Analyzing Authentic Texts for Language Learning: Web-based Technology for Input Enrichment and Question Generation

Dissertation

zur

Erlangung des akademischen Grades

Doktor der Philosophie

in der Philosophischen Fakultät

der Eberhard Karls Universität Tübingen

vorgelegt von

Maria Chinkina

aus

Moskau, Russland

2018

Gedruckt mit Genehmigung der Philosophischen Fakultät der Eberhard Karls Universität Tübingen

Dekan(in): Prof. Dr. Jürgen Leonhardt

Hauptberichterstatter(in): Prof. Dr. Detmar Meurers

Mitberichterstatter(in): Prof. Dr. Barbara Kaup

Tag der mündlichen Prüfung: 18.07.2018

Universitätsbibliothek Tübingen, TOBIAS-lib

Abstract

Acquisition of a language largely depends on the learner's exposure to and interaction with it. Our research goal is to explore and implement automatic techniques that help create a richer grammatical intake from a given text input and engage learners in making form-meaning connections during reading.

A starting point for addressing this issue is the automatic input enrichment method, which aims to ensure that a target structure is richly represented in a given text. We demonstrate the high performance of our rule-based algorithm, which is able to detect 87 linguistic forms contained in an official curriculum for the English language. Showcasing the algorithm's capability to differentiate between the various functions of the same linguistic form, we establish the task of tense sense disambiguation, which we approach by leveraging machine learning and rule-based methods.

Using the aforementioned technology, we develop an online information retrieval system FLAIR that prioritizes texts with a rich representation of selected linguistic forms. It is implemented as a web search engine for language teachers and learners and provides effective input enrichment in a real-life teaching setting. It can also serve as a foundation for empirical research on input enrichment and input enhancement. The input enrichment component of the FLAIR system is evaluated in a web-based study that demonstrates that English teachers prefer automatic input enrichment to standard web search when selecting reading material for class.

We then explore automatic question generation for facilitating and testing reading comprehension as well as linguistic knowledge. We give an overview of the types of questions that are usually asked and can be automatically generated from text in the language learning context. We argue that questions can facilitate the acquisition of different linguistic forms by providing functionally driven input enhancement, i.e., by ensuring that the learner notices and processes the form. The generation of well-established and novel types of questions is discussed and examples are provided; moreover, the results from a crowdsourcing study show that automatically generated questions are comparable to human-written ones.

Acknowledgements

I am grateful to each and every person who has shared this journey with me. Although your names are already immortalized in the examples used throughout this thesis, I would like to use this page to express my deep gratitude, too.

First and foremost, I would like to thank my advisor Prof. Dr. Detmar Meurers for guiding me through the rough paths of academia and providing me with a perfect balance of supervision and freedom to pursue my professional interests and goals. I am particularly grateful to my second advisor Prof. Dr. Barbara Kaup for her professional and personal support. I would also like to thank Prof. Dr. Doreen Bryant, Prof. Dr. Andrea Weber, and Prof. Dr. Kurt Eberle for being on my thesis committee.

It would have been much harder for me to do my PhD – from formulating the research questions to analyzing the studies and writing everything up – if it were not for the excellent support of the LEAD Graduate School & Research Network. I would like to acknowledge the hard work of the scientific coordination, the research center, and the LEAD faculty and thank them for making me feel at home.

I would like to thank Johann Jacoby, Michael Grosz, and Zarah Weiß for sharing their expertise and insights in the field of statistical analysis; Simón Ruiz for productive work together and his eagerness to help; Xiaobin Chen for his advise, support, and our long discussions during several conference visits; Heiko Holz for his enthusiasm and the 'visible fun'. I am extending my gratitude to the friends and colleagues that I met at the department of computational linguistics: Sowmya Vajjala, Chris Culy, Bjorn Rudzewitz, Adriane Boyd, Ramon Ziai, Eyal Schejter, Velislava Todorova, Sabrina Galasso, Jochen Saile, Kordula DeKuthy, Johannes Dellert, and Magdalena Wolska. Thank you for the constructive feedback, (technical) support, thought-provoking discussions, and fun Friday lunches. Special thanks to Madeeswaran Kannan and Ankita Oswal, computational linguistic students and research assistants at the University of Tübingen, for their support in optimizing *FLAIR* and developing the online experimental set-up.

My warmest regards go to my LEAD colleagues and friends: Cansu Atlay, Karo Mikus, Anne Knauf, Michèle Sulhmann, Ulrich Ludewig, Adam Ayaita, Molly Hammer, Christine Postema, Thomas Lösch, Juliane Kant, and many others with whom I have shared the LEAD experience, a paper, a lunch, or even a smile.

Finally, I owe the biggest thank-you to the people who have surrounded me with love, joy, support, understanding, and patience: my dearest friends in Russia, the Czech Republic, and Germany, my incredible partner, and my faraway family who are with me wherever I am. Thank you for encouraging me to strive to be a better version of myself every day.

Funding

This research was funded by the LEAD Graduate School & Research Network [GSC1028], a project of the Excellence Initiative of the German federal and state governments.

Publications

Parts of the work discussed in this dissertation have been published in the following peer-reviewed papers and theses:

1. Chinkina, M., Oswal, A., & Meurers, D. (2018). Automatic Input Enrichment for Selecting Reading Material: An Online Study with English Teachers. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications.* New Orleans, LA, pages 35-44. `http://www.aclweb.org/anthology/W18-0504`

2. Chinkina, M., & Meurers, D. (2017). Question Generation for Language Learning: From ensuring texts are read to supporting learning. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications.* Copenhagen, Denmark, pages 334-344. `http://www.aclweb.org/anthology/W17-5038`

3. Chinkina, M., Ruiz, S., & Meurers, D. (2017). Automatically generating questions to support the acquisition of particle verbs: evaluating via crowdsourcing. *CALL in a climate of change: adapting to turbulent global conditions–short papers from EUROCALL 2017*, 73. `https://tinyurl.com/qg-crowdsourcing`

4. Chinkina, M., Kannan, M., & Meurers, D. (2016). Online information retrieval for language learning. *ACL 2016: System demonstrations.* Berlin, Germany, pages 7-12. `http://www.aclweb.org/anthology/P16-4002`

5. Chinkina, M., & Meurers, D. (2016). Linguistically Aware Information Retrieval: Providing Input Enrichment for Second Language Learners. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications.* San Diego, CA, pages 188–198. `http://www.aclweb.org/anthology/W16-0521`

6. Chinkina, M. (2015). *Form-focused Language-aware Information Retrieval* (Master's thesis, Eberhard Karls Universität Tübingen). `http://www.sfs.uni-tuebingen.de/~mchnkina/downloads/Chinkina_Maria_thesis.pdf`

Contents

List of Figures

List of Tables

Abbreviations

CL	Computational Linguistics
SLA	Second Language Acquisition
FLTL	Foreign Language Teaching and Learning
QG	Question Generation
iCALL	Intelligent Computer-Assisted Language Learning
FLAIR	Form-focused Linguistically-Aware Information Retrieval
NLP	Natural Language Processing
POS	Part Of Speech
WSD	Word Sense Disambiguation
TSD	Tense Sense Disambiguation
IR	Information Retrieval
NER	Named Entity Recognition
TF	Term Frequency
IDF	Inverted Document Frequency
ARI	Automated Readability Index
ITS	Intelligent Tutoring System
API	Application Program Interface
QE	Query Expansion
CEFR	Common European Framework of Reference for Languages
GS	Gold Standard
SVM	Support Vector Machine
AUC	Area Under Curve
ROC	Receiver Operating Characteristic
CV	Cross-validation
M	Mean
SD	Standard Deviation
SE	Standard Error
CI	Confidence Interval

ICC	Intra-class Correlation Coefficient
TOST	Two One-Sided Test of equivalence
SESOS	Smallest Effect Size Of Interest

Chapter 1

Introduction

1.1 Motivation

Recent years have seen a dramatic growth in freely available natural language text data, including web pages, news articles, and social media, which have opened up entirely new theoretical, economic and educational opportunities. For the language learning context, this means easy access to a large variety of authentic documents that provide valuable input for language learners, which is crucial for language acquisition. Indeed, the importance of *input* in language learning has been repeatedly emphasized by the proponents of major second language acquisition (SLA) theories (Gass and Varonis, 1994; Swain, 1985).

Krashen (1977) argued that exposing learners to comprehensible input containing target constructions (*i+1*) is the single most important component of SLA. However, Nagy and Herman (1985) found that a single incidental encounter of a word seldom leads to its acquisition. They argued that a sufficient amount of written language exposure is needed for successful language learning. This approach is also been supported by other SLA theories that have further advanced our understanding of the role of input, specifically the frequency and perceptual salience of constructions required for learners to acquire a language (Slobin, 1985). This, in turn, can be aided by *input flood* (Trahey and White, 1993) and the corresponding computational linguistic (CL) method of *input enrichment* (Chinkina and Meurers, 2016), which ensure that a targeted structure is richly represented in the input.

At the same time, second language acquisition research has emphasized that language input alone is not sufficient to ensure successful language acquisition. Learners must also notice linguistic forms and grammatical categories (Schmidt, 1990), and teaching can facilitate such noticing through what is known as a *focus on form* (Doughty and Williams, 1998), which is designed to draw the learner's attention to relevant linguistic features of the language as they arise, while keeping the overriding focus on meaning and communication (Long, 1991, pp. 45f). For written input, the corresponding method of *visual input enhancement* has been used to make target linguistic forms more salient with the help of, e.g., boldfacing or highlighting. However, while such methods do draw the learner's attention to a linguistic form, this increased salience by itself cannot ensure that the learner processes and fully understands its meaning.

As Hulstijn (1989) pointed out, orientation towards semantic traits of linguistic forms requires directed attention – either by making statements or asking questions about these forms. VanPatten and colleagues put this idea to practice by proposing *processing instruction* (VanPatten, 2004), which Wong (2004) defines as a type of focus on form instruction. After being provided with explicit information about the target linguistic form and processing strategies, language learners engage in so-called *structured input activities* – ranging from matching exercises to questions – that ensure extensive processing of the form in a communicative setting. In line with this body of research, we argue that language activities, and questions in particular, can not only test but also facilitate the acquisition of different linguistic forms by providing *functionally driven input enhancement*, i.e., by ensuring that the learner notices and processes a given form.

1.2 Addressing Research Gaps

Most of the research gaps arise from the lack of CL research for the purposes of foreign language teaching and learning (FLTL). Indeed, CL tasks are somewhat different from those in FLTL: While computational linguists focus on inferences about temporal relations (Lapata and Lascarides, 2006), which is undoubtedly relevant for FLTL as well, foreign language teachers can benefit greatly from being able to *search texts for linguistic forms*, such as grammatical tenses and conditionals. While required by the curriculum, they cannot be easily found by simply typing their names into a search engine. We address this gap by developing a detection algorithm that identifies 87 linguistic forms specified in an official curriculum for the English language.

As language learning is not about learning forms but rather form-meaning connections (VanPatten and Oikkenon, 1996), automatically distinguishing between different meanings of the same form is an important endeavor in the field of CL. Although *disambiguation* of word senses is a well-developed CL field, its performance remains at 70-80% (Navigli, 2009). While also relevant in the FLTL context, this CL task can be extended to making distinctions between the different senses of grammatical constructions. For instance, *going to* can be used as part of present progressive (*I am going to the US tomorrow*) or a standalone phrase expressing intention (*I am going to call you tomorrow*). The present perfect tense can be used to emphasize the duration of an action (*I have lived here for three years*) or a result (*I have finished writing my thesis*). In our study on the disambiguation of tense senses, we experiment with leveraging rule-based and machine learning approaches to improve the state-of-the-art performance on this task.

Some research on *input enrichment*, or input flood, has demonstrated that learners benefit from the exposure to target linguistic forms richly represented in the language input (Trahey and White, 1993; Williams and Evans, 1998). However, recent iCALL research has mainly focused on the general readability of documents derived from machine learning algorithms using traditional CL features, such as the number of noun phrases and length of T-units. We argue that retrieving documents containing *i* and *i+1* linguistic constructions (Krashen, 1977) relevant for language learners and filtering out documents containing unknown constructions is crucial for any system providing reading material for language learners. We develop such a system and design a study in which we investigate whether English teachers need and prefer automatic input enrichment as opposed to using a standard web search engine to select reading material for class.

Once the teacher has an enriched text containing a sufficient number of target linguistic forms, they might want to make those forms more salient for their students via intonation (in speech) or highlighting (in text). This SLA technique of *input enhancement* (Sharwood Smith, 1993) was automated using state-of-the-art natural language processing technology in the WERTi system (Meurers et al., 2010). Its multilingual extension VIEW is implemented as a browser add-on and is able to turn any web page into an interactive exercise by enhancing the specified linguistic forms on the page either by coloring them or replacing them with fill-in-the-blank or multiple choice items. One problem with *visual* input enhancement is that making a form visually more visually salient does not ensure that it is noticed and cognitively processed more thoroughly nor do we know which aspect of that form the reader will notice and how it will be interpreted. For example, coloring the form *has been raining* in a text may draw the reader's attention to any aspect of it: the number or length of the words, the *-ing* suffix of the last word, etc. In addition, noticing the form does not necessarily mean that a learner will be able to map it to its present perfect progressive interpretation, which emphasizes the duration of an action that started in the past. As a solution, we propose the method of *functionally driven input enhancement*, which uses automatically generated questions about target linguistic forms in the text to ensure their understanding and processing.

In conclusion, we argue that CL research should support FLTL more actively. The main goal of this thesis is, therefore, to leverage and improve existing NLP tools and CL methods to develop efficient, theory-motivated applications for language teaching and learning. This thesis makes the approach concrete by providing teachers support in:

- searching for linguistically-rich reading material (*input enrichment*) with the option of highlighting the target linguistic forms in the text (*input enhancement*), and
- generating text-based questions given a list of target linguistic forms (*functionally driven input enhancement*).

1.3 Research Questions

Here we present the general research questions that are relevant for the thesis as a whole and are addressed in Chapters 4, 5, and 6. Concrete research questions and hypotheses for each of the individual studies are presented in the corresponding sections: 4.2, 5.4, 6.7, and 6.8.

1. How well can we automatically *detect linguistic forms* and *disambiguate their senses* by leveraging rule-based, crowdsourcing, and machine learning approaches?

2. Does *input enrichment* succeed in giving teachers material that is (i) rich in the linguistic forms they care about, (ii) relevant to their topic of interest, and (iii) suitable as a reading assignment for their students?

3. Can *automatically generated questions* provide functionally driven input enhancement and be used alongside the human-written questions in FLTL?

1.4 Terminology

By **linguistic forms**, we mean lexical items and grammatical constructions that are of pedagogical interest to language teachers. Examples of ***lexical items*** include words and word expressions, while examples of ***grammatical constructions*** include tenses, conditionals, and other forms that can be derived by using the syntactic rules of a language.

Input enrichment is an automatic method of maximizing the number of occurrences of target linguistic forms in a collection of texts. While text manipulation and editing is a viable alternative, we approach this task as an information retrieval one and prioritize the texts containing the best representation of target linguistic forms.

Input enhancement is an umbrella term for techniques used to make linguistic forms more salient to the learner. ***Visual input enhancement***, also known as textual enhancement, uses written or typographical cues, such as boldfacing and highlighting, to draw learners' attention to target linguistic forms. ***Functionally driven input enhancement*** not only draws learners' attention to linguistic forms but also facilitates processing and deeper understanding by asking either meaning-driven questions about their interpretation or factual questions about their immediate context.

Automatic question generation is approached in this thesis as the computational linguistic task of generating questions from declarative sentences by applying a set of rules and constraints or by using templates.

1.5 Approach and Key Contributions

In our work, we combine insights from SLA research with state-of-the-art computational linguistic, machine learning, and statistical methods to support language teachers and learners in selecting topically and linguistically appropriate reading materials and creating activities that ensure that target linguistic forms are noticed and processed. The key contributions of this thesis are:

- We developed the web-based *FLAIR* system, which, as an input enrichment tool theoretically grounded in SLA research, provides linguistically rich reading material for FLTL: `www.purl.org/icall/flair`.
- We conducted an online study that demonstrated that English teachers prefer automatic input enrichment over standard web search when selecting authentic reading materials for their students.
- We implemented an algorithm for detecting 87 linguistic forms specified in the official curriculum for the English language in the state of Baden-Württemberg, Germany.
- We compiled a dictionary of grammatical tenses and their coarse-grained senses and created a corpus of sentences from news articles containing 4089 instances of grammatical tenses annotated with those senses.
- We trained state-of-the-art machine learning models for the task of tense sense disambiguation that outperformed a strong baseline and the state of the art for this task.
- We introduced two novel types of FLTL questions that provide functionally driven input enhancement by drawing learners' attention to target linguistic forms and facilitating their processing and understanding of these.
- We designed an automatic question generation tool that generates text-based questions providing functionally driven input enhancement for FLTL: `www.purl.org/qg`.

1.6 Outline

The rest of the thesis is organized as follows:

Chapter 2 provides theoretical support from the field of SLA and discusses standard CL approaches and methods for developing iCALL tools. To contextualize our approach, we give an overview of some existing systems that provide reading material and exercises for language learners.

Chapter 3 introduces the *FLAIR* system, which retrieves appropriate reading materials for language learners and generates questions targeting the specified linguistic forms, thereby providing automatic input enrichment and functionally driven input enhancement to language teachers and learners.

Chapters 4, 5, and 6 discuss and evaluate each of the components in the *FLAIR* pipeline:

Chapter 4 is subdivided into two parts. The first part focuses on the rule-based detection of the linguistic forms relevant for language teachers and learners. The second part adopts a machine learning approach to address the task of tense sense disambiguation in order to provide a richer variety of contexts in which grammatical tenses are used.

Chapter 5 discusses the selection of appropriate reading material for FLTL and the information retrieval algorithms supporting it. It also presents an online study investigating English teachers' need and preference for automatic input enrichment as opposed to using standard web search when selecting authentic reading materials for their students.

Chapter 6 introduces two novel types of questions providing functionally driven input enhancement and the algorithms used to automatically generate them. The quality of computer-generated questions is compared to that of human-written ones in two crowdsourcing studies.

Finally, Chapter 7 summarizes the main results generated by the work that went into this thesis, discusses the limitations, and suggests directions for future research.

Chapter 2

Background

Parts of the work discussed in this chapter appeared in the following thesis:

1. Chinkina, M. (2015). *Form-focused Language-aware Information Retrieval* (Master's thesis, Eberhard Karls Universität Tübingen).

Our work lies at the intersection of second language acquisition, computational linguistics, and intelligent computer-assisted language learning. In the following sections, we review the theoretical and practical research done in these fields that is relevant to our work.

2.1 Second Language Acquisition

Second Language Acquisition (SLA) research is of an interdisciplinary nature since it has evolved from several disciplines, namely, linguistics, psychology, cognitive science, and education. SLA hypotheses have roots in these fields and shed light on particular aspects of the language learning process. As input, noticing, and form-meaning mapping are key SLA concepts that are directly related to our work, we review them in this section.

2.1.1 Input and Noticing

Acquisition of a language directly depends on the learner's exposure to it. Hence, SLA research consistently emphasizes the importance of input in second language (L2) learning (Krashen, 1977; Swain, 1985; Gass and Varonis, 1994). *The input hypothesis* (Krashen, 1977), also termed *i+1*, is driven by the notion of comprehensible input, i.e., language input at a slightly more advanced level than learners' current foreign language (L2) competence. Krashen (2003) saw comprehensible input as the cause of language acquisition and argued that exposing learners to language input containing *i* as well as *i+1* structures is a better method of developing grammatical accuracy than explicit grammar instruction. Nevertheless,despite being a prominent SLA paradigm in the late 1970s, this theory left some gaps in our understanding of the cognitive aspects of SLA processes that other theories tried to fill in later.

Connectionism theory, which evolved from the field of cognitive psychology, advocates taking a data-driven approach to language learning. Connectionists see the brain as a statistical recorder of the frequencies of words and structures. They stress the importance of input, which is considered the source of both the units and the rules of language. However, Schmidt (1990) points out that not all input − however rich − can become intake for language learning. This insight led Schmidt to emphasize the importance of attention and noticing in language acquisition. According to his *noticing hypothesis*, the learner must consciously notice L2 forms in order to acquire them. This hypothesis has been confirmed by numerous studies of attention and awareness in L2 learning (Robinson, 2001, 2003).

Consequently, Long (1991) introduced the concept of *focus on form* in education. Focus on form refers to pedagogical instruction aimed at directing the learner's attention to particular structures in the input by making them more salient in order to promote their acquisition. This method has proved to be superior to purely communicative instruction (Leeman et al., 1995). While the various practical applications of this approach are discussed further in this section, they all try to address the following question: How can one ensure conscious noticing − or even processing − of target linguistic forms, and is this at all necessary for language acquisition?

2.1.2 Input Flood, Enrichment, and Enhancement

Practical applications of the noticing hypothesis and focus-on-form instruction had to face the problem of finding texts with a sufficient number of target structures, which was addressed by the method of *input flood* (Trahey and White, 1993). Its goal is to ensure learners' incidental exposure to a large number of target linguistic forms. Benati (2016) reviews the research testing the effects of input flood on SLA and comes to the conclusion that it might increase learners' awareness of the different possibilities a language offers but cannot guarantee that target linguistic forms are noticed.

In the field of computational linguistics, the corresponding technique of automatically ensuring that a target structure is frequently represented in a text is referred to as *input enrichment* (Chinkina and Meurers, 2016) and is approached as an information retrieval task (see Section 2.2.3 for a formal definition of information retrieval). That is, given a collection of texts and a grammatical query consisting of one or more linguistic forms, an input enrichment system prioritizes the texts containing the best representation of the target forms. Our system implementing this approach is introduced in Chapter 3 and discussed in detail in Chapter 5.

Input flood, or enrichment, is also in line with the perceptual salience approach by Slobin (1985), who considers the frequency and salience of constructions in input crucial for how L2 learners process and learn the language. Increasing the salience of richly represented linguistic forms is the goal of *input enhancement* (Sharwood Smith, 1993), which can be seen as an instantiation of focus-on-form instruction (Long, 1991). While input enhancement can be used to draw learners' attention to the occurrences of linguistic forms in speech (White et al., 1991), SLA researchers mainly investigate the effects of its textual form referred to as textual or *visual input enhancement*. As the meta-analysis by Lee and Huang (2008) shows, the results of studies on the isolated effect of visual input enhancement on language acquisition have been mixed. One option for pushing this research further is to investigate other types of input enhancement or in combination with other input activities. Whether these activities should provide focus on form, focus on meaning, or a combination of both to facilitate SLA is an empirical question that is reviewed further in this section.

2.1.3 Form, Meaning, and Redundancy

The debate over the relative importance of accuracy (*form*) and communication (*meaning*) in the FLTL classroom has been around for decades (Leeman et al., 1995; Seedhouse, 1997; Chang, 2011). Several studies show that learners process input for meaning before processing it for form (VanPatten, 1990; Wong, 2001). However, Norris and Ortega (2000) argued that simultaneously directing the learner's attention to form and meaning in the input does not hinder L2 development or reading comprehension. Leow et al. (2008) came to the same conclusion after revisiting the methodology used in the replication studies mentioned above and conducting a new study. They did not find any statistically significant differences in comprehension between different intervention groups. Finally, a study by Morgan-Short et al. (2012) demonstrated that learners who attended to and processed linguistic forms while reading for meaning actually scored higher on comprehension than those who only read for meaning.

The four stages believed to be involved in the learning of form-meaning connections are: (i) initial connection, (ii) subsequent processing of the connection, (iii) continual encounters of it in the input, and (iv) accessing form-meaning connections for use (VanPatten and Oikkenon, 1996; VanPatten, 2002). A pedagogical intervention following these steps and designed to ensure that learners make form-meaning connections during reading was introduced by VanPatten and Cadierno (1993) as *processing instruction*. Work in this paradigm provides insights on the relative importance of each of its components – explicit instruction, practice activities (so-called structured input activities, which are discussed further in the section), and learner production – for SLA (e.g., Farley, 2001; Wong, 2004; Marsden and Chen, 2011; DeKeyser et al., 2002).

In his update on the principles of processing instruction, VanPatten (2002) revises the concept of the communicative value of a linguistic form. The idea behind this concept is that other lexical items in the context may express the same meaning as the target linguistic form, thus introducing semantic redundancy. For example, the sentence *John went to school yesterday* provides two past tense cues: the past simple form *went* and the adverb *yesterday*. VanPatten's primacy of meaning principle states that learners process input for meaning before they process it for form and prefer lexical items (*yesterday*) to grammatical ones (*went*). An attempt to draw students' attention to the past simple tense using this sentence may not be successful: As there is semantic redundancy introduced by other linguistic cues in the sentence, the students will not have to rely on the target linguistic form to understand the meaning. However, linguistic forms do not always co-occur with temporal expressions: The question *Where did he go?* provides no cues of the past simple tense except for the past form *did*. Thus, the communicative value of a linguistic form can increase or decrease depending on the context. VanPatten (2002) even argues that a form with no or consistently little communicative value is the least likely to get processed and may never get acquired without help. However, more empirical studies are needed to support this argument.

2.1.4 Structured Input Activities

Structured input is defined as "input that is manipulated in particular ways to push learners to become dependent on form and structure to get meaning" (Lee and VanPatten, 1995). Structured input activities can be seen as an umbrella term for a wide range of communicative language-teaching techniques. They provide learners with enriched input and prompt them to attend to and process the target linguistic form in order to understand the meaning and complete the activity.

One of the key components of processing instruction (VanPatten and Cadierno, 1993), *structured input practice*, has been identified as particularly effective in fostering L2 development. Pointing out the importance of a systematic focus on target linguistic forms, VanPatten and Oikkenon (1996) found that contextualized structured input activities were more effective than explicit explanations of rules for intermediate learners of Spanish. Benati (2004) and Wong (2004) replicated the study targeting different linguistic forms and came to the same conclusion. Although a number of other studies did not find the same effect (e.g., DeKeyser and Sokalski, 1996; Collentine, 1998), VanPatten (2002) reviews their findings as complimentary rather than contradictory to the principles of processing instruction.

VanPatten (2002) distinguishes between referential and affective structured input activities. The former have a target correct answer that the learner is expected to produce. Neupane (2009) translates VanPatten's examples into English and presents a referential listening activity targeting the causative versus non-causative use of the verb *to make*: First, the students are presented with several questions: *Who prepared an omelet?*, *Who did the homework?* etc. Then the teacher reads out the same number of sentences containing the answers to the questions: *The teacher made the student do his homework*, *Ram made an omelet*, etc. Such a task encourages students to attend to the target linguistic form in every sentence and rely on their understanding of its meaning to produce the correct answer. Note that there are no other cues pointing to the correct answer in the sentences except for the target linguistic form itself. Affective structured input activities, on the other hand, have learners engage in actively processing information about the real world and expressing their opinions. An example of such an activity targeting the same linguistic form (*to make somebody do something*) is selecting one option from a set of alternatives for a list of sentences with the target form: *An adult made [a child/a dog] bark. An adult made [a child/a dog] eat meat. An adult made [a child/a dog] read a story.* Students complete the activity and then share and discuss their answers. Note how both options are reasonable in the second sentence and there are no correct or incorrect answers in this activity.

Marsden and Chen (2011) conduct a study comparing the effects of referential and affective structured input activities on the acquisition of the *-ed* past tense inflection. Their results confirm the conclusion that the use of processing instruction, and structured input activities in particular, leads to learning gains. In addition, they suggest that participants mostly gained explicit knowledge and argue that this was induced from the referential activities. Finally, they conclude that affective activities, either alone or following referential activities, do not have any impact on learning the target linguistic construction.

To conclude, a large body of research has been done on testing the effects of input enrichment, input enhancement, and processing instruction. However, many studies have yielded mixed results. Some have argued that the original studies were not properly replicated (VanPatten, 2002) leading to inconsistent results. Certainly, the choice of target linguistic forms, types of input enhancement, and structured input activities are important factors that can influence the design of a study and its final results.

Whether by simply exposing the learner to certain linguistic forms or by providing activities targeting those forms, all of the aforementioned approaches rely on the existence of appropriate reading materials with a rich representation of linguistic forms for effective language acquisition. Manually searching for such reading material takes a lot of time and effort so teachers often fall back on schoolbook texts designed to introduce and practice the relevant constructions. The limitations this puts on the choice of texts and other considerations are discussed below.

2.1.5 Authentic reading material

School textbook material is criticized for using artificial language, i.e., containing stilted and unnatural vocabulary and grammatical constructions. Using up-to-date jargon and slang in textbooks, on the other hand, makes little sense since their lifespan is quite short and might even be considered obsolete by the time the textbook is published. At the end of the day, when students leave the classroom, unless they have been exposed to the foreign culture directly, they might lack some knowledge of real-life language usage.

Foreign language teaching and learning (FLTL) professionals (e.g., Peacock, 1997; Morrison, 1989; Swaffar, 1985) have experimented with practical methods of teaching English courses with authentic texts of various types and levels. They demonstrate positive outcomes overall, both with respect to motivating learners to learn the target language and developing their communicative competence. Interestingly, learners in some studies reported authentic materials to be significantly less interesting than curated materials (Peacock, 1997). Though this finding seems to speak against using authentic texts in the educational context, in our opinion, it instead demonstrates just how unprepared schoolchildren are to read real-life texts. Given that they will mostly come across such texts outside their learning environment, it actually provides an additional ground for introducing authentic materials into the everyday language learning process.

With that in mind, more and more teachers are turning to the Web in search of authentic texts. The sources for authentic texts on the Web include a wide range of texts from books, newspaper articles, and blogs up to (or rather down to) short social media posts. As authentic texts are becoming an integral part of the FLTL classroom, teachers and learners alike need support in finding appropriate reading material to foster learners' language acquisition via focus on form (Meurers, 2012). Thus, whether one can make use of the aforementioned findings from the field of SLA to find materials that are appropriate for learners' competence level and meet their interests at the same time remains an open question. This issue is addressed throughout the thesis and is one of the motivations of the *FLAIR* system described in Chapter 3.

2.2 Computational Linguistics

While input enrichment, input enhancement, and processing instruction have been traditionally implemented manually, computational linguistic methods can support their automation. In this section, we overview the basic concepts and tasks of computational linguistics (CL), which is an interdisciplinary field that studies computational approaches to addressing linguistic questions. Natural Language Processing (NLP) is an area of CL dealing with natural language data and its efficient modeling.

2.2.1 Natural Language Processing Tasks

Tokenization, *POS tagging*, and *lemmatization* are the building blocks of NLP and follow the process of assigning words to increasingly more fine-grained categories. First, a parser recognizes a string of characters as a word, then it assigns a part-of-speech (POS) tag to it, e.g., to differentiate between *evening* as a noun (late afternoon) and as a gerund or present participle (making more even), as shown in (1) and (2). This information is then used to find the lemma of the word, which for the noun *evening* will be *evening* and for the gerund or present participle *evening* will be *even*. The performance of POS taggers is evaluated using accuracy per token and is generally very high, at 97%.[1]

(1) Sowmya was working late in the *evening*.

(2) Vladimir has finished *evening* the platform.

Given this information about the individual tokens, *constituency parsers* resolve a sentence into its components, i.e., phrases and words, that form a parse tree and represent the syntactic structure of a sentence (see Figure 2.1). This process has traditionally relied on probabilistic context-free grammars (PCFG), with recent approaches implementing combinations of PCFG and neural networks (Socher et al., 2013) and reporting an F$_1$-measure of 90-92% McClosky et al. (2006). *Dependency parsing*, on the other hand, focuses on the grammatical relations between individual tokens rather than their grouping into phrases (see Figure 2.2). The performance of state-of-the-art dependency parsers has recently been boosted through

[1] `https://aclweb.org/aclwiki/POS_Tagging_(State_of_the_art)`

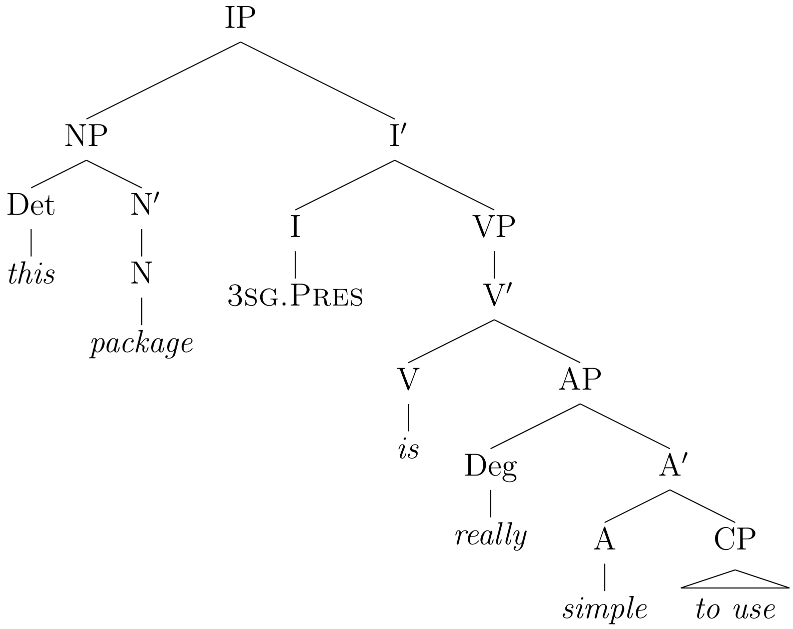


FIGURE 2.1: Output of a constituency parser for the sentence *This package is really simple to use.*

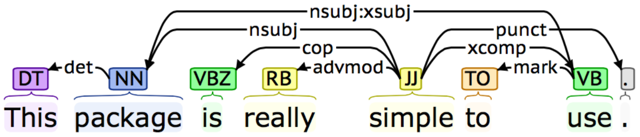
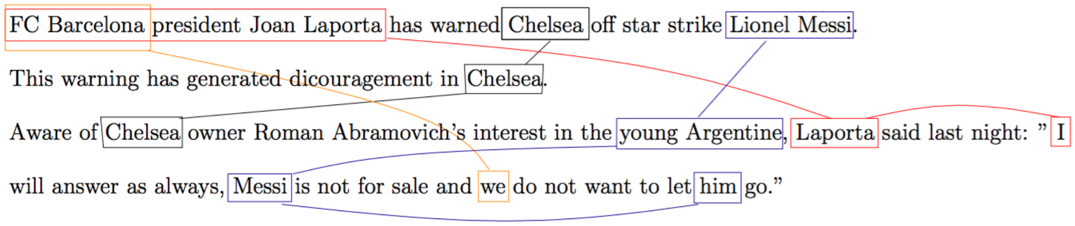


FIGURE 2.2: Output of a dependency parser for the sentence *This package is really simple to use.*

the use of neural networks achieving an accuracy of 92% (Chen and Manning, 2014; Pei et al., 2015), although it has traditionally been addressed with graph-based approaches relying on hand-crafted features. The most commonly used lists of POS tags and dependencies are from the Penn Treebank Project (Santorini, 1990) and the Universal Dependencies framework,[2] respectively.

NLP tools not only take into account the syntactic structure of a sentence but also integrate the semantic information provided either by the direct context of a word or dictionaries and ontologies. Tasks such as *named entity recognition* (NER) and sentiment analysis are examples of a hybrid implementation of syntactic and semantic features. The goal of NER is to single out proper nouns, time expressions, and other named entities from a text and tag each of them with one of the available NER tags (person, organization, time, location, etc.). *Sentiment analyzers* usually assign positive, negative, or neutral tags to whole sentences and are built on the

[2] http://universaldependencies.org



FC Barcelona president Joan Laporta has warned Chelsea off star strike Lionel Messi.

This warning has generated dicouragement in Chelsea.

Aware of Chelsea owner Roman Abramovich's interest in the young Argentine, Laporta said last night: " I

will answer as always, Messi is not for sale and we do not want to let him go."

Figure 2.3: A text excerpt with resolved coreferences produced by TALP Research Center (`http://www.talp.upc.edu`).

assumption that there are positive and negative words, such as *amazing* and *horrible.* Additional rule-based or machine learning algorithms take care of negation (e.g., *not great*), adverbs (e.g., *rarely good*), changes in the meaning of certain words from negative to positive and vice versa (e.g., *decadent*), and even sarcasm (e.g., *My mom would sure love it, heh*). The last of these has become particularly popular with the rise of social media data mining (Maynard and Greenwood, 2014; Ghosh et al., 2015).

Coreference resolution is considered to be one of the most challenging NLP tasks. Its goal is finding all linguistic expressions that refer to the same real-world entity in a text or speech, as shown in Figure 2.3 below. While coreference resolution is crucial for natural language modeling and understanding, the best supervised methods only achieve an accuracy of 64-76% depending on the evaluation metric and the dataset (Rahman and Ng, 2009). However, neural network approaches produce slightly superior results of 70-79% on the same datasets (Clark and Manning, 2016).

All of the NLP components mentioned above are implemented in the state-of-the-art Stanford CoreNLP toolkit (Manning et al., 2014) and are used as a pre-processing step for the system described in this thesis.

2.2.2 Disambiguation of Senses

In the *evening* example at the beginning of the previous section (1 and 2), both humans and parsers should not have any difficulty distinguishing the noun and the verb uses of the word *evening* in different contexts. However, ambiguity may also arise within one POS category as in (3) and (4). Such semantic ambiguity may not influence the accuracy of a parser but is detrimental for other CL tasks such as machine translation and information retrieval.

(3) Mareike saw a *seal* lying on the shore.

(4) Sophie broke the *seal* of the letter.

Distinguishing between different meanings of a word in context is the goal of the CL task of *word sense disambiguation* (WSD). This is a classification task currently best solved with supervised methods (Navigli, 2009), which require labeled training data to train and fit machine learning classifiers. Semi-supervised knowledge-based approaches (Yarowsky, 1995; Mihalcea, 2004) have also been used to address the task of WSD, but most fail to achieve the performance accuracy of supervised methods. Unsupervised methods do not have access to labeled data and approach the task of WSD as a sense discrimination task by clustering all instances into groups with no predefined senses. Ensembles of unsupervised WSD algorithms are reported to outperform single unsupervised systems (Brody et al., 2006) but they cannot be directly compared to supervised and semi-supervised systems due to the difference in task definitions.

A SensEval/SemEval task[3] (Kilgarriff and Rosenzweig, 2000) was organized as an initiative to evaluate and compare WSD systems. A WSD system's performance is expected to lie within the percentages determined by its lower and upper bounds. The upper bound specifies the best performance a WSD can achieve and is usually calculated as the inter-annotator agreement. The lower bound is provided by a simple baseline selected either randomly or by marking all instances of a word with its most frequent sense (Gale et al., 1992). The latter has proved to be quite hard to beat: While experts manually annotating linguistic corpora for word senses achieve agreement of 80-90% for WSD (Palmer et al., 2007; Navigli et al., 2007), the most-frequent-sense baseline in the SemEval-2007 shared task achieved 78% (Agirre and Soroa, 2007). This indicates that WSD systems need to focus on infrequent senses to improve performance.

