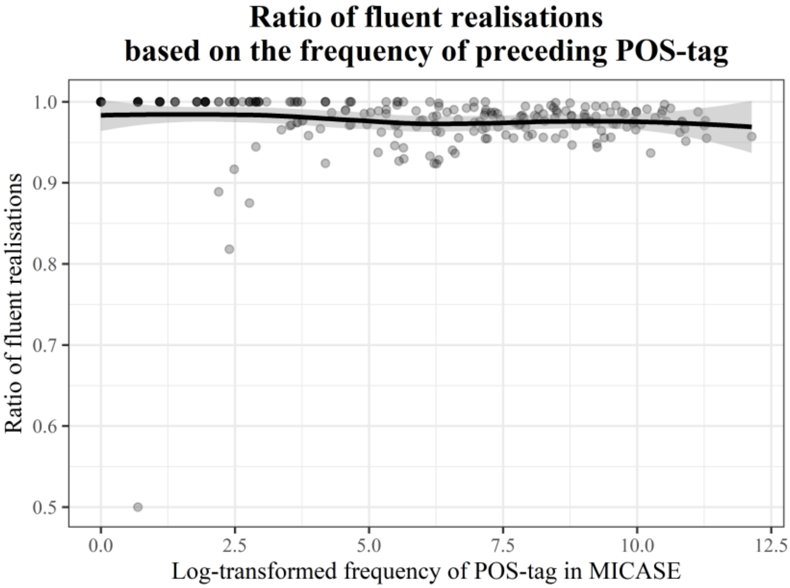
Sentence-initial : n-gram surprisal	-7.79×10^{-02}	1.41×10^{-02}	-5.53	<0.001	***
Sentence-initial : syntactic surprisal	9.92×10^{-01}	1.94×10^{-01}	5.11	<0.001	***
MI : G	-1.04×10^{-02}	1.16×10^{-03}	-8.99	<0.001	***
MI : TP-D	-5.45	3.23×10^{-01}	-16.84	<0.001	***
MI : TP-B	2.17×10^{-01}	1.05×10^{-01}	2.07	0.039	*
MI : bigram frequency	-1.48×10^{-06}	9.94×10^{-08}	-14.88	<0.001	***
MI : surprisal difference	-7.96×10^{-03}	2.83×10^{-03}	-2.81	0.005	**
MI : surprisal	-1.02×10^{-01}	1.22×10^{-02}	-8.39	<0.001	***
MI : n-gram surprisal	2.16×10^{-04}	1.89×10^{-03}	0.11	0.909	
MI : syntactic surprisal	1.20×10^{-01}	1.12×10^{-02}	10.77	<0.001	***
G : TP-D	3.67	2.33×10^{-01}	15.73	<0.001	***
G : TP-B	-4.07×10^{-02}	3.21×10^{-02}	-1.27	0.206	
G : bigram frequency	5.16×10^{-07}	4.49×10^{-08}	11.49	<0.001	***

G : surprisal difference	-1.46×10^{-03}	1.16×10^{-03}	-1.26	0.207	
G : surprisal	-3.71×10^{-03}	7.00×10^{-03}	-0.53	0.596	
G : n-gram surprisal	-9.79×10^{-03}	7.41×10^{-04}	-13.21	<0.001	***
G : syntactic surprisal	3.35×10^{-03}	6.58×10^{-03}	0.51	0.610	
TP-D : TP-B	$-2.86\times10^{+01}$	$1.44\times10^{+01}$	-1.98	0.048	*
TP-D : bigram frequency	-3.32×10^{-04}	7.94×10^{-05}	-4.18	<0.001	***
TP-D : surprisal difference	4.21×10^{-01}	1.84×10^{-01}	2.29	0.022	*
TP-D : surprisal	-2.78×10^{-01}	1.01	-0.28	0.782	
TP-D : n-gram surprisal	2.84	2.97×10^{-01}	9.53	<0.001	***
TP-D : syntactic surprisal	-1.60	8.57×10^{-01}	-1.87	0.062	.
TP-B : bigram frequency	-7.11×10^{-07}	5.71×10^{-07}	-1.25	0.213	
TP-B : diff	-1.61×10^{-02}	1.08×10^{-01}	-0.15	0.881	