Reichart and Rappoport (2010) point out that according to the construction grammar framework (Goldberg, 1995), words, multi-word expressions, and syntactic forms are all valid constructions comprised of a form and a meaning. Thus, the WSD task can be generalized to the disambiguation of all linguistic constructions that have more than one meaning. Grammatical tenses are one example of syntactic forms: Tenses can have different grammatical meanings depending on the context in which they are used. For example, the present perfect form *has lived* can

[3] `www.senseval.org`

be used to express the following grammatical meanings (adapted from Murphy, 2012):

- Norbert *has lived* in three different cities. (Grammatical meaning: Finished action emphasizing the experience)
- Maria *has lived* here for the last 5 years. (Grammatical meaning: Duration of an action or a state)
- Alexander *has lived* a good life. (Grammatical meaning: Finished action emphasizing the result)

When it comes to disambiguation of the senses of syntactic forms such as grammatical tenses, experts have an agreement of 84.2%, while the most-frequent-sense baseline only achieves 46.7%, according to Reichart and Rappoport (2010). This implies that the distributions of the senses of grammatical tenses are less skewed than those of words. Consequently, different kinds of machine learning algorithms could be suitable for grammatical tenses. We discuss our findings from a tense sense disambiguation task and compare them to the work done by Reichart and Rappoport (2010) in Section 4.2.7.

The main CL applications where sense disambiguation can be of great benefit are machine translation and information retrieval. As the latter is particularly relevant to our work, it is discussed in more detail in the next section.

2.2.3 Information Retrieval

The research field of Information Retrieval (IR) addresses the problem of efficiently obtaining relevant information from a collection of resources. Formally, the problem of IR can be defined as follows:

$$\begin{aligned} V &= \{w_1, w_2, \dots, w_N\}; \\ q &= q_1, \dots, q_k | q_i \in V; \\ d_i &= d_{i1}, \dots, d_{imj} | d_{ij} \in V; \\ C &= \{d_1, \dots, d_M\}; \\ R(q) &\subseteq C \end{aligned} \tag{2.1}$$

where V is a language vocabulary, q is a query, d is a document, C is a collection of documents and $R(q)$ is a set of relevant documents given query q. The task of IR is to compute $R'(q)$, which is an approximation of $R(q)$. This can be achieved using two strategies, document selection or document re-ranking, the dangers of the former being low precision at high recall or retrieving an empty list of results. Web search is the most common instantiation of the IR strategy of document re-ranking. However, there are certain peculiarities when it comes to retrieving documents from the Web rather than from a preprocessed database. Given unstructured data and ambiguous queries, the retrieved documents do not match the query precisely but are rather relevant (or irrelevant) to it (van Rijsbergen, 1979).

Different types of models representing documents in the collection can be used to design an effective ranking function $f(q, d)$, many of which rely on features such as bag-of-words representation, term frequency, and document length. The ranking function assigns each document a weight based on either its probability of being relevant given a query (probabilistic models) or the similarity between the query and the document (algebraic models) and ranks the documents accordingly. While a comprehensive overview of IR algorithms is given in Grossman (2004), we focus on the most effective methods relevant to our research.

The simplest yet powerful IR model is *tf-idf*, which has also given rise to many more sophisticated heuristics. While $tf_{t,d}$ represents the frequency of a query term in a document, $idf_{t,D}$ filters out the terms that are extremely frequent in the whole collection and are barely informative. The *tf-idf* of a term is a product of the term frequency and inverse document frequency, each of which has a number of possible implementations. Thus, *tf-idf* increases with the number of occurrences of a query term within a document and with the rarity of the term in the collection, as the following instantiation of the *tf-idf* formula demonstrates:

$$tfidf(t, d, C) = (1 + log(tf_{t,d})) \times log_{10}(\frac{N_C}{df_t}) \tag{2.2}$$

where t is a query term, d is a document in a collection C of size N, $tf_{t,d}$ is the number of occurrences of t in d and df_t is the number of documents that contain t, or the document frequency of t.

The main limitation of Formula 2.2 is that it does not take document length into consideration. This is addressed in optimizations of the formula, such as *BM25* (Robertson and Walker, 1994) and pivoted length normalization (Singhal et al., 1996). The trick behind these formulae is a tunable parameter b introduced to control the amount of length normalization required for a particular retrieval purpose, as demonstrated in the most common instantiation of BM25:

$$BM25(q,d) = \sum_{t \in q \cap d} \frac{(k+1) \times tf_{t,d}}{tf_{t,d} + k \times (1 - b + b \times \frac{|d|}{avdl})} \times idf_t \tag{2.3}$$

where t is a query term, d is a document, $tf_{t,d}$ is the number of occurrences of t in d, $|d|$ is document length, $avdl$ is the average document length in the collection, idf_t is an instantiation of inverted document frequency, and k and b are free parameters.

Importantly, both algorithms allow for any document length unit (e.g., words, tokens, characters): the denominator in Formula 2.3 includes a ratio of the current document length to the average document length in the collection and is thus unit-independent. BM25 also controls for the upper-bound of tf, thus, avoiding dominance by one term in a document. In Section 5.3.3, we provide reasons for our use of an instantiation of BM25 as the ranking formula for our IR system *FLAIR*.

Evaluating IR systems usually requires a set of documents, a set of queries, and a set of relevance judgments. Manning et al. (2008) give an overview of common evaluation metrics, such as the F_1-measure, R-precision, and a ROC curve, most of which are derived from measures of precision and recall.

2.2.4 Automatic Question Generation

A typical text-based question generation (QG) system consists of three components: target selection (sentences and words), generation of questions and answers, and generation of distractors, which is applicable for a multiple choice answer format. Most work on *target selection* follows a top-down perspective on the text: First, a set of suitable sentences is selected based on different criteria (e.g., Pino et al., 2008; Pilán et al., 2013). Then the target words or linguistic forms are selected from within the set of suitable sentences (e.g., Becker et al., 2012). However, as our focus is on input enhancement for language learning, we pursue a bottom-up approach instead: Given one or more target linguistic forms (e.g., the passive voice or the present perfect tense), we automatically select all candidate sentences in a text containing the target forms, apply basic constraints to filter out unsuitable sentences (such as those containing unresolvable pronouns), and then generate questions for the remaining ones.

Once the target sentence has been selected, it can be used to *generate questions* addressing particular linguistic forms contained in the sentence. The QG methodology goes back to Wolfe (1976), who used simple pattern matching to generate questions from texts. As the researcher expected, advances in the field of NLP further improved the accuracy of this method by encoding meta-linguistic information in the patterns. Although similar to pattern matching, the method of using templates is approached slightly differently in the QG task. The templates are either created manually (Liu et al., 2010) or learned from a large amount of data (Curto et al., 2012). Most of the attempts at QG have also probably included some kind of transformation – syntax-based transformation rules (Heilman, 2011), transformations based on semantic labeling (Mannem et al., 2010; Chali and Hasan, 2012), or semantic transformations that make use of text representations (Yao and Zhang, 2010). To achieve a higher accuracy, researchers usually combine several methods, such as transformation rules and statistical ranking (Heilman and Smith, 2009). Finally, QG is not an exception to the wave of neural networks, with Du et al. (2017) recently developing an approach for the automatic generation of reading comprehension questions on that basis. All of the aforementioned QG systems either assess vocabulary or reading comprehension, which contrasts with the focus of our work on functionally supporting focus on form in language learning.

Distractor generation is a separate complex task that has received some attention in the QG community. It supports the provision of answer options in a multiple-choice setup by ensuring that the choice of distractors is closely tied to what is intended to be assessed by the question. Traditionally, distractors are selected among words that are semantically related to the correct answer (Mitkov et al., 2006; Araki et al., 2016). Brown et al. (2005) select the distractors among the most frequent words that have the same part of speech as the correct answer. Pino and Eskenazi (2009) use wrong answers provided by the users of their system to inform the distractor generation component. Given that our focus is not on the multiple-choice answer format, distractor generation is not discussed further in this thesis.

QG systems are commonly evaluated by humans, who rate the generated questions on a set of predefined scales. These scales can represent a range of aspects, such as grammaticality, semantic ambiguity, and relevance to the task. As annotating a large amount of data is a costly task, the evaluation of IR and QG systems as well as other linguistic and CL tasks is sometimes outsourced to many non-expert workers, which has proved to be comparable to expert annotation (Snow et al., 2008). The next section introduces crowdsourcing as a method for collecting, annotating, and evaluating linguistic data.

2.2.5 Crowdsourcing in CL

Viewing crowdsourcing as the process of outsourcing work to a large number of people, Howe (2008) distinguishes between four primary strategies of crowdsourcing: crowd wisdom, crowd creation, crowd funding, and crowd voting. Indeed, crowdsourcing comes in different forms: several research teams taking part in shared tasks, hundreds of people editing Wikipedia[4] entries, answering questions on Quora[5] or StackOverflow[6], and sending money to support promising start-ups on Kickstarter.[7]

Estellés-Arolas and González-Ladrón-De-Guevara (2012) provide an overview of crowdsourcing tasks and develop a definition of crowdsourcing as an online activity undertaken voluntarily by a heterogeneous group of contributors, who are rewarded for their contributions either monetarily or by means of social recognition and the development of their own skills.

Although crowdsourcing annotations have generally proven to be comparable to expert ones (Snow et al., 2008), Hsueh et al. (2009) raises concerns about outsourcing linguistic tasks to non-experts as they are not specifically trained for annotation and might not want to invest enough time and effort into producing high-quality data. The results of their study on sentiment analysis of political blogs suggest that crowd workers have lower inter-annotator agreement than experts. They show that the quality of labels can be improved by eliminating noisy annotators and ambiguous examples and point out that even noisy data from several crowd workers can still be successfully used to build statistical models.

Callison-Burch and Dredze (2010) provide an overview of language-related crowdsourcing tasks conducted as part of a shared task in which participants were given

[4] `www.wikipedia.org`

[5] `www.quora.com`

[6] `www.stackoverflow.com`

[7] `www.kickstarter.com`

$100 of seed money to analyze linguistic data. As a result, they used crowdsourcing to annotate textual and visual data, recognize textual entailment, and evaluate information extraction algorithms and machine translation outputs.

Munro et al. (2010) propose using crowdsourcing to collect linguistic and psycholinguistic data in tasks that go beyond the scope of annotation. They give an overview of seven tasks conducted via crowdsourcing and compare them to their lab counterparts. These tasks range from semantic judgments and cloze tasks to the audio segmentation of words in artificially constructed languages. Munro et al. (2010) report non-significant differences between crowd workers' and lab subjects' ratings and high correlations between the two experimental settings. They also used the collected judgments to fit mixed-effect logistic regression models for syntactic tasks. The most significant factor yielded by the model was in line with the results of other experimental models. They conclude that the quality of crowdsourcing linguistic tasks is comparable to that of controlled laboratory experiments.

Following the work of Parent and Eskenazi (2010), who utilized crowdsourcing for the task of word sense disambiguation, we rely on this method to annotate the senses of grammatical tenses and report the results in Section 4.2. In line with Heilman and Smith (2010), we also evaluate the quality of automatically generated questions via crowdsourcing and present the findings of two studies in Section 6.6.

2.3 Intelligent Computer-Assisted Language Learning

Intelligent computer-assisted language learning (iCALL) focuses on leveraging available NLP technology to create theoretically and pedagogically informed and efficient FLTL applications that foster learners' awareness of linguistic forms and categories and provide individual feedback (Amaral and Meurers, 2011). Advances in technology and the rise of Web 2.0 have not only proved to help enhance various aspects of learning (Ehrmann, 2002) but also have caused major changes in the way courses are developed and delivered. Technology no longer has to be used in an "ad hoc and disconnected fashion" (Warschauer and Healey, 1998) but is instead an integral part of the everyday learning process. Consequently, the role of the teacher has shifted from an instructor to a facilitator who must be aware of a variety of appropriate material for improving students' language skills.

Moreover, language learners may want or need to search for reading materials themselves. In his book *Education and Ecstasy*, Leonard (1968) observes that students prefer being in control of the learning process rather than remaining mere recipients of information. Lepper (1985) also concluded that control gives learners the opportunity to make choices and makes them feel more competent and intrinsically interested in the activity. Rather than being a passive recipient of information, a learner who uses technology becomes an active user.

One objective of iCALL is to allow the learner to interact with the system as independently as possible while acquiring linguistic knowledge and developing skills. However, iCALL systems can also target language educators and assist them in selecting appropriate teaching material for class. When it comes to finding texts at an appropriate level of language proficiency, readability measures come into play. These are designed to predict the grade level that a text corresponds to (e.g., elementary school, B2 level). The next section provides an overview of traditional and state-of-the-art approaches to automatic readability assessment.

2.3.1 Text Readability

According to the input hypothesis discussed in Section 2.1, reading at the appropriate level is crucial for language learning in general and developing grammatical accuracy in particular. Standard text readability measures rely on the predictive power of the morphological and syntactic structure of a language, the two most traditional features for computing readability being word length and sentence length (Kincaid et al., 1975).

Average syllable count has also been used as a measure of text readability. However, in the corpus used by Si and Callan (2001), web pages written for Grades 3 to 5 had more polysyllable words than those written for Grades 6 to 7. This indicates that the number of polysyllables does not necessarily increase with difficulty. Another counterargument is made by Bennöhr (2005), who remarks that longer words might look familiar to adult L2 learners due to their background knowledge and their mother tongue, differently from young L1 learners. By contrast, adult L2 learners may find short words with no similarity to other languages much harder to remember. Moreover, sentence length has been found to be a more reliable feature than syllable count (Uitdenbogerd, 2003).

Doing research at the intersection of language, computation and education, Vajjala and Meurers (2012) demonstrate that insights and findings from SLA research can improve the performance of readability scoring systems. Nevertheless, finding material that is appropriate in terms of both form and content is still seen as a challenge. The problem itself is two-sided as inappropriate content can either be uninteresting or not match the reader's level of language competence. With respect to text interestingness, most SLA approaches agree on the importance of discovering and making use of the learner's interests to increase their motivation. Thus, it is critical to find reading material that is both interesting for the learner and at their level of language competence. While readability-based systems take care of the latter, the former can be addressed simply by letting the reader search text collections themselves.

In another paper, Vajjala and Meurers (2013) analyze the top 100 documents for 50 queries retrieved from a standard web search engine to see if form-appropriate results can be found among the content-appropriate ones, which proved to be the case. It should also be noted that some readability measures do not always apply to web retrieval due to the ubiquity of malformed web pages. Sentence length, parse tree height, and other syntactic sentential features might fail on web pages containing little text or unstructured captions and links. Therefore, special attention should be paid to reducing the amount of boilerplate text in the corpora to a minimum.

Algorithms for calculating a text readability score differ in the number and the level of features used in the formulae. The processing cost of each algorithm is directly correlated with the complexity of its features. While the automated readability index (ARI) proposed by Smith and Senter (1967) only takes word and sentence length into consideration and can be calculated easily and fast, the sophisticated readability formula discussed in Vajjala and Meurers (2014) requires deeper parsing as it makes use of about 151 text complexity features in a supervised machine learning approach. ARI is calculated as follows:

$$ARI(d) = 4.71 \times (\frac{chars}{words}) + 0.5 \times (\frac{words}{sentences}) - 21.43 \tag{2.4}$$

where *chars* are the number of letters and digits; *words* are the number of dependencies as obtained from the parser, without punctuation;[8] and *sents* is the number of sentences as obtained from the sentence splitter. The final score is then rounded up to the next whole number. It falls within a range from 1 to 12 and roughly aligns with the corresponding US school grade level. Because of its low computational cost, ARI was our IR algorithm of choice for the current version of the *FLAIR* system presented in Chapter 3.

2.3.2 IR systems for FLTL

We understand *IR for FLTL* as a task involving searching a document collection to retrieve texts that (a) satisfy the learner's information need, (b) are linguistically appropriate given the learner's language proficiency level and (c) assist the teacher in achieving their current pedagogical goal. Much of the recent research has focused on satisfying the criterion of text appropriateness (Collins-Thompson et al., 2011; Vajjala and Meurers, 2014). Consequently, the algorithms used in state-of-the-art learner-oriented IR systems are mainly based on lexical properties and readability features, although some mention the integration of grammar modules as a goal for the future (Brown and Eskenazi, 2004; Ott and Meurers, 2011). Table 2.1 provides a comparison of three learner-oriented IR systems: *TextFinder* by Bennöhr (2005), *REAP* by Brown and Eskenazi (2004), and *LAWSE* by Ott and Meurers (2011). This comparative overview should set up the correct context for critically approaching the problem of IR for FLTL as well as our approach, which we compare to the aforementioned ones in Appendix A.

Educator-oriented IR systems, on the other hand, focus more on pedagogical intent and usually delegate full control over the reading material to the educator, which is justified for language test designers but might not be appropriate for teachers. For example, *SourceFinder* (Sheehan et al., 2007) is an authoring tool targeting test developers that supports the retrieval of suitable source material for developing reading comprehension passages. Its text acceptability model is built on linguistic constructs and other textual cues that are detected in order to estimate complexity features such as degrees of narrativity and argumentation. Among the IR systems mentioned in this section, LAWSE (Ott and Meurers, 2011) appears to be the closest in spirit to our work since it retrieves authentic documents from the Web

[8] Although the original ARI does contain punctuation.

rather than from a preprocessed text database. Its authors emphasize the need for a more language-informed retrieval to facilitate the search for texts containing particular grammatical constructions, which is one of the objectives of this thesis.

2.3.3 Systems Generating Questions and Activities for FLTL

This section introduces research on automatic question generation (QG) for FLTL. The most prominent approach to QG for FLTL, which is also the closest to our work, is taken by Heilman (2011). He utilizes the syntactic phrase structure of input sentences to transform them into factual wh- questions targeting noun phrases, prepositional phrases, and subordinate clauses. As this results in a large number of generated questions, he implements a ranking machine learning algorithm to prioritize the most well-formed questions. As future work, he emphasizes the need for a larger system providing both the IR and QG functionality as well as a grading interface for language teachers.

	Textfinder (Bennöhr, 2005)	***REAP*** (Brown and Eskenazi, 2004)	***LAWSE*** (Ott and Meurers, 2011)
Database	Offline database	Offline database	The Web
Source	Texts from online newspapers	Web materials	Web materials
Third-party tools	Lucene	AltaVista	Lucene
Main focus	Text complexity and user modeling	Lexical language modeling	Text complexity, lexical language modeling
Target users	English L2 adult learners	English learners, teachers, researchers (L1 and L2)	English L2 learners
Readability	Yes: word and sentence length + conjunctions (regression)	Yes: word histograms (machine learning)	Yes: readability measures + lexical frequency profiles
Learner model	Yes: writing sample	Yes: pre-defined ability levels	No
Grammar difficulty	Partially: conjunctions	No	No
Vocabulary difficulty	No	Yes: word lists	Yes: lexical frequency profiles
Reading interface	No	Yes: dictionary definitions	No
Evaluation	Teacher ranking, learner questionnaire	Empirical study, learner questionnaire	Against a corpus of graded texts
Stated future work	Readability formula optimization	Grammar difficulty, text cohesiveness	Syntactic features, grammatical constructions for visual input enhancement

TABLE 2.1: A comparison table of learner-oriented IR systems

While the REAP system (Brown and Eskenazi, 2004) was reviewed as an IR tool in the previous section, it can also generate certain types of questions (Brown et al., 2005). In particular, it generates cloze and definition exercises for vocabulary training. The main focus of the system is on the selection of appropriate target lexical items and distractors, such as synonyms, antonyms, hypernyms, and hyponyms, which are obtained from WordNet (Miller, 1995). The system features a learner model that represents learners' vocabulary knowledge and gets updated every time a learner completes an exercise.

Mostow et al. (2004) generate similar types of exercises for assessing reading comprehension and vocabulary knowledge. They conduct rigorous empirical research on their system with native English speakers and report that it could predict the participants' word identification, word comprehension, and passage comprehension scores on the Woodcock Reading Mastery Test (Woodcock et al., 1987) with a high reliability.

Language Muse^{SM} by Burstein et al. (2012) is another system that generates exercises of different forms targeting vocabulary, syntactic structure, and discourse relations. It is designed for content-area teachers and supports them in developing relevant language-based instructional scaffolding for their students. Language Muse^{SM} makes use of a large number of resources and manually-created lists to create a wide range of activities for FLTL. The reported resources include morphological and discourse analyzers, a distributional thesaurus, and paraphrase generation tools. The researchers conducted a teacher survey (Madnani et al., 2016) and a user study with K-12 English teachers (Burstein et al., 2013) to evaluate the effectiveness of the tool. The results show that teachers' linguistic awareness increases in the post-test measure.

It is important to note that while QG systems utilize state-of-the-art NLP resources to generate pedagogically and theoretically-motivated activities for FLTL, they do not necessarily analyze learners' answers to the automatically generated questions, with the exception of multiple choice questions, where the correct answer is presented along with several distractors. On the other hand, when a learner provides an answer that does not match the words in the text verbatim, it may be challenging to assess its correctness. Although some work has been done in the field of matching learner answers to target ones (Ziai et al., 2012), there is to the best of our knowledge no system that generates exercises and analyzes the answers at the same time. The FeedBook system (Rudzewitz et al., 2017) is taking a step in this direction, as it is able to automatically provide scaffolding linguistic feedback to learners on the basis of their answers. However, we do not compare this system to the other systems in this section because it does not automatically generate activities but rather digitizes those found in printed English workbooks.

To conclude, systems for generating questions for FLTL still require supervision and are currently mostly being developed for language teachers as the point of connection between materials and learners. Further development of NLP and statistical algorithms as well as the integration of a scaffolding feedback component into QG systems are needed to fully implement an end-to-end tutoring system for language learners.

2.3.4 Intelligent Tutoring Systems for FLTL

Intelligent tutoring systems (ITS) focus on providing individualized instruction to learners. They do not necessarily generate activities but adjust their presentation based on learners' psychological states, be it general knowledge, current comprehension of the material, or motivation.

A meta-analysis of the effects of ITS on learning outcomes by Ma et al. (2014) revealed that ITS are superior to teacher-led, large-group, and non-ITS computer-based instruction as well as working with textbooks or workbooks. However, no significant differences were found between ITS and learning from a human tutor or working in small groups. Tsiriga and Virvou (2004) examined language learners' attitudes to the individualized instruction and compared their learning outcomes when using an intelligent and a non-intelligent system. While it took the participants longer to familiarize themselves with the intelligent system, they appreciated the individualized support. Objectively, it also led them to take more exploratory, rather than linear, learning paths and resulted in higher learning outcomes.

While ITS and iCALL research has mostly focused on the interaction between the learner and the system, the development of teacher interfaces, such as ITS authoring tools (Ainsworth and Grimshaw, 2004) or teacher professional development tools (Burstein et al., 2012), allows researchers to look into the ways teachers create learning environments and to evaluate their effectiveness. We contribute to this line of research by developing the iCALL system *FLAIR* introduced in the next chapter and conducting an online study with English teachers presented in Section 5.4. The results provide insights into English teachers' preferences when selecting reading material for their students.

Chapter 3

The *FLAIR* Approach

Parts of the work discussed in this chapter appeared in the following peer-reviewed publications and theses:

1. Chinkina, M., & Meurers, D. (2017). Question Generation for Language Learning: From ensuring texts are read to supporting learning. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications.* Copenhagen, Denmark, pages 334-344.
2. Chinkina, M., Kannan, M., & Meurers, D. (2016). Online information retrieval for language learning. *ACL 2016: System demonstrations.* Berlin, Germany, pages 7-12.
3. Chinkina, M., & Meurers, D. (2016). Linguistically Aware Information Retrieval: Providing Input Enrichment for Second Language Learners. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications.* San Diego, CA, pages 188–198.
4. Chinkina, M. (2015). *Form-focused Language-aware Information Retrieval* (Master's thesis, Eberhard Karls Universität Tübingen).

While second language acquisition (SLA) research advances our understanding of effective methodology and strategies for language teaching and learning, the field of computational linguistics (CL) can support this endeavor by automating some SLA methods, such as input enrichment, input enhancement, and processing instruction. This motivated us to develop our own approach, which was practically implemented in the *FLAIR* system (`www.purl.org/icall/flair`) that we present, discuss, and evaluate in this thesis.

3.1 Motivation and Need for ***FLAIR***

Section 2.1 introduced input enrichment, input enhancement, and processing instruction as for SLA. All of these methods rely on the existence and availability of comprehensible input for language learners. Apart from the curated texts found in textbooks, language teachers often turn to the Web for additional authentic materials. While it is the largest freely available text database in the world, the Web has the disadvantage of being unstructured and thus difficult to search. In fact, the larger the database, the more time-consuming a search for appropriate reading material gets – both for the user and for the machine. Consequently, the development of efficient web search tools has been a major concern in the field of computer science in the last couple of decades. Sophisticated ranking algorithms in the leading search engines, such as Google[1] and Yahoo![2], include an estimated up to 200 features[3] that influence the final ranking of the retrieved results, all of which work to ensure that the user's information need is satisfied as fast and with as few clicks as possible.

At the core of any web search engine lies vocabulary retrieval: One obtains an appropriate text containing target lexical items by including them in a search query. Grammar retrieval, on the other hand, requires an extension of web search as the user is unlikely to find appropriate texts by simply searching for *texts containing regular and irregular verbs*. In fact, the top search results may contain very few to no occurrences of the linguistic form of interest. The heat map at the top of Figure 3.1 demonstrates that although regular and irregular verbs are highly frequent, they are not evenly represented across the top 60 search results retrieved by Microsoft Bing.[4] The heat map at the bottom of Figure 3.1 shows the results following automatic input enrichment: a reordered list of the search results retrieved from Bing with those containing the best representation of both regular and irregular verbs closer to the top (i.e., to the left in the figure). This approach allows for the retrieval of texts that address content of interest to the learners and at the same time are rich in terms of the linguistic forms to be taught. This is the basis of our information retrieval system *FLAIR* (Chinkina and Meurers, 2016), which supports the retrieval of documents containing 87 linguistic

[1] `www.google.com`

[2] `www.yahoo.com`

[3] `http://backlinko.com/google-ranking-factors`

[4] `www.bing.com`

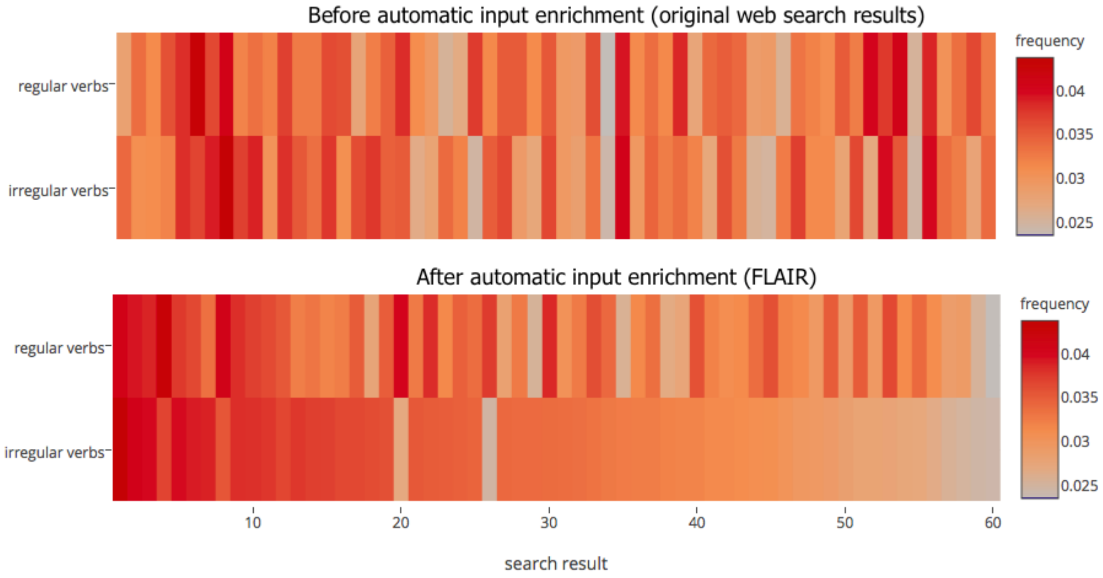


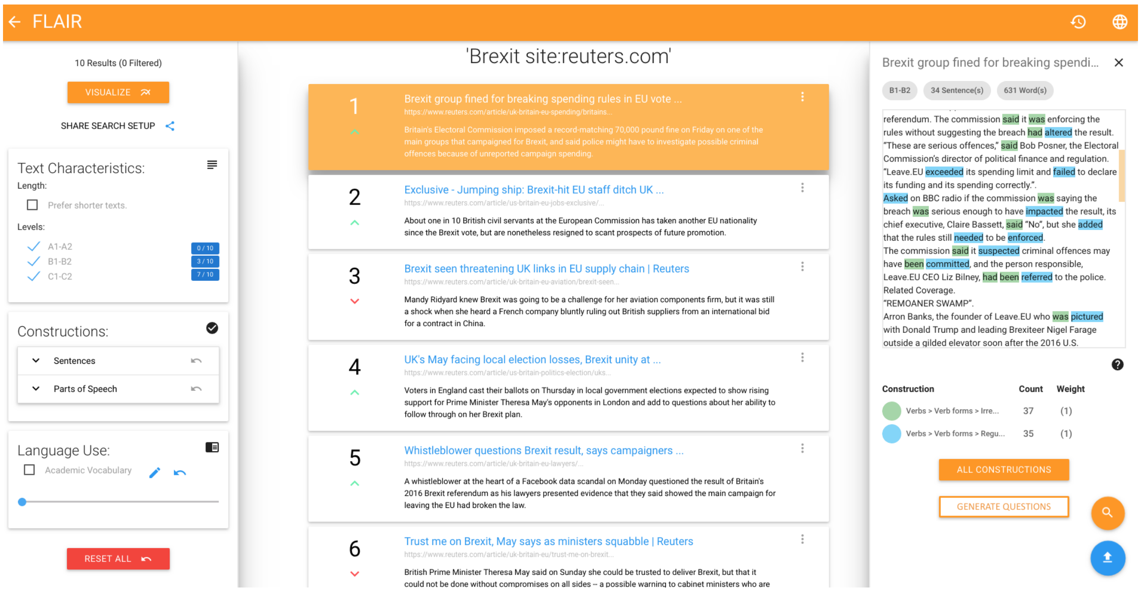
Figure 3.1: Comparison of the top results retrieved by a standard web search engine before and after automatic input enrichment. The 60 search results are plotted along the X axis, and the two target linguistic forms, regular and irregular verbs, are plotted on the Y axis.

forms specified in the official English language curriculum for schools in Baden-Württemberg, Germany, such as different verb forms, grammatical tenses, and conditionals. Concretely, it re-ranks texts in a document collection based on the frequency of the selected linguistic forms.

3.2 FLAIR Interface and Functionality

Figure 3.2 demonstrates the general layout of *FLAIR*. It consists of three elements: a settings panel on the left, a results field in the middle, and a text panel on the right. A ***search field*** opens when the user clicks on a magnifying glass icon in the bottom-right corner. The search language and number of results to be retrieved can be specified right away. In another scenario, the user can click on the upload icon next to the search icon and is prompted to upload their own collection of texts in English or German. Both scenarios result in a specified number of texts displayed in the original order in the ***results field***.

In the ***settings panel*** on the left, the user can select linguistic forms and adjust their weights using a slider to retrieve the documents containing the most optimal linguistic representations. *FLAIR* currently can detect 87 linguistic forms, which are grouped into sentence-level and word-level forms. The first group includes different types of questions, clauses, and sentences, while the second group lists the remaining linguistic forms by part of speech. This functionality is central to *FLAIR* and requires the use of a number of natural language processing (NLP) resources and algorithms. We discuss the detection of linguistic forms and associated challenges in Section 4.1.



10 Results (0 Filtered)

Length:
□ Prefer shorter texts.

Levels:
✓ A1-A2 0 / 10
✓ B1-B2 3 / 10
✓ C1-C2 7 / 10

□ Academic Vocabulary

'Brexit site:reuters.com'

1 Brexit group fined for breaking spending rules in EU vote ...
https://www.reuters.com/article/uk-britain-eu-spending/britains...
Britain's Electoral Commission imposed a record-matching 70,000 pound fine on Friday on one of the main groups that campaigned for Brexit, and said police might have to investigate possible criminal offences because of unreported campaign spending.

2 Exclusive - Jumping ship: Brexit-hit EU staff ditch UK ...
https://www.reuters.com/article/us-britain-eu-jobs-exclusive/...
About one in 10 British civil servants at the European Commission has taken another EU nationality since the Brexit vote, but are nonetheless resigned to scant prospects of future promotion.

3 Brexit seen threatening UK links in EU supply chain | Reuters
https://www.reuters.com/article/uk-britain-eu-aviation/brexit-seen...
Mandy Ridyard knew Brexit was going to be a challenge for her aviation components firm, but it was still a shock when she heard a French company bluntly ruling out British suppliers from an international bid for a contract in China.

4 UK's May facing local election losses, Brexit unity at ...
https://www.reuters.com/article/us-britain-politics-election/uks...
Voters in England cast their ballots on Thursday in local government elections expected to show rising support for Prime Minister Theresa May's opponents in London and add to questions about her ability to follow through on her Brexit plan.

5 Whistleblower questions Brexit result, says campaigners ...
https://www.reuters.com/article/uk-britain-eu-lawyers/...
A whistleblower at the heart of a Facebook data scandal on Monday questioned the result of Britain's 2016 Brexit referendum as his lawyers presented evidence that they said showed the main campaign for leaving the EU had broken the law.

6 Trust me on Brexit, May says as ministers squabble | Reuters
https://www.reuters.com/article/uk-britain-eu/trust-me-on-brexit...
British Prime Minister Theresa May said on Sunday she could be trusted to deliver Brexit, but that it could not be done without compromises on all sides -- a possible warning to cabinet ministers who are

Brexit group fined for breaking spendi...

B1-B2 34 Sentence(s) 631 Word(s)

referendum. The commission said it was enforcing the rules without suggesting the breach had altered the result.
"These are serious offences," said Bob Posner, the Electoral Commission's director of political finance and regulation.
"Leave.EU exceeded its spending limit and failed to declare its funding and its spending correctly.".
Asked on BBC radio if the commission was saying the breach was serious enough to have impacted the result, its chief executive, Claire Bassett, said "No", but she added that the rules still needed to be enforced.
The commission said it suspected criminal offences may have been committed, and the person responsible, Leave.EU CEO Liz Bilney, had been referred to the police.
Related Coverage.
"REMOANER SWAMP".
Arron Banks, the founder of Leave.EU who was pictured with Donald Trump and leading Brexiteer Nigel Farage outside a gilded elevator soon after the 2016 U.S.

Construction	Count	Weight
🟢 Verbs > Verb forms > Irre...	37	(1)
🔵 Verbs > Verb forms > Regu...	35	(1)

Figure 3.2: FLAIR interface.

In addition, the panel includes settings for preferred text length and readability level. The latter provides the information about the number of texts corresponding to each of the three CEFR levels: A1-A2, B1-B2, and C1-C2. The configuration does not change for consecutive searches within a session and can be shared with others via a unique link by clicking on *Share Search Setup*. In the most common scenario, the teacher would adjust the settings and share them with their students, who will in turn use the system as a search engine with pre-configured settings.

Based on the feedback from English teachers, we also implemented a language-use component. This contains an *academic vocabulary search* that uses the Academic Word List (Coxhead, 2000) to estimate the register of documents on-the-fly and re-rank them accordingly. In addition, the user has the option of searching for and highlighting the occurrences of words from *manually created vocabulary lists*.

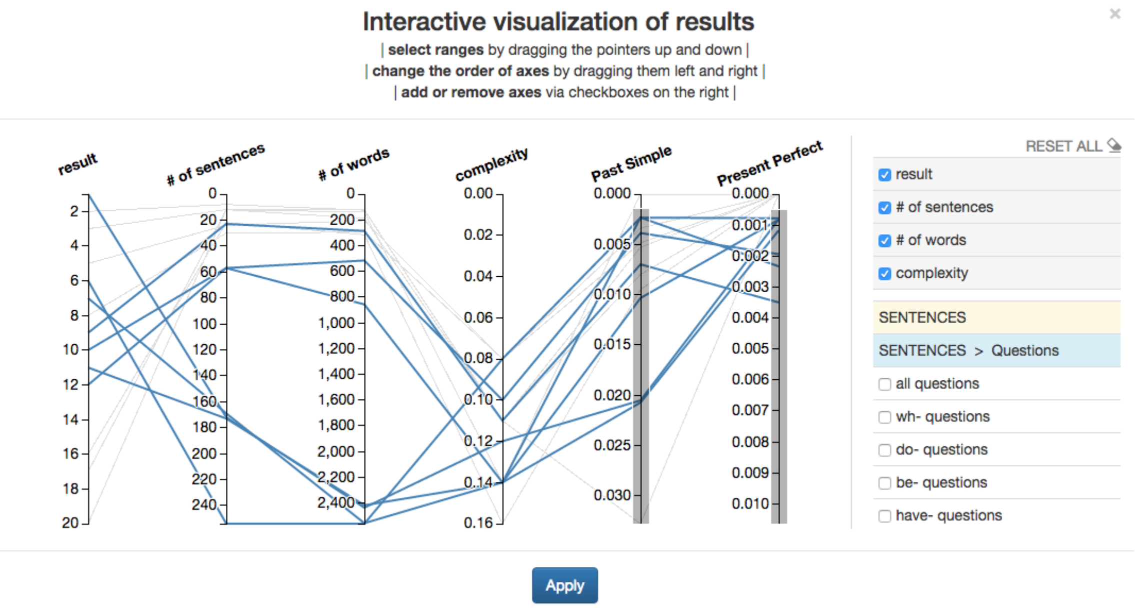
When the user clicks on a search result, the text of the corresponding web page is shown in the ***text panel*** on the right with the occurrences of the selected constructions highlighted. Information about the text readability level, the approximate number of sentences and words in the text, and the frequencies of all linguistic forms found in it are also presented. Finally, there are two *FLAIR* components that we describe further in this section: the visualization component (*Visualize* in the settings panel; Section 3.2.1) and the question-generation component (*Generate Questions* in the text panel; Section 3.3).

3.2.1 Interactive visualization

On the basis of the feedback from language test designers at the development stage, we enhanced the system by adding a visual component that allows for a stricter selection of documents and delegates more control over different parameters to the user. This element makes it possible to inspect and further select documents on the basis of the multi-faceted nature of the retrieved documents.

The interface illustrated in Figure 3.3 is based on the visualization technique of parallel coordinates used for visualizing multivariate data. Vertical axes represent parameters, such as any linguistic forms selected by the user, the number of sentences, the number of words, and a global readability score. Each polyline stands for one document and records its linguistic characteristics by going through different points on the parameter axes. The interface supports mouse interaction, allowing the user to restrict the range of values permitted for particular parameters, with other documents becoming grayed out in the interface and removed from the search results. In the figure, only documents with a non-zero frequency for both *past simple* and *present perfect* are selected. The numbers on the vertical axes for the grammatical constructions correspond to their relative frequencies in the documents. Once the *Apply* button is clicked, the search result list is restricted to those documents satisfying the constraints specified in the visualization module.

This visualization makes it possible to get an overview of the distribution of linguistic characteristics in the set of documents to be re-ranked. The interface also supports interaction with the visualization, providing fine-grained control over a user-selected set of linguistic characteristics. Users can select a range of values for one or more constructions to precisely identify and retrieve documents.



Interactive visualization of results

| **select ranges** by dragging the pointers up and down |
| **change the order of axes** by dragging them left and right |
| **add or remove axes** via checkboxes on the right |

RESET ALL

☑ result
☑ # of sentences
☑ # of words
☑ complexity

SENTENCES

SENTENCES > Questions

□ all questions
□ wh- questions
□ do- questions
□ be- questions
□ have- questions

FIGURE 3.3: FLAIR visualization component. Documents with a non-zero frequency for both *past simple* and *present perfect* are selected.

3.3 Question Generation Component

As mentioned in Section 2.2.4, most of the work on QG has dealt with vocabulary (Brown et al., 2005) and comprehension questions (Mostow et al., 2004) rather than grammar. Among the approaches to automatically generate exercises that facilitate grammar acquisition and practice, cloze sentences are the most ubiquitous type. These are generated by substituting the target linguistic form with a gap: the challenge usually lies in the selection of appropriate sentences and gaps (Becker et al., 2012; Niraula and Rus, 2015):

(5) The advisory group had ________ a list of all the different territorial arrangements in the EU. (draw up)

Meta-linguistic questions, which are designed to test learners' explicit knowledge of the language system, have not received much attention in the CL community. This is because they require the use of a limited number of templates and only a minimal amount of NLP. For example, in order to generate the question *From which verb is the noun 'generation' derived?*, one would only need a POS tagger and the WordNet database (Miller, 1995). Teachers' frequent use of meta-linguistic questions is also widely criticized by educators and researchers alike, mainly because they do not serve a communicative goal. In our work, we combine cloze sentences with open-ended wh- questions to leverage their advantages and cancel out the drawbacks.

The questions that we generate are text-based, and the target linguistic forms come from the source text out of which they are detected. Therefore, the goal of asking questions in our case is to foster learners' processing of the target linguistic forms, in line with VanPatten's (2002) work on structured input activities. The two main types of questions that we generate are form-exposure and grammar-concept questions. *Form-exposure questions*, such as (6a), have the form of a wh- question followed by a cloze sentence, or a gap sentence, and are similar to local comprehension questions. They are generated by transforming the original declarative sentence into an interrogative one. The current version of our system can generate form-exposure questions for subjects, objects, and predicates.

(6) Indeed, Semel and the media executives he brought in by all accounts turned a scrappy young internet startup into a highly profitable company.

a. *Form-exposure question*: Who turned a scrappy young internet startup into a highly profitable company? Semel and the media executives he _______.

Grammar-concept questions, such as (7a), draw learners' attention to the semantics of the target linguistic form and encourage them to rely on it to get to the meaning of the sentence. Grammar-concept questions can currently target grammatical tenses (present perfect, past simple, etc.) and are template-based. We discuss the generation of form-exposure and grammar-concept questions further in Chapter 6.

(7) Chinese retailers have cut staff.

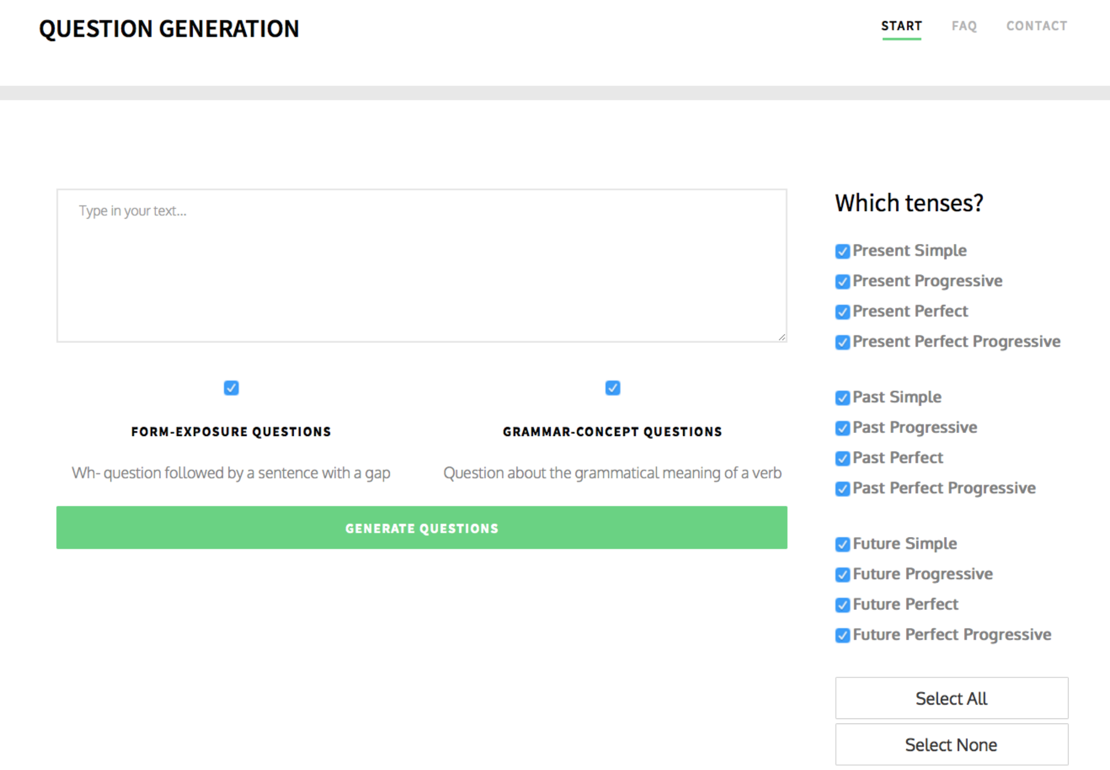
a. *Grammar-concept question*: Are Chinese retailers still cutting staff?

The current implementation of the QG component in *FLAIR* allows the user to generate questions about the grammatical tenses previously specified in the settings. The overall procedure is as follows:

1. Type in a query or upload a collection of texts.
2. Obtain a list of search results.

3. Specify one or more linguistic forms including grammatical tenses.
4. Obtain a re-ranked list of search results.
5. Select a text by clicking on it.
6. In the right-hand panel, click on *Generate Questions*.
7. Obtain the questions targeting the selected grammatical tenses.

We have developed a prototype of *FLAIR* incorporating the QG component and provide a simple system-independent interface for trying out the QG tool (`www.purl.org/qg`). The user can type in a text and generate form-exposure and grammar-concept questions for any grammatical tense automatically detected in the text. Figure 3.4 demonstrates the functionality of the demo interface.



☑ FORM-EXPOSURE QUESTIONS

Wh- question followed by a sentence with a gap

☑ GRAMMAR-CONCEPT QUESTIONS

Question about the grammatical meaning of a verb

Which tenses?

☑ Present Simple
☑ Present Progressive
☑ Present Perfect
☑ Present Perfect Progressive

☑ Past Simple
☑ Past Progressive
☑ Past Perfect
☑ Past Perfect Progressive

☑ Future Simple
☑ Future Progressive
☑ Future Perfect
☑ Future Perfect Progressive

Figure 3.4: A standalone application demonstrating the functionality of our question generation tool.

3.4 Technical Implementation and Profiling

FLAIR is written in Java and implemented as a Java EE web application. The core architecture is based on a client-server implementation that uses WebSocket (Fette and Melnikov, 2011) and Ajax (Garrett et al., 2005) technologies for full-duplex, responsive communication. All server operations are performed in parallel, and each operation is divided into subtasks that are executed asynchronously. Operations initiated by the client are dispatched as asynchronous messages to the server. The client then waits for a JSON[5] response from the server. By using WebSockets to implement the server endpoint, we were able to reduce most of the overhead associated with HTTP responses.

The sequence of operations performed within the *client* boundary is described as follows:

1. Send search query to server and initiate web search
2. Wait for completion signal from server
3. Initiate text parsing
4. Wait for completion signal from server
5. Request parsed data from server
6. Cache parsed data
7. Re-rank results according to parameters

The sequence of operations performed within the *server* boundary is described as follows:

1. Receive search query from client
2. Begin web search operation:
    (a) Fetch top N valid search results
    (b) For each search result, fetch page text
    (c) Signal completion

[5] `http://json.org`

3. Wait for request from client
4. Begin text parsing operation:
    (a) For each valid search result, parse text and collate data
    (b) Signal completion
5. Wait for request from client
6. Send parsed data to client

Parallelization of the tool allowed us to reduce the overall processing time. However, due to the highly parallel nature of the system, its performance is largely dependent on the hardware on which it is deployed. Amongst all the different operations performed by the pipeline, web crawling and text annotation prove to be the most time-consuming and resource-intensive tasks. We conducted several searches and calculated the relative time each operation took. Fetching the results and extracting the documents (from entering the query till displaying a list of results) took around 50-65% of the total time and parsing them took around 20-30% of the total time. The *FLAIR* algorithm for detecting linguistic forms builds upon the results of the Stanford shift-reduce constituency parser while adding negligible overhead. The technical evaluation of the *FLAIR* algorithm for detecting linguistic forms is presented in Section 4.1.1.

As for the effectiveness of the tool in a real-life setting, full user studies with language teachers and learners are necessary for a proper evaluation of the distinctive components of *FLAIR*. We took first steps in this direction by conducting an online study with English teachers on automatic input enrichment, reported in Section 5.4, and two crowdsourcing studies on automatic question generation, reported in Sections 6.7 and 6.8.

3.5 Use cases for *FLAIR*

First and foremost, *FLAIR* expands the scope of empirical studies that can test the effects of SLA phenomena such as input enrichment, input enhancement, and processing instruction. In Section 5.4, we present one such study conducted with English teachers. One of its objectives was to see whether teachers saw a need for automatic input enrichment systems and preferred the results provided by them over those retrieved from a standard web search engine, which proved to be the case.

In terms of envisaged use cases, in the most straightforward case, *FLAIR* helps teachers identify appropriate reading materials for a class or individual students in terms of form, content, and reading level. The system can also feed into platforms that provide input enhancement, such as *WERTi* by Meurers et al. (2010), or generate exercises from text, such as *Language Muse*^{SM} by Burstein et al. (2012), ensuring that the form targeted in enhancement or exercise generation is as richly represented as possible given the text base used. Finally, the QG component of *FLAIR* assists teachers in automatically creating questions of different types to facilitate processing the text as a whole and the linguistic forms in it.

In scenarios placing more value on learner autonomy or data-driven learning, *FLAIR* makes it possible to divide up the specification of the form and content criteria between the teacher and the learner: The teacher uses their pedagogical background in foreign language teaching and learning and their knowledge of the learner's abilities to configure *FLAIR* in a way that prioritizes the texts that best satisfy these form specifications. Using the teacher-configured *FLAIR*, the learner then takes control and enters search queries in line with their personal interests or information needs. The outcome is a collection of documents retrieved on the basis of the learner's search query, with the results ranked according to the pedagogical language learning needs defined by the teacher. One potential scenario includes the teacher obtaining information about the texts being read and using the system to automatically generate questions about these specific texts. The teacher can then select questions that are in line with their pedagogical goals and present them to the learner. In another potential scenario requiring a learner model and 100% accuracy of the QG tool, the process of selecting the target linguistic forms, retrieving the appropriate texts, and generating questions is fully automated, and is supervised but not fully controlled by the teacher. As our ultimate goal is expanding *FLAIR* into an intelligent tutoring system for students supplemented with an authoring tool for teachers, we present our high-level ideas about the functionality of such a system in the following section.

3.6 Towards the Development of an Intelligent Tutoring System for FLTL

As intelligent tutoring systems (ITS) provide individualized instruction to learners (see Section 2.3.4 for background information about ITS), a learner model is the key component implemented in most state-of-the-art ITS. Ideally, the model should store and process information about the learner's current psychological state in addition to their linguistic knowledge and reading comprehension. It gets updated after each interaction the learner has with *FLAIR*, that is, after each text read and assignment completed. To improve the efficiency and accuracy of the learner model, learners are also asked for explicit feedback about the texts, exercises, and difficult vocabulary and grammatical constructions they encounter. The flow of *FLAIR* as an ITS can be envisaged as follows:

1. The teacher provides configured settings either individually or to all students.
2. A student searches for a topic of interest.
3. The system re-ranks the texts based on:
    - the teacher's configuration and
    - the student's model.

    If the student has difficulties with a certain linguistic form, this is automatically selected by the system. If the student is tired, shorter texts will be automatically prioritized.
4. The student reads the text (with or without input enhancement of the target linguistic forms).
5. The teacher gets information about the text that each student has read and uses the system to automatically generate (and potentially manually select) questions and exercises.
6. The learner gets the exercises and completes them with the support of scaffolding feedback.
7. The teacher gets the report about the texts read and exercises completed by each student.

8. The system updates each student's model, with the results grouped into several classes: general reading comprehension, vocabulary skills, grammar knowledge, and potentially other more specific classes for every grammatical construction.

Before *FLAIR* can be further developed into an ITS, its performance, usability, and general effectiveness need to be evaluated. The next three chapters address this issue by presenting separate components of *FLAIR* and evaluating them both technically and empirically. Chapter 4 deals with developing NLP technology relevant for FLTL, such as the detection of linguistic forms and their different uses in a variety of contexts.

Chapter 4

Leveraging NLP Technology

Parts of the work discussed in this chapter appeared in the following peer-reviewed publications and theses:

1. Chinkina, M., & Meurers, D. (2016). Linguistically Aware Information Retrieval: Providing Input Enrichment for Second Language Learners. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications.* San Diego, CA.
2. Chinkina, M. (2015). *Form-focused Language-aware Information Retrieval* (Master's thesis, Eberhard Karls Universität Tübingen).

The linguistic constructions relevant for language teachers and learners are somewhat different from traditional computational linguistic (CL) forms: while computational linguists prefer to deal with noun and prepositional phrases, these are not as richly represented in English textbooks. At the same time, linguistic forms such as conditionals and grammatical tenses are of interest for language teachers because their instantiation differs across languages, causing difficulties for learners. In this chapter, we discuss the methods used to approximate and automatically detect the linguistic forms relevant for foreign language teaching and learning (FLTL) and distinguish their different contextual interpretations. Given an input sentence, we run it through the Stanford CoreNLP pipeline (Manning et al., 2014) and obtain the part of speech (POS) and the lemma of every token in the sentence, a syntax tree with constituency information, a semantic graph with dependency information, resolved coreferences, annotated named entities, and the sentiment of the sentence. This information is then used in rule-based, machine learning, and hybrid algorithms. The first part of this chapter discusses the rule-based detection of a list of 87 linguistic forms specified in the official curriculum for the English language in the state of Baden-Württemberg, Germany. The second part of the chapter focuses on one of the most challenging aspects of the English grammar for language learners, grammatical tenses, and explores the detection of their different meanings using machine learning and hybrid approaches.

4.1 Detection of Linguistic Forms

NLP makes use of different approaches, from string matching to sophisticated grammar formalisms and machine learning. While string matching works fine for some structures (e.g., articles, prepositions), the detection of other constructions requires deeper syntactic analysis, going well beyond the word level. In the case of *pronouns*, for example, differentiating between cases and forms, i.e., retrieving subjective, objective, reflexive as well as possessive pronouns, requires that a simple look-up be supplemented with dependency parsing in order to distinguish the subjective from the objective *you* as well as the objective from the possessive *her*.

When dealing with the construction *going to*, one needs to be able to distinguish it from the identical form of the verb *to go* in the progressive aspect. Thus, it is necessary to identify the part of speech of the word following this construction in addition to a regular expression matching the *going to* pattern. The colloquial *gonna* is easier to detect in this case and does not require any additional part-of-speech tagging. It is worth pointing out, though, that due to the frequent usage of such colloquial structures in web texts (e.g., *gonna do*, *wanna go*), one needs to decide whether the algorithm should check for the POS tag *TO* (and annotate *na go* as a to-infinitive) or for the word *to* (and not license these constructions at all) preceding an infinitive. This choice mainly depends on the application, and, keeping our end user (the language learner) in mind, we decided to treat such colloquial uses as to-infinitives in order to expose the learners to real-life language rather than let them gloss over unfamiliar structures.

The algorithm for detecting the *used to* construction referring to a habitual action in the past takes this one step further. After making sure that the following word is a to-infinitive, and thus excluding the option of licensing the constructions *to be used to doing* and *to get used to doing*, one is still left with an ambiguous structure that can be either interpreted as the target construction, as in (8a), or as a passive structure, as in (8b). Such ambiguity can then be solved by checking the POS tag of the verb *used* or the presence of an auxiliary *to be* preceding *used.*

(8) a. Eyal *used to come* here every day.
b. It is *used to build* rockets.

The identification of *conditional sentences* poses another challenge. A conditional sentence contains two clauses, the conditional clause and the consequent clause, which are dependent on each other. A large majority of conditional clauses are introduced by the subordinating conjunction *if* or *unless.* School grammar books (e.g., Murphy, 2012) divide conditionals into:

- Zero Conditional (as in *"If you heat ice, it melts."*)
- First Conditional (as in *"Unless it rains, Karo will go jogging."*)
- Second Conditional (as in *"If Cansu had more time, she would write one more paper."*)
- Third Conditional (as in *"If Anne had known about it, she would have told Molly."*)

Narayanan et al. (2009) proposed a POS-based approach to identifying conditional types for the task of sentiment analysis. It mapped sequences of POS tags to tenses (VBD + VBN = Past Perfect) as well as conditional types (If + Past perfect, MD + Present Perfect = Third Conditional). However, two different types of conditionals can be mixed in the same sentence, producing the fifth type not covered by this taxonomy, Mixed Conditional. It is ubiquitously used in everyday speech and can be easily found in web texts, especially in interviews or transcripts. Consider the following real example sentence taken from the Web:

(9) I had a Yorkshire Terrier, and if I **was** a rat, it definitely **would have eaten** me.

Puente and Olivas (2008) proposed a more granular classification of conditional sentences and an algorithm for detecting them but pointed out that authentic texts containing conditionals pose a challenge since some retrieved sentences do not conform to their taxonomy. This was exactly our concern when constructing the *FLAIR* parsing module. For this reason, we use a constraint grammar to differentiate between *real* (Type 0 and 1) and *unreal* (Type 2 and 3) conditionals. In a class setting, it can be beneficial for learners to be exposed to the full variety of real-life conditional usages rather than stilted textbook examples.

Conditionals make use of tense-like grammatical forms of verbs to describe hypothetical situations or implications. However, these forms do not express the time reference like tenses in indicative sentences do: *if they were more patient* refers to the present, not the past, and *unless you eat your lunch* makes an assumption about the future. As conditionals are studied after most of the tenses, they are likely to be misinterpreted by the learner if accidentally found in a text. That is, we argue that if learners with no knowledge of conditionals search for the past simple tense, they should not obtain a text containing a conditional clause, such as *if they were more patient*. Therefore, the detection of *tenses* is conducted after the detection of conditionals in *FLAIR*, and the learner is given an option to deprioritize a certain linguistic form (conditionals, in this case) to ensure that the retrieved texts do not contain it. A complete list of the implemented grammatical constructions with the corresponding detection methods can be found in Appendix B.

4.1.1 Evaluation and Error Analysis

Linguistic target	**Precision**	**Recall**	**F$_1$**
Yes/no questions	1.00	1.00	1.00
Irregular verbs	1.00	0.96	0.98
used to	0.83	1.00	0.91
Phrasal verbs	1.00	0.61	0.76
Tenses (Present Simple, ...)	0.95	0.84	0.88
Conditionals (real, unreal)	0.65	0.83	0.73
Mean (81 targets)	0.94	0.90	0.91
Median (81 targets)	1.00	0.97	0.95

TABLE 4.1: Performance of the *FLAIR* detection algorithm.

Before evaluating the identification of the target linguistic forms, we inspected the performance of the Stanford shift-reduce parser for the constructions our patterns depend on. Among the biggest challenges were *gerunds*, which were mistakenly annotated as nouns (*NN*). *Phrasal verbs*, such as *settle in*, also appeared to be problematic for the parser and were sometimes not presented as a single entity in the list of dependencies.

To evaluate the performance of linguistic form detection on the basis of the parsed output, we used a corpus of news articles as a common type of data analyzed by *FLAIR*. We submitted three search queries and saved the top three results for each, obtaining nine news articles with an average length of 28 sentences. Table 4.1 shows the precision, recall, and F-measure for selected linguistic constructions identified by *FLAIR* and the medians and means across the 81 constructions, the details of which are included in Appendix B.

As the numbers show, some constructions are easily detectable (yes/no questions) while others are less reliably identified by the parser (phrasal verbs). There are different reasons for lower performance: the ambiguity of the construction (real conditionals) and problems of the Stanford Parser (*-ing* verb forms) discussed above as well as erroneous output from the text extractor module and some limitations to the *FLAIR* patterns used for identification (unreal conditionals). *Conditionals* were identified with an average F_1 score of 0.73, partially due to the difficulty of their disambiguation, as discussed in Section 4.1.3, and partially because of a particular choice we made: In order to avoid exposing learners to an unknown grammatical construction, we disambiguated all unclear cases of conditionals as the one appearing later in the curriculum, *unreal conditionals* (Grade 8). This way, any potential instances of this construction in texts at a lower level can be avoided (e.g., in Grade 6, when *real conditionals* are introduced).

4.1.2 Frequency Distribution of Linguistic Forms

The algorithm for detecting grammatical constructions allowed us to analyze the distribution of the constructions in the top results retrieved by Bing. We searched Bing to retrieve the top 60 documents for each of 40 queries and processed each of the 2,400 documents using the *FLAIR* algorithm. We could then divide the detected linguistic forms into five frequency groups, presented in Table 4.2: extremely frequent (91-100%), relatively frequent (71-90%), of average frequency (31-70%), relatively infrequent (11-30%) and extremely infrequent (0-10%).

Extremely frequent (91-100%)

simple aspect	positive d. of adverbs	advanced prepositions	plural regular nouns	pronouns
present simple	long auxiliary verb forms	*to* infinitive	irregular verbs	pronouns as subjects
regular verbs	simple prepositions	articles *a*, *the*	copular verbs	relative clauses
prepositions	*-ing* verb forms	present time	simple conjunctions	positive d. of adjectives
direct object	compound sentences	complex sentences	auxiliaries *be*, *do*, *have*	
past time	possessive pronouns	past simple tense	subordinate clauses	

Relatively frequent (71-90%)

modals	reduced relative clauses	pronouns as objects	adverbial clauses	negation
negation *not*	passive voice	advanced conjunctions	simple sentences	phrasal verbs

Of average frequency (31-70%)

perfect aspect	indefinite article *an*	*-ing* noun forms	short aux. verb forms	future time
present perfect	plural irregular nouns	progressive aspect	conditionals	future simple tense
can, *could*	negation *n't*	superl. d. of short adj.	direct questions	compar. d. of short adj.
present progr.	complex prepositions	compar. d. of short adv.	existential *there*	determiner *some*

Relatively infrequent (11-30%)

past progr.	reflexive pronouns	determiner *any*	real conditionals	*may*, *might*
there is / *there are*	past perfect tense	*do*, *be* questions	superl. d. of short adv.	*there was* / *there were*
determiner *many*	indirect object	*must*, *have to*	compar. d. of long adj.	yes/no questions
wh- questions	superl. d. of long adj.	*going to*	questions	perfect progr. aspect

Extremely infrequent (0-10%)

what, *who* question	present perfect progr.	unreal conditionals	determiner *much*	future perfect
tag questions	emphatic *do*	partial negation	*how*, *why* questions	*able*, *ought*, *need*
imperatives	superl. d. of long adv.	past perfect progr.	absolute poss. pronouns	compar. d. of long adv.
used to	*when*, *where* questions	*which*, *whose* questions	modal verb questions	*have* questions

Table 4.2: Classification of constructs based on their document frequency normalized by the total number of documents in the web collection.

While the table shows that grammatical constructions are highly heterogeneous, the heat map in Figure 4.1 demonstrates that they are not evenly distributed across the different documents: white cells represent documents with no occurrences of a given form, and red and black cells stand for documents with a very rich linguistic representation of that form. However, conclusions should not been drawn based solely on this image since grammatical constructions of different levels are mixed here, i.e., the word and sentence levels. In order to get a clearer picture of less frequent constructions, we replotted the data consisting for sentence-level constructions only. Complex (subordinate) sentences appeared to be the most frequent sentence type followed by compound (coordinate) sentences. Simple sentences, which usually correspond to lower levels of text complexity, are not easily found in every document, which complicates the search for reading material at lower levels of language proficiency.

4.1.3 Discussion of Challenges and Solutions

A statistical parser is typically not informed by a deeper knowledge of linguistic properties. When two constructions are identical in form, additional analysis of the target form in context can be required. For instance, Meurers et al. (2010) employ about 100 constraint grammar rules to disambiguate ***gerunds*** and ***participles***, which poses a challenge for both English language learners and parsers. ***Real conditionals*** and ***answers to indirect questions*** are another example of such ambiguities:

(10) a. Ramon doesn't *want* to come if Jochen is coming.
b. (Does Kordula know if Detmar is coming?)
Kordula doesn't *know* if Detmar is coming.

(10a) and (10b) are almost indistinguishable on the basis of the constituency and dependency structure provided by the parser. A simple solution based on a list of transitive verbs followed by an object *if*-clause (e.g., *know*, *see*) can help tackle this case, but may not generalize well to other cases where multiple interpretation are possible.

In the education context, not differentiating between several interpretations of the same form can mean exposing the learner to unfamiliar constructions far beyond

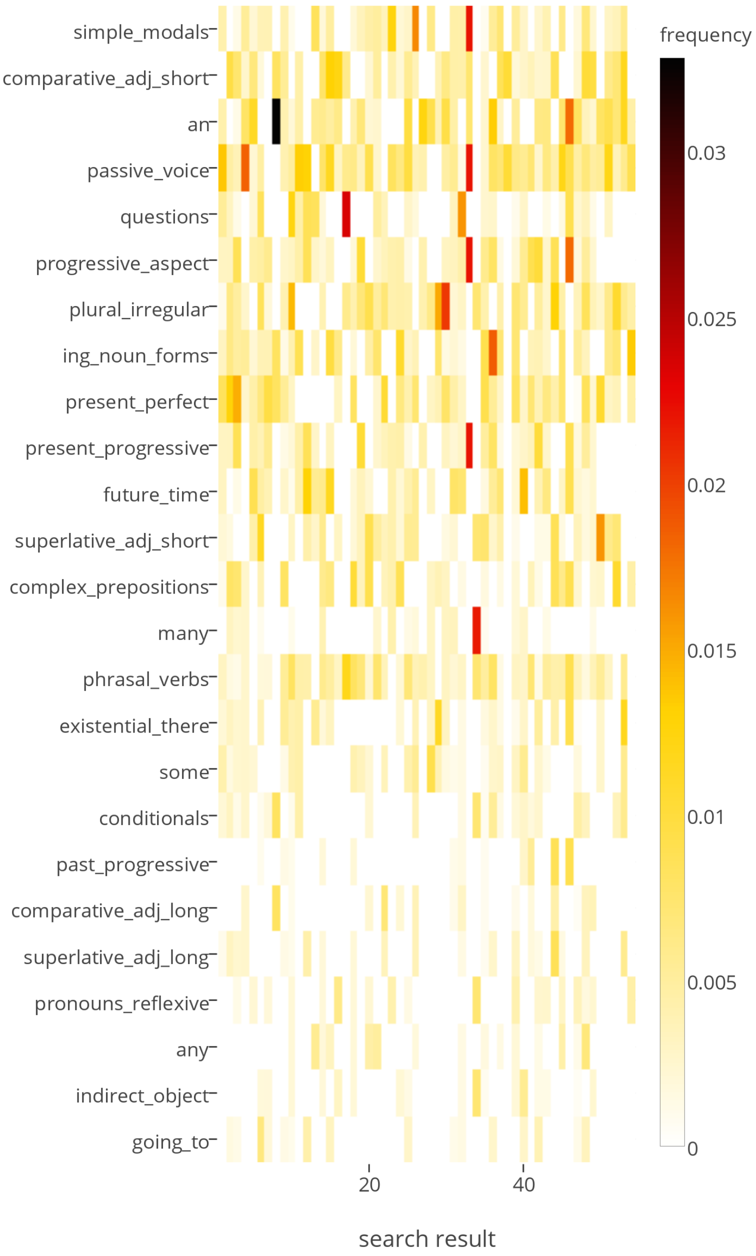


FIGURE 4.1: A heat map showing the distribution of grammatical constructions across the top 60 results: average for 40 queries of type *People* and *News*.

their current level. According to the English curriculum, *real conditionals* are introduced in the sixth grade while *answers to indirect questions* only in the eighth. Different parts of speech ending in *-ing*, such as *gerunds* and *present participle* forms, are taught in Grades 2, 8, and 10. Finally, the primary meaning of the *present progressive* as an action taking place at the moment of speaking (as in 11a) is introduced in the second grade. However, it is only six years later, in the eighth grade, that school children are expected to use this linguistic form to express the meaning of an arranged action in the future, as in (11b).

(11) a. Sebastian and Michèle are *waiting* for Michael.

b. Simón and Xiaobin are *leaving* next week.

The last phenomenon is not limited to the *present progressive* but generalizes to all grammatical tenses and some other linguistic forms. Therefore, it is both important to identify and disambiguate forms as well as distinguish among their different interpretations. In the next section, we approach this task with computational and statistical methods.

4.2 Towards the Disambiguation of Tense Senses

Sentences (11a) and (11b) above demonstrate two meanings, or senses, of the present progressive tense. In fact, most tenses have more than one sense. For instance, ***present simple*** can be used to express an action in the present (e.g., *Björn walks to work*), a future arrangement (e.g., *The train leaves at 9 a.m.*), or even a past event (e.g., in the headline *Earthquake Hits Iran*). ***Present perfect*** can express an experience (e.g., *Daria has been to Portugal once*), a finished action emphasizing the result (e.g., *Anna has finished her PhD*), or an ongoing action emphasizing the duration (e.g., *Natalie has lived in this house for 3 years*). This is reflected in the definition of tenses by Salaberry and Shirai (2002, p 2): They describe a tense as "a deictic category that places a situation in time with respect to some other time, usually the moment of speech". Deictic is the key word in this definition as it emphasizes the importance of the context in which a tense is used. Like lexical items, tenses are polysemous and are therefore appropriate subjects for the task of sense disambiguation.

Tenses are also crucial in the FLTL context, as they are challenging for language learners (Bardovi-Harlig, 1999). Learners particularly require support in establishing form-meaning connections for tenses: English grammar books make a distinction between the different meanings of the present progressive, present perfect, and other tenses (e.g., Murphy, 2012). In English schoolbooks, different types of activities are designed to introduce and practice different tense senses – in line with Ellis (2016) comments about the importance of tailoring the type of input enhancement to a particular linguistic form. Therefore, tutoring systems and iCALL applications offering activities that focus on grammar and tenses can benefit greatly from the use of an automatic component for disambiguating tense senses in order to more efficiently select appropriate texts and exercises. Outside of the FLTL context, this technology can improve the performance of CL tasks such as machine translation and automatic textual entailment. As different languages use different means of expressing time, individual tense senses may or may not match those expressions. In Section 4.2.8.3, we discuss the various possible translations of the present perfect tense into Russian and how the task of tense sense disambiguation (TSD) can both benefit from this linguistic phenomenon and support machine translation approaches.

In line with the only other work on this topic to our knowledge by Reichart and Rappoport (2010), we approach the task of TSD with a machine learning approach. In the following sections, we describe the research questions and task design, explain how the data were collected and annotated, review the machine learning classifiers we experimented with, present the results, and make suggestions to improve the performance of statistical models.

4.2.1 Task Definition and Design

As research on word sense disambiguation (WSD) has repeatedly shown, the disambiguation of senses is a classification task best solved with supervised methods (Navigli, 2009). This approach requires a large number of annotated instances, which have traditionally been obtained by asking expert judges to annotate a corpus using a given taxonomy. While Reichart and Rappoport (2010) followed this procedure for their TSD task, we propose a crowdsourcing approach already explored by the WSD research community (Hong and Baker, 2011).

The key difference between WSD and TSD is that the former is concerned with the semantics of lexical items, while the latter aims at disambiguating the *grammatical* meanings of *syntactic* forms. This difference is best demonstrated with an example: Sentences (12a) and (12b) present two lexical meanings of the verb *to work*, namely, *to do something for a living* and *to function.* On the other hand, Sentences (12c) and (12d) exemplify two grammatical meanings of the present progressive tense expressed by the verb *to work*, namely, *an action taking place at the moment of speaking* and *an arranged future event.*

(12) a. Magdalena *is working* from home today. (lexical meaning: to do something for a living; grammatical meaning: action taking place at the moment of speaking)

b. The machine *is* not *working.* (lexical meaning: function; grammatical meaning: action taking place at the moment of speaking)

c. Eran *is working* on a large project. (lexical meaning: to do something for a living; grammatical meaning: action taking place at the moment of speaking)

d. Adriane *is* not *working* tomorrow. (lexical meaning: to do something for a living; grammatical meaning: an arranged future action)

Automatic disambiguation of senses implies that the linguistic form itself has already been detected, which highlights another difference between TSD and WSD. While the detection of lexical items is possible with tokenization and POS tagging, the detection of grammatical tenses requires the use of constituency and dependency parsing (see Section 4.1 and Appendix B). As an algorithm for detecting linguistic forms, including grammatical tenses, is implemented in the *FLAIR* system, we used it to search and process corpora prior to addressing the TSD task itself.

The study aimed at addressing the following research questions with this study:

1. Can we develop TSD statistical models that outperform a strong most-frequent-sense (majority) baseline for each tense?

2. Do TSD statistical models for different grammatical tenses make use of different features?

4.2.2 Data Collection

First, we searched *Newsela*,[1] an American news website for language learners, for sentences containing instances of all grammatical tenses. Although we attempted to retrieve the same number of instances per tense, some tenses were highly infrequent, which resulted in a somewhat skewed distribution of instances. Having collected 1000 sentences, we ran a pilot study and added more sentences for a

[1] www.newsela.com

selected list of tenses afterwards. All in all, we collected 4,089 instances of grammatical tenses from *Newsela*. Interestingly, the future perfect progressive tense was not represented in the corpus, so we excluded it from further analysis. The distribution of tenses is presented in Table 4.3.

4.2.3 Data Annotation

4.2.3.1 Dictionary of Tense Senses

Kilgarriff (1997) argues against using a predetermined set of senses because of the constant deviation of the contextual meaning of words from their dictionary definition. At the same time, the author points out that a taxonomy of senses should not be a universal convention but should rather be designed for the task at hand. We are addressing the task of TSD for two reasons: (i) to provide English teachers and learners with a wider variety of contexts where linguistic forms occur in their different senses and (ii) to be able to automatically generate questions asking about different interpretations of linguistic forms (see Chapter 6 and Section 6.4 in particular). After consulting several English grammar books and textbooks (Murphy, 2012; Jespersen, 2013), we compiled a dictionary of tense senses whose disambiguation is relevant in the FLTL context (see Appendix E).

Similarly to the task of WSD, whether to make the tense sense classification coarse- or fine-grained is a design decision. We had several considerations when choosing the number of senses for each tense. The first came from the results of SensEval/SemEval shared tasks, which show that coarser-grained taxonomies lead to better inter-annotator agreement and, consequently, higher algorithm accuracy (Navigli et al., 2007; Pradhan et al., 2007), as already discussed in Section 2.2.2. This was confirmed by our pilot crowdsourcing annotation round on the present perfect tense and its senses: Changing the number of senses from three to four by splitting one sense into two led to a decrease in average agreement from 76% to 72%. This motivated the coarse-grained nature of our dictionary, in which every tense had at most four senses. On the other hand, for the sake of the disambiguation task, we wanted every tense to have at least two senses, which turned out to be difficult for some infrequent tenses. In these cases, we opted for a more fine-grained classification.

4.2.3.2 Gold Standard Tense Sense Annotation

We obtained gold standard annotations of tense senses from two experts, both doctoral students in CL, and one of whom was the author of this thesis. They first read the instructions and familiarized themselves with the dictionary of tense senses described in Section 4.2.3.1 and presented in Appendix E. They were then presented with a single sentence from the data set described in Section 4.2.2. The target tense instance was highlighted, and the senses of this tense were listed next to the sentence. The annotators selected the most appropriate tense sense from the list, or the option *None of the above* if they thought that none of the senses matched the grammatical meaning expressed by the highlighted form. Each expert annotated 77 instances.

The inter-rater agreement across all tenses calculated using Cohen's Kappa was 70%. As Kappa is sensitive to a lack of variability in the ratings, we do not report it for individual tenses due to the highly imbalanced representation of senses discussed below. Instead, we opted for an overall agreement measure, calculated as the number of items both annotators agreed on over the total number of items. The overall agreement was comparable to the one calculated using Kappa: 71%. Interestingly, the agreement for some tenses was much lower than for others: Present perfect senses posed the biggest challenge to the annotators yielding an agreement of 55%. Future simple and present perfect progressive, on the other hand, proved to be the easiest tenses to annotate, with an agreement of 83-88%. This was due to the fact that these tenses exhibited a highly imbalanced representation of senses, which was also true for some other tenses: All progressive-aspect tenses had one predominant sense of *action in progress*; and all instances of future simple, present perfect progressive, and future perfect were used to represent only one meaning. Therefore, we only included a small number of those tense instances into the crowdsourcing annotation study.