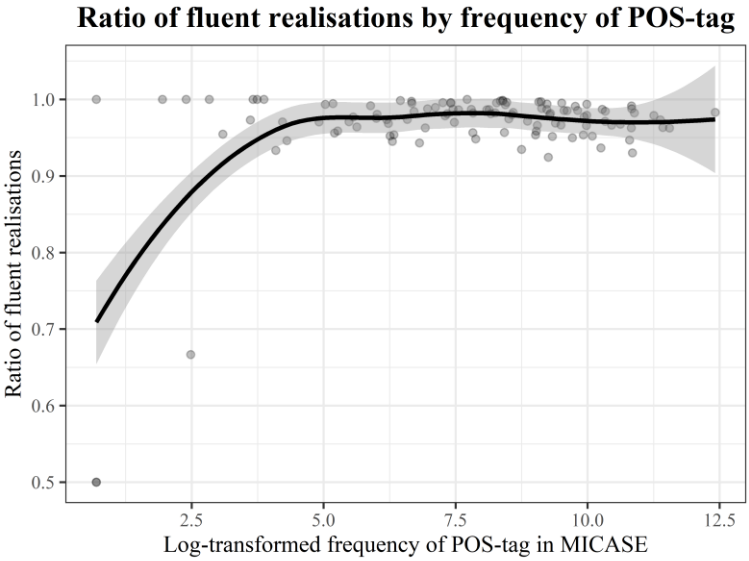
TP-B : surprisal	5.31×10^{-02}	1.59×10^{-01}	0.33	0.739	
Bigram frequency : n-gram surprisal	2.68×10^{-07}	7.47×10^{-08}	3.59	<0.001	***
Bigram frequency : surprisal difference	-5.73×10^{-08}	4.10×10^{-08}	-1.40	0.162	
Bigram frequency : surprisal	3.88×10^{-08}	3.70×10^{-07}	0.11	0.917	
Bigram frequency : syntactic surprisal	-3.17×10^{-07}	3.01×10^{-07}	-1.05	0.292	
Surprisal difference : surprisal	-4.98×10^{-02}	1.34×10^{-02}	-3.71	<0.001	***
Surprisal difference : n-gram surprisal	-6.83×10^{-03}	1.86×10^{-03}	-3.68	<0.001	***
Surprisal difference : syntactic surprisal	3.97×10^{-02}	1.25×10^{-02}	3.18	0.001	**

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

A.3 Full model parameters of a simple logistic regression disfluency prediction model on the MICASE dataset.



A.4 Disfluency rates as related to the frequency of the POS-tag before the potential disfluency location.



A.5: Disfluency rates as related to the frequency of the POS-tag before the potential disfluency location: ditto tags were simplified.

	β	Std.Error	z	p-value	
(Intercept)	-3.86	5.33×10^{-02}	-72.46	<0.001	***
B-NP	2.55×10^{-01}	4.83×10^{-02}	5.29	<0.001	***
B-PP	-4.85×10^{-01}	7.53×10^{-02}	-6.45	<0.001	***
B-VP	-9.11×10^{-02}	7.93×10^{-02}	-1.15	0.251	
I-NP	-3.96×10^{-01}	7.37×10^{-02}	-5.37	<0.001	***
I-PP	$-1.08\times10^{+02}$	$5.22\times10^{+02}$	-0.21	0.836	
I-VP	-5.34×10^{-01}	9.64×10^{-02}	-5.54	<0.001	***