4.2.3.3 Crowdsourcing Tense Sense Annotation

Annotating a collection of several thousand sentences is costly in terms of time and effort but is necessary for the reliable performance of statistical methods. As crowdsourcing has proved to be successful at simple linguistic tasks, including WSD (Hong and Baker, 2011), we opted to use it to annotate our data set with tense senses. As previously mentioned, *FLAIR* detected the tenses automatically, so the crowd workers only had to select the most appropriate sense for a target tense. Following the best practice guidelines suggested by Sabou et al. (2014) and utilizing the functionality of the Figure-Eight platform[2] (formerly CrowdFlower), we:

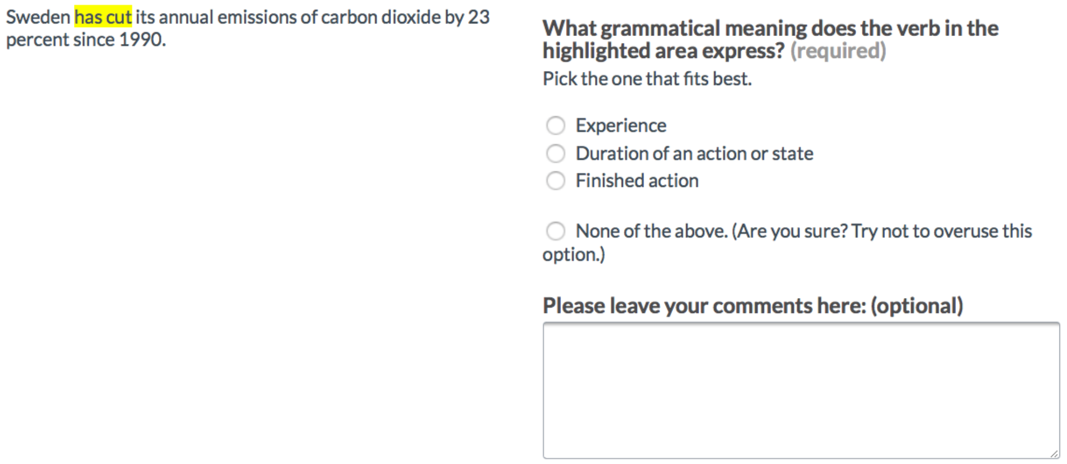
- wrote clear instructions, which were rated 4.2 out of 5 by the crowd workers in the main study,
- displayed a short text containing only one sentence,
- presented a selection of no more than four options followed by the option *None of the above*,
- designed a clear interface,
- provided reasonable test questions, which were rated 3.7 out of 5 by the crowd workers in the main study,
- used the testing service provided by Figure-Eight to get feedback from several trusted crowd workers about the task before launching it, and
- ran pilot studies to adjust the instructions, examples, test questions, and payment.

The Tasks We designed and ran three crowdsourcing studies to obtain enough data to develop and test our TSD statistical models. Two pilot studies investigated the level of granularity of tense sense taxonomy and only included instances of present perfect. The main study included all grammatical tenses except future perfect progressive, as no instances of this tense were found in the collection of 5,000 documents. As previously mentioned, the expert annotation revealed that some tenses had highly imbalanced sense classes, so we included fewer instances of those in the main study.

Procedure In each study, participants were presented with items consisting of a sentence with a highlighted predicate and a list of senses of the grammatical tense represented by it. Figure 4.2 demonstrates an item from the main study.

[2] `www.figure-eight.com`

The items were displayed in groups of five, and one of the items in each group was a test question to ensure the participant's reliability. We discuss the importance of test questions later in this section.



Sweden has cut its annual emissions of carbon dioxide by 23 percent since 1990.

What grammatical meaning does the verb in the highlighted area express? (required)
Pick the one that fits best.

○ Experience
○ Duration of an action or state
○ Finished action

○ None of the above. (Are you sure? Try not to overuse this option.)

Please leave your comments here: (optional)

FIGURE 4.2: An item from the main crowdsourcing task on TSD.

We used 100 items with ten test questions for each of the pilot studies focusing on present perfect and 2062 items with 50 test questions for the main study. We collected at least three judgments per item resulting in a total of 6526 judgments.

Ensuring Reliability of Participants To ensure the quality of the responses, we first selected a list of participating countries, including both English-speaking countries and some European countries, where English proficiency is high according to the EF English Proficiency Index (First, 2017): the Netherlands, Denmark, Norway, Sweden, Finland, Germany, and Austria. The official guidelines on the Figure-Eight platform list three ways of collecting good-quality judgments: instructing, training, and testing crowd workers. The first is achieved by providing clear instructions and examples, while the latter two are achieved via test questions.

As Hong and Baker (2011) noted, when designing a crowdsourcing study, one should avoid unclear terminology in the instructions and the task itself and explain everything in layman's terms. Along the same lines, Munro et al. (2010) pointed out that crowdsourcing is most successful when an annotation task is designed to be as simple as possible. We followed this advice and avoided linguistic terminology in the instructions and the task, which made the task accessible for a larger number of crowd workers (see Appendix F for the instructions).

Each study had two parts: the quiz and the main task. To make the quiz possible and ensure the proper training and testing of crowd workers, we included a sufficient number of test questions in each study. First, participants had to answer four out of the five quiz questions correctly in order to proceed to the main task, where they had to keep their accuracy at 70% by correctly answering test questions randomly inserted among the other items. While some of the test questions looked exactly like the items in the main task, another type of questions tested the crowd workers's attentiveness, as illustrated in Example 13:

(13) Please, select the third option:
  a. Experience
  b. Finished action
  c. Duration of an ongoing action

Munro et al. (2010) emphasize the importance of including such test questions, arguing that they not only filter out unreliable workers but also generally prompt workers to be more attentive. Another type of test question can be used to draw workers' attention to concepts relevant to the task at hand. For example, asking workers to select the correct rephrasing of (14) can be used to draw their attention to the concept of hypothetical actions. However, we did not use this type of test question in the current study in order to keep the task short and feasible.

(14) Yulia could have fallen down.
  a. Real action (*Yulia fell down*)
  b. Hypothetical action (*Yulia did not fall down*)

Inter-rater agreement To measure the agreement of a large number of crowd workers providing annotations on an imbalanced set of classes for some tenses (see Table 4.3 for the distribution), we opted to report simple agreement. This was calculated as the number of judgments agreeing with the most commonly selected annotation per item divided by the total number of judgments per item, averaged across all items. The average agreement across tenses was 87%, and the agreement for individual tenses is reported in the corresponding sections below.

Grammatical tense	**No. of instances**	**Distribution of senses**
Present simple	409	77 / 16 / 3 / 2 %
Present progressive	395	93 / 6 / 1 %
Present perfect	207	14 / 21 / 57%
Present perfect progr.	92	100 / 0 %
Past simple	363	9 / 87 / 2 / 0 %
Past progressive	94	96 / 3 / 1%
Past perfect	357	4 / 6 / 89 %
Past perfect progr.	56	39 / 61%
Future simple	93	97 / 0 / 3%
Future progressive	72	98 / 2%
Future perfect	24	100 / 0 %
Future perfect progr.	0	0 / 0 %

Table 4.3: Distribution of grammatical tenses and their senses in the dataset.

4.2.4 Annotated Data Set

Only the sentences where the majority of the annotators agreed on a tense sense were included into the analyses, thus, eliminating ambiguous items as well as those where the tense was incorrectly detected. Having thoroughly examined the annotations, we discuss the data collected for each individual tense in this section. Table 4.3 provides an overview of the number of instances of each tense in our dataset and the distribution of their senses, as annotated by the crowd workers.

4.2.4.1 Present tenses

Present simple As discussed in Section 4.2.3.1, we opted for a coarse-grained taxonomy with at most four senses per tense. Present simple was represented by four senses:

1. State in the present (with verbs that do not denote action: be, know, have) (*Joscha knows everything about programming.*)

2. Repeated action (habit or routine) (*Johann drinks coffee at 8 a.m. every morning.*)

3. Future scheduled event (*The train leaves at 7 p.m. next Monday.*)

4. Past event in a report, storytelling (*Hurricane destroys several cities.*)

The first sense, *state in the present*, was the most frequent (77% of instances), followed by *repeated action* (16%). Only 5% of instances were annotated as having the sense of *a future arrangement* or *past event*. Overall, crowd workers achieved an agreement of 83% when annotating the senses of present simple. However, one test question was labeled incorrectly in 50% of cases, making it a common challenge for annotators:

(15) "Sledding *is* a risky activity," Dubuque's city attorney wrote in proposing the ban.

Although the verb *is* expresses the *state* of sledding being risky, half of the annotators were presumably misled by its nature as an *activity*, interpreted the sentence as a whole, and labeled this occurrence of present simple as *a repeated action in the present*. We thus revisited the instructions and examples and replaced all references to 'sentences' with 'highlighted words' in order to avoid confusion: e.g., *What grammatical meaning do the highlighted words express?*

We only used the first two senses of present simple in the statistical analyses when training the statistical models. To cover all four senses, simple rules were used to detect the infrequent *future scheduled event* (via a time or date expression referring to the future as a dependent of the target tense) and *past event* (by checking whether all words in the sentence were capitalized, as is often the case in headlines).

Present progressive This tense had three senses: *action in progress*, *future arrangement*, and *repeated annoying action*. The annotators selected the first sense, *action in progress*, for 93% of items, with all three annotators agreeing on most of them (88%). However, some items with 100% agreement were incorrectly annotated:

(16) A key part of their plan *is casting* Republicans as uncooperative.

This example shows that the *FLAIR* algorithm incorrectly detected an instance of the present progressive tense because of the ambiguous parse (VP (VBZ is) (VP (VBG casting))), where the copula *is* is annotated as an auxiliary dependent on *casting*. Pointing out this possibility in the instructions in lay terms may encourage the annotators to be more attentive and select the *None of the above* option in a similar situation.

As the sense classes for present progressive were highly imbalanced, we concluded that a rule-based algorithm was the optimal solution for TSD in this case. The second sense of present progressive, *future arrangement*, can be detected if one of the direct dependents of the main verb in the target predicate is a named entity *time* or *date*. Instances of the third sense, *repeated annoying action*, can be identified by searching for the adverbs *constantly* or *always* as direct dependents of the main verb in the predicate.

Present perfect Expert annotation, two pilot studies, and the main study showed that this tense achieved the lowest inter-annotator agreement of 55-76%. The two pilot studies both targeted present perfect and differed in the number of senses for this tense: three and four, respectively. Making the taxonomy of senses more fine-grained in the second study lowered the inter-annotator agreement from 76% to 72%, so we opted for the first setting with three senses for the main study:

1. Experience (*Huan has never been to Barcelona.*)
2. Ongoing action or state (*I have known Barbara for 3 years / since 2015.*)
3. Finished action (*Heiko has finished debugging and can read the comics now.*)

While agreement on the second and third senses was at 80%, the main challenge occurred with the first sense, *experience* (71% agreement). Example (17) below received the label *experience* with the lowest agreement and demonstrates the difficulty of the annotation task for some items. It is not clear from the context whether the process of promoting 'these new incentives' has already finished. Thus, *has been* may express either *experience* or *an ongoing state.*

(17) Bolund, one of six Green Party cabinet members in Sweden, *has been* a key figure in promoting these new incentives.

Present perfect progressive As in the case of present perfect, not every context made it clear whether an action described by the present perfect progressive tense was still ongoing, as can be seen in the following example:

(18) Another reason might be that doctors *have been telling* parents not to give small children allergens, like peanuts.

There are not enough cues in this sentence to make inferences about what the doctors are telling parents now. However, this lack of evidence did not stop the crowd workers from annotating all 92 pilot instances of present perfect progressive with its first sense, *ongoing action.* Presumably, the intuition that most people follow is that a progressive aspect always denotes an action in progress. We did not collect more data on this tense and concluded that the detection of the second sense of present perfect progressive, *finished action that stopped very recently*, requires a context larger than one sentence. We leave this for future work.

4.2.4.2 Past tenses

Past simple This tense was represented by four senses:

1. State in the past (*Ulrich and Adam were very busy last year.*)
2. Single action in the past (*Chris and Lee saw a good film last night.*)
3. Repeated action in the past (*Veli and Ico went dancing every week.*)
4. Social distancing (*I just wanted to ask you...*)

The second sense, *single action in the past*, was the most frequent, receiving 87% of all annotations, while instances of *social distancing* did not appear in the dataset at all. As this sense is represented by a narrow range of expressions, such as *I just wanted to...*, we can use this simple heuristic to detect it. Therefore, we only used three senses when training a statistical model for past simple.

Past progressive As in the case of past simple, the *social distancing* sense of past progressive did not appear in the dataset at all. Moreover, as in the case of the other progressive aspect tenses, the most frequent sense, *action in progress in the past*, received the absolute majority of annotations (96%). Therefore, we did not build a statistical model for this tense and assumed that simple heuristics can be used to detect the infrequent senses of past progressive: e.g., the presence of the conjunctions *when* or *while* for the sense of *ongoing action interrupted by* *another action in the past*, the presence of the adverbs *always* or *constantly* for the sense of *repeated annoying action in the past*, and a list of expressions, such as *to be wondering*, *to be thinking*, for the sense of *social distancing.*

Past perfect This tense received an agreement of 88%, with 89% of instances annotated with its last sense, *finished action.* The issue already discussed above regarding the completeness of an action led to some disagreement among annotators. This can be exemplified with the following item:

(19) "I think I just realized at that moment – I *had never been* there," he said.

The distribution of answers for this item was:

- Experience at a time point in the past: 22%
- Duration of an action or a state at a time point in the past: 22%
- Finished action at a time point in the past: 44%

One person commented: "I agree that 'have never been' would be an experience at a time point in the past, to me the verb phrase "HAD never been" is a finished action as had implies that the action is over - eg 'I had never been there until that day'." Intuitively, a past perfect predicate containing negation (*had never been*, *hadn't seen*) does indeed imply that this action actually happened shortly before the moment of speaking or is happening then: *Jason had never seen a tiger before* would probably be said in a situation when he finally saw a tiger. However, one could also come across the following context: "Martí's daughter asked him what a Yeti looked like. Martí had never seen one so he did not know what to say." This is similar to present perfect, where *Johannes has never seen a tiger* may or may not mean that he sees a tiger now. These examples illustrate yet again that a broader context is needed to differentiate between the senses of some linguistic forms.

Past perfect progressive The overall agreement among crowd workers for this tense was 70%, and its two senses received a comparable number of annotations: 39% and 61%. However, after more closely inspecting the instances of this tense, we concluded that most of the sentences did not provide enough context to properly differentiate between a finished and an ongoing action in the past, as demonstrated in the example below.

(20) That violin *had been missing* for more than two years.

The distribution of senses for this item was:

- Ongoing action at a time point in the past: 57%
- Finished action at a time point in the past: 43%

One person commented on this item: "Sounds to me that it is finished. Had been missing means just that, had, which is no longer." The sentence does not say that the violin was found, but the annotator believes that the semantics of the past perfect progressive tense suggest it. However, the sentence could also continue in the following way: "..., and still nobody could find it." For many items, the assignment of the first or the second sense exhibited a random nature. Due to this ambiguity in a narrow context, we leave this tense for future work, in which we plan to conduct a similar study but provide a larger context for each item.

4.2.4.3 Future tenses

All instances of future tenses were annotated with their first senses with overwhelmingly high frequencies: 97% of future simple items were labeled as *a future event or state*, 98% of future progressive items were annotated with the sense of *an action in progress at a point of time in the future*, and 100% of future perfect items were tagged as *an action completed before a future point of time.* No instances of future perfect progressive were found in a corpus of 5,000 news articles.

Taking into account this imbalanced distribution of senses, we concluded that machine learning was not a suitable approach for the disambiguation of the senses of future tenses. As was the case for present progressive and some other tenses, simple heuristics could be implemented to detect the instances of infrequent senses, such as the presence of the preposition *for* followed by a time expression to detect the second sense of future progressive and future perfect, namely, *duration of an action or a state in the future.*

To conclude, we selected four tenses for machine learning experiments: present simple, present perfect, past simple, and past perfect. The next section presents a list of features designed to differentiate the senses of these tenses.

4.2.5 Features Distinguishing the Senses of Grammatical Tenses

We compiled a list of surface, lexical, syntactic, and discourse features taking into consideration the distinguishing characteristics of various tense senses. ***Surface features*** include sentence length, which has been found to be a more reliable feature than syllable count (Uitdenbogerd, 2003), as well as the length of the predicate containing the target tense and its position in the sentence. All ***lexical features*** are binary and check for the presence of prepositions, adverbs, articles, negation, and different types of verbs in the sentence. ***Syntactic features*** check for instances of other tenses in the sentence, commas surrounding the target tense instance, and dependents of the main verb in the target tense, including date and time expressions. Finally, ***discourse features*** range from the sentiment of the sentence to the presence of adverbs and conjunctions of time. All features were extracted using Stanford CoreNLP and additional simple algorithms and are listed in Table 4.4.

In line with some WSD researchers (Mihalcea, 2002; Martínez et al., 2002), we assume that different tenses benefit from different features: For instance, the presence of the preposition *since* in a sentence may help differentiate between the senses of present perfect more so than the senses of present simple. In the following sections, we first experiment with all 22 features and then train models using different sets of features for different tenses.

4.2.6 Methodology

4.2.6.1 Learning Algorithms

The choice of the learning algorithm was motivated by several criteria defined according to the TSD task. To meet our requirements, an algorithm had to:

Surface (all continuous)	Sentence length
	Length of the tense instance
	Tokens between the tense instance and the previous punctuation mark
	Tokens between the tense instance and the next punctuation mark
Lexical (all binary)	The main verb of the tense instance is stative
	Presence of any form of *to be* in the tense instance
	Presence of articles in the sentence
	Presence of *for* or *since* related to the tense instance
	Presence of the adverb *already* in the sentence
	Presence of *now* or *at the moment* in the sentence
	Presence of the adverb *every* or a marker of frequency in the sentence
Syntactic	The closest instance of another tense in a sentence (categorical)
	The tense instance is followed by a comma (binary)
	The tense instance is preceded by a comma (binary)
	Number of dependents of the main verb (continuous)
	Date or time expression is a dependent of the tense instance (binary)
Discourse	Sentiment of the sentence (categorical)
	Sentence contains reported speech (binary)
	Presence of the conjunction *after* in the sentence (binary)
	Presence of the conjunction *before* in the sentence (binary)
	Presence of the conjunctions *when* in the sentence (binary)
	Presence of the conjunctions *while* in the sentence (binary)

Table 4.4: Features used in TSD statistical models

1. Be suitable for a multi-class classification task (logistic regression, k-nearest neighbors, gradient boosted decision trees, SVM, random forest)
2. Perform well with a small number of observations (logistic regression, Naive Bayes, SVM)
3. Handle a mixture of feature types, such as binary and categorical, with or without additional encoding (decision trees, gradient boosted decision trees, random forest, SVM)
4. Be interpretable to some extent (no neural networks)
5. Handle imbalanced data well (decision trees, random forest)

Taking these criteria into account, we experimented with decision trees, linear SVM, and random forest for each tense and selected the best performing one, achieving a balance between weighted F_1 and precision scores in a 10-fold stratified cross-validation. We implemented the algorithms using the *sklearn* Python library (Pedregosa et al., 2011) and provided the models along with the parameters in Appendix G. The following section presents the motivations behind our choice of evaluation metrics.

4.2.6.2 Evaluation metrics

There are several common metrics for evaluating the performance of machine learning algorithms, including accuracy, recall, precision, F_1-score, and AUC. While accuracy appears to be commonly used for standard data sets, it is not suitable for our case as the sense classes for almost every tense are highly imbalanced, which will most certainly lead to an overestimation. Thus, we selected the optimal evaluation metrics by analyzing the task at hand and the desired outcomes.

In the FLTL context, high precision is favored over high recall as learners' exposure to erroneous language should be minimized. However, the balance of precision and recall is also an important measure of algorithm effectiveness, so we opted for an F_1-score as the measure of comparison between our models and a strong majority (most-common-sense) baseline. As weighted averaging takes the skewed distribution of labels into account, we calculated a weighted F_1-score and precision when selecting the best performing model for each tense.

To properly evaluate the models, we first split the data into a training set (75%) and a held-out test set (25%). To assess the predictive performance and the generalizability of our models, we performed 10-fold cross-validation on the training set. This evaluation method repeatedly splits the data into a training subset (9/10 of the data) and a test subset (1/10 of the data), trains the selected model on the training subset, and tests it on the test subset. It repeats this for each of the ten folds and reports the average. As our data set contains a rather small number of instances and imbalanced classes, we opted for a stratified 10-fold cross-validation, which ensures a balanced representation of all classes in the test subset for each individual fold. Importantly, we did not use the held-out test set to tune the parameters of the models when performing cross-validation. The performance of our models on the held-out test set is reported in Table 4.5 along with the cross-validation results.

4.2.7 Results

4.2.7.1 Performance

Unlike Reichart and Rappoport (2010), who trained one model for all grammatical tenses, we treat each tense individually. However, although our models cannot be directly compared, the authors' results can be seen as another general baseline. While Reichart and Rappoport (2010) do not report the classification accuracy for each tense, they mention the accuracy gain over the baseline for several tenses. In line with our results, present perfect senses seem to be the most difficult to classify, with an accuracy of only about 57%. Past perfect and present simple, on the other hand, achieve accuracies of 77.3% and 75.8%, respectively. For the sake of comparison, we note that our models achieved an accuracy of 62% for present perfect, 82% for present simple, and 92% for past perfect, as measured in a 10-fold stratified cross-validation. However, as argued in Section 4.2.6.2, we only use the F_1-score evaluation metric due to the highly imbalanced data set.

Table 4.5 compares the performance of a strong majority baseline with that of the best performing algorithms used to detect the senses of four grammatical tenses. The results demonstrate that our models outperform the baseline both on the held-out test set and on the 10-fold cross-validation performed on the training set. The biggest gain in F_1-score was observed for the present perfect model ($p = .02$, 95%

Tense	**Model**	**10-fold CV**		**Held-out Test Set**	
		Baseline F_1	Model F_1	Baseline F_1	Model F_1
Present simple	Random forest	$M = .75$ $(SD = .02)$	$M = .78$ $(SD = .04)$	.73	.78
Present perfect	Linear SVM	$M = .48$ $(SD = .03)$	$M = .61^{*}$ $(SD = .13)$	.48	.56
Past simple	Decision tree	$M = .84$ $(SD = .04)$	$M = .91^{**}$ $(SD = .06)$	.82	.88
Past perfect	Decision tree	$M = .90$ $(SD = .03)$	$M = .91$ $(SD = .02)$	.86	.88

* $p \leq .05$, ** $p \leq .01$: significant differences between a TSD model and the baseline

Table 4.5: Performance of our TSD statistical models on crowd-annotated data using all 22 features compared to the strong majority baseline.

CI $[-0.23; -0.02]$) and the past simple model (p = .01, 95% CI $[-0.12; -0.02]$), which significantly outperformed the majority baseline as measured by a Student's t-test. These results provide strong evidence for the predictive power of our models.

TSD has proved to be a complex task, in which achieving high agreement is difficult even for humans. As this certainly influenced the performance of our statistical models, we decided to re-annotate part of the dataset ourselves and investigate (i) how much we (dis)agreed with the crowd and (ii) how much the performance of the statistical models could be improved by improving the data quality. Concretely, the author of this thesis annotated the 207 instances of present perfect and re-ran the machine learning algorithms using these new data.

The agreement between the author and the crowd workers was 60%, which is lower than the agreement among the crowd workers (76%) but higher than that among expert annotators on a smaller dataset (55%). This may indicate that the instructions and examples were too simplistic, as crowd workers had a higher agreement among themselves than with the expert. On the other hand, it could be an indication of the crowd workers' stronger bias towards the most common sense: They selected the third sense of present perfect, *a finished action*, 57% of the time, as compared to the expert's 46%. The results presented in Table 4.6 demonstrate that the statistical models trained and tested using expert annotation significantly outperformed the majority baseline; p = .006, 95% CI $[-0.30; -0.06]$. However, the difference between the crowdsourcing and expert models was not statistically significant; $p = .2$, 95% CI $[-0.24; 0.05]$.

Annotation	Model	10-fold CV		Held-out Test Set	
		Baseline F_1	Model F_1	Baseline F_1	Model F_1
Crowdsourcing	Linear SVM	$M = .48$ $(SD = .03)$	$M = .61^*$ $(SD = .13)$	.48	.56
One expert	Decision tree	$M = .52$ $(SD = .06)$	$M = .71^{**}$ $(SD = .15)$	.46	.70

* $p \leq .05$, ** $p \leq .01$: significant differences between a TSD model and the baseline

Table 4.6: Performance of TSD statistical models for present perfect on crowd-annotated versus expert-annotated data using all 22 features.

4.2.7.2 Important Features

All three algorithms we experimented with provide ways of determining the relative informativeness of different features: either using weight (Linear SVM) or feature importance (decision tree, random forest) vectors. Across all four tenses, *surface* (or length) features and the syntactic feature *number of dependents* were the most informative. This is in line with the findings for readability assessment (Vajjala and Meurers, 2012) and TSD itself (Reichart and Rappoport, 2010), where surface features have proved to be extremely predictive.

Having used the whole list of 22 features presented in Section 4.2.5 to select the best performing model for each tense, we then experimented with non-surface features to see whether a smaller number of features we considered relevant for each tense could outperform the all-features models. Below we present the experimental results and provide the top most informative features of the tenses for which we built statistical models.

Present Simple To differentiate between the two most frequent senses of present simple, the selected-features random forest model ($F_1 = .77$, $SD = .04$) made use of eight features: presence of the conjunction *when*, presence of the auxiliary *to be* in the tense instance, the tense instance is followed by a comma, the tense instance is preceded by a comma, the main verb is stative, presence of articles in the sentence, sentiment of the sentence, and the closest instance of another tense in a sentence. This model slightly outperformed the strong majority baseline but did not achieve the performance of the all-features model, nor did any other algorithm. The most informative features for differentiating between the senses of present simple were:

1. Sentiment of the sentence (24%)
2. Main verb is stative (20%)
3. Presence of articles in the sentence (14%)

Present Perfect Linear SVM was the best performing model both when including all 22 features included and when only including six non-surface features (presence of the conjunction *after*, presence of the conjunction *when*, date or time expression is a dependent of the tense instance, the main verb is stative, presence of the adverb *already*, presence of *for* or *since*). The selected-features model even slightly outperformed the all-features model ($F_1 = .62, SD = .12$), albeit not significantly. The most discriminative features for present perfect were:

1. Main verb is stative (29%)
2. Date or time expression is a dependent of the tense instance (24%)
3. Presence of *already* (20%)

Past Simple For this tense, the selected-features random forest model including five non-surface features (the main verb is stative, date or time expression is a dependent of the tense instance, the closest instance of another tense, presence of the conjunction *before*, presence of the conjunction *while*) outperformed the majority baseline ($F_1 = .87, SD = .04$), but fell below that of the all-features model. The two other models did not outperform the baseline. Using an all-features model to differentiate between the senses of past simple appears to be the optimal solution. The most important features differentiating between the senses of this tense were:

1. Main verb is stative (45%)

2. Date or time expression is a dependent of the tense instance (24%)
3. Closest tense (14%)

Past Perfect All selected-features models outperformed the majority baseline for past perfect. They made use of six non-surface features: presence of the conjunction *after*, presence of the conjunction *before*, presence of the conjunction *when*, presence of the adverb *already*, presence of *for* or *since*, date or time expression is a dependent of the tense instance. Random forest achieved the best performance, with the same F$_1$-score in cross-validation as the best performing all-features model ($F_1 = .91, SD = .04$). Consequently, we can conclude that for past perfect, the performance of a model containing carefully selected non-surface features is comparable to that of a model including all features. The most informative features for differentiating between the senses of this tense were:

1. Presence of *for* or *since* (54%)
2. Date or time expression is a dependent of the tense instance (27%)
3. Presence of *after* or *before* (22%)

To conclude, all-features TSD models seem to be more robust and have higher predictive power than the strong most-common-sense baseline. A careful selection of features for each tense may not pay off, as such models do not significantly outperform the ones using all features. In future work, we plan to experiment with supervised methods using larger feature sets and unsupervised methods using tense vectors to improve the performance of the TSD models for different tenses. We discuss these and other challenges and possibilities further on in the section.

4.2.8 Discussion of Challenges and Solutions

Fellbaum et al. (1997) noted that training humans to tag senses is far more difficult than training them to assign parts of speech to words. Similarly, machine learning approaches to sense disambiguation do not yet reach the accuracy of POS taggers either. Partially on the basis of our own work and partially inspired by Navigli's (2009) overview of related research on WSD, we emphasize the following challenges of the task of TSD.

4.2.8.1 Training Data and Features

The WSD research community estimated that building a high accuracy domain-independent WSD model requires between 900 and 1400 occurrences of each word (Ng and Lee, 1996; Ng, 1997), which is extremely costly. However, the finite number of tenses and the existence of high-performing tense-detection algorithms such as *FLAIR* allow for extensive data collection for the task of TSD. Data annotation can be aided by the use of so-called bootstrapping, a statistical method inducing a classifier from a small set of labeled data and a large set of unlabeled data (Abney, 2002).

Defining separate feature sets for individual words has also been proposed: Mihalcea (2002) and Martínez et al. (2002) designed algorithms to automatically select the most informative features in each cross-validation fold to either train or adjust the final system. Such automatic approaches, as well as dimensionality reduction approaches such as principal component analysis and linear discriminant analysis, can facilitate the selection of the most informative features on-the-go.

4.2.8.2 Design Decisions

Using a fine-grained or coarse-grained taxonomy of senses. Dictionaries and thesauri seek to provide the most comprehensive list of senses for every word. For instance, the two main meanings of the verb *lie* are usually listed as separate entries. Within the first entry (with the general meaning of *being positioned*), there are more fine-grained meanings, such as *to be in a horizontal position*, *reside*, or *be situated*. The second entry refers to the meaning of misinforming someone and is not divided into sub-meanings. But how should a list of senses be compiled for a sense disambiguation task? Should it only consist of the two main senses; include the more fine-grained senses as sub-meanings, resulting in a hierarchical structure; or list them at the same level as the main senses? While the objective truth is probably – as it oftentimes is – somewhere in the middle, this issue also has a more practical value that is highly dependent on the task at hand. Research has shown that this decision influences the accuracy of data annotation and the inter-annotator agreement: In the Senseval-2 shared task, which used fine-grained word sense distinctions, human annotators agreed on only 85% of word occurrences, while Navigli et al. (2007) report an inter-annotator agreement of 86-94% for a coarse-grained word sense taxonomy in SemEval-2007.

At the same time, the task of TSD has achieved an inter-annotator agreement of 84% on a taxonomy of 103 senses (Reichart and Rappoport, 2010). Along with the twelve grammatical tenses, the authors also included other linguistic forms, such as conditionals and reported speech, in the task. This resulted in 18 linguistic forms and an average number of 5.7 senses per linguistic form. The authors do not present the senses and inter-annotator agreement for every linguistic form, but based on the list of 11 senses of the present simple tense presented in the paper, we conclude that they used a fine-grained taxonomy for some tenses and a coarse-grained one for others.

Compiling a finite list of senses or clustering senses on-the-go: Supervised or unsupervised algorithms. Compiling a list of senses is compulsory for supervised machine learning methods as they need annotated data for training models. In unsupervised learning, on the other hand, clustering techniques are used to automatically group word senses. Thus, the decision about the level of granularity of a sense taxonomy can be partially delegated to the algorithm, but one will still need to specify the number of clusters or a threshold of sense similarity.

In WSD, the most prominent unsupervised approach is context clustering. This is based on the assumption that the semantics of a word can be determined by the context in which it occurs, in line with the famous saying *“Tell me who your friends are and I will tell you who you are”.* Every target word is represented as a vector, the dimensions of which are determined by its co-occurrences with other words in the immediate context (a sentence or paragraph). In attempting to apply this approach to the task of TSD, the main question that arises is what kind of linguistic forms should be treated as ‘friends’ of grammatical tenses.

On the one hand, some tense senses may be distinguishable by their co-occurrence with certain lexical items. For instance, when present progressive is used with the adverb *always*, it is most likely to express a repetitive annoying action, such as *He is always complaining.* On the other hand, when the same adverb is used with present simple, an unsupervised clustering algorithm may not be able to distinguish between the two senses of the present simple tense: a repeated action (e.g., *Alina always takes a train to work*) and a state in the present (e.g., *Katharina is always very polite*). However, when the main verb in the tense is annotated as active (*take*) or stative (*is*), the distinction becomes more prominent. This is only possible when using a supervised approach.

Other grammatical tenses may also serve as 'friends' to the target tense and thus form vector dimensions. Indeed, when past progressive co-occurs with past simple in a sentence, it is most likely to express an action interrupted by another action in the past, not simply an action in progress in the past. For example, compare the sentences (21a) and (21b):

(21) a. Madeesh *was sleeping* when the phone rang and woke him up.
b. Ankita *was sleeping* at 12 a.m. yesterday.

This certainly does not apply to all co-occurring tenses, as can be seen in the sentences (22a) and (22b) below, where present simple and present progressive are used together but have different grammatical meanings:

(22) a. Marina *understands* why Slava *is working* so hard. (Present simple: state in the present; present progressive: action in progress)
b. Christine *walks* to work every day but tomorrow she *is cycling.* (Present simple: repeated action; present progressive: future arrangement)

As the *closest tense* feature included in our statistical TSD models proved to be informative for some tenses, we plan to experiment with unsupervised machine learning algorithms using tense vectors.

Choosing a sentence or paragraph-level context. Yarowsky and Florian (2002) demonstrate that the performance of WSD models is sensitive to the size of the context on which they are trained. They conclude that an increasing amount of context actually lowers the accuracy of discriminative WSD algorithms and propose using an optimal context window of ±10 neighboring words. Although this is more applicable to bag-of-words WSD models, one might assume that TSD models would similarly not benefit from the inclusion of features representing dependents of higher degrees. While leaving the empirical confirmation of this premise for future work, we include only the immediate dependents of the target tense into our feature set (see Section 4.2.5).

Another scenario where the size of the context is of importance concerns the human annotation of data. Our study showed that one sentence is not always sufficient for humans to differentiate between tense senses. However, in light of the previously mentioned finding from WSD research that more context may lead to lower performance (Yarowsky and Florian, 2002), future TSD studies need to test the effects of larger contexts on the inter-annotator agreement and the performance of machine learning algorithms for the task of TSD.

Using a large-scale data set or collecting your own data. Disambiguation of word senses ultimately serves other higher-level CL tasks – be it machine translation, natural language understanding, or automatic textual entailment. Consequently, Navigli et al. (2007) expresses concern about the lack of an end-to-end evaluation of WSD systems. Along the same lines as this discussion in the WSD research community, we argue that TSD systems should be designed, implemented, and evaluated for the task at hand. As already discussed in this section, linguists, computational linguists, and language teachers have different expectations when it comes to tense senses and their taxonomies. While linguists might arguably prefer the most fine-grained distinction of tense senses, language teachers may only want to present the most prominent ones to their students.

This has an implication for the design of the whole TSD task, from data collection to the choice of statistical methods. The type of corpora on which TSD models are trained could also be tailored to the end users: If graded readers for sixth grade do not use present progressive to refer to future arrangements, as in *He is driving home tomorrow*, it is because this sense is not introduced till grade eight. Training a model on the British National Corpus instead of texts appropriate for sixth graders may not only be unnecessary but also lower the accuracy of the model due to the larger variety of contexts. Therefore, we argue that although large data sets are great sources of easily accessible annotated data, one should ideally collect and annotate data for the task at hand.

Definition of the task When running a TSD task on a crowdsourcing platform, it is important to point out that the annotators should take into consideration only the highlighted part of the sentence and not try and annotate the sentence as a whole. For instance, one crowd worker left the following comment when annotating (23), providing a correct answer but also expressing the need for clearer task definition: “While only using one word to try and categorize, it is a bit difficult. Said is an action verb, and it’s a past tense verb.”

(23) “I’ve lost many nights of sleep trying to figure out where we’re going to get funding, and in recent months I just haven’t thought of any place left to go,” she *said.*

In general, researchers can greatly benefit from the comments that annotators leave, and we strongly believe that a constant feedback loop is crucial for successful data collection and annotation.

4.2.8.3 Use of Bilingual Corpora

Another interesting approach to TSD would be to use aligned bilingual corpora to disambiguate the tenses that have been translated to other languages. This was initially proposed by Brown et al. (1991) for the task of WSD and has been successfully used ever since, achieving state-of-the-art WSD performance (Ng et al., 2003; Bovi et al., 2017).

For instance, the senses of the English present perfect tense in our study include *experience*, *finished action*, and *duration of an ongoing action.* In Russian, a finished and an ongoing action would be translated using different tenses, past and present, respectively:

- **Experience:**
  Thomas *has been* to Italy twice. – Томас *был* в Италии дважды. (past)
- **Finished action:**
  Corina *has completed* the task. – Корина *выполнила* задание. (past)
- **Duration of an ongoing action:**
  Joel *has known* Jill for many years. – Джоел *знает* Джил много лет. (present)

This approach can also be used with unsupervised methods: a tense instance in a multilingual corpus can be represented as a context vector consisting of all the possible ‘tense’ translations of the target tense in other languages.

4.2.8.4 Addressing the Redundancy Principle

As discussed in Section 2.1.3, VanPatten's (2002) redundancy principle states that in order to draw learners' attention to the form of a grammatical construction and connect it to its meaning, no other markers expressing the same meaning, such as adverbs of time or dates, should be present in the sentence. That is, the sentence *Daniil is writing a paper* is preferable to *Daniil is writing a paper now* when practicing the present progressive tense in a communicative language-learning environment.

Unfortunately, the soundness of this principle is impeded by the limitations of its practical application. Even if the immediate context of one sentence does not introduce redundancy, it is likely to occur in a larger context: the time setting, event probabilities and order as well as gender and plurality can oftentimes be inferred due to the abundance of verbal and non-verbal contextual cues. In fact, WSD research heavily relies on such redundancy (Yarowsky, 1995) to design features for supervised machine learning approaches (Reichart and Rappoport, 2010; Lee, 2011). This is also the case for our system. To address this issue, we propose a feasible approach to ensuring that the learner notices and processes target linguistic forms even when they are redundant. In line with the SLA research discussed in Section 2.1, we implement automatic input enrichment, visual input enhancement, and functionally driven input enhancement in the *FLAIR* system by:

- retrieving texts containing a high number of occurrences of target linguistic forms (see Chapter 5),
- highlighting the instances of target linguistic forms in the text, and
- generating questions targeting those linguistic forms, which helps learners build form-meaning connections (see Chapter 6).

4.3 Summary

NLP makes use of different approaches to characterize language data, from shallow matching to deep grammar formalisms and machine learning. These are equally well-motivated for language learning as an application domain (Meurers, 2015, sec. 3.2). While some grammatical constructions in the English language support relatively straightforward characterizations based on the syntactic analysis provided by the Stanford CoreNLP, the detection of other constructions, and especially their different interpretations, or senses, requires methods going beyond this level.

As machine learning algorithms are able to learn from data and make predictions about unseen data, they are suitable for classification tasks that cannot be solved by specifying explicit rules, such as the disambiguation of tense senses. Having conducted a rigorous analysis of different grammatical tenses, we conclude that while some senses can be differentiated rather easily by implementing heuristic rules, others require the use of machine learning algorithms to account for the interaction of various features. Although the task of tense sense disambiguation is far from being solved, we were able to show that our statistical models for four tenses (present simple, present perfect, past simple, and past perfect) outperformed a strong most-frequent-sense baseline and a state-of-the-art model by Reichart and Rappoport (2010). To conclude, analyzing the tense senses in authentic data is a complex task that highlights the need to revisit traditional FLTL notions.

The algorithms for detecting linguistic forms and their senses discussed in this chapter open up a range of opportunities for iCALL applications. Searching for appropriate reading materials containing these forms is one such application, which has its roots in the SLA notions of input flood and input enrichment. We approach this as an information retrieval task, discuss its implementation, and present an online study investigating the benefits of input enrichment for language teachers in the next chapter.

Chapter 5

Automatic Input Enrichment

> "An ideal Web search site for language learners [...] would provide sophisticated querying capabilities to ensure highly relevant results, not only matching characters, but also parts of speech and even syntactic structures. [...] Above all, a search site for language professionals would stress quality and relevance of search results over quantity."
>
> Fletcher (2004)

Parts of the work discussed in this chapter appeared in the following peer-reviewed publications and theses:

1. Chinkina, M., & Meurers, D. (2018). Automatic Input Enrichment for Selecting Reading Material: An Online Study with English Teachers. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications.* New Orleans, LA.
2. Chinkina, M., Kannan, M., & Meurers, D. (2016). Online information retrieval for language learning. *ACL 2016: System demonstrations.* Berlin, Germany.
3. Chinkina, M., & Meurers, D. (2016). Linguistically Aware Information Retrieval: Providing Input Enrichment for Second Language Learners. In *Proceedings of the 11th Workshop on Innovative Use of NLP for Building Educational Applications.* San Diego, CA.
4. Chinkina, M. (2015). *Form-focused Language-aware Information Retrieval* (Master's thesis, Eberhard Karls Universität Tübingen).

5.1 A Search for Reading Material

The information needs of people searching for texts vary greatly and require different factors to be taken into account. In the educational context alone, one can think of several groups of target users, namely, students, teachers, and language test designers, each of whom will have their own search preferences. Indeed, the information needs of a student writing an essay are quite different from those of an educator designing a language test, as is their willingness to take time to configure the search settings in order to get the best results. While a student may be satisfied with interesting content and gloss over some unknown words or grammatical constructions, a language test designer will most likely have a clearer idea of appropriateness factors and, if given the opportunity, will make use of a larger number of text complexity settings.

When it comes to acquiring appropriate texts, sites such as *Newsela*[1] offer help to users searching for reading material of a certain length and at a desired language proficiency level. Their scope is narrowed to news articles that are manually adapted and grouped into thematic categories and grade levels, which facilitates search. *Adapted readers* and *schoolbook texts* are another example of readily-available enriched texts produced by language learning specialists. They either simplify an already existing story or write an entire text from scratch, fortifying it with grammatical constructions and vocabulary. Such books do not usually require any additional searching as texts either represent a single story or are grouped by grammar topics.

When searching for additional reading material and interactive activities, foreign language teachers often turn to the Web. We chose them as the target group for assessing the need for a web search engine for educational purposes, and asked them about their use of web search in their teaching practice. After conducting literature research on effective survey instruments (e.g., Sudman and Bradburn, 1982; Rattray and Jones, 2007), we presented the teachers with a survey consisting of 14 questions. The majority of the teachers who took part in the survey were in their 20s or 30s and had six to fifteen years of teaching experience (90%). The majority were English teachers (80%) but some teachers of other languages, such as Japanese (10%) and German (10%), were also represented. Two questions in the survey asked whether teachers used web search engines to look for adapted

[1] `https://newsela.com`

or authentic reading materials on the Web to use later in class and, if yes, at which level (A1-A2, B1-B2, C1-C2). The results show that teachers search the Web for both adapted and authentic texts at all complexity levels, with Level B1-B2 receiving the largest number of responses (87.5%). Among the factors that discourage them from using web search for educational purposes, 100% of teachers selected vocabulary that does not correspond to the words students learn at school (e.g., slang). Grammatical constructions that do not follow the rules taught at school were among the most discouraging factors for 60% of teachers. Although grammar difficulty as such was not selected as a discouraging factor by many teachers, when asked about the desired functionality of a web search engine, all of them expressed the need to find texts appropriate for a certain reading level and containing particular grammatical constructions.

5.2 Input Enrichment Strategies: From the Web to Corpora

While web search is a common method of acquiring authentic texts, there is an abundance of other resources on specialized websites and language learning forums. Teachers also have reading materials that they frequently use in class and will benefit from the ability to search and automatically enrich such corpora. Therefore, we argue that input enrichment systems – although similar in nature to web search engines – should be designed to search both the Web and other text collections. Figure 5.1 presents an overview of input enrichment strategies for ensuring a sufficient representation of target linguistic forms in reading materials.

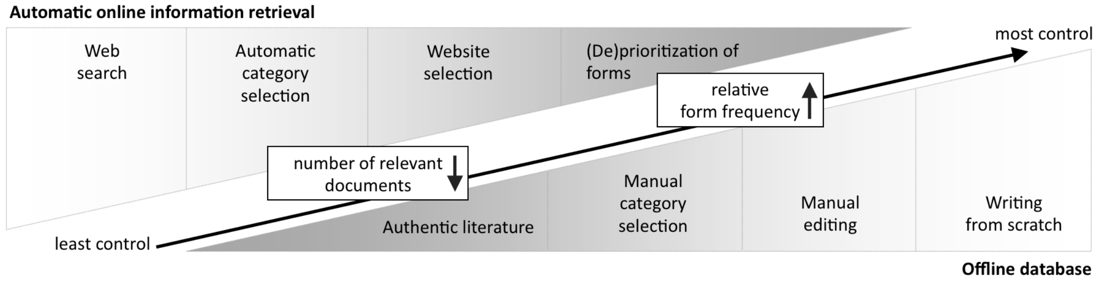

Automatic online information retrieval
Web search
Automatic category selection
Website selection
(De)prioritization of forms
most control
relative form frequency
number of relevant documents
least control
Authentic literature
Manual category selection
Manual editing
Writing from scratch
Offline database

FIGURE 5.1: Strategies for automatic input enrichment of an already existing corpus or during web search.

Automatic input enrichment starts with a search for a topic of interest. Restricting the search to a particular category, such as FAQs and tutorials, may not only narrow down the semantic scope of the results but also ensure the occurrences of certain linguistic constructions, such as *questions* and *imperatives*, respectively. Similarly, when searching a particular news website (e.g., Reuters[2]), one can expect the retrieved news articles to have a higher or lower representation of formal vocabulary, such as the words on the Academic Word List (Coxhead, 2000). These techniques can be configured in the interface of a search engine or by expanding the search query with words (e.g., *FAQ*) or advanced search operators (e.g., *site:reuters.com*). One can also make use of a standard lexical query expansion to maximize the probability of the occurrence of the target linguistic forms in the retrieved documents (see Section 5.5.3). The amount of further control over the search results depends on the implemented settings and the available meta information about the documents. While a text corpus can be annotated in advance, the automatic annotation of web texts can be conducted using a pipeline of information retrieval (IR) and natural language processing (NLP) algorithms. Once the documents have been annotated, the user can configure the settings to rank the documents by (de)prioritizing linguistic forms to ensure the retrieval of linguistically rich reading material.

In the next section, we discuss the implementation of the input enrichment component of the *FLAIR* system introduced in Chapter 3. Originally designed as a web search tool, it can be used equally well to search through *Project Gutenberg*[3], hand-curated text repositories for children, such as Time for Kids[4], OneStopEnglish[5], or any other manually created and uploaded text corpora.

5.3 Implementation of Automatic Input Enrichment Systems

5.3.1 FLAIR Components

The original *FLAIR* pipeline can be broadly reduced to four primary operations – web search, text crawling, parsing and ranking. As demonstrated by the diagram in Figure 5.2, the first three operations are delegated to the server, as they require

[2] `http://www.reuters.com`
[3] `https://www.gutenberg.org`
[4] `http://www.timeforkids.com/news`
[5] `http://onestopenglish.com`

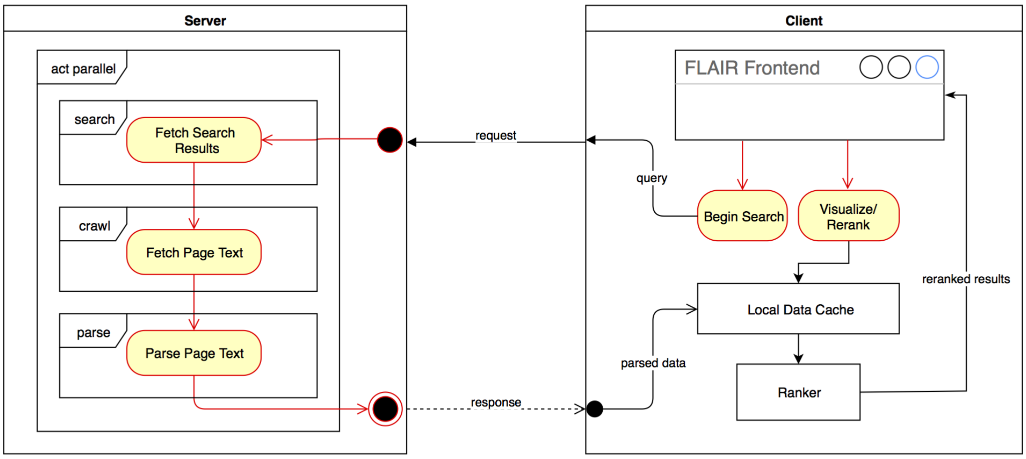


FIGURE 5.2: FLAIR architecture.

the most resources. Ranking, however, is performed locally on the client endpoint to reduce latency. The following sections cover the main design decisions made and challenges faced when implementing each of the four modules.

Web Crawler The implementation of the *FLAIR* web search module relies on the Microsoft Bing API. Thus, some of the *FLAIR* functionality is built upon the search options provided by Bing, such as searching web pages in a particular language or searching only selected websites and categories. Given the increasingly large number of web pages retrieved by Bing for every search, a cut-off value must be specified to limit the document space for further re-ranking. *FLAIR* offers the retrieval of 10 to 50 web search results. Alternatively, the user can upload their own collection of texts of reasonable size. Once the user has specified the search query and the number of results, the web crawler module uses the Bing API to retrieve search results for a given query. If a document is not on our manually-created black list and its URL is not identical to another already retrieved document, it is stored in memory for further processing by the text extractor module.

Text Extractor After the specified number of results for a query have been retrieved, the Boilerpipe library by Kohlschütter et al. (2010) is utilized for text extraction. This library provides several algorithms for the extraction of the main textual content from different types of web pages. We tested DefaultExtractor, ArticleExtractor and LargestContentExtractor on a test collection of 50 documents. The results indicate that DefaultExtractor is the best choice for this task, with the other two options extracting too little text in some cases when the main content was divided into several parts. It is worth mentioning that ArticleExtractor should be a better choice for searching news articles and is indeed reported to have higher accuracy in this scenario. *FLAIR* does not currently differentiate between categories of web documents (e.g., news) but introducing this functionality may improve both the performance of the text extractor module and the user's search experience.

The decision to use DefaultExtractor comes with a trade-off: it sometimes fetches links and advertisements that are accumulated together to form very long sentences. This can pose a problem at the next processing stage of *FLAIR*. To address this issue and facilitate parsing, we add a full stop to the end of each line that does not end with a punctuation mark. While decreasing the parsing speed, this has the drawback of creating short sentences that are later identified as *simple sentences* or *incomplete sentences* by the parser, and should be taken into account when interpreting text features.

Parser The parser module employs Stanford CoreNLP (Manning et al., 2014) to identify numerous linguistic forms using the syntactic category and dependency information obtained from it. Long sentences are quite frequent in web texts, so we employed the Stanford shift-reduce parser, which is less sensitive to sentence length. This parser has also been reported to outperform the older Stanford constituency parsers.[6] After obtaining the output from the parser, we use a detection algorithm to identify 87 linguistic forms listed in an official curriculum for the English language. The detection of linguistic forms and their senses is described in detail in Chapter 4, and the full list of the implemented grammatical constructions and corresponding detection methods can be found in Appendix B.

Ranker The ranker module is responsible for re-ranking the retrieved documents on the basis of the statistical analysis of the data received from the previous modules. When making a decision about the parameters of the ranking model, we adhered to the common IR practice of assuming that high-ranked documents should balance the occurrences of all the items in the search query. That is, the most relevant document would ideally contain the same number of occurrences of all query items. Documents containing all query items but considerably more

[6] `http://nlp.stanford.edu/software/srparser.shtml`

instances of one than the others would be ranked lower. Finally, the documents containing only one item, even if the number of occurrences is higher than in any other document, would be considered the least relevant. The types of ranking features as well as the ranking algorithm itself are discussed further on in the section.

5.3.2 Ranking Features

Ranking features in CL retrieval tasks range from standard statistical retrieval measures, such as term frequency or document frequency, as discussed above, to those driven by linguistics and NLP, e.g., the number of advanced conjunctions or verb phrases per sentence. In our approach, we differentiate among three groups of features, namely, *static*, *dynamic*, and *FLAIR query* features, which are taken into consideration either simultaneously or separately depending on the end user of the tool. It is important to note that since we work with the results already retrieved from a commercial web search engine given some search term(s), a *FLAIR query* does not consist of words but is a set of grammatical constructions specified by the user to get a better ranking based on their learning or pedagogical needs.

Static features depend only on the document, not on the query. They include document position in the initial ranking, document length and document readability score. These features are precomputed during indexing. *Dynamic* features depend both on the contents of the document and the query. In *FLAIR*, they are represented by the term frequency (tf) and document frequency (*df*) of the target constructions. The BM25 algorithm discussed further in this section incorporates both tf, *idf* as well as *relative document length*. Finally, *FLAIR query* features only depend on the query. When formulating a *FLAIR* query, i.e., selecting the grammatical constructions for re-ranking, a user can assign a weight to each construction. These weights then become a significant factor in the overall re-ranking.

While characteristics such as document length, position in the initial ranking, and term weight are straightforward to calculate, *readability score* and *tf-idf* require additional processing. In the current version of *FLAIR*, we focus on the retrieval of grammatical constructions and provide the user with an approximation of the readability level of each retrieved document by using simple surface features, such as word and sentence length.

5.3.3 FLAIR Ranking Algorithm

Our system utilizes the IR bag-of-words model, which is in fact a *bag of linguistic constructions* in the *FLAIR* context, where each document in the collection is represented as a set of grammatical constructions. The user can then provide a *FLAIR query* that contains the weights corresponding to how much preference should be given to each construction. Intuitively, the documents containing a larger number of occurrences of the specified construction should be ranked higher.

For this, the *term frequency (tf)* measure is used, which is simply the number of occurrences of a grammatical construct in a document. However, if several constructions with different frequency distributions are specified, e.g., frequent *definite articles* and infrequent *phrasal verbs*, they will still get the same weights given the identical user configuration. The more frequent constructions will be rewarded more than the less frequent one, which is not desirable. This problem is solved by making use of the *inverse document frequency (idf)* measure, which rewards constructions that are infrequent across the whole collection. *Document length* is another parameter that is introduced into all transformations of tf-*idf* to ensure a balance between short and long documents in the set of retrieved results.

BM25 (Robertson and Walker, 1994) is an example of such a transformation, and has proved to be highly successful in traditional IR systems. An important advantage of BM25 is the fact that it allows for units of any length and helps to avoid the dominance of a single term over the others, which is particularly useful in the input enrichment context. There are two free parameters k and b in BM25, which control the upper bound of tf and the document length normalization, respectively. Although evidently useful, the optimal combination of their values has been found difficult to predict for different text collections (Zobel and Moffat, 1998). In the book *Introduction to Information Retrieval*, Manning et al. (2008) suggest that $k \in [1.2; 2.0]$ and $b = 0.75$. We experimented with these values on different collections of documents and decided to use $k = 1.7$ and $b = 0$ as the default values in the end. In addition, we used b to extend the functionality of the tool by giving the user control over the b parameter, which can take values from 0 to 1. When it is set to zero, no length normalization is carried out whatsoever, so longer documents are ranked higher due to producing larger (non-normalized) tfs. The bigger b is, the more long documents are penalized.

Finally, the idea of filtering out *stop words*, short and extremely frequent units, is accepted by many standard IR systems because it saves processing power when constructing the *idf*. But how well does this apply to our bag-of-constructions approach? Following the frequency logic, candidate *stop constructions* could include articles, prepositions or personal pronouns – these constructions usually appear in every document in the collection. But, as opposed to a standard search query, the constructions included in a *FLAIR query* can contain those on the stop list. Indeed, it is easy to imagine a user seeking out texts with as many articles as possible: either a teacher wanting to refresh their students' memory about article usage or a researcher setting up an experiment to test the impact of visual input enhancement on the acquisition of articles (Ziegler et al., 2017). However, if articles are on the stop list, the user's configuration will not change the order of the documents and will not result in the most appropriate ranking. Consequently, this standard IR technique might not be as beneficial for grammar retrieval.

Thus, the *grammatical score* of each document is an instantiation of BM25 and is calculated as follows:

$$G(q,d) = \sum_{t \in q \cap d} \frac{(k+1) \times tf_{t,d}}{tf_{t,d} + k \times (1 - b + b \times \frac{|d|}{avdl})} \times log \frac{N+1}{df_t} \tag{5.1}$$

where q is a *FLAIR* query containing one or more linguistic forms, t is a linguistic form, d is a document, $tf_{t,d}$ is the number of occurrences of t in d, $|d|$ is document length, *avdl* is the average document length in the collection, and k and b are free parameters set by default to 1.7 and 0, respectively.

To demonstrate the advantages of this approach over simple *tf-idf* ranking, let us assume we have used a web search engine to search for *the 2015 Pulitzer Prize* and retrieved the top six documents for further re-ranking. We then configured the settings by assigning the highest weights to two grammatical constructions, *the definite article* and *phrasal verbs*. The order in which the documents are represented in (24a) – (24f) corresponds to their *FLAIR* ranking using the grammatical score presented above. Table 5.1 contains the scores for three ranking functions: tf (normalized by document length), tf-*idf* (normalized and smoothed) and BM25 (k=1.7; b=0). In each column, the score of the document ranked first is in bold.

(24) a. ... they **put out** an annual ranking of the most dangerous states for women based on the rate of women killed by men. South Carolina **came out** at ...

b. ... forming a coalition with Labour, however, Miliband did **come out** himself to **rule out** the ...

c. ... The drama award, ... according to the guidelines. The play **beat out** “Marjorie Prime," by Jordan Harrison ...

d. ... the LA Times won the Writing Prize for the stories of how the whole draught in the USA had influence on ...

e. ... exploration of the impact of human behavior on the natural world. David I. Kertzer’s “The Pope and Mussolini: The Secret History of Pius XI ...

f. ... at this point is that the outcome of the presidential race will likely determine control of the Senate ...

Several observations can be made when comparing the scores in the table, the most obvious one being the highest-ranked document. Because the simple *tf* algorithm assigned identical weights to both constructions, its final score is simply the ratio of the total number of constructions in the document to the document length. The *tf-idf* algorithm took the document frequency of each construction into consideration as well and rewarded a shorter document with a larger number of occurrences of the infrequent construction. More accurately, it did not reward shorter documents but rather penalized longer ones, which is clear from the scores for documents (24a) and (24e). Finally, BM25 refrained from over-penalizing longer documents and penalized the extremely high *df* of *the definite article* instead, thus rewarding the document with a balanced number of occurrences of the two constructions.

5.3.4 Other Potentially Useful Features

When using the web search module of *FLAIR*, the user provides their information need or current interest, and it is the task of a standard web search engine to rank the most relevant results higher. Intuitively, the position of a document in the initial ranking can be a good estimate of how well it satisfies the user’s needs. However, the results of our online study with English teachers discussed in Section 5.4 showed that there was no significant correlation between the original

d	|d|	TF score	TF-IDF score	BM25
24a	27	0.15	0.26	**5.1**
24b	15	0.2	**0.39**	4.57
24c	16	**0.25**	0.36	4.34
24d	21	0.24	0.28	2.35
24e	24	0.16	0.2	2.21
24f	18	0.16	0.2	2.01

TABLE 5.1: Comparison of the *tf*, *tf-idf* and *BM25* weighting for grammar retrieval given a collection of six documents.

rank of the result and its content rating (Pearson's $r = .1, p = .27$). This implies that a ranking algorithm may not profit from a feature representing the original rank of the document before re-ranking, although more studies are needed to confirm this. Nevertheless, other measures of content relevance explored in the NLP and IR research, such as the semantic overlap between the search query and a document, may be beneficial for prioritizing topically relevant documents.

The background information necessary for understanding a text, or prior knowledge of the topic, has proved to be a strong predictor of the reader's comprehension of a text (Kendeou and Van Den Broek, 2007; Ozuru et al., 2009). This can be automatically approximated using named entity recognition, a subtask of NLP with robust state-of-the-art systems producing output with a high F-measure of over 90% (Marsh and Perzanowski, 1998). The count of the *person*, *organization* and *place* tags that appear infrequently in the document collection and are not followed by an apposition can be used to calculate the amount of background knowledge the writer expects their readers to have. Consider the sentences (25a) and (25b):

(25) a. *Joe Kelly* and *Matt Barnes* each pitched a perfect inning of relief for *the Red Sox* to complete the shutout.

b. *Erdogan*, 64, the most popular – yet divisive – leader in modern *Turkish* history, told jubilant, flag-waving supporters there would be no retreat from his drive to transform *Turkey*, a *NATO* member and, at least nominally, a candidate to join *the European Union*.

The first excerpt requires the reader to have the knowledge about three named entities – Joe Kelly, Matt Barnes, the Red Sox – because no additional information is provided about any of them in the immediate context. At the same time, although the second sentence contains more named entities, the writer elaborates on two (or three) of them by using appositions, thus making it easier for a lay reader to understand it. This observation is in line with the feedback from the English teachers who took part in our online study discussed in the next section: When selecting reading materials for class, they discarded texts with a high number of unknown names that were not familiar to their students.

5.4 Online Study with English Teachers

In order to assess teachers' experience and satisfaction with automatic input enrichment as a method for retrieving topically relevant and linguistically rich texts, the current study focuses on teachers as the conduit between students and reading materials. The results should inform the computational linguistic and language teaching communities about the characteristics of appropriate reading materials for language learners and the use of automatic input enrichment to retrieve such material.

The *research questions* of the study address the importance of content and linguistic form as well as teachers' attitudes towards their optimal balance: Does automatic input enrichment succeed in giving teachers the material that:

- is enriched with the linguistic forms relevant in the FLTL context,
- is in line with the information need expressed via a search query, and
- is suitable as a reading assignment for their students?

We designed an online study to operationalize these research questions. In the study, English teachers compared news articles retrieved by the standard web search engine Microsoft Bing to those provided by the automatic input enrichment system *FLAIR*. The following *hypotheses* guided the design and content of our study:

H1: Teachers prefer texts provided by *FLAIR* over those provided by Bing when choosing a reading assignment for their students.

H2: Texts provided by *FLAIR* are perceived to be less relevant to the topic than those provided by Bing.

H3: The more infrequent the target linguistic forms are, the more teachers prefer texts provided by *FLAIR* over those provided by Bing.

5.4.1 Design

In order to address the aforementioned hypotheses, we designed an online study where the participants were asked to rate and compare pairs of news articles based on (i) their relevance to a given topic and (ii) the representation of given linguistic forms in them. One of the news articles was the top search result from a standard search engine, while the other was a search result prioritized by *FLAIR* after specifying the target linguistic forms. We opted for a repeated-measures within-subjects design and ensured a random order presentation of the news articles retrieved from Bing and *FLAIR* as well as a random combination of topics and pairs of linguistic forms in the main task. The study proceeded as follows:

Procedure Participants received a message with the link to the online study and were asked to carefully read the information for participants and the consent form before registering. Upon registration, they filled out a short questionnaire asking for their age, gender, native language(s), English language proficiency, highest degree in teaching, and the proficiency level(s) of their students. They were also asked whether they used web search to look for reading materials for their classes. Once they submitted their answers to the questionnaire, they were able to read the detailed instructions, which were displayed on every login.

The flow of the main task is displayed in Figure 5.3: Participants were presented with a topic and a pair of target linguistic forms. They read and rated each of the two provided news articles by answering two questions: 1) How relevant is the article to the topic? 2) How rich is the representation of the two target linguistic forms in the article? Answers to both questions were submitted on a five-point Likert scale. Finally, participants were asked to pick one article as a reading assignment for their students with a preference scale from *Definitely Text 1* to *Definitely Text 2*.

After completing the ten topics, participants filled out a debriefing questionnaire where they explained their general strategies for answering each of the questions in the main task (e.g., *How did you decide on the relevance of an article to a given topic?*). Finally, they submitted their email address and received a 20 Euro voucher as reimbursement.

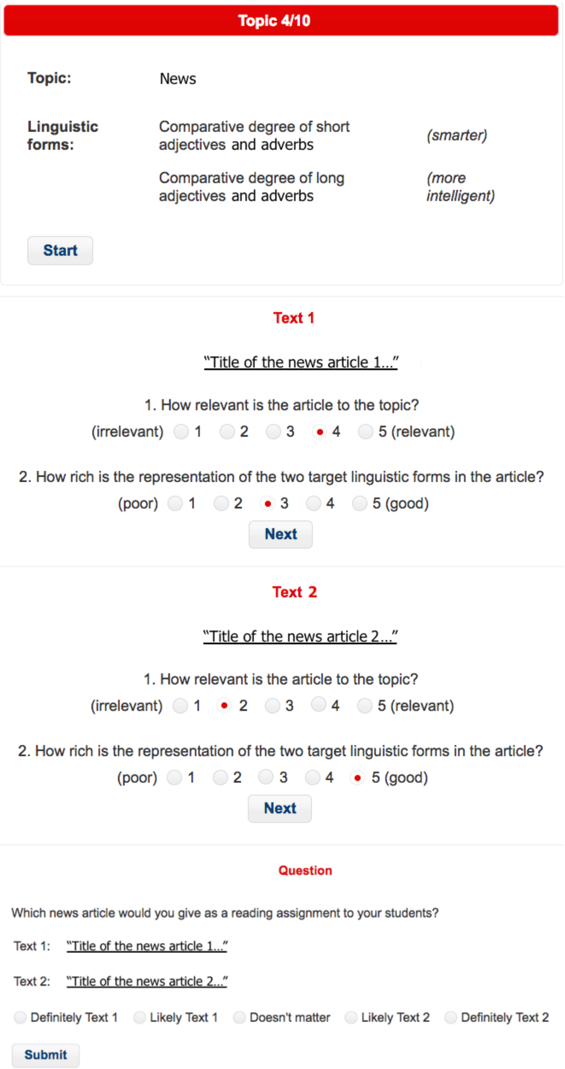
Implementation We implemented the online study as a Java J2EE web application. To ensure anonymity, personal information on users obtained from the questionnaire was stored separately from their responses. Upon registration, each user was assigned a list of ten topics in a random order. Each topic was matched with one of the three types of linguistic forms (see Section 5.4.2 below), one news article provided by *FLAIR* and one news article retrieved by Bing. For each topic, the two articles were displayed in a random order, and participants could not change their rating of the first news article once the second was displayed.

5.4.2 Data and Participants

A total of 60 news articles were used in the study. The texts were presented in pairs concerning shared the same topic and the same pair of target linguistic forms (e.g., *the present simple* and *present progressive tenses*). One article in each pair was obtained by submitting a search query to the web search engine Microsoft Bing and selecting the top search result. The other article in each pair was obtained by submitting the same query to *FLAIR*, configuring the settings to prioritize texts with the two target linguistic forms and selecting the top search result from the reordered list. As *FLAIR* relies on Microsoft Bing for retrieving the original search results, the only variable that differed between the two conditions was the automatic input enrichment component implemented in *FLAIR*.

Linguistic forms For the current study, we selected three pairs of linguistic forms (frequent, mixed, and infrequent) based on their document co-occurrence frequency in a corpus of 2400 news articles. Table 5.2 provides the distribution of their mean relative term frequencies across the texts provided by Bing and *FLAIR*.

The *frequent* pair consisted of regular (e.g., *typed*) and irregular (e.g., *wrote* – *written*) verb forms. It had a high document co-occurrence frequency of 95%. This



Topic 4/10

Topic: News

Linguistic forms: Comparative degree of short adjectives and adverbs *(smarter)*

Comparative degree of long adjectives and adverbs *(more intelligent)*

Start

Text 1

"Title of the news article 1..."

1. How relevant is the article to the topic?

(irrelevant) ○ 1 ○ 2 ○ 3 ● 4 ○ 5 (relevant)

2. How rich is the representation of the two target linguistic forms in the article?

(poor) ○ 1 ○ 2 ● 3 ○ 4 ○ 5 (good)

Next

Text 2

"Title of the news article 2..."

1. How relevant is the article to the topic?

(irrelevant) ○ 1 ● 2 ○ 3 ○ 4 ○ 5 (relevant)

2. How rich is the representation of the two target linguistic forms in the article?

(poor) ○ 1 ○ 2 ○ 3 ○ 4 ● 5 (good)

Next

Question

Which news article would you give as a reading assignment to your students?

Text 1: "Title of the news article 1..."

Text 2: "Title of the news article 2..."

○ Definitely Text 1 ○ Likely Text 1 ○ Doesn't matter ○ Likely Text 2 ○ Definitely Text 2

Submit

FIGURE 5.3: The main task included reading and rating two news articles and selecting one of them as a reading assignment for class.

means that these two linguistic forms occur together in 95 out of 100 documents on average. Both constructions are also highly frequent: in the texts chosen for our study, regular and irregular verbs both had an average relative term frequency of 0.016. We did not count those forms when they occurred in modifier positions (e.g., *is interested*, *colored balloons*).

The *mixed* pair of linguistic forms consisted of two grammatical tenses, present simple (e.g., *Kate plays guitar*) and present progressive (e.g., *Kate is playing guitar now*). Their relative term frequencies in the study were 0.012 and 0.003, respectively, with a document co-occurrence frequency of 50%. Predicates containing modal verbs were not counted as the present simple tense (e.g., *He can swim*), with the exception of the verbs *have to*, *need*, and *want*. When a form constituted part of a conditional sentence, it was not counted either (e.g., *I will not go out if it is still raining*).

The *infrequent* pair consisted of the comparative degree of short adjectives and adverbs (e.g., *nicer*) and that of long adjectives and adverbs (e.g., *more beautiful*). In addition to only co-occurring in 4% of documents, these linguistic forms had low term frequencies of 0.002 and 0.001. When the comparative form *more* occurred as part of a longer form (e.g., *more intelligent*), the whole expression was counted as a long form, and *more* was not additionally counted as a short form.

Texts Using Microsoft Bing, we did a web search for Reuters news articles by expanding the search query with *site:reuters.com.* The following ten topics popular on Bing at the time served as search queries: Game of Thrones, health care, street artists, Roger's Cup 2017, SpaceX, electric cars, Bitcoin, Venezuela coup, Brexit, opioid epidemic. The top result for each topic was stored in our database as a Bing result, and the top 20 results were used for further reordering – in line with Lewandowski (2008), who retrieved the top 20 results per query to evaluate the relative performance of major commercial web search engines. This decision was also partly based on the results of several case studies demonstrating that users only look at the top 10-20 results retrieved by a web search engine.

For each topic, we repeatedly configured the *FLAIR* settings to prioritize texts containing each of the three pairs of linguistic forms presented above and stored the three top hits as *FLAIR* results. In the end, we had three pairs of news articles per topic: One was the top web search result from Bing, while the other was the top result from *FLAIR*. The two texts for a given topic and a given pair of linguistic forms were of comparable length (the difference was at most 50% of the shortest article) and at the same or adjacent readability levels calculated using a simple Automated Readability Index (Senter and Smith, 1967).

Participants Twelve English teachers working with upper-intermediate and advanced learners of English in Germany were recruited through university and social media channels. Each participant was reimbursed with a 20 Euro voucher, and all responses (n = 240) were anonymized. The participants' ages ranged from 25 to 59 years old, and 91% of them were women. The first language of the majority of the participants was German (75%) followed by English (8%), French (8%),

		Bing	**FLAIR**
frequent (95%)	**regular verbs**	0.012	0.020
	irregular verbs	0.012	0.019
mixed (50%)	**present simple**	0.011	0.014
	present progressive	0.001	0.005
infrequent (4%)	**comparative d. of short adj. and adv.**	0.001	0.003
	comparative d. of long adj. and adv.	0	0.001

Table 5.2: Mean relative term frequencies of three pairs of linguistic forms across the texts provided by Bing and *FLAIR*.

and Spanish (8%). All participants had an advanced level of English proficiency and a degree in teaching English. They worked at a secondary school (50%), a high school (42%), or a university (8%). The majority (75%) specified that they currently used web search to look for reading materials for their students, and 25% said they sometimes used web search for this purpose.

5.4.3 Results

All analyses were conducted using R version 3.2.1 (R Development Core Team, 2008). Packages for individual tests and models are specified in the footnotes. We first compared the general preference for *FLAIR* to that for Bing. As previously mentioned and as presented in Figure 5.3, each item consisted of two articles and a final question. This question asked which of the two articles the participant would choose as a reading assignment for their students and their level of certainty in doing so. The option *Doesn't matter* was selected 25% of the time. These responses were not included in the analysis presented below as we were interested in the cases where teachers expressed some preference.

All in all, a chi-square test[7] revealed a significant preference for *FLAIR*: Participants chose it over Bing 71% of the time; $\chi^2(1) = 16.04$, $p < .001$. They were also more confident in choosing *FLAIR*: The answer *Definitely* was selected three times more for *FLAIR* than for Bing; $\chi^2(1) = 12.60$, $p < .001$. Thus, our first hypothesis could be confirmed: Teachers indeed preferred the linguistically enriched texts provided by *FLAIR* over those provided by Bing when choosing a reading assignment for their students.

We conducted two logistic regression analyses[8] to investigate how texts provided by *FLAIR* and Bing compared in terms of (i) their representation of linguistic forms and (ii) the relevance of the content to the topic. In line with the descriptive statistics in Table 5.2, the logistic regression models showed that *FLAIR* ($M = 3.22$, $SD = 1.07$) was significantly more likely to be rated higher in terms of representation of linguistic forms than Bing ($M = 2.51$, $SD = 1.15$); $b = 1.89$, $SE = 0.51$, $p < .001$. Moreover, texts provided by *FLAIR* ($M = 3.67$, $SD = 1.08$) were perceived to be slightly more relevant to the topic than those provided by Bing ($M = 3.58$, $SD = 1.00$), although the difference failed to reach statistical significance; $b = 0.53$, $SE = 0.74$, $p = .470$.

In order to test whether the absence of statistical significance was due to chance or whether the texts provided by *FLAIR* and Bing were indeed comparable with regard to content, we conducted two one-sided tests of equivalence (Schuirmann, 1987).[9] The results were statistically significant, $t_1 = 4.55$, $t_2 = -3.19$, $p_1 < .001$, $p_2 < .001$, 90% CI $[-0.13; 0.31]$, so we could confirm that the samples were equivalent with a medium effect size of 0.5 and an alpha level of .05.

Finally, we used a two-way repeated-measures analysis of variance[10] to test whether the preference for *FLAIR* depended on the type of linguistic form. We hypothesized that the more infrequent the target linguistic forms were, the more teachers would prefer texts provided by *FLAIR*. The first factor was the preference for *FLAIR* (a five-point scale), and the second factor was the type of linguistic forms (frequent, mixed, or infrequent). The ANOVA did not show the tendency that we expected; $F(2, 90) = 0.87, p = .419$; so we inspected the means of all three groups and performed paired samples t-tests.

[7] R native stats package, method *chisq.test()*

[8] R native stats package, method *glm()*

[9] R package *TOSTER*, method *TOSTtwo()*

[10] R native stats package, method *aov()*

The biggest mean preference for *FLAIR* was found for the mixed pair of linguistic forms (present simple and present progressive; $M = 3.92$, $SD = 1.99$), followed by the infrequent group (comparative degree of short adjectives and adverbs; $M = 3.69$, $SD = 1.30$) and the frequent one (regular and irregular verbs; $M = 3.46$, $SD = 1.39$). When we turned the five-point scale into a binary outcome variable (i.e., either selecting *FLAIR* as a reading assignment or not) and calculated the percentage of responses favoring *FLAIR*, we found 76% of positive responses in the infrequent group, 75% of responses in the mixed group, and 65% in the frequent one.

As the data for the three groups were not normally distributed (Shapiro-Wilk's normality test[11] yielded significant differences from a normal distribution), we opted for paired two-samples Wilcoxon tests.[12] The paired tests revealed that there was no significant difference between the groups with regard to preference for *FLAIR*: infrequent and mixed groups, $Z = 128$, $p = .352$; mixed and frequent groups, $Z = 157$, $p = .643$; infrequent and frequent groups, $Z = 217$, $p = .727$.

5.4.4 Discussion

English teachers demonstrated an overall preference for *FLAIR* over a standard web search engine when choosing a reading assignment for their students. This is in line with our first hypothesis and a strong argument in support of automatic input enrichment tools for language teachers.

Feedback from teachers suggested that the relevance of the article to the topic and the content of the article were the decisive factors in choosing one article over the other as a reading assignment. We were therefore particularly interested in whether there was a trade-off between the content and the representation of linguistic forms in the articles, because a large number of the news articles retrieved by *FLAIR* (40%) were not among the top ten original search results. Thus, we hypothesized that the texts retrieved by *FLAIR* would be rated as less relevant to the topic but have a richer representation of linguistic forms.

As the number of occurrences of the given linguistic forms in the texts retrieved by *FLAIR* was higher (see Table 5.2), this indeed resulted in significantly higher

[11] R package *dplyr*, method *shapiro.test()*

[12] R native stats package, method *wilcox.test()*

teachers' ratings for the representation of linguistic forms. However, counter to our expectations, the texts provided by *FLAIR* were neither inferior nor superior to those originally retrieved by Bing in terms of content: They were rated slightly, but not significantly, more relevant to the given topic. This suggests that the most appropriate texts for language learners may not appear within the top web search results, and those texts that are not ranked high by standard web search engines can have higher linguistic and pedagogical potential than the top hits.

As the study showed, automatic input enrichment is particularly beneficial for retrieving texts containing target linguistic forms of lower frequency levels, although the differences were non-significant. This can be explained by document and term frequencies: The high term and document frequencies of frequent linguistic forms make it likely that every retrieved text contains at least several instances of each form. In this case, the texts prioritized by an automatic input enrichment system may not differ from the original top hits with regard to their linguistic characteristics. Other frequently co-occurring pairs of linguistic forms relevant for language teaching are, for example, adjectives and adverbs (co-occurring in 97% of documents), the definite and indefinite articles (96%), present simple and past simple (93%), and *to* infinitives and *ing* verb forms (90%). In the next section, we propose a way to improve the functionality of automatic input enrichment systems targeting frequent linguistic forms.