Sentence-initial	1.45	3.88×10^{-02}	37.29	<0.001	***
MI	-2.52×10^{-01}	1.27×10^{-02}	-19.82	<0.001	***
G	3.26×10^{-02}	6.49×10^{-03}	5.03	<0.001	***
TP-D	6.02	4.33	1.39	0.165	
TP-B	5.96×10^{-01}	9.74×10^{-01}	0.61	0.541	
Bigram frequency	-9.46×10^{-06}	1.03×10^{-06}	-9.20	<0.001	***
Surprisal difference	-7.16×10^{-03}	1.20×10^{-02}	-0.60	0.551	
Surprisal	1.17×10^{-01}	1.81×10^{-02}	6.46	<0.001	***
POS-tag frequency	-1.06×10^{-06}	5.49×10^{-07}	-1.94	0.053	.
Content word	-6.46×10^{-01}	4.92×10^{-02}	-13.15	<0.001	***
B-NP : Sentence-initial	7.14×10^{-02}	3.66×10^{-02}	1.95	0.051	.
B-PP : Sentence-initial	3.54×10^{-01}	6.34×10^{-02}	5.60	<0.001	***
B-VP : Sentence-initial	3.41×10^{-01}	7.35×10^{-02}	4.64	<0.001	***
B-NP : MI	1.01×10^{-02}	1.19×10^{-02}	0.85	0.393	
B-PP : MI	2.73×10^{-02}	1.68×10^{-02}	1.62	0.105	
B-VP : MI	4.88×10^{-02}	1.79×10^{-02}	2.72	0.006	**
I-NP : MI	-4.52×10^{-02}	1.54×10^{-02}	-2.93	0.003	**

I-PP : MI	$-1.21\times10^{+01}$	$5.11\times10^{+01}$	-0.24	0.812	
I-VP : MI	-5.36×10^{-03}	2.87×10^{-02}	-0.19	0.852	
B-NP : G	-5.13×10^{-03}	4.85×10^{-03}	-1.06	0.291	
B-PP : G	1.46×10^{-03}	7.44×10^{-03}	0.20	0.844	
B-VP : G	1.30×10^{-02}	7.50×10^{-03}	1.73	0.083	.
I-NP : G	-3.63×10^{-02}	8.01×10^{-03}	-4.54	<0.001	***
I-PP : G	$1.27\times10^{+01}$	$6.43\times10^{+01}$	0.20	0.843	
I-VP : G	7.31×10^{-03}	1.11×10^{-02}	0.66	0.510	
B-NP : TP-D	1.61	8.19×10^{-01}	1.96	0.050	*
B-PP : TP-D	2.39×10^{-01}	9.01×10^{-01}	0.27	0.791	
B-VP : TP-D	3.16	1.79	1.77	0.078	.
I-NP : TP-D	1.13	1.05	1.08	0.281	
I-PP : TP-D	$2.67\times10^{+03}$	$1.13\times10^{+05}$	0.02	0.981	
I-VP : TP-D	-1.81	3.09	-0.58	0.559	
B-NP : TP-B	-1.86×10^{-01}	9.90×10^{-01}	-0.19	0.851	
B-PP : TP-B	-1.90×10^{-01}	1.15	-0.17	0.869	
B-VP : TP-B	-3.28×10^{-01}	1.30	-0.25	0.801	
I-NP : TP-B	-1.92×10^{-01}	1.15	-0.17	0.868	
I-PP : TP-B	$-1.13\times10^{+04}$	$1.06\times10^{+04}$	-1.07	0.283	

I-VP : TP-B	-1.81	2.26	-0.80	0.423	
B-NP : bigram frequency	5.40×10^{-06}	9.06×10^{-07}	5.96	<0.001	***
B-PP : bigram frequency	-4.71×10^{-06}	1.51×10^{-06}	-3.11	0.002	**
B-VP : bigram frequency	-2.90×10^{-06}	1.16×10^{-06}	-2.51	0.012	*
I-NP : bigram frequency	1.24×10^{-06}	1.38×10^{-06}	0.90	0.369	
I-PP : bigram frequency	-5.30×10^{-04}	5.75×10^{-03}	-0.09	0.927	
I-VP : bigram frequency	-3.40×10^{-06}	1.19×10^{-06}	-2.85	0.004	**
B-NP : surprisal difference	3.41×10^{-02}	1.17×10^{-02}	2.91	0.004	**
B-PP : surprisal difference	4.37×10^{-02}	1.73×10^{-02}	2.52	0.012	*
B-VP : surprisal difference	9.84×10^{-02}	1.94×10^{-02}	5.08	<0.001	***
I-NP : surprisal difference	4.15×10^{-02}	2.18×10^{-02}	1.91	0.056	.