Infrequent linguistic forms, on the contrary, appear together in few texts and have a small number of occurrences within each text. The advantage of automatic input enrichment in this case is that it can detect the few texts that contain the target infrequent linguistic forms. Other pairs of linguistic forms with low document co-occurrence frequencies as well as low term frequencies are, for example, the modal verbs *can* and *may* (14%), past perfect and past progressive (12%), future simple and *going to* (9%), wh- questions and yes/no questions (7%), and real and unreal conditionals (4%).

In the case of mixed pairs of linguistic forms (i.e., those consisting of one frequent and one infrequent form), the reordering algorithm pushes the few texts containing the infrequent form to the top. These texts are at the same time also likely to contain several occurrences of the frequent form due to its high term and document frequencies. Other mixed pairs of linguistic forms relevant for teaching English are past simple and present perfect (63%), positive and comparative degrees of short adjectives (58%) and adverbs (45%), present simple and future simple (40%), and past simple and past progressive (30%).

The aforementioned results show that, while relying on a standard web search engine to retrieve the results, automatic input enrichment succeeds in providing the texts that are a) rich in the specified target linguistic forms, b) in line with the information need expressed via a search query or a topic, and c) suitable as a reading assignment. The results also provide insights about which linguistic forms benefit most from automatic input enrichment.

It is important to note that our goal was not to compare automatic input enrichment to web search but to show that the linguistically motivated re-ranking of texts leverages the content and form aspects of the retrieved material. With the abundance of authentic texts available on the internet, such reordering does not prioritize texts of low quality but selects the most linguistically appropriate ones from the pool of relevant texts. This means that systems such as *FLAIR* can rely on standard web search engines for retrieving texts with sound content. In fact, *FLAIR* also allows users to upload their own corpora and prioritize the most appropriate texts from among those that they have preselected. Whether automatic input enrichment systems also provide an effective learning environment for language learners should be tested in further end-to-end empirical studies.

5.5 Challenges and Solutions

The findings of the online study described above, numerous discussions with language teachers, previous theoretical as well as empirical work in second language acquisition (SLA), and our own understanding and practical knowledge of the field helped us to identify the main challenges of designing a system providing automatic input enrichment.

5.5.1 The Web as Corpus for FLTL

When approaching the implementation of an input enrichment system for FLTL as a web search task, two types of challenges arise. The first concerns the appropriateness of the Web as a corpus in general, while the other concerns its appropriateness as a corpus *for FLTL*.

Is the Web representative? If so, is it representative enough for language teaching and learning? In their introduction to the special issue of the Web as Corpus, Kilgarriff and Grefenstette (2003) argue that the Web is as representative as any other corpus, with its own characteristics and limitations that need to be explicitly stated and discussed. Although they acknowledge several constraints of web search engines for language researchers, the authors also call the Web "a fabulous linguists' playground". However, whether the Web is an appropriate playground for language learners as well is a more complicated question. One difference between language researchers and learners is that the latter do not necessarily benefit from exposure to all the varieties of language (at least not at lower levels of proficiency). The solution we propose is in line with the position taken by Kilgarriff and Grefenstette (2003) but applied to FLTL: When selecting the Web as a source of additional reading materials for class, teachers should introduce it to learners and explain its peculiarities. Different genres of text, different language dialects and varieties, reliable and unreliable sources of information, blogs, social media language – they all require a special introduction and careful exploration but do not have to be banned from the FLTL classroom.

How erroneous is the language on the Web? Is it too erroneous for language learners? While referring to the Web as a useful source of frequently occurring, authentic, and contextualized linguistic forms, Wu et al. (2009) also point out that it does not necessarily represent exemplary models of language. Indeed, web texts may contain typos, grammatical errors, and unconventional collocations. What can one do to minimize learners' exposure to them?

One scenario involves the user making a mistake or a typo themselves. This has already been addressed by standard web search engines, which have the functionality of correcting misspelled words in a query. A search for *I beleave in you* will automatically yield the results for its corrected version, *I believe in you.* This ensures that the user will not receive any web pages with the erroneous spelling of this word. That being said, web search engines also provide the option of searching for the results of the original query in case the corrected words are neologisms or proper names. However, in another scenario, occasional misspellings and errors may occur in articles, blogs, and social media posts. As parsing is an integral part of any input enrichment system, the integration of an additional spell-checking step at this stage may filter out some unacceptable documents and will not lead to overhead.

Is the vocabulary appropriate (no swear words)? Is it appropriate for FLTL (no slang)? In response to the first concern, standard web search engines have implemented a safe search option that, when activated, hides websites containing inappropriate language from users. While this is useful for parents and language teachers alike, the latter also express a need for stricter filtering options. In particular, the teachers who took part in our online study were concerned about lexical items in the texts that their students may not be familiar with. In line with the approach taken by Wu et al. (2009), who used a word list from the British National Corpus to remove non-words and website names, one can use a pre-compiled word list to ensure that the retrieved texts contain only or at least a certain percentage of known lexical items.

5.5.2 Relative Importance of Content and Forms

What do teachers care about when selecting reading material for class: the content or an appropriate level of vocabulary and grammar? More interestingly, what can foster or hinder learners' language acquisition? Teachers' comments in our online study may provide some insights into the first question.

When the teachers were asked how they selected news articles for a reading assignment, the relevance of the content to the topic was an important factor. Teachers were particularly sensitive to the amount of irrelevant information in the text (including names and tedious details) and looked out for texts that could spark further discussion in class. The majority viewed the content as superior to the representation of linguistic forms. This demonstrates that reading activities are commonly designed to be communicative and serve as a basis for further interactive activities.

On the other hand, the SLA research on processing input for meaning and form reviewed in Section 2.1 provides evidence that reading and grammar can be presented together. In fact, in one study, learners who read for meaning *and* form did better on a reading comprehension test than those who only focused on comprehending the text (Morgan-Short et al., 2012). While more SLA research on this topic will advance our understanding of the relative importance of reading for meaning and form, automatic input enrichment systems can support such research by logging users' activity to identify the characteristics of topically relevant and linguistically appropriate reading material.

5.5.3 Frequent and Infrequent Linguistic Forms

Another potential benefit of the logs obtained from the users of an input enrichment system is in exploring whether different topics and categories of web texts differ in the frequencies of various linguistic forms. To test the viability of this question, we compared 2400 documents retrieved for two types of queries – *people* and *events*. A chi-square test was used to compare the average relative frequencies of linguistic forms in the two categories of documents. The results showed that there were no statistically significant differences between the documents of category *people* compared to those of category *events* ($p > 0.05$ for each linguistic form). However, it would be interesting to compare the distribution of grammatical constructions in a larger document collection retrieved for other types of queries obtained from real learners' search logs. Such analyses can be useful for the teacher to ensure that all learners get the same exposure to target grammatical constructions, no matter what topics they search for.

While there may not be much variation in the distribution of one linguistic form in the texts of the two aforementioned categories, there are certainly some differences in the relative frequencies of individual linguistic forms (see Section 5.4.2 and Figure 4.1). Interestingly, both frequent and infrequent linguistic forms pose challenges for input enrichment. As the results of our online study show, when the target linguistic forms are highly frequent (such as regular and irregular verb forms), language teachers' preference for an input enrichment system is not overwhelmingly higher than that for a standard web search engine. As discussed in Section 5.4.4, this has to do with the fact that frequent forms are very likely to be richly represented in any text. But how can input enrichment systems maximize their usefulness and effectiveness in this case?

The first solution is counting types instead of tokens or at least offering this option to the user. A good example of a linguistic form where this change will make a difference is the category of irregular plural nouns, such as *people*, *children*, *women*, and *men*. Although irregular nouns are an exception to the rule of adding the inflection *-s* to singular nouns, they are treated as a frequent linguistic form because the aforementioned four words are highly frequent across web documents. Counting each unique word only once will reduce the term frequency of this linguistic form, but increase its weight in the *FLAIR* ranking algorithm because it prioritizes infrequent constructions. For grammatical constructions, such as tenses, this solution can be complemented with the disambiguation of tense senses, as discussed in Section 4.2 and further in this section.

Some infrequent forms, some of them have an extremely low document frequency as well as a low term frequency. This means that even when *FLAIR* finds a few documents containing the construction, their number of occurrences within each document may not be enough for language practice. Simply retrieving more documents prior to re-ranking may not solve the issue in this case. As discussed above, the assumption that a certain category of texts will have a richer representation of an infrequent form has not been empirically proved, either. We thus propose a more fine-grained classification of texts that is tailored to every infrequent linguistic form.

For instance, one can expect Frequently Asked Questions (FAQs) and interviews to contain a higher number of questions than news reports. When looking for imperatives, one can consider nowadays ubiquitous How-To articles, cooking recipes, and user manuals. This solution can be put to practice using an ad-hoc IR method of query expansion (QE). For instance, when the user specifies *questions* as a target construction, the system will automatically expand the query with the term *FAQ* or *interview* before sending a request to a standard web search engine, thus ensuring a higher number of occurrences of the target construction in the top results that are to be further re-ranked by the tool. Currently, we assume that the user will first type in a search query, get the required number of search results, and only then configure the grammatical settings by selecting the target linguistic forms. However, another scenario is possible: The system can first ask the user to select the target forms and suggest the topics and categories of texts that are most likely to contain it. Finally, QE can also ensure that a web search engine classifies the query sent by *FLAIR* as an informational one. This is needed in order to retrieve web documents containing enough text material as opposed to, for instance, transactional websites. In the current version of *FLAIR*, we use QE for this purpose by adding the word *about* to every query. Importantly, this should not alter the user's informational request and can be seen as the initial step of QE, which can then be followed by any of the techniques described above.

5.5.4 Tense Sense Disambiguation for Input Enrichment

While the detection of forms is the key component of input enrichment, differentiating between the possible interpretations of the same form can result in a richer and more varied linguistic representation. Consider the following two excerpts, (26a) and (26b), targeting the past simple tense:

(26) a. "Of course, the details are incredibly complex and, as in any negotiation, there will be compromises," she said. But she added she was setting out a path to deliver the Brexit people had voted for. "I will need your help and support to get there," she told the Sunday Times.

b. The divisions inside her government over the customs issue were laid bare on Tuesday when Foreign Minister Boris Johnson said proposals for a customs partnership with the European Union after Britain leaves the bloc were "crazy".

Although both excerpts contain the same number of occurrences of the target linguistic form, the first one presents only one function of past simple, namely, reporting something in the past (*said*, *added*, *told*). The second one, on the other hand, presents a wider range of functions, such as a state in the past (*were*) and a passive action in the past (*were laid*). As we have proved that statistical models are capable of detecting different functions of the same tense (see Section 4.2), the integration of such functionality into input enrichment systems will ensure learners' exposure to richer linguistic input. Due to a number of important conceptual and design decisions about its concrete implementation, we leave this for future work.

5.6 Summary

This chapter outlined motivations for input enrichment and presented an online information retrieval system for ensuring effective input enrichment in a real-life teaching setting. An online study showed that the *FLAIR* system succeeds in selecting reading material that (i) is in line with the teacher's pedagogical goal (that is, enriched with target linguistic forms), (ii) offers content of interest to the learner, and (iii) is suitable as a reading assignment.

Systems providing input enrichment can be utilized in a language learning classroom setting or – as shown in Section 5.4 – as a basis for studies on input enrichment, input enhancement, and other SLA methods relying on enriched input. Apart from that, such systems can also support the functionality of state-of-the-art iCALL systems that generate exercises from text, such as Language Muse^{SM} (Burstein et al., 2012). We contribute to this line of research by developing a question generation system, evaluating it, and integrating it into *FLAIR*. Chapter 6 presents the rationale behind the system and different types of questions that can be generated in the FLTL context.

Chapter 6

Automatic Question Generation for FLTL

Parts of the work discussed in this chapter appeared in the following peer-reviewed publications and theses:

1. Chinkina, M., & Meurers, D. (2017). Question Generation for Language Learning: From ensuring texts are read to supporting learning. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications.* Copenhagen, Denmark, pages 334-344.
2. Chinkina, M., Ruiz, S., & Meurers, D. (2017). Automatically generating questions to support the acquisition of particle verbs: evaluating via crowdsourcing. *CALL in a climate of change: adapting to turbulent global conditions–short papers from EUROCALL 2017.*

A growing body of computational linguistic (CL) research supports automatic question generation (QG) as a means of assisting teachers in constructing practice exercises and tests. For example, Heilman (2011) developed a prominent approach to the generation of factual questions suitable for beginner or intermediate students. His goal is to assess the reader's knowledge of the information in the text, which is relevant for both content and language teaching. Other QG systems have been developed to assist students in reading (Mazidi and Nielsen, 2014) and vocabulary learning (Brown et al., 2005; Mostow et al., 2004), or to identify weaknesses in students' knowledge (Cheng et al., 2009). When implemented properly, a question generation system can support language teachers by saving them hours of time or facilitate learners' self-evaluation.

In this chapter, we broaden the perspective on the different functions questions can play in foreign language teaching and learning (FLTL) and discuss how automatic QG can support these different uses. Complementing the focus on comprehension, we highlight the fact that questions can also be used to make learners notice form aspects of the linguistic system and their interpretation. Furthermore, we discuss and illustrate the generation of well-established and novel types of questions, and present the results of a crowdsourcing study showing that the questions automatically generated by our system are comparable to human-written ones. The current standalone implementation of our question generation system is available at `www.purl.org/qg`.

6.1 Questions in FLTL

Text-based questions allow the teacher to not only check reading comprehension, but also notice gaps in the learner's linguistic knowledge. In a FLTL setting, questions can be asked to serve a broad range of different goals:

1. One can ask about the learner's experience or general knowledge: e.g., the question *What do you know about Japan?* serves a purely communicative goal.
2. Comprehension or recall questions can be asked to check whether the learner has understood a text or read it at all.
3. Questions can also be asked with the goal of eliciting a linguistic form from the learner: e.g., the question *What would you do if you won the lottery?* requires the learner to produce conditionals.
4. One can use questions to draw the learner's attention to the linguistic forms used in a text, providing functionally driven input enhancement: e.g., The question *Which happened first: Zarah finished talking or Sabrina said something?* asks about the interpretation of the past perfect tense in the sentence *Sabrina didn't say anything until Zarah had finished talking.*
5. Finally, there are meta-linguistic questions checking the learner's explicit knowledge of the language system: e.g., *From which verb is the noun decision derived?* or *What is a synonym for staff?*

Our work focuses on several types of questions that have multi-faceted goals: checking the learner's comprehension of the text, drawing their attention to particular linguistic forms in the presented reading material, and eliciting those forms.

6.2 Questions as Functionally Driven Input Enhancement

In line with Ellis (2016) remarks about focus on form, we assume that different kinds of activities are needed to facilitate the acquisition and practice of different linguistic forms. Typical exercises targeting lexical items are multiple-choice questions asking learners to select a synonym for a target word in the text in order to check their understanding of its meaning. Grammatical forms, such as grammatical tenses, also require the learner's understanding of their underlying semantics in context (e.g., which action happened first, is the action already finished) as well as their morphology (e.g., *ed* for the regular verb forms in past simple). In our work, we combine insights from the research on second language acquisition (SLA) and CL to generate text-based questions that help learners create form-meaning connections. Concretely, we propose to generate two novel types of questions creating a functional need to process the target linguistic forms, thus providing *functionally driven input enhancement.*

The first type of questions we generate are content questions about the clause containing the target form. These are factual questions targeting particular linguistic forms to be acquired and ensuring their increased activation. The goal of these questions is to ensure greater exposure to the forms, so we will refer to them as ***form-exposure questions***. See (27a) for an example of a form-exposure question targeting the present perfect tense expressed by the predicate *has scaled back* in the sentence (27). Form-exposure questions are discussed in more detail in Section 6.3.

(27) The Indian government <u>has scaled back</u> the urgency of its search for a new governor of the Reserve Bank of India.

a. *Form-exposure question*: According to the article, what has the Indian government done? The Indian government _______ the urgency of its search for a new governor of the Reserve Bank of India.

The second type of functionally driven input enhancement is designed to also ensure the correct interpretation of the target form in a given context. For this, the nature of the question that is generated must change from asking about the content of the text to asking about the semantics of the form being targeted. The semantics of linguistic forms like grammatical tenses can be targeted via a combination of comprehension and meta-linguistic questions. While keeping the interaction between the teacher and the learner communicative, they draw the learner's attention to linguistic forms and ensure the understanding of their semantics. In the spirit of Workman's (2008) concept questions, we refer to such questions as ***grammar-concept questions***. See (28a) for an example of a grammar-concept question asking about the interpretation of the present perfect tense form *have cut* in the sentence (28). Grammar-concept questions are discussed in more detail in Section 6.4.

(28) Chinese retailers have cut staff.

a. *Grammar-concept question*: Are Chinese retailers still cutting staff?

In essence, automatically generating questions that target grammatical categories in a text supports incidental focus on form (Loewen, 2005) in a meaning-focused reading task. In the following sections, we discuss the two types of questions that serve this purpose and report on a crowdsourcing evaluation comparing automatically generated and manually written questions targeting phrasal verbs, a challenging linguistic form for learners of English.

6.3 Form-exposure Questions

Form-exposure questions focus on a particular linguistic form, which can either be part of the question or expected in the answer produced by the learner. For example, questions about source text (29) could conceivably address different linguistic targets: relative clauses, past forms of irregular and regular verbs, etc. Both questions (29a) and (29b) target the past simple form expressed by the phrasal verb *brought in* and prompt the learner to produce it. The difference between these two questions is that (29a) is a question about the subject and (29b) is a question about the predicate.

(29) Indeed, Semel and the media executives he brought in by all accounts turned a scrappy young internet startup into a highly profitable company.

a. Who turned a scrappy young internet startup into a highly profitable company? Semel and the media executives he _______.

b. What did Semel do? He _______ media executives.

Form-exposure questions can take the form of a wh-, yes/no, or an alternative question. In our work, we focus on the first type, and as the examples above show, supplement it with a sentence where the target answer is replaced by a gap. This decision is motivated by the complementary nature of wh- questions and gap sentences. While questions are communicative, they may be too general, as can be seen in example (29b). From a computational perspective, *What did Semel do?* is a safe question to generate as it does not contain any adverbial clauses or prepositional phrases, which normally pose challenges for the task of QG (see Section 6.9). However, this question has at least two target answers: *turned a scrappy young internet startup into a highly profitable company* and *brought in media executives.*

A gap sentence can then be added to the question in order to guide the learner to produce the expected target linguistic form by narrowing down the context of the question. In this case, we wanted to target the phrasal verb *brought in*, so we picked the corresponding part of the sentence. The results of a crowdsourcing study reported in Section 6.8 confirm our intuition: Wh- questions are perceived as better-formed and can be answered more easily given the source text when they are followed by a gap sentence.

6.3.1 Generation of Form-Exposure Questions

We generate form-exposure questions about subjects, objects, and predicates. The main linguistic form we focus on is grammatical tense, so our form-exposure questions target verbs and verb phrases. We use the Java implementation of Stanford CoreNLP 3.7.0, a natural language processing toolkit by Manning et al. (2014) for sentence splitting, tokenizing, lemmatizing, constituency and dependency parsing, and resolving coreferences. After extracting a sentence or a clause containing the target form, we perform the following steps: adjust and normalize the auxiliaries, resolve pronouns and other referential expressions, and detect quotation sources, if any.

Unlike the overgenerate-and-rank approach (Heilman, 2011), we first apply a set of constraints to minimize the probability of generating ambiguous, unanswerable, or ungrammatical questions. Concretely, we eliminate sentences with any unresolved pronouns (to avoid questions like *What did he do?* or *What happened to this man?*), sentences expressing hypothetical events or wishes (e.g., *I wish I had*, *if only I knew*), conditionals (e.g., *if there were*, *unless he said so*), imperative sentences with no subject (e.g., *read this*), and sentences containing reported speech (e.g., *they said they knew*). The regular expressions used for detecting these constructions can be found in Appendix C. Such strict filtering leads to fewer generated questions but also a lower percentage of ill-formed ones, which is especially important in the FLTL setting.

Once the unsuitable sentences are filtered out, the algorithm proceeds to detect specific syntactic components of the sentence and modify them if necessary. Finally, transformation rules are used to turn a sentence into a question. Let us inspect the algorithm for generating questions using the example of a simple sentence *Chinese retailers have cut staff*:

Question about Subject (e.g., *Who or what has cut staff?*)

1. Replace the subject with *Who or what* and move to the beginning of the sentence.
2. Detect the grammatical time of the predicate, and if it is *Present*, replace the auxiliary (if any) or the main verb with the form of 3rd person singular (does, has, is).

Question about Object (e.g., *Who or what have Chinese retailers cut?*)

1. Replace the object with *Who or what* and move to the beginning of the sentence.
2. Detect or generate an auxiliary verb.
    - If there is an auxiliary verb modifying the main verb, detect it.

- If there is no auxiliary modifying the main verb, detect the grammatical tense of the main verb, generate an appropriate auxiliary verb, and replace the main verb with its base form.

3. Move the auxiliary verb right after *Who or what.*

4. Remove the rest of the sentence after the verb.

Question about Predicate (e.g., *What have Chinese retailers done?*)

A. Active
(e.g., *What have Chinese retailers done?*)

   1) Insert the question word *What* at the beginning of the sentence.
   2) Identify or generate an auxiliary verb.
      - If there is an auxiliary verb modifying the main verb, identify it.
      - Otherwise, identify the grammatical tense of the main verb and generate an appropriate auxiliary verb.
   3) Move the auxiliary verb to right after *What.*
   4) Identify the grammatical form of the main verb and replace the rest of the sentence with the same form of the verb *do.*

B. Passive
(e.g., *What happened to the staff?*)

   1) Insert the question word *What* at the beginning of the sentence.
   2) Identify the grammatical tense of the main verb and replace the whole predicate with the same form of the verb *happen* (including the auxiliary verb, if any).
   3) Insert the preposition *to* to left of the subject.
   4) Remove the rest of the sentence.

As previously mentioned and demonstrated, in addition to generating questions, we also generate gap sentences (e.g., for phrasal verbs, *Chinese retailers have* _____ *staff.*). As the syntactic components of the sentence have already been extracted to form a question, the generation of a gap sentence boils down to combining those in the same order as they appear in the source sentence and replacing the target linguistic form with a gap.

Form-exposure questions can be used as fill-in-the-blank or multiple choice exercises. In the latter case, one can ensure deeper processing of the target linguistic form by having not the linguistic form itself but a synonym as the solution, for example, and using semantically related words as distractors. While we do not discuss the automatic generation of distractors in this thesis, we propose a novel type of question that ensures that the learner processes the target linguistic form by asking about its interpretation in a given context, which is discussed in the next section.

6.4 Grammar-concept Questions

Questions about grammar can focus the reader's attention on either the form or the meaning of grammatical constructions. In addition to testing the learner's understanding of the text, meaning-driven questions also raise the learner's (meta-)linguistic awareness and help them read and learn the language in a focused way. Rephrasing and form manipulation are one example of such meaning-driven grammar questions. The passive voice, for instance, is normally substituted with the active voice (or vice versa) to have the learner make inferences based on semantics: Given the sentence *The crew paved the highway*, a grammar activity may prompt learners to produce *The highway was paved by the crew.*

Similarly, grammar-concept questions make the learner infer information by isolating defining semantic characteristics of linguistic forms. Once the grammatical concept of a linguistic form is broken down into a series of semantic statements, yes/no or alternative questions can be asked about each of them. Consider the following example by Workman (2008):

Sentence: He *used to* play football.

Concept: *Used to* expresses a discontinued past habit. It highlights the fact that the person does not do this anymore in the present.

Concept questions:

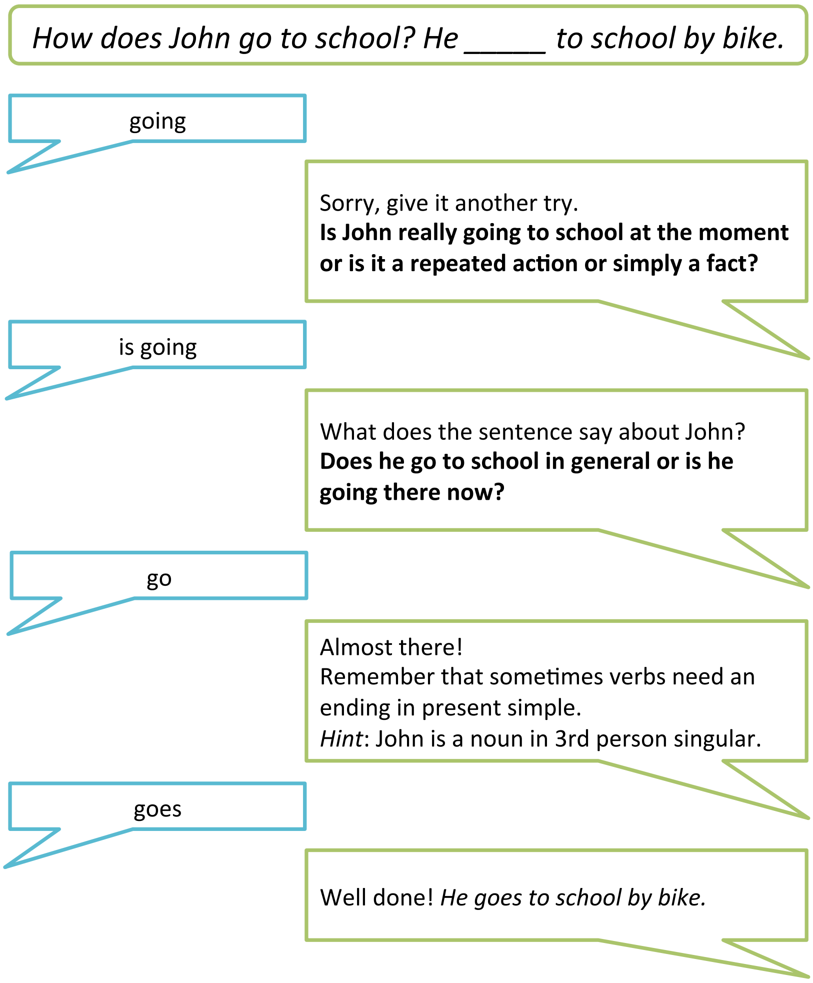
1. Does he play football now? (No)

2. Did he play football in the past? (Yes)

3. Did he play once or often? (Often)

6.4.1 Providing Scaffolding Feedback

One important application of grammar-concept questions in FLTL is scaffolding feedback. Such questions can incrementally guide the learner towards task completion by scaffolding the use of correct forms. Figure 6.1 demonstrates a possible interaction between a learner and a tutoring system providing scaffolding feedback – similar to the approach implemented by Rudzewitz et al. (2017) in their FeedBook system. Grammar-concept questions not only make learners aware of the target linguistic form and its semantics, but also guide them towards producing the form.



How does John go to school? He _____ to school by bike.

going

Sorry, give it another try.
Is John really going to school at the moment or is it a repeated action or simply a fact?

is going

What does the sentence say about John?
Does he go to school in general or is he going there now?

go

Almost there!
Remember that sometimes verbs need an ending in present simple.
Hint: John is a noun in 3rd person singular.

goes

Well done! *He goes to school by bike.*

Figure 6.1: A possible interaction between a language learner and a tutoring system providing scaffolding feedback. Grammar-concept questions are in bold.

6.4.2 Generation of Grammar-concept Questions

Prior to generating grammar-concept questions, we subject each candidate sentence to the same set of constraints discussed in Section 6.3.1 and exemplified in Appendix C. We then use different templates depending on the linguistic form to generate grammar-concept questions. Let us consider the present perfect tense as an example. Its two key characteristics are (i) the finished state of the action and (ii) the irrelevance of the exact time in the past when the action took place. Templates (30a) and (30b) are used to generate grammar-concept questions about each of these aspects. The sentence *Chinese retailers have cut staff* is again taken as an example.

(30) a. `Be-form` `subject` still `verbing` (`particle`) (`dir-obj`) (`indir-obj`)?

e.g., *Are Chinese retailers still cutting staff?*

b. Is it more important when exactly `subject` `verb-past` (`particle`) (`dir-obj`) (`indir-obj`) or that `verbing` (`dir-obj`) (`indir-obj`) took place at all?

e.g., *Is it more important when exactly Chinese retailers cut staff or that cutting staff took place at all?*

The correct answers for each template are known, so they can be hard-coded. As the templates show, a target sentence should always contain a subject and a verb, while the object elements are optional. Importantly, the *particle* element is included in order to be able to target phrasal verbs.

6.5 Motivation for the Studies

> "I don't think it is, but if that's the computer then WOW"
>
> A rater in a crowdsourcing study commenting on one of the automatically generated questions

> "If this was written by a teacher, please stop teaching."
>
> Another rater in a crowdsourcing study commenting on another automatically generated question

For questions to be effective in real-life FLTL, they must be reasonably well-formed and answerable. The performance of QG systems has been assessed either by using automatic measures, such as BLEU (Papineni et al., 2002), or by collecting human judgments. For instance, Zhang and VanLehn (2016) recruited students to judge the comparability of computer-generated, web-crawled and human-written biology questions on several 5-point scales (relevance, fluency, ambiguity, pedagogy, depth). Heilman and Smith (2010) conducted a crowdsourcing study to assess the quality of computer-generated questions on a 5-point scale and used the collected judgments to train a statistical ranker for their QG system.

Crowdsourcing is an attractive option for evaluating QG systems due to its time and cost effectiveness along with its similarity to expert judgments (Snow et al., 2008; Benoit et al., 2016). Using crowdsourcing to compare computer-generated and human-written questions seems like a logical next step in this line of research. Therefore, we conducted two crowdsourcing studies using the Figure-Eight platform[1] to answer the following research questions:

1. Are computer-generated questions comparable to those written by an English teacher in well-formedness and answerability?
2. Are wh- questions followed by a gap sentence perceived better with respect to well-formedness and answerability than open-ended wh- questions?
3. Do wh- questions followed by a gap sentence elicit more correct responses and target phrasal verbs than open-ended wh- questions?

With respect to the first research question, there is no previous research comparing computer-generated and human-written questions via crowdsourcing. However, based on related work comparing the two types of questions offline in a university setting (Zhang and VanLehn, 2016) and research evaluating similar QG systems (Heilman, 2011), we expect that the questions produced by a computer and an English teacher will be comparable in terms of the two aspects under investigation.

Given that the question items we generate are of a novel type (i.e., wh- question followed by a gap sentence), we only have intuitive predictions about the second and the third questions. For the second research question, we predict that a wh-question and a gap sentence may cancel out each other's potential disadvantages,

[1] `www.figure-eight.com`

and thus their combination will be rated higher with respect to both perceived well-formedness and answerability than a wh- question alone. For the third research question, we predict that the addition of a gap sentence will limit the participants' choice of an answer phrase to the phrasal verb given in the text. Thus, the combination of a wh- question and a gap sentence will increase the likelihood of obtaining a correct response and have a higher probability of containing the target phrasal verbs from the source text as part of the answer than simple open-ended wh- questions.

To address the aforementioned questions, we used the QG module of *FLAIR* to generate wh- questions and gap sentences about phrasal verbs in order to draw learners' attention to this form. Phrasal verbs were our linguistic form of choice as they represent a considerable teaching and learning load, as discussed by Garnier and Schmitt (2016). For instance, given the source text (31), our program automatically generated the form-exposure question (31a).

(31) Cancellations "ticked up slightly and unexpectedly" in early April amid press coverage about the coming increases, the Netflix letter said.

a. According to the Netflix letter, what did cancellations do? Cancellations __________ slightly and unexpectedly in early April amid press coverage about the coming increases.

6.6 Using Crowdsourcing to Evaluate QG Systems

Heilman and Smith (2010) demonstrated that data collected via crowdsourcing can be used to successfully train a machine learning algorithm to rate automatically generated questions. The authors argued that a crowdsourcing platform can be a useful evaluation tool for natural language generation, summarization, and translation. Indeed, Callison-Burch (2009) and Zaidan and Callison-Burch (2011) used crowdsourcing to evaluate machine translations and compare the quality of the data acquired from crowd workers to that of professional annotators.

Becker et al. (2012) made use of a crowdsourcing evaluation to select the best candidate gaps when generating gap-fill questions. They controlled the quality of annotations post-hoc by eliminating the judgments from unreliable crowd workers and items with low inter-annotator agreement. While an integral part of any study, quality control is especially crucial when running crowdsourcing tasks. An additional measure implemented by the Figure-Eight platform is test questions, which limit the set of workers to those who continue to satisfy the requirements and make it possible to verify that the workers are paying attention and following the instructions. We discuss such test questions in more detail in Section 6.7.1.

So far, crowdsourcing has only been used to collect judgments about automatically generated wh- and gap-fill questions, not to compare those to human-written ones. In Sections 6.7 and 6.8, we explore whether the questions automatically generated by our system are perceived similarly to manually written questions with regard to their well-formedness and answerability.

6.7 Study on the Quality of Automatically Generated Questions

6.7.1 Design

Data We started with a corpus of 40 news articles and 96 questions designed to test learners' knowledge of phrasal verbs written by Simón Ruiz, an English teacher and SLA researcher. We used the question generation approach introduced in Chapter 4 to generate 69 form-exposure questions about phrasal verbs. To obtain an equal number of questions for the experiment, we randomly selected 69 of the manually created questions. To provide an illustration of the data set, (32a) and (32b) below are instances of well-formed questions by a human and a computer, respectively, asking about the same source text (32). On the other hand, (33a) is a well-formed human-written question, while (33b) is an ill-formed computer-generated question.

(32) Beijing's drive to make the nation a leader in robotics through its "Made in China 2025" initiative launched last year has set off a rush as municipalities up and down the country vie to become China's robotics center.

a. *Human (+):* What has the "Made in China 2025" initiative done since it was launched last year? It has __________ a rush for municipalities to become China's robotics center.

b. *Computer (+):* According to the article, what has Beijing's drive done? Beijing's drive has __________ a rush as municipalities up and down the country vie to become China's robotics center.

(33) Twitter is also working to better define its role in the social media landscape. This week it rolled out a video ad that showed it as the place to go for live news, updates and discussion about current events.

a. *Human (+):* What is Twitter doing to better define its role in the social media landscape? It __________ a video ad this week.

b. *Computer (−):* According to the article, what did this week do? This week __________ a video ad that showed it as the place to go for live news.

Participants While the main advantage of crowdsourcing is that it provides access to a large number of people all around the world, this also carries the risk of recruiting unsuitable contributors (see Stewart et al. (2017) for a recent review on the use of crowdsourcing in behavioral research). For this study, we needed judgments that were as close as possible to those of experts. The following steps helped us achieve this. First, we used the functionality of the crowdsourcing website to select only English-speaking countries, thereby maximizing the probability that the contributors were native speakers of English. However, when we only received one response in the first five hours, we extended the list of participating countries to some European ones where English proficiency is high according to the EF English Proficiency Index (First, 2017): the Netherlands, Denmark, Norway, Sweden, Finland, Germany, and Austria.

We then asked test questions to further filter out unsuitable contributors. In order to proceed to the main task, each contributor first took a quiz in which they had to correctly answer four out of five test questions. The test questions looked exactly like the questions in the main task, except that eight of them were manually edited to either be ungrammatical or unanswerable. To obtain test questions on the clearly grammatical and answerable side of the spectrum, we ran a pilot study and selected sentences that were rated as well-formed and answerable with a high agreement (more than 70%) among the contributors. Four human-written and four computer-generated ones were chosen as good examples of well-formed and answerable test questions. Eight ungrammatical or unanswerable question items were fabricated by editing four of the 27 human-written questions not used in the study and four automatically generated questions to make them either ungrammatical or unanswerable.

Finally, a small number of test questions required the participants to specify whether they were in fact proficient speakers of English and whether their answers were reliable. In this way, we made sure that the contributors understood the task at hand, that they were able to distinguish between a well-formed and an ill-formed question, and that their language skills were advanced enough to answer a question given a source text. In order to perform the main task, participants had to keep their accuracy rate above 70% by correctly answering test questions randomly inserted among the other question items. In total, 364 reliable contributors took part in this study.

Procedure We investigated whether computer-generated questions are on par with human-written ones on the basis of two criteria, well-formedness and answerability; in other words, whether the question is written in acceptable English and whether it can be answered given the information in the source text. As corpus studies suggest that the concept of well-formedness or grammaticality is gradient (Wasow, 2007), we used a five-point Likert scale for both criteria. In addition, we asked the crowd workers whether they thought the question was written by an English teacher or generated automatically by a computer. Concretely, each task presented to the crowd workers consisted of an excerpt from the source news text and a human-written or automatically generated question. The workers were asked to answer four questions and could leave an optional comment:

1. How well-formed is this question item? Is it written in good English? (5-point Likert scale)
2. Can this question item be completed with the information from the source text? (5-point Likert scale)
3. Please, answer this question – in your words, in as few words as possible – based on the information from the source text. (free input)
4. Do you think this question was written by an English teacher or generated by a computer? (binary choice)

6.7.2 Results

We received 1,384 judgments by 364 crowd workers who were classified as reliable, identified as proficient English speakers, and passed the quiz mode with the test questions. The means for the well-formedness scale were 4.53 for human-written questions and 4.40 for computer-generated questions. The means for the answerability scale were 4.44 and 4.47, respectively. We calculated the intra-class correlation (ICC) for the contributors and obtained the values of 0.08 and 0.09 for well-formedness and answerability, respectively. The low contributor ICC ($<$ .1) implies that the contributors provided different ratings for different question items, so we can ignore the dependencies among the observations and did not need a multi-level analysis.

We ran Welch's t-test to find out whether the difference in ratings between computer-generated and human-written questions was statistically significant. On the well-formedness scale, the results turned out to be statistically significant, but the effect size was small: $t(913) = 2.06$, $p = .03$, Cohen's $d = 0.13$. On the answerability scale, the results were non-significant: $t(944) =$ -0.42, $p \geq .1$, Cohen's $d = 0.02$.

However, the absence of evidence does not imply the evidence of absence. To test whether the computer-generated and human-written questions were equivalent in quality (well-formedness and answerability), we used TOST, Schuirmann's two one-sided test (Schuirmann, 1987). The TOST is commonly used in medical research to determine whether one treatment is as effective as another. To prove our alternative hypothesis that computer-generated and human-written questions are comparable in quality, we needed to reject both parts of the null hypothesis:

$H0_1$: Computer-generated questions are inferior in quality to human-written ones.

$H0_2$: Computer-generated questions are superior in quality to human-written ones.

In statistical terms, the null hypothesis is that there is a true effect larger than a Smallest Effect Size of Interest (SESOS) between the two samples (Lakens, 2014). For this task, we opted for an SESOS of 0.5, a medium effect size according to Cohen (1977), and an alpha level of .05 (Lakens, 2017). We used the R package TOSTER[2] to conduct TOST testing for the equivalence of the samples. All results

[2] `https://cran.r-project.org/web/packages/TOSTER/`

were statistically significant on both scales ($p \leq .001$), so we could reject the null hypothesis (for more details, see Table 6.1).

Scale	t_1	t_2	p_1 and p_2	90% CI
well-formed	9.81	-5.68	$\leq$.001	[0.02;0.22]
answerable	7.32	-8.17	$\leq$.001	[-0.13;0.08]

Table 6.1: Results of Schuirmann's TOST for equivalence of computer-generated and human-written questions. Effect size d = 0.5; alpha = 0.05.

The results indicate that any difference between the human-written and computer-generated questions in the ratings for well-formedness and answerability is of an effect size smaller than the SESOS. In line with this finding, the contributors' answers guessing whether a question had been written by an English teacher or generated by a computer were similar for both question classes: 74% of human-written and 67% of computer-generated questions were thought to be written by an English teacher. Our goal at this stage was to identify whether the computer-generated questions can effectively be used on par with manually written questions – which indeed seems to be the case.

6.7.3 Discussion

The results of this first study imply that the questions automatically generated by our system are comparable to those written by a human with respect to well-formedness and answerability, although the questions written by the English teacher were rated as slightly better-formed. Interestingly, most of the well-formed and answerable questions were thought to be written by the English teacher, even if they had, in fact, been generated automatically. This indicates that people do not expect computers to be able to produce coherent output and expect automatically generated questions to be more ungrammatical and unnatural.

6.8 Study on Types of Questions and Answers They Elicit

In the second crowdsourcing study, we wanted to find out (i) whether the addition of a gap sentence to an otherwise open-ended wh- question influences a question rating and (ii) whether wh- questions followed by a gap sentence elicit more phrasal verbs than open-ended wh- questions. The task and the procedure were the same as in the first study but the selection criteria for both data and participants differed.

6.8.1 Design

Data For each source sentence, we generated two types of questions, namely, an open-ended wh- question and a wh- question followed by a gap sentence. As we did not intend to evaluate our system in this study, we excluded all ungrammatical and unanswerable computer-generated questions. Overall, the data consisted of 96 human-written and 96 computer-generated questions. These were randomized in such a way that the two types of questions (with and without a gap sentence) for the same source sentence were never shown together on the same page. We collected five judgments per question item.

Participants For the second study, we selected contributors with a high reliability, as specified on the crowdsourcing page, but did not limit participation based on their level of English. Our assumption was that users must have the necessary English skills to work on an English-language crowdsourcing website. In this study, we collected judgments from 477 contributors.

Procedure As in the first study, participants were asked to answer the presented questions and rate them on two separate five-point Likert scales, one for well-formedness and one for answerability. In this study, we analyzed both the ratings and the responses to the questions. This time, however, we did not ask participants to guess whether the question had been written by a teacher or generated by a computer.

6.8.2 Results

In contrast to the first study, as mentioned earlier, participants in the second study were not selected based on their English proficiency level. Consequently, there was less agreement among subjects on rating questions regarding well-formedness and answerability ($ICC = 0.34$ and 0.37, respectively). Hence, we used mixed-effect models to account for the dependencies across observations.

The analysis was conducted using the lme4 package version 1.1-12 in the R environment version 3.2.1 (R Development Core Team, 2008). We estimated a model for each of the two continuous dependent variables: the perceived well-formedness and answerability of question items. The models included fixed effects for the source of a question item (human or computer) and the item type (with or without a gap sentence) as well as crossed random effects for both participants and items (Baayen et al., 2008). An effect was considered significant if the absolute value of the t statistic was greater than or equal to 2.0 (Gelman and Hill, 2006; Baayen et al., 2008). First, we found that participants did not rate computer-generated questions significantly lower than human-written ones: well-formedness, $b = 0.024, SE = 0.047, t = 0.500$; answerability, $b = 0.065, SE = 0.060, t = 1.080$. These results are in line with those of our first study with proficient English speakers.

The addition of a gap sentence did indeed influence the rating of a question item. The results showed that this had an effect on both the perceived well-formedness, $b = 0.158, SE = 0.054, t = 2.930$, and answerability, $b = 0.127, SE = 0.055, t = 2.300$. In other words, the addition of a gap sentence to a simple open-ended wh- question rendered the question better-formed and more easily answerable.

Finally, we conducted logistic regression analyses (Jaeger, 2008) to investigate which type of questions elicited more correct responses and phrasal verbs. In the first model, the dependent variable was analyzed as a binary outcome (correct vs. incorrect answer). In the second model, the dependent variable was also treated as a binary outcome (presence vs. absence of the phrasal verb from the source text). We selected a random sample of 20% of responses and excluded nonsensical (e.g., “good!”) and non-English (e.g., “konuşma”) answers from the data. Out of 359 answers, 277 contained an exact match to the phrasal verb given in the source text. Only 12 contained rephrasings of phrasal verbs, and the remaining 70 answers were marked as incorrect. As expected, the linear regression results showed that questions followed by a gap sentence had a higher probability of eliciting correct responses, $b = 0.791$, $SE = 0.278$, $p = .004$, and containing the target phrasal verbs from the source text, $b = 2.577$, $SE = 0.484$, $p < .001$, than simple wh- questions.

6.8.3 Discussion

The results of the second study showed that the question ratings were in line with those of the first study: computer-generated and human-written questions were rated similarly regarding well-formedness and answerability. This confirms our hypothesis as well as the findings of previously conducted studies on automatic question generation.

Importantly, we found that wh- questions followed by a gap sentence were rated higher than open-ended ones on both the well-formedness and answerability scales. Our assumption was that a gap sentence can render an otherwise ambiguous question more specific and thus better-formed and easier to answer. Indeed, there seems to be a trade-off between ambiguity and ill-formedness in the case of wh- questions: The more specific a wh- question is, the more syntactic elements it contains, raising the probability of a question being ungrammatical. When the number of syntactic elements is kept to a minimum, there is a risk that a question will be too general or ambiguous. On the other hand, gap sentences are typically grammatical and unambiguous (Becker et al., 2012), but they do not serve a communicative goal. Combining a general wh- question with a more specific gap sentence helps avoid the aforementioned pitfalls of both question types: It maximizes the grammaticality and minimizes the ambiguity of the whole question item while keeping the task communicative.

Finally, wh- questions followed by a gap sentence also elicited more correct responses and more phrasal verbs. Our intuition that the gap narrows the reader's focus to the target linguistic form in the source sentence was confirmed.

6.9 Challenges and Limitations of Question Generation

6.9.1 NLP Challenges

The quality of the automatically generated questions relies greatly on the accuracy of the natural language processing tools that our QG system is built on. In fact, the main causes of ill-formed questions were erroneous coreference resolution (43%) and incorrect parses (28%) of the source sentences (out of the questions with an average rating of below three on a five-point scale). Other factors influencing question quality are discussed and exemplified below.

As previously noted, a question item may not contain enough information to be answered correctly (e.g., a missing restrictive clause) or may be too specific compared to the more general context of the paragraph:

(34) Chinese retailers have also cut staff and seen inventories pile up, luxury sector growth has dried up, and fast-food giants such as KFC-parent Yum Brands Inc and McDonald's Corp are grappling for growth.

a. *Computer:* According to the article, what do inventories do? Inventories _______.

A question item may have superfluous information (e.g., a non-restrictive phrase or a clause) making it too long:

(35) Musk on Wednesday sketched out his vision for an integrated carbon-free energy enterprise offering products and services beyond electric cars and batteries.

a. *Computer:* According to the article, what did Musk on Wednesday do? Musk on Wednesday _______ his vision for an integrated carbon-free energy enterprise offering products and services beyond electric cars and batteries.

While our QG produces a high number of well-formed questions (85%), it also produces ungrammatical or unanswerable questions. We plan to adapt the algorithm to detect errors and minimize the generation of ill-formed questions.

6.9.2 No Questions or Ungrammatical Questions

There is a two-stage process for identifying the main syntactic components, POS- and dependency-based, and both are obligatory for the system to be able to generate a question. If there is an error, a syntactic component may not be detected. For instance, in (36), *Skype* was identified as a verb by the statistical parser. Consequently, no subject was detected, and it was not possible to generate a question.

(36) Skype was snapped up by eBay Inc.

The most challenging case that results in the generation of ungrammatical questions is when the parser incorrectly identifies secondary parts of speech, which does not prevent the system from generating a question. For example, question (37a) was generated from source text (37) below. The erroneous parse tree of the source includes the noun phrase *(NP (VBG meaning) (NNS fans))*, which was then identified as the subject of the sentence.

(37) Internet access in the Communist-ruled island is restricted, meaning fans can not easily look up series and mangas on the web.

a. What can meaning fans not do? Meaning fans can not ______ series and mangas on the web.

6.9.3 Coreference Resolution

Another type of error occurs when the coreference resolution component maps a referring expression to the wrong noun phrase. Given the source sentence (38), the program generated question (38a): in the question, *the manager* is resolved incorrectly as Dean Saunders instead of Chris Coleman.

(38) Former Wales striker Dean Saunders says his country will struggle to hang on to Chris Coleman after their startling run to the Euro 2016 semi-finals and believes the manager could be tempted away soon.

a. According to the article, what could happen to former Wales striker Dean Saunders? Former Wales striker Dean Saunders could be ______ soon.

For questions about subjects and objects, coreference resolution was originally used to determine the question word, *Who* or *What*. However, the error rate was high for rare names that occasionally occur in news articles at the beginning of sentences. Thus, we now combine the two question words into one question phrase *Who or what*. The English teachers we consulted preferred this solution over erroneously generated question words.

To further minimize the effect of errors caused by coreference resolution, we do not substitute the subject of a gap sentence with a pronoun, which often leads to repetition of subject noun phrases. All computer-generated examples in this section demonstrate this limitation. However, importantly, the feedback from the participants in our crowdsourcing studies indicated that questions may be perceived as less well-formed if the subject in the gap sentence repeats the subject in the wh-question.

6.9.4 Tense Sense Disambiguation for Question Generation

For template-based grammar-concept questions, the number of templates is limited to one or two per tense if there is no information about a specific interpretation (or sense) of the target grammatical tense in the given context. For instance, the Cambridge Dictionary gives the following definition of past simple: "Past simple is the form of a verb used to describe an action that happened before the present time and is no longer happening."[3] So, given sentence (39), we can only ask a grammar-concept question similar to (39a). However, like other tenses, past simple can express more than one meaning (see Section 4.2). The dictionary of tense senses we compiled (see Appendix E) differentiates between four meanings of past simple:

- State in the past (*Jason was aware of the consequences.*)
- Single action in the past (*Alfonso read a great book last night.*)
- Repeated action in the past (*It snowed every day last winter.*)
- Social distancing (*I just wanted to know...*)

[3] `https://dictionary.cambridge.org/dictionary/english/past-simple`

With this list of senses and an algorithm that detects the most suitable tense sense given the context, we can generate a more concrete grammar-concept question (39b).

(39) At first, Stefanie *felt* confused about this.

a. Does Stefanie feel confused about this now? (no)

b. Did Stefanie repeatedly feel confused about this? (no, just once, at first; as it is not a repeated action)

The task of tense sense disambiguation (TSD) is very relevant to our work on QG as it can facilitate the creation of more fine-grained templates. We address this in Section 4.2 by leveraging machine learning and rule-based algorithms. However, the performance of TSD models is still too suboptimal to be used in such a high-stakes task as QG, where precision has to be 100% in order to avoid exposing learners to erroneous linguistic input.

6.9.5 Adverbial Clauses and Prepositional Phrases

Finally, there is a challenge concerning including different types of clauses and prepositional phrases or removing them from the question. In fact, the computer-generated questions that received the highest scores were concise, highlighting the importance of considering the syntactic structure of a sentence. For instance, removing non-restrictive clauses (usually separated by commas or other punctuation) and keeping restrictive types usually led to well-formed questions:

(40) Meanwhile, LeEco has spun out sports and cloud units, bringing in private equity capital from conglomerate HNA Group, Alibaba boss Jack Ma's Yunfeng Capital, and others.

a. *Computer (+):* According to the article, what has LeEco done? LeEco has __________ sports and cloud units.

This computer-generated question received the highest score on both the well-formedness and answerability scales. Interestingly, this seemed to be the case even when not enough information was provided in the question to answer it correctly. For example, the following question does not specify the conditions under which

Jia might be forced to put up more collateral. Nevertheless, the question also received the highest scores on both scales:

(41) Such share pledges can be risky: if Leshi Internet stock fell sharply, Jia might be forced to put up more collateral or sell down his stake.

a. *Computer (+):* According to the article, what might Jia be forced to do? Jia might be forced to _________ his stake.

We only removed non-restrictive clauses from a gap sentence when they were separated by commas, which was the case for 33% of computer-generated questions, and we never removed prepositional phrases when they were in the same clause as the target form. Subordinate and coordinate clauses were not removed when they followed the main clause and were not separated by a comma:

(42) Nomura pegs Mohan as neutral in his monetary policy stance. The oldest of the field of candidates, he has just taken up a position at Yale University *although a source familiar with his plans indicated he was reluctant to take on the post.*

a. *Computer:* According to the article, what has the oldest of the field of candidates done? The oldest of the field of candidates has _________ a position at Yale University *although a source familiar with his plans indicated he was reluctant to take on the post.*

To provide statistical evidence supporting our intuition about the superiority of the questions with removed non-restrictive clauses, we conducted the following analyses. First, we filtered out obviously ungrammatical computer-generated questions in which the errors were propagated by the parser and the coreference resolution module. We then annotated the 60 computer-generated questions ($M_{well-formedness} = 4.59$, $M_{answerability} = 4.62$) and conducted Welch's t-tests. The results showed that the perceived well-formedness of the computer-generated questions with non-restrictive clauses removed ($M = 4.73$, $SD = 0.23$) was higher than that of the ones where no part of the sentence following the target form was removed ($M = 4.52$, $SD = 0.49$), and the difference was significant, $t(58) = 2.30$, $p = .02$, 95% CI $[4.73; 4.52]$. In case of answerability, on the other hand, the questions with removed clauses ($M = 4.59$, $SD = 0.79$) were rated as more difficult to answer than those that did not undergo any additional modifications ($M = 4.64$,

$SD = 0.54$). However, the difference was non-significant, $t(28) = -0.23$, $p = .82$, 95% CI $[0.45; 0.36]$.

Although the heuristics of splitting the sentence into clauses separated by commas seems to be working well, QG systems could benefit from statistical models trained on larger data sets containing pairs of sentences – with and without certain clauses and prepositional phrases – that could automatically classify a clause as necessary or not.

6.9.6 Evaluation of Question Generation Systems

First and foremost, it should be noted that any kind of human evaluation is subjective. This issue becomes particularly salient when raters encounter test questions. The Figure-Eight guidelines recommend an even answer distribution for test questions, i.e., testing both good and bad questions in our case. While ill-formed or unanswerable questions were not difficult to write and did not receive criticism from the participants even when they rated them incorrectly (we allowed for ratings of 1, 2, and 3 for such questions), good questions proved to be more challenging. Some participants argued that when a question is well-formed and answerable, the rating it receives in the end is quite subjective. To address this problem, one could only test the participants on ill-formed questions or introduce a binary choice *Is this question grammatical or ungrammatical?* instead of a scale *How well-formed is this question?*.

The fact that open-ended questions did not elicit significantly more rephrasings of phrasal verbs, as the results of our second study showed, may also be due to the limitations of a crowdsourcing experiment design. First, there were simply too few rephrasings (3%) produced by the participants. In crowdsourcing studies, participants are less encouraged to take their time to process every question they are not being tested on. As there were no correctness checks for the answers they submitted, many used the same wording as in the source text for the sake of time. When designing a similar study in the future, one could try blocking the copy-paste functionality in order to prevent participants from copying the text without reading it more thoroughly.

Finally, it is also worth pointing out that a crowdsourcing study with non-proficient English speakers took about four hours to complete. When we introduced the filtering mechanism to select proficient English speakers, the time went up to about 39 hours. This could indicate a potential demographic and linguistic bias of some crowdsourcing studies. That being said, while crowdsourcing platforms are not magic boxes that always produce perfect data, they do provide quality control mechanisms that allow researchers and other users to acquire annotations and judgments comparable to those of experts.

6.10 Summary

To conclude, answering questions is an integral part of both developing and testing reading comprehension as well as vocabulary and grammar knowledge in the context of reading. In this chapter, we provided an overview of different types of questions used in the context of foreign language teaching and learning and discussed the generation of two novel types of questions providing functionally driven input enhancement: form-exposure and grammar-concept questions.

The results of the two crowdsourcing studies described in this chapter showed that automatically generated form-exposure questions are comparable to human-written ones. This finding is in line with previous research in which expert judges evaluated the quality of computer-generated and human-written questions (Zhang and VanLehn, 2016). We also found that the addition of a gap sentence to a wh-question significantly improves its well-formedness and answerability. Moreover, the responses elicited by wh- questions followed by a gap sentence contain significantly more correct answers, as well as target linguistic forms, than open-ended wh- questions. From a CL perspective, these findings imply that QG systems can benefit from leveraging and combining different types of questions instead of opting either for wh- questions or gap sentences.

Overall, we demonstrated that both proficient and non-proficient English speakers rate the quality of computer-generated questions just as high as human-written ones. Interestingly, proficient speakers of English thought that most of the questions were written by an English teacher, although the proportion of computer-generated and human-written questions in the study was the same. This shows that people are often not aware of the state-of-the-art in computational linguistic technology. This is true not only for crowd workers but also English teachers, who could benefit more from intelligent computer-assisted language learning tools.

Chapter 7

Conclusion and Outlook

> "Learning is goal-oriented . . . Teaching therefore becomes an active thinking and decision-making process in which the teacher is constantly assessing what students already know, what they need to know, and how to provide for successful learning."
>
> O'Malley and Chamot (1990)

In this thesis, we addressed the research gaps identified in the introductory chapter by improving the existing computational linguistic technology and leveraging it to support language teaching and learning. The *FLAIR* system (`www.purl.org/icall/flair`) we developed implements algorithms for the automatic detection of linguistic forms, input enrichment, and question generation. Its components have been theoretically grounded and evaluated in separate studies, the results of which demonstrate the feasibility and effectiveness of our approach. In this chapter, we summarize our work, discuss the implications and limitations of our approach, and conclude with an outlook for the future.

7.1 Summary of the Results

The research goal of this thesis was to explore and implement automatic techniques that help create a richer grammatical intake from a given text input and engage learners in making form-meaning connections during reading. We successfully applied rule-based, crowdsourcing, and machine learning approaches to develop theory-motivated and efficient tools for foreign language teaching and learning (FLTL) and to address the research questions listed in Chapter 1.

A technical evaluation demonstrated a high state-of-the-art performance of our rule-based algorithm that *detects 87 linguistic forms* specified in the official curriculum for the English language. It allowed us to explore the representation, distribution, and co-occurrences of linguistic forms across web documents and develop the *FLAIR* system, which provides automatic input enrichment for FLTL.

The task of *automatic input enrichment* – as we approached it – consisted of selecting an information retrieval algorithm and the features to be included in it. We opted for a variation of the classical BM25 algorithm that includes weighted linguistic forms and a normalization parameter controlling for text length. The results of our online study with English teachers showed that *FLAIR* succeeds in providing reading materials that are (i) rich in the linguistic forms teachers care about, (ii) topically relevant, and (iii) suitable as a reading assignment for language learners. Teachers' feedback indicated that input enrichment systems need to take into account not only a rich representation of linguistic forms but also their variety.

With that in mind, we explored the task of *tense sense disambiguation*, for differentiating between the variety of contexts in which grammatical tenses occur. Having created a taxonomy of tense senses, we crawled a corpus of 5000 documents and annotated 4089 instances of tenses with their senses. We then designed a set of machine learning features and successfully trained several statistical models to select the optimal ones for the target tenses. Each of our selected models outperformed a strong majority baseline as well as the state of the art in the task of tense sense disambiguation. We showed that models for different tenses benefit from different features, but the best-performing ones made use of all the provided features.

Finally, we discussed *question generation* for FLTL and proposed that, in addition to the typical focus on comprehension, questions can also play an important role in *functionally driven input enhancement*. We proposed two novel types of questions, form-exposure and grammar-concept questions, that help language learners process relevant linguistic forms and draw form-meaning connections during a communicative activity. We discussed our transformation- and template-based question generation approaches and evaluated the quality of the output of our system in two crowdsourcing studies. The results confirmed that automatically generated questions are comparable to human-written ones with regard to their well-formedness and answerability and can therefore be used in the FLTL context with minimal supervision. To showcase this, we incorporated the question generation component into the new version of *FLAIR* and designed a standalone interactive web application that automatically generates questions for an input text (`www.purl.org/qg`).

7.2 Limitations

The pragmatic nature of our work allowed us to leverage several approaches and algorithms to achieve our research goals, thus downplaying technical pitfalls such as erroneous output by third-party NLP tools. While this is discussed and exemplified throughout the thesis, in this section, we consider the methodological limitations of our research and make suggestions for minimizing or avoiding them in future work.

Generalizability We chose news article as our development data for all of the *FLAIR* components: from the detection of linguistic forms and disambiguation of their senses to input enrichment and question generation. However, issues of generalizability and representativeness need to be taken into consideration when drawing conclusions from the experimental results. Firstly, news articles are likely to include only a limited number of certain linguistic forms or their senses, such as a spontaneous decision for future simple (*I'll pay with a card*). Secondly, they may or may not appeal to language teachers and learners, depending on their interests. Finally, they are usually written for educated readers of higher language proficiency. We plan to investigate whether the performance of our tools differs on texts of lower complexity and from other genres, such as news articles for children, graded readers, and corpora of spoken English.

Sample size The online study testing automatic input enrichment employed a repeated-measured design, which allowed us to collect 240 judgments from 12 teachers. The fact that we could not reject the null hypothesis that the preference for *FLAIR* did not depend on the frequency level of the target linguistic forms (see Section 5.4.3) might have been caused by the small sample size. A larger number of participants would be desirable in order to have more statistical power and – potentially – find significant effects.

Another case where more data could improve the results was our study on tense sense disambiguation. A low inter-annotator agreement led us to exclude many items from the training set, which had an impact on the performance of our machine learning algorithms. There could be several reasons for a low inter-annotator agreement. If it is caused by the intrinsic ambiguity of the items, carefully examining the dataset and making decisions on whether to leave in or exclude such items may improve the results. If the cause is poor performance by the annotators, both crowdsourcing and traditional annotation scenarios can benefit from providing more training and support to the annotators as well as implementing stricter quality control mechanisms, such as in-task test questions.

Subjectivity Another limitation leading to a low inter-annotator agreement is the subjective nature of any human evaluation or annotation. For classification tasks, broader categories may minimize the disagreement among annotators. In the case of evaluation, discrete scales will ensure the independent nature of judgments, which is necessary for statistical analyses. Importantly, every category or criterion should be richly illustrated with examples in the instructions and in the task itself, e.g., via tooltips, to reduce the chance of ungrounded subjective judgments.

7.3 Outlook

We conclude this thesis with a discussion of several potential future directions for our research at the intersection of second language acquisition, computational linguistics, and intelligent computer-assisted language learning.

As discussed in Section 3.6, our ultimate goal is to further develop *FLAIR* into an intelligent tutoring system. The first steps in this direction will include implementing a learner model along with the functionality of marking certain linguistic forms as difficult. Eventually, we plan to model learners' reading behavior as well as linguistic knowledge in order to automatically suggest appropriate material accompanied by activities providing functionally driven input enhancement and scaffolding feedback.

The comparability of computer-generated and human-written questions paves the way for an external evaluation of the learning outcomes that can be achieved using the approach of functionally driven input enhancement. With NLP technology integrated into web-based tools supporting the intervention, a large-scale randomized controlled field study can be set up and run over an entire semester or school year, the time span in which real-life foreign language learning takes place. Crucially, such a setup could also include a collection of measures on individual differences and other relevant factors. In addition, the insights gained from such a study could further improve the learner model by parameterizing the selection of reading materials and the generation of questions targeting the linguistic forms that are particularly relevant for a given user.

From a computational linguistic perspective, the task of generating questions is feasible yet challenging and is interestingly intertwined with other NLP tasks. For instance, the tasks of named entity recognition and coreference resolution can be used to make questions more precise. The integration of semantic roles – and potentially, other semantic representations – into the question generation system can lead to a larger variety of generated questions, as has been previously demonstrated by, e.g, Heilman (2011) and Flor and Riordan (2018). Further advances in the task of tense sense disambiguation can broaden the scope of grammar-concept questions asking about the interpretation of tenses in various contexts.

Finally, one important strand of future work is the generalization of our system to other languages. As our system is rule-based, this task requires writing different rules for every language both for the detection of linguistic forms and for the generation of questions. Currently, we have a full English system and an input enrichment module for German in place.

To conclude, we demonstrated that the methodology extensively explored and evaluated in the field of second language acquisition can be automated using state-of-the-art computational linguistic technologies. Our work thus proves that developing theory-motivated and efficient applications for language teaching and learning is a feasible enterprise. Feedback loops between researchers, developers, and teachers can ensure the constant improvement of such applications and encourage teachers to integrate them into their pedagogical practice.

Appendix A

FLAIR and Other IR Systems

	FLAIR	***TextFinder*** (Bennöhr, 2005)	***REAP*** (Brown and Eskenazi, 2004)	***LAWSE*** (Ott and Meurers, 2011)
Database	The Web or a corpus	Offline database	Offline database	The Web
Source	Web materials or the user's own texts	Texts from online newspapers	Web materials	Web materials
3rd-party tools	Bing, CoreNLP	Lucene	AltaVista	Lucene
Main focus	Re-ranking, question generation	Text complexity and user modeling	Lexical language modeling	Text complexity, lexical language modeling
Target users	English and German L2 teachers and learners	English L2 adult learners	English learners, teachers, researchers (L1 and L2)	English L2 learners
Readability	Yes: word and sentence length	Yes: word and sentence length + conjunctions	Yes: word histograms	Yes: readability measures + lexical frequency profiles
Learner model	No	Yes: writing sample	Yes: pre-defined ability levels	No
Grammar	Yes: 87 linguistic forms	Partially: conjunctions	No	No
Vocabulary	Yes: academic vocabulary and custom word lists	No	Yes: word lists	Yes: lexical frequency profiles
Reading interface	Yes: highlighted forms	No	Yes: dictionary definitions	No
Evaluation	Online study with teachers, crowdsourcing	Teacher ranking, learner questionnaire	Empirical study, learner questionnaire	Against a corpus of graded texts
Future work	Intelligent tutoring system, other languages	Readability formula optimization	Grammar difficulty, text cohesiveness	Syntactic features, grammatical constructions

Table A.1: A comparison table of *FLAIR* and a selection of IR systems.

Appendix B

List of 87 Linguistic Forms Identified by ***FLAIR***

Abbreviations:

lookup = look-up in a predefined list
regex = regular expression
Tregex = tree-based regular expression
Semgrex = dependency-based regular expression
POS = part of speech
CP = constituency parse
DP = dependency parse

Construct	**Detection method**
Simple constructions	
Existential *there* (present and past)	DP
Sentence structure	
Imperative sentences	Tregex
Subordinate clauses	CP
Relative clauses	CP
Adverbial clauses	CP
Direct object	DP

Indirect object	DP
Questions	
Direct questions	regex
Indirect questions	CP
General questions (yes/no)	regex + POS
Be-, do- or *have-* questions	regex + CP + DP
Wh- questions	regex + POS
Tag questions	regex + POS + DP
Conditionals	
Real conditionals (Type 0 and Type 1)	Tregex + lookup
Unreal conditionals (Type 2, Type 3 and Mixed)	Tregex + lookup
Tenses and Aspects	
Future perfect, future perfect progressive	Semgrex
The simple aspect, present simple, past simple, future simple	DP + POS
The progressive aspect, present progressive, past progressive, future progressive	DP + CP + POS + lookup
The perfect aspect, present perfect, past perfect, future perfect	DP + CP + POS
The perfect progressive aspect, present perfect progressive, past perfect progressive, future perfect progressive	DP + CP + POS
The Voice	
Passive voice	DP
Verb forms	
irregular verbs (in the 2nd or the 3rd form)	POS + regex
regular verbs (in the 2nd or the 3rd form)	POS + regex
phrasal verbs	Tregex

going to, used to (habitual, past)	regex + POS
short verb forms ('s, 're, 'm, 's, 've, 'd)	POS + regex
full verb forms *(is, are, am, has, have, had)*	POS + regex
auxiliaries *be, have* in declarative sentences	POS + regex + DP
-ing verb forms (gerund AND participle)	POS
to- infinitives	POS
emphatic *do*	POS + DP
Modal verbs	
simple modals *(can, must, need, may)*	POS + lookup
advanced modals *(might, ought to, able)*	POS + lookup
Negation	
no, not, never	DP
other negative adverbs *(rarely, hardly, etc.)*	POS + lookup
Conjunctions	
simple conjunctions *(and, but, or)*	POS + lookup
advanced conjunctions *(therefore, because, etc.)*	POS + lookup
Prepositions	
basic prepositions *(in, at, on, with, after, to)*	POS + lookup
advanced prepositions *(during, through, etc.)*	POS
complex prepositions *(because of, etc.)*	POS + DP
Pronouns	
subjective pronouns *(I, you, etc.)*	DP + POS + lookup
objective pronouns *(me, you, them, etc.)*	DP + POS + lookup
possessive pronouns *(my, your, their, etc.)*	POS
absolute possessive pronouns *(mine, yours, etc.)*	lookup
reflexive pronouns *(myself, themselves, etc.)*	POS + lookup

Quantifiers	
some, any, much, more	DP + regex
Articles	
a, an, the	POS + regex
Adjectives	
positive degree	POS + regex
comparative or superlative degree - simple	POS
comparative or superlative degree - compound	POS + regex
irregular comparative or superlative	POS + lookup
Adverbs	
positive degree	POS + regex
comparative or superlative degree - simple	POS
comparative or superlative degree - compound	POS + regex
irregular comparative or superlative	CP + lookup

Appendix C

Regular Expressions for Tree and Dependency Graphs

C.1 Imperatives

Tregex:

```
/^S.*/ > (ROOT !<<- /\?/) < (/^V.*/ !,, /^N.*/
<<, /VB|VBP/=imperativeVerb)
```

Explanation: A declarative sentence that is just 1 level away from the root, immediately dominates a VP starting with an infinitive verb form and not preceded by any NP. Note that in Java, the question mark has to be escaped twice.

Tricky cases where it works:

- When the sentence does not start with a VP: “Please, close the door”, “And then turn on the lights"

Tricky cases where it does not work:

- When the imperative VP is preceded by a pronoun, usually *you*: “You go and think about what you’ve done.”

C.2 Wishes

Semgrex:

```
{lemma:wish;tag:/^V.*/} >ccomp ({} >cop {tag:/VBD/} |
>aux {tag:/VBD/} | >aux {word:could} | >aux {word:would} |
>aux {word:might}) | >ccomp {tag:/VBD/}
```

Tricky cases where it works:

True negatives (not detected, and rightly so):

- “He wished her a happy birthday."
- “He wished her to always be happy."
- “He wished to always be happy."

True positive (correctly detected):

- “I am wishing you were here."

C.3 Hypothetical events

Semgrex:

```
{lemma:if} <mark ({tag:/VBD|N/} !</advcl:if/ {}) | <mark
({} >cop {tag:/VBD|N/} | >aux {tag:/VBD|N/} !</advcl:if/ {})
| <mwe {lemma:as}
```

Explanation: *If only* followed by a verb in a past tense with no subjunctive clause. *As if* may or may not be followed by a verb in a past tense (e.g., “As if I cannot sing!")

C.4 Reported speech

Semgrex:

```
{lemma:/say|tell|add|write|report/;tag:/VB.*/}
!>ccomp ({tag:/^VB.*/} $++ {tag:’’} $-- {tag:‘‘})
```

Explanation: A reporting verb that is not governing a clause inside quotation marks. If there is no verb inside the quotation marks, it is not considered a full quote, and the sentence can still be counted as reported speech.

Decisions on borderline cases: Although reported speech is mainly studied in the language learning setting when something was reported in the past (e.g., “He said that..."), we also detect the present-tense use of reported speech (e.g., “He is saying that ...").

Tricky cases where it works:

- “The company also said that it will continue with its application to the Department of Trade and Industry to operate a ‘personal communications network’."

C.5 Conditionals

Tregex:

```
/^S.*/=ifClause <+ (/^CONJP/) (/^IN/ < if|If|unless|Unless !, as|As
!. necessary !. (not !.. /^V.*/)) !>+ (/SBAR/) (/^V.*/
<<# (know|see|show|remember|tell|remind|advise|notice|recognize|
recognise|doubt|check|investigate|consider|decide|explain|feel|find|
guess|mention|repeat|knew|known|saw|seen|showed|shown|remembered|
told|reminded|advised|noticed|recognized|recognised|doubted|
checked|investigated|considered|decided|explained|felt|found|
guessed|mentioned|repeated
!.. (that .. if|unless)) !$++ /^NP/) !$, (/.*/ <<# (sure|certain|aware
!.. (that ..if|unless))) (>+ (@VP|S|SBAR) (/^S.*/=mainClause < (/^NP/
$.. (/^VP/ (?<<, would|could|might=would | ?<<, /^VBD$/=realPast))) |
<+ (!SBAR) (SBAR ?<< would|could|might=would) | <+ (!SBAR) (S << /,/
?<< would|could|might=would)) <+ (/^S$/) (/^V.*/
(<<, needed|would|could|might=ifPast | <<, need|will|can|may|
wo=ifPresent | <<, @VBZ|VBP=ifPresent | <<, @VBN=ifPresent |
<<, (@VBD=ifPast ,, /^N.*/) | <<, (@VBD=ifPresent !,, /^N.*/)))
```

Explanation:

Mark a clause that has *if* or *unless* as a descendant, which satisfies the following conditions:

If/*unless* is not preceded by *as* (to exclude *as if*), is not followed by *necessary* (to exclude *if*/*unless necessary*), and is not followed by *not* if it is followed by a non-VP (to exclude *if not altogether horrible* but not to exclude *if not treated correctly*).

This clause should also not be a sister-node of a VP heading the verbs like know, see, check, etc. (to exclude "I don't know if he is coming", "He will immediately see if you are making enough effort", etc.) and also not a sister-node of any subtree heading *aware*, *certain*, or *sure* (to exclude "I am not certain if he will come"). Both the special verbs and *aware*, *certain*, and *sure*, should not be followed by *that if*, immediately or not (to exclude "I am sure that if you try hard, you can succeed.")

There are three ways to find the main clause:

1. It can be the first S-parent of the if-clause (via an unbroken chain of VP, S, or SBARs, if any) that contains an NP followed by an VP (not necessarily immediately followed).
2. It can be the first SBAR-child of the if-clause (via an unbroken chain of VP or S's, if any).
3. It can be the first S-child (via an unbroken chain of any nodes except for SBAR, if any) that contains a comma.

In the main clause, find the main verb: a VP whose first child is either would/-could/might or a verb in the past tense (VBD). If a verb in the past tense is found, the sentence belongs to the real-past conditionals (“If I studied hard, I always got good marks.”). If would/could/might is found, mark it as *would* for later analysis. Always inspect the first element of the retrieved trees and omit the rest.

In the if-clause, find the main verb: a VP (dominated by the if-clause-S/SBAR through a chain of S, if any), whose left-most descendant is either:

– a modal in its present form (need/can/may/will): mark it as *ifPresent*

– a modal in its past form (needed/could/might/would): mark it as *ifPast*

– a verb in its present form (VBZ or VBP): mark it as *ifPresent*

– a verb in its past form (VBD) preceded by a noun: mark it as *ifPast*

– a verb in its past participle form (VBN: “if accompanied by parents, . . . ”): mark it as *ifPresent*

– a verb in its past form (the verb may mistakenly be parsed as VBD instead of VBN) not preceded by a noun: mark it as *ifPresent*

Then, follow the logic:

if *realPast*: real past

else if *would* && *ifPresent*: mixed

else if *would* && *ifPast*: unreal (or real past with would/cold/might)

else if (!*would*) && *ifPresent*: real present/future

else if (!*would*) && *ifPast*: mixed

else: mixed

Decisions on borderline cases:

- *Even if* is always counted as conditional: either real (e.g., “Even if he fights, he will need..."), unreal (e.g., “Even if it was not right..."), or mixed (e.g., “Even if it is, some state would need...").
- If the verb in the main clause is in its past form, it is counted as a real past conditional (e.g., “If I studied hard, I always got good marks.")

- All conditionals where the verb in the if-clause is in its past form and the verb in the main clause is would/could/might are counted as unreal.

Tricky cases where it works:

True positives (correctly detected):

- “He knew that if she..."
- “He always tells me that if I..."

True negatives (not detected, and rightly so):

- “He does not know if she..."
- “He will not come if she..."

Tricky cases where it does not work:

- Real conditional in the past, especially with reported speech (e.g., “He said that if he were to..., he would...")
- e.g., "You will not find out their surnames if they are not related.": If the if-clause projects to the main S via a chain of VP's, the object dependent of the verbs from the list (know, see, find, etc.) will not be matched because it will not be at the same level as the if-clause.
  However, this only influences recall, not precision. It could be solved by adding a Semgrex.