I-PP : surprisal difference	5.24×10^{-01}	6.80×10^{-01}	0.77	0.442	
I-VP : surprisal difference	-4.80×10^{-02}	2.88×10^{-02}	-1.67	0.095	.
B-NP : surprisal	3.72×10^{-02}	1.75×10^{-02}	2.13	0.033	*
B-PP : surprisal	5.64×10^{-02}	3.05×10^{-02}	1.85	0.064	.
B-VP : surprisal	4.26×10^{-02}	2.53×10^{-02}	1.69	0.092	.
I-NP : surprisal	-8.38×10^{-02}	2.80×10^{-02}	-2.99	0.003	**
I-PP : surprisal	-1.06	1.01	-1.05	0.295	
I-VP : surprisal	-2.08×10^{-02}	3.99×10^{-02}	-0.52	0.603	
B-NP : POS-tag frequency	2.65×10^{-06}	4.77×10^{-07}	5.56	<0.001	***
B-PP : POS-tag frequency	-6.81×10^{-07}	7.49×10^{-07}	-0.91	0.363	
B-VP : POS-tag frequency	3.84×10^{-06}	7.07×10^{-07}	5.43	<0.001	***
I-NP : POS-tag frequency	3.43×10^{-07}	5.37×10^{-07}	0.64	0.524	

I-PP : POS-tag frequency	1.22×10^{-03}	6.38×10^{-03}	0.19	0.848	
I-VP : POS-tag frequency	2.41×10^{-06}	9.66×10^{-07}	2.50	0.013	*
B-NP : content word	-1.05×10^{-01}	5.04×10^{-02}	-2.08	0.038	*
B-PP : content word	8.31×10^{-02}	8.34×10^{-02}	1.00	0.319	
B-VP : content word	-3.82×10^{-01}	6.55×10^{-02}	-5.83	<0.001	***
I-NP : content word	1.21×10^{-01}	7.31×10^{-02}	1.65	0.099	.
I-PP : content word	2.22	$5.09\times10^{+02}$	0.00	0.997	
I-VP : content word	2.50×10^{-01}	9.69×10^{-02}	2.58	0.010	**
Sentence-initial : MI	1.39×10^{-01}	1.11×10^{-02}	12.62	<0.001	***
Sentence-initial : G	1.65×10^{-02}	4.09×10^{-03}	4.03	<0.001	***
Sentence-initial : TP-D	-1.47	7.44×10^{-01}	-1.98	0.048	*

Sentence-initial : TP-B	-7.17×10^{-01}	5.37×10^{-01}	-1.34	0.182	
Sentence-initial : bigram frequency	6.53×10^{-07}	2.75×10^{-07}	2.37	0.018	*
Sentence-initial : surprisal difference	2.03×10^{-02}	1.08×10^{-02}	1.88	0.061	.
Sentence-initial : surprisal	-1.86×10^{-01}	2.18×10^{-02}	-8.52	<0.001	***
Sentence-initial : POS-tag frequency	-9.18×10^{-07}	4.67×10^{-07}	-1.97	0.049	*
Sentence-initial : content word	4.02×10^{-01}	4.53×10^{-02}	8.87	<0.001	***
MI : G	-2.01×10^{-03}	1.04×10^{-03}	-1.94	0.052	.
MI : TP-D	-1.41	2.74×10^{-01}	-5.15	<0.001	***
MI : TP-B	1.90×10^{-01}	1.16×10^{-01}	1.64	0.102	
MI : bigram frequency	-1.07×10^{-06}	1.27×10^{-07}	-8.38	<0.001	***
MI : surprisal difference	-4.90×10^{-03}	2.97×10^{-03}	-1.65	0.099	.