Appendix D

Document Co-occurrence Frequencies of Linguistic Forms

Construction 1	**Frequency (% docs)**	**Construction 2**
	High frequency	
Positive adjective (nice, difficult)	96.70	Positive adverb (easily, fast)
The	95.80	A
Irregular verbs (form 2, 3)	95.20	Regular verbs (form 2, 3)
Present Simple	93.20	Past Simple
To infinitives	90.00	-ing verb forms
	Medium frequency	
Pronouns as subject	73.50	Pronouns as object
The	67.80	An
A	67.30	An
-ing verb forms	67.10	-ing nouns

Long verb forms (have, are)	64.80	Short verb forms ('ve, 're)
Plural nouns - regular	63.30	Plural nouns - irregular
Present Perfect	63.20	Present Simple
Past Simple	62.90	Present Perfect
Comparative adjective short (nicer)	57.80	Positive adjective (nice, difficult)
Superlative adjective short (the nicest)	55.40	Positive adjective (nice, difficult)
Present Simple	50.20	Present Progressive
Comparative adverb short (faster)	45.30	Positive adverb
Not	41.60	N't
Future Simple	40.10	Present Simple
Present Progressive	38.10	Present Perfect
Comparative adjective short (nicer)	38.00	Superlative adjective short (the nicest)
Comparative adverb short (faster)	33.00	Comparative adjective short (nicer)
Past Simple	30.00	Past Progressive
Future Simple	26.00	Present Progressive
Past Perfect	24.70	Past Simple
Superlative adverb short (fastest)	23.70	Positive adverb (easily, fast)
Direct object (transitive verb)	23.50	Indirect object (ditransitive verb)
Present Perfect	21.60	Past Progressive
Can	20.50	Could

Past Perfect	19.30	Present Perfect
Comparative adjective long (more difficult)	19.00	Positive adjective (nice, difficult)
Superlative adjective long (the most difficult)	18.40	Positive adjective (nice, difficult)
Past Progressive	17.20	Present Progressive
Comparative adjective short (nicer)	15.80	Comparative adjective long (more difficult)
Superlative adverb short (fastest)	15.10	Comparative adverb short (faster)
Superlative adjective short (the nicest)	13.80	Superlative adjective long (the most difficult)
Can	13.70	May
Some	13.70	Any
Could	13.10	May
Past Perfect	12.10	Past Progressive
	Low frequency	
Past Simple	10.20	Present Perfect Progressive
Future Simple	9.50	Going to
Can	8.90	Might
Present Perfect Progressive	8.70	Present Perfect
Might	8.60	Could
Present Progressive	7.10	Present Perfect Progressive
Wh- questions	6.90	Yes/No questions
Comparative adjective long (more difficult)	6.10	Superlative adjective long (the most difficult)

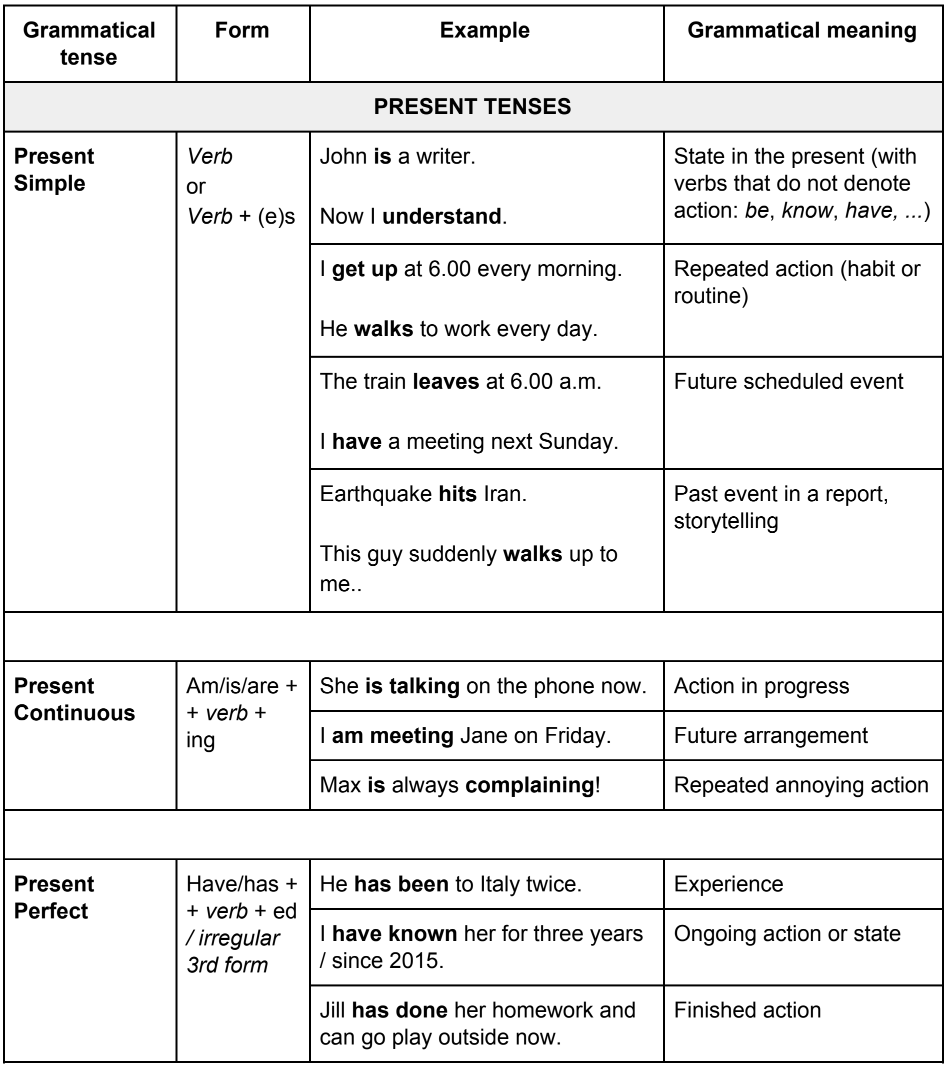
Explicit negation (no, not, never)	5.10	Partial negation (barely, hardly)
Might	5.10	May
Real conditional	4.30	Unreal conditional
Comparative adverb long (more easily)	4.10	Positive adverb (easily, fast)
Comparative adverb long (more easily)	4.10	Comparative adverb short (faster)
Present Perfect Progressive	3.80	Past Progressive
Past Perfect	3.60	Present Perfect Progressive
Many	3.00	Much
Superlative adverb long (the most easily)	2.90	Positive adverb (easily, fast)
Superlative adverb short (fastest)	2.90	Superlative adverb long (most easily)
Past Perfect Progressive	2.60	Past Simple
Possessive pronoun (my)	2.40	Possessive absolute pronoun (mine)
Past Perfect Progressive	2.00	Present Perfect
Comparative adjective long (more difficult)	2.00	Comparative adverb long (more easily)
Past Perfect Progressive	1.80	Past Progressive
Past Perfect Progressive	1.70	Present Progressive
Past Perfect	1.40	Past Perfect Progressive
Past Perfect Progressive	0.50	Present Perfect Progressive
Can	0.50	Able

Comparative adverb long (more easily)	0.20	Superlative adverb long (most easily)
Might	0.00	Able
Able	0.00	May

Table D.1: Pairwise document co-occurrence frequencies of grammatical constructions calculated on the basis of a corpus of 2400 web texts.

Appendix E

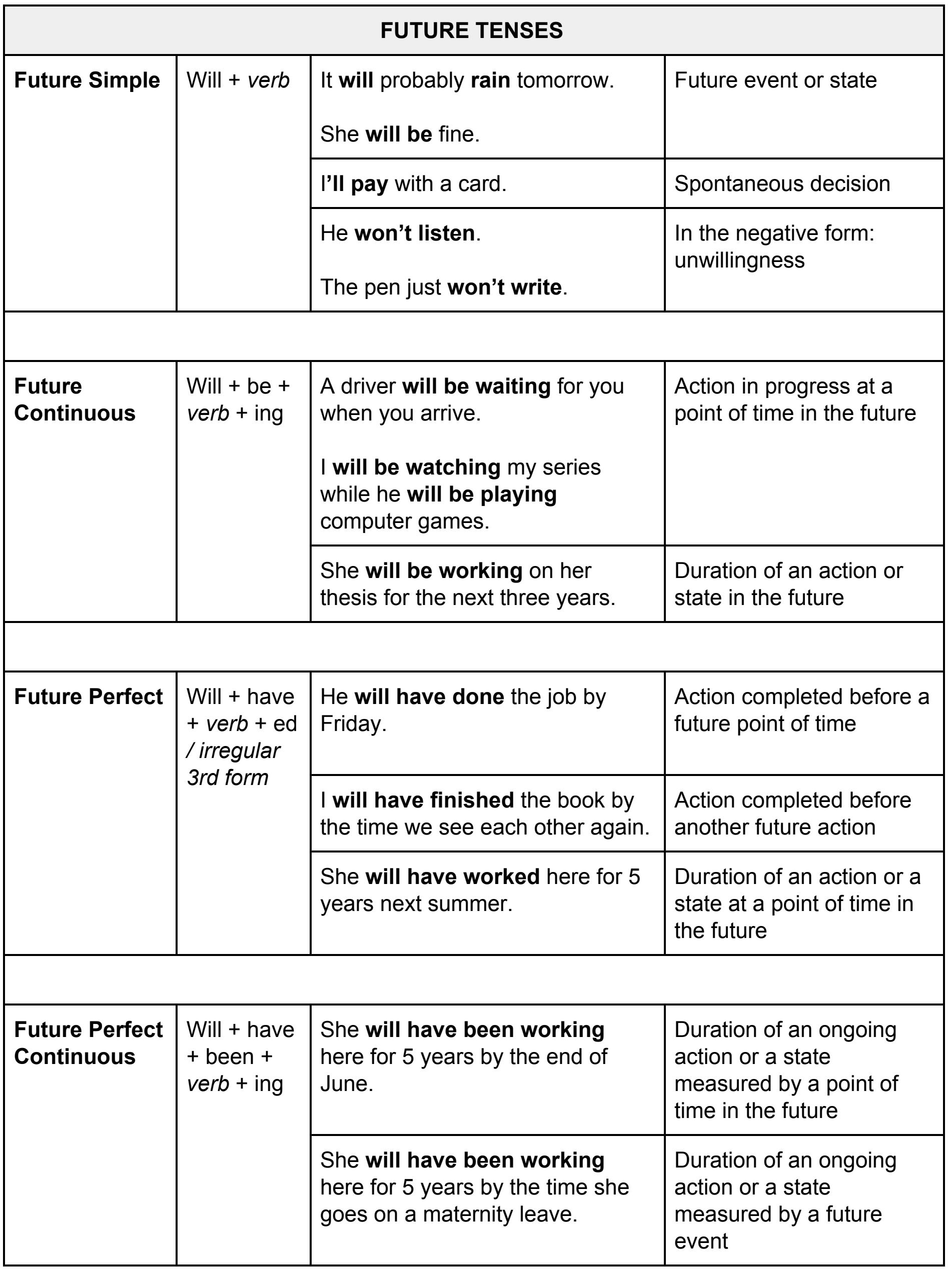
Dictionary of Tense Senses



Grammatical tense	Form	Example	Grammatical meaning
PRESENT TENSES			
Present Simple	*Verb* or *Verb* + (e)s	John **is** a writer. Now I **understand**.	State in the present (with verbs that do not denote action: *be*, *know*, *have, ...*)
		I **get up** at 6.00 every morning. He **walks** to work every day.	Repeated action (habit or routine)
		The train **leaves** at 6.00 a.m. I **have** a meeting next Sunday.	Future scheduled event
		Earthquake **hits** Iran. This guy suddenly **walks** up to me..	Past event in a report, storytelling
Present Continuous	Am/is/are + + *verb* + ing	She **is talking** on the phone now.	Action in progress
		I **am meeting** Jane on Friday.	Future arrangement
		Max **is** always **complaining**!	Repeated annoying action
Present Perfect	Have/has + + *verb* + ed */ irregular 3rd form*	He **has been** to Italy twice.	Experience
		I **have known** her for three years / since 2015.	Ongoing action or state
		Jill **has done** her homework and can go play outside now.	Finished action

Present Perfect Continuous	Have/has + been + *verb* + ing	Luke **has been working** since 8 a.m.	Ongoing action
		Here you are. I **have been waiting** for you.	Finished action that stopped very recently

PAST TENSES			
Past Simple	*Verb* + ed / *irregular 3rd form*	Last winter **was** extremely cold. Mr. Jones **had** three brothers.	State in the past (with verbs that do not denote action: *be*, *know*, *have*)
		Jane **saw** a good film last night.	Single action in the past
		It **snowed** every day last winter.	Repeated action in the past
		I just **wanted** to ask you…	Social distancing
Past Continuous	Was/were + + *verb* + ing	Chris **was hiking** at 2 p.m. yesterday.	Action in progress in the past
		I **was watching** TV when the phone rang. While Alex **was traveling** in Europe, he ran into an old friend.	Ongoing action in the past interrupted by another action
		He **was coughing** all night long.	Repeated annoying action in the past
		I **was wondering** ...	Social distancing
Past Perfect	Had + + *verb* + ed / *irregular 3rd form*	Tim **had only been** to France once by then.	Experience at a time point in the past
		They **had known** each other for years.	Duration of an action or a state in the past
		They **had** already **moved** to Atlanta by 2010. She **had left** when he arrived.	Finished action at a time point in the past
Past Perfect Continuous	Had + been + *verb* + ing	They **had been swimming** in the water for 2 hours when they finally saw a whale.	Unfinished action interrupted by another action but still continuing
		I saw the wet streets: It **had been raining**.	Finished action at a time point in the past



FUTURE TENSES			
Future Simple	Will + *verb*	It **will** probably **rain** tomorrow. She **will be** fine.	Future event or state
		I'll pay with a card.	Spontaneous decision
		He **won't listen**. The pen just **won't write**.	In the negative form: unwillingness
Future Continuous	Will + be + *verb* + ing	A driver **will be waiting** for you when you arrive. I **will be watching** my series while he **will be playing** computer games.	Action in progress at a point of time in the future
		She **will be working** on her thesis for the next three years.	Duration of an action or state in the future
Future Perfect	Will + have + *verb* + ed */ irregular 3rd form*	He **will have done** the job by Friday.	Action completed before a future point of time
		I **will have finished** the book by the time we see each other again.	Action completed before another future action
		She **will have worked** here for 5 years next summer.	Duration of an action or a state at a point of time in the future
Future Perfect Continuous	Will + have + been + *verb* + ing	She **will have been working** here for 5 years by the end of June.	Duration of an ongoing action or a state measured by a point of time in the future
		She **will have been working** here for 5 years by the time she goes on a maternity leave.	Duration of an ongoing action or a state measured by a future event

Appendix F

Instructions for the Crowdsourcing Task on TSD

OVERVIEW

Important information! This job is designed for **proficient speakers of English**. We kindly ask you to only do this job if you are fluent in English and are familiar with the English grammar.

We want to prove that crowd workers are as good as expert human judges at linguistic tasks! Since we only need high-quality responses, there will be many test questions that will ensure your reliability. We appreciate your time and effort!

The task: Please, remember your English lessons, read the instructions carefully and help us select the most appropriate grammatical meaning of a verb in a sentence. The examples are provided below, in the grammar reference, and in the task itself.

WHAT IS A GRAMMATICAL MEANING?

Let's take the verb *to live* as an example.

We all know what it means and can find its definition in a dictionary. But depending on the grammatical construction and the context in which it is used, it can have different *grammatical meanings*. For example, just one form *has lived* can be used to express the following grammatical meanings:

- I ***live*** in the US. (Grammatical meaning of *live* here: State in the present)
- I ***have lived*** in three different cities. (Grammatical meaning of *have lived* here: Experience)
- I ***am living*** with my parents now. (Grammatical meaning of *am living* here: Action in progress)
- I ***have been living*** here for the last 5 years. (Grammatical meaning of *have been living* here: Ongoing action that started in the past and continues in the present)

STEPS

1. Read the sentence, pay attention to the highlighted area, especially to the **verb** in it.

2. Select the most appropriate **grammatical meaning of the verb** in this context (*(See the examples of grammatical meanings in the Grammar Reference below and when you mouse over the meanings in the task).*

   If the verb in the sentence is **negated**, select the corresponding 'positive' meaning from the list. For example:

   – Sentence: He has completed the task. – Grammatical meaning of the verb: Finished action

   – Sentence: He has not completed the task. – Grammatical meaning of the verb: Finished action

   Select the option **None of the above** If the highlighted words describe not a real but a hypothetical action as in:

   – If only we could *turn back* time! (But we cannot.)

   – She would *love* to go skiing. (But she does not love it now.)

   – I wish they *did not argue* all the time. (But they do.)

   – He must *have forgotten* about our meeting. (But maybe he did not.)

3. (Optional) Leave a comment about the current sentence in the corresponding field. Is there anything peculiar about the sentence that makes it hard to judge the tense or its meaning?

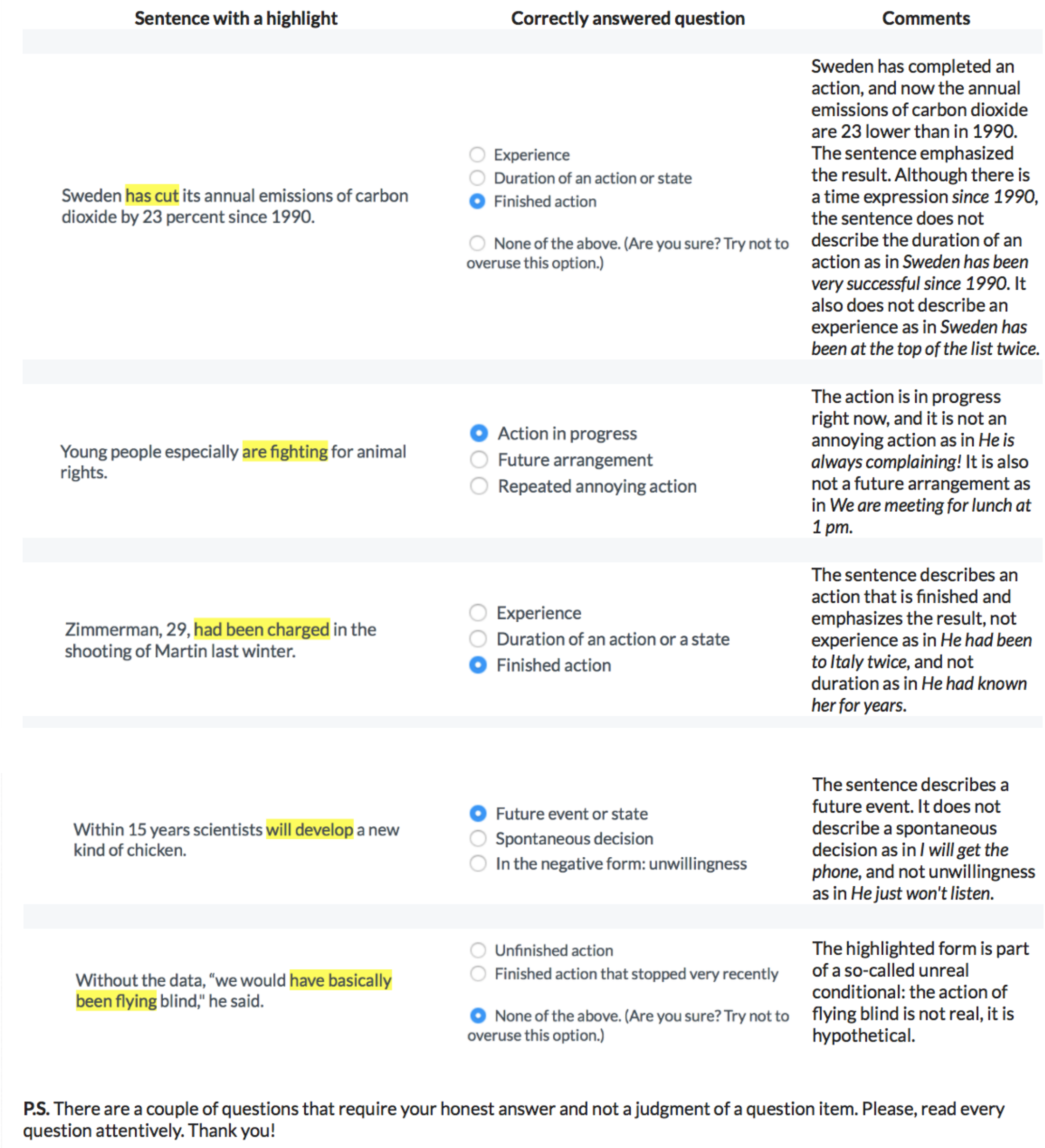
GRAMMAR REFERENCE

You will find the examples of each grammatical meaning when mousing over it in the task.

Here is a link to the grammar reference containing **all grammatical constructions** with their **forms**, **examples**, and **grammatical meanings**:

>>> **Grammar Reference (opens in a new tab)** <<<

EXAMPLES



Sentence with a highlight	Correctly answered question	Comments
Sweden has cut its annual emissions of carbon dioxide by 23 percent since 1990.	○ Experience ○ Duration of an action or state ● Finished action ○ None of the above. (Are you sure? Try not to overuse this option.)	Sweden has completed an action, and now the annual emissions of carbon dioxide are 23 lower than in 1990. The sentence emphasized the result. Although there is a time expression *since 1990*, the sentence does not describe the duration of an action as in *Sweden has been very successful since 1990*. It also does not describe an experience as in *Sweden has been at the top of the list twice*.
Young people especially are fighting for animal rights.	● Action in progress ○ Future arrangement ○ Repeated annoying action	The action is in progress right now, and it is not an annoying action as in *He is always complaining!* It is also not a future arrangement as in *We are meeting for lunch at 1 pm*.
Zimmerman, 29, had been charged in the shooting of Martin last winter.	○ Experience ○ Duration of an action or a state ● Finished action	The sentence describes an action that is finished and emphasizes the result, not experience as in *He had been to Italy twice*, and not duration as in *He had known her for years*.
Within 15 years scientists will develop a new kind of chicken.	● Future event or state ○ Spontaneous decision ○ In the negative form: unwillingness	The sentence describes a future event. It does not describe a spontaneous decision as in *I will get the phone*, and not unwillingness as in *He just won't listen*.
Without the data, "we would have basically been flying blind," he said.	○ Unfinished action ○ Finished action that stopped very recently ● None of the above. (Are you sure? Try not to overuse this option.)	The highlighted form is part of a so-called unreal conditional: the action of flying blind is not real, it is hypothetical.

P.S. There are a couple of questions that require your honest answer and not a judgment of a question item. Please, read every question attentively. Thank you!

Appendix G

Model Instantiations and Parameters for TSD

G.1 Preprocessing

All preprocessing, training, and testing was done in Python using the *sklearn* library (Pedregosa et al., 2011). We followed the following procedure for every model:

1. Excluded the items with agreement lower than 60%: e.g., when each of the three annotators selected a different sense of a tense:

   ```
   data_frame = data_frame.loc[data_frame[‘agreement’] >= 0.6]
   ```

2. One-hot encoding for categorical features:

   ```
   int_encoded = LabelEncoder().fit_transform(
                 data_frame[‘feature_name’].values)
   onehot_encoder = OneHotEncoder(sparse=False)
   int_encoded = int_encoded.reshape(len(int_encoded), 1)
   onehot_encoded = onehot_encoder.fit_transform(int_encoded)
   data_frame[‘feature_name_label’] = onehot_encoded
   ```

3. Split the data set into a training and a test set:

```
X = data_frame[[‘feature_name_1’,‘feature_name_2’]]
y = data_frame[‘sense_label’]
X_train, X_test, y_train, y_test = train_test_split(
                                   X, y, random_state=0)
```

4. Scaling, normalization of training and test sets:

```
scaler = MinMaxScaler()
X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

G.2 Model parameters

Majority baseline

```
clf = DummyClassifier(strategy = ‘most_frequent’).
                      fit(X_train, y_train)
```

Present simple

```
clf = RandomForestClassifier(max_features = 9,random_state=0).
                             fit(X_train, y_train)
```

Present perfect

```
clf = LinearSVC(C=1, random_state=0, class_weight=‘balanced’).
                fit(X_train_scaled, y_train)
```

Past simple

```
clf = DecisionTreeClassifier(max_depth=2).fit(X_train, y_train)
```

Past perfect

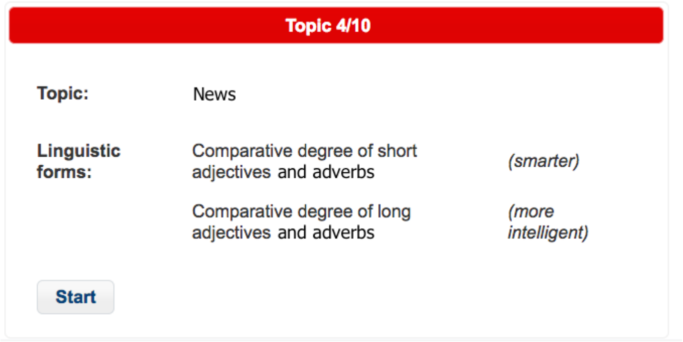
```
clf = RandomForestClassifier(n_estimators = 29, max_features = 12,
                             random_state=0).fit(X_train, y_train)
```

Appendix H

Online Study on Automatic Input Enrichment

INSTRUCTIONS

You will read and evaluate 10 pairs of articles on 10 different topics. For each pair of articles, you will first be presented with a topic and two target linguistic constructions:



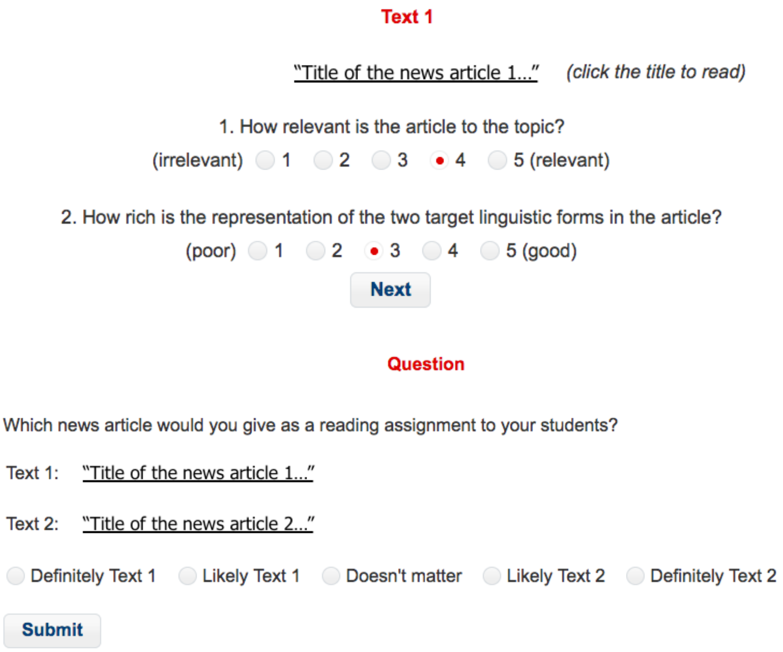
Topic 4/10

Topic:	News	
Linguistic forms:	Comparative degree of short adjectives and adverbs	*(smarter)*
	Comparative degree of long adjectives and adverbs	*(more intelligent)*

You will then read the first article and answer two questions about its relevance to the given topic and the representation of the given grammar constructions in it:

Follow the same procedure with the second article and click Next to answer the final question.

The final question presents the topic, the two target grammar constructions, and both articles (which you can open and read again if you want to). You need to decide which article you would choose as a reading assignment for an English class:



Text 1

"Title of the news article 1..." (click the title to read)

1. How relevant is the article to the topic?

(irrelevant) ○ 1 ○ 2 ○ 3 ● 4 ○ 5 (relevant)

2. How rich is the representation of the two target linguistic forms in the article?

(poor) ○ 1 ○ 2 ● 3 ○ 4 ○ 5 (good)

Question

Which news article would you give as a reading assignment to your students?

Text 1: "Title of the news article 1..."

Text 2: "Title of the news article 2..."

○ Definitely Text 1 ○ Likely Text 1 ○ Doesn't matter ○ Likely Text 2 ○ Definitely Text 2

Once you have read and answered questions about all 10 pairs of articles, you will be requested to answer a few questions about your experience during the experiment. Finally, you will be requested to submit your email address, to which the amazon gift voucher will be sent.

In case you encounter any problem, you can let us know by clicking the 'Report a problem' form in the left-side panel of the main webpage. Alternatively, you can also send us an email.

We thank you in advance for participating in the experiment.

Appendix I

Crowdsourcing Study on Automatically Generated Questions

INSTRUCTIONS

Important information! This job is designed for **proficient speakers of English**. We kindly ask you to only do this job if you are fluent in English. Thank you for your understanding!

In this job, you will be presented with **questions asked about excerpts from news articles**. The questions are intended for learners of the English language, and some of them are automatically generated by a computer program.

Your task is to **rate** each question on **two scales** from 1 to 5:

1. "How well-formed is this question?" (1 - very ill-formed ... 5 - very well-formed)
2. "Can this question be answered by the source text?" (1 - no, not at all ... 5 - yes, easily)

Then **answer** the question given the information provided in the source text. Please, write your responses in **English**.

Finally, make a **guess** whether the presented question is written by a human or is generated by a computer.

Optional: Once you have answered a question, you can **leave a comment** in the corresponding box.

STEPS

- **Read** the source text, the question asked about it and the gap sentence that answers it
- **Rate** the question item (the question and the gap sentence) based on how well-formed it is
- **Rate** the question based on how easily it can be answered by the source text
- **Answer** the question with as few words as possible
- **Guess** whether the question was written by an English teacher or generated by a computer. It does not matter if your guess is wrong.
- If you want, **leave a comment** about the question (optional)

EXAMPLES

Source text: The housing boom in Canada's hottest cities has spilled over into the suburbs, where builders say they are working as fast as they can to meet soaring demand and get homes to market before a much-feared housing bust.

Question items:

	NO	**YES**
Well-formed? (first scale)	What the housing boom in Canada's hottest cities has? ***Reason:*** *The question is ungrammatical. The word "done" is missing, the auxiliary verb "has" should be used right after the question word "What".*	What has the housing boom in Canada's hottest cities done? It has _____ into the suburbs.
Can be answered? (second scale)	What has the economic boom in England's hottest cities done? ***Reason:*** *The question cannot be answered. The text mentions a housing boom in Canada, not an economic one in England.*	Possible answers: spread / spilled out

P.S. There are a couple of questions that require your honest answer, and not a judgement of a question item. Please, read every question attentively. Thank you!

Bibliography

Steven Abney. Bootstrapping. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 360–367. Association for Computational Linguistics, 2002.

Eneko Agirre and Aitor Soroa. Semeval-2007 task 02: Evaluating word sense induction and discrimination systems. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 7–12. Association for Computational Linguistics, 2007.

Shaaron Ainsworth and Shirley Grimshaw. Evaluating the redeem authoring tool: can teachers create effective learning environments? *International Journal of Artificial Intelligence in Education*, 14(3, 4):279–312, 2004.

Luiz A Amaral and Detmar Meurers. On using intelligent computer-assisted language learning in real-life foreign language teaching and learning. *ReCALL*, 23 (1):4–24, 2011.

Jun Araki, Dheeraj Rajagopal, Sreecharan Sankaranarayanan, Susan Holm, Yukari Yamakawa, and Teruko Mitamura. Generating questions and multiple-choice answers using semantic analysis of texts. In *COLING*, pages 1125–1136, 2016. URL `http://aclweb.org/anthology/C16-1107`.

R Harald Baayen, Douglas J Davidson, and Douglas M Bates. Mixed-effects modeling with crossed random effects for subjects and items. *Journal of memory and language*, 59(4):390–412, 2008.

Kathleen Bardovi-Harlig. From morpheme studies to temporal semantics: Tense-aspect research in sla. *Studies in second language acquisition*, 21(3):341–382, 1999.

Lee Becker, Sumit Basu, and Lucy Vanderwende. Mind the gap: learning to choose gaps for question generation. In *Proceedings of the 2012 Conference of* *the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 742–751. Association for Computational Linguistics, 2012. URL `http://aclweb.org/anthology/N12-1092`.

Alessandro Benati. The effects of structured input activities and explicit information on the acquisition of the italian future tense. *Processing instruction: Theory, research, and commentary*, pages 207–225, 2004.

Alessandro Benati. Input manipulation, enhancement and processing: Theoretical views and empirical research. *Studies in Second Language Learning and Teaching*, 6(1):65–88, 2016. URL `https://doi.org/10.14746/ssllt.2016.6.1.4`.

Jasmine Bennöhr. A web-based personalised textfinder for language learners. Master's thesis, University of Edinburgh, 2005.

Kenneth Benoit, Drew Conway, Benjamin E Lauderdale, Michael Laver, and Slava Mikhaylov. Crowd-sourced text analysis: Reproducible and agile production of political data. *American Political Science Review*, 110(2):278–295, 2016.

Claudio Delli Bovi, Jose Camacho-Collados, Alessandro Raganato, and Roberto Navigli. Eurosense: Automatic harvesting of multilingual sense annotations from parallel text. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 594–600, 2017.

Samuel Brody, Roberto Navigli, and Mirella Lapata. Ensemble methods for unsupervised wsd. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 97–104. Association for Computational Linguistics, 2006.

Jonathan Brown and Maxine Eskenazi. Retrieval of authentic documents for reader-specific lexical practice. In *InSTIL/ICALL Symposium 2004*, 2004.

Jonathan Brown, Gwen Frishkoff, and Maxine Eskenazi. Automatic question generation for vocabulary assessment. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 819–826, Vancouver, British Columbia, Canada, October 2005. URL `http://aclweb.org/anthology/H05-1103`.

Peter F Brown, Stephen A Della Pietra, Vincent J Della Pietra, and Robert L Mercer. Word-sense disambiguation using statistical methods. In *Proceedings of the 29th annual meeting on Association for Computational Linguistics*, pages 264–270. Association for Computational Linguistics, 1991.

Jill Burstein, Jane Shore, John Sabatini, Brad Moulder, Steven Holtzman, and Ted Pedersen. The language musesm system: Linguistically focused instructional authoring. *ETS Research Report Series*, 2012(2):i–36, 2012.

Jill Burstein, John Sabatini, Jane Shore, Brad Moulder, and Jennifer Lentini. A user study: Technology to increase teachers' linguistic awareness to improve instructional language support for english language learners. *NLP4ITA 2013*, page 1, 2013.

Chris Callison-Burch. Fast, cheap, and creative: evaluating translation quality using amazon's mechanical turk. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 286–295. Association for Computational Linguistics, 2009.

Chris Callison-Burch and Mark Dredze. Creating speech and language data with amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 1–12. Association for Computational Linguistics, 2010.

Yllias Chali and Sadid A Hasan. Towards automatic topical question generation. *Proceedings of COLING 2012*, pages 475–492, 2012.

Shih-Chuan Chang. A contrastive study of grammar translation method and communicative approach in teaching english grammar. *English Language Teaching*, 4(2):13, 2011.

Danqi Chen and Christopher Manning. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 740–750, 2014.

Shu-Chen Cheng, Yen-Ting Lin, and Yueh-Min Huang. Dynamic question generation system for web-based testing using particle swarm optimization. *Expert systems with applications*, 36(1):616–624, 2009.

Maria Chinkina and Detmar Meurers. Linguistically-aware information retrieval: Providing input enrichment for second language learners. In *Proceedings of the*

11th Workshop on Innovative Use of NLP for Building Educational Applications, pages 188–198, San Diego, CA, 2016. URL `http://aclweb.org/anthology/W16-0521.pdf`.

Kevin Clark and Christopher D Manning. Deep reinforcement learning for mention-ranking coreference models. *arXiv preprint arXiv:1609.08667*, 2016.

Jacob Cohen. *Statistical power analysis for the behavioral sciences.* Academic Press, New York, 1977.

Joseph Collentine. Processing instruction and the subjunctive. *Hispania*, pages 576–587, 1998.

Kevyn Collins-Thompson, Paul N Bennett, Ryen W White, Sebastian de la Chica, and David Sontag. Personalizing web search results by reading level. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 403–412. ACM, 2011.

Averil Coxhead. A new academic word list. *TESOL quarterly*, 34(2):213–238, 2000.

Sérgio Curto, Ana Cristina Mendes, and Luísa Coheur. Question generation based on lexico-syntactic patterns learned from the web. *Dialogue & Discourse*, 3(2): 147–175, 2012. URL `https://doi.org/10.5087/dad.2012.207`.

Robert DeKeyser, Rafael Salaberry, Peter Robinson, and Michael Harrington. What gets processed in processing instruction? a commentary on bill vanpatten's "processing instruction: An update". *Language Learning*, 52(4):805–823, 2002.

Robert M DeKeyser and Karl J Sokalski. The differential role of comprehension and production practice. *Language Learning*, 46(4):613–642, 1996.

Catherine Doughty and John Williams, editors. *Focus on form in classroom second language acquisition.* Cambridge University Press, Cambridge, 1998.

Xinya Du, Junru Shao, and Claire Cardie. Learning to ask: Neural question generation for reading comprehension. *arXiv preprint*, 2017. URL `https://arxiv.org/pdf/1705.00106`.

Stephen C Ehrmann. Improving the outcomes of education: Learning from past mistakes. *Educause Review*, 37:54–55, 2002.

Rod Ellis. Focus on form: A critical review. *Language Teaching Research*, 20(3): 405–428, 2016.

Enrique Estellés-Arolas and Fernando González-Ladrón-De-Guevara. Towards an integrated crowdsourcing definition. *Journal of Information science*, 38(2):189–200, 2012.

Andrew P Farley. Authentic processing instruction and the spanish subjunctive. *Hispania*, pages 289–299, 2001.

Christiane Fellbaum, Joachim Grabowski, and Shari Land. Analysis of a hand-tagging task. *Tagging Text with Lexical Semantics: Why, What, and How?*, 1997.

Ian Fette and Alexey Melnikov. The websocket protocol. 2011.

Education First. Ef english proficiency index-a comprehensive ranking of countries by english skills. Technical report, Retrieved 2017-01-28, from http://www.ef.se/epi, 2017.

William H Fletcher. Facilitating the compilation and dissemination of ad-hoc web corpora. *Corpora and Language Learners*, 271, 2004.

Michael Flor and Brian Riordan. A semantic role-based approach to open-domain automatic question generation. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 254–263, 2018.

William Gale, Kenneth Ward Church, and David Yarowsky. Estimating upper and lower bounds on the performance of word-sense disambiguation programs. In *Proceedings of the 30th annual meeting on Association for Computational Linguistics*, pages 249–256. Association for Computational Linguistics, 1992.

Mélodie Garnier and Norbert Schmitt. Picking up polysemous phrasal verbs: How many do learners know and what facilitates this knowledge? *System*, 59:29–44, 2016.

Jesse James Garrett et al. Ajax: A new approach to web applications. 2005.

Susan M Gass and Evangeline Marlos Varonis. Input, interaction, and second language production. *Studies in second language acquisition*, 16(03):283–302, 1994.

Andrew Gelman and Jennifer Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Analytical Methods for Social Research. Cambridge University Press, 2006. doi: 10.1017/CBO9780511790942.

Aniruddha Ghosh, Guofu Li, Tony Veale, Paolo Rosso, Ekaterina Shutova, John Barnden, and Antonio Reyes. Semeval-2015 task 11: Sentiment analysis of figurative language in twitter. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 470–478, 2015.

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

David A Grossman. *Information retrieval: Algorithms and heuristics*, volume 15. Springer Science & Business Media, 2004.

Michael Heilman. *Automatic factual question generation from text*. PhD thesis, Carnegie Mellon University, 2011.

Michael Heilman and Noah A. Smith. Question generation via overgenerating transformations and ranking. Technical report, DTIC Document, 2009.

Michael Heilman and Noah A Smith. Rating computer-generated questions with mechanical turk. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's mechanical turk*, pages 35–40. Association for Computational Linguistics, 2010.

Jisup Hong and Collin F Baker. How good is the crowd at real wsd? In *Proceedings of the 5th linguistic annotation workshop*, pages 30–37. Association for Computational Linguistics, 2011.

Jeff Howe. *Crowdsourcing: How the power of the crowd is driving the future of business*. Random House, 2008.

Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. Data quality from crowdsourcing: a study of annotation selection criteria