MI : surprisal	9.37×10^{-03}	4.01×10^{-03}	2.34	0.019	*
MI : POS-tag frequency	2.57×10^{-07}	8.43×10^{-08}	3.05	0.002	**
MI : content word	-1.01×10^{-02}	1.21×10^{-02}	-0.84	0.404	
G : TP-D	9.69×10^{-01}	1.85×10^{-01}	5.22	<0.001	***
G : TP-B	-2.72×10^{-02}	4.24×10^{-02}	-0.64	0.521	
G : bigram frequency	4.57×10^{-07}	5.41×10^{-08}	8.46	<0.001	***
G : surprisal difference	1.44×10^{-03}	1.16×10^{-03}	1.25	0.213	
G : surprisal	-3.88×10^{-03}	1.65×10^{-03}	-2.35	0.019	*
G : POS-tag frequency	-2.12×10^{-08}	3.72×10^{-08}	-0.57	0.569	
G : content word	-2.33×10^{-02}	5.15×10^{-03}	-4.53	<0.001	***
TP-D : TP-B	$-3.37\times10^{+01}$	$1.73\times10^{+01}$	-1.95	0.052	.
TP-D : bigram frequency	-9.78×10^{-05}	8.35×10^{-05}	-1.17	0.241	
TP-D : surprisal difference	1.74×10^{-01}	2.00×10^{-01}	0.87	0.384	

TP-D : surprisal	5.62×10^{-01}	3.87×10^{-01}	1.45	0.147	
TP-D : POS-tag frequency	1.01×10^{-05}	1.00×10^{-05}	1.00	0.316	
TP-D : content word	1.26×10^{-01}	1.79	0.07	0.944	
TP-B : bigram frequency	-2.10×10^{-07}	5.33×10^{-07}	-0.39	0.694	
TP-B : surprisal difference	9.91×10^{-03}	1.14×10^{-01}	0.09	0.931	
TP-B : surprisal	-1.50×10^{-02}	1.80×10^{-01}	-0.08	0.933	
TP-B : POS-tag frequency	-5.25×10^{-06}	3.85×10^{-06}	-1.36	0.172	
TP-B : content word	9.03×10^{-01}	5.07×10^{-01}	1.78	0.075	.
Bigram frequency : surprisal difference	-1.72×10^{-08}	4.28×10^{-08}	-0.40	0.687	
Bigram frequency : surprisal	2.40×10^{-07}	1.06×10^{-07}	2.28	0.023	*
Bigram frequency :	-6.02×10^{-11}	4.85×10^{-12}	-12.41	<0.001	

POS-tag frequency					
Bigram frequency : content word	3.08×10^{-06}	7.77×10^{-07}	3.96	<0.001	
Surprisal difference: surprisal	-8.08×10^{-03}	2.06×10^{-03}	-3.93	<0.001	***
Surprisal difference: POS-tag frequency	-1.88×10^{-07}	1.01×10^{-07}	-1.87	0.062	.
Surprisal difference: content word	-3.44×10^{-03}	1.36×10^{-02}	-0.25	0.800	
Surprisal : POS-tag frequency	3.31×10^{-08}	1.35×10^{-07}	0.25	0.806	
Surprisal : content word	7.60×10^{-02}	1.86×10^{-02}	4.08	<0.001	***
POS-tag frequency : content word	-3.18×10^{-06}	4.00×10^{-07}	-7.95	<0.001	

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

A.6 Full parameters of the disfluency prediction model using both the predictors from Study IIa and the structural features from Study IIb.

Until recently, disfluencies in human language were outside of the focus of linguistic research. However, with the advent of technologies such as digital personal assistants, this approach changed. In order to mimic natural conversation, it is necessary to create a natural sounding output, including phenomena deemed undesirable in an idealized view of the language, such as disfluencies.

This thesis presents two novel approaches to disfluency prediction. It extends the list of known predictors of disfluencies with surprisal, a measure of processing complexity derived from psycholinguistic and information-theoretic observations. Additionally, it presents a computational-linguistic approach in which a machine translation architecture (encoder-decoder) is used for the prediction of disfluencies.

Die Publikationsreihe NIHIN – New Ideas in Human Interaction – entstand 2010 und ist ein Kooperationsprojekt zwischen der Hermann Paul School of Linguistics (HPSL) und der Universitätsbibliothek Freiburg (UB).

NIHIN bietet eine moderne, frei zugängliche Plattform für wissenschaftliche Essays erfahrener WissenschaftlerInnen sowie Prädikatsdissertationen, Textsammlungen zum Thema Sprache in der Interaktion und multimodale Sprachkorpora.

eucor
The European Campus

Universität
Basel

UNI
FREIBURG

ISBN 978-3-928969-84-0

9 783928 969840 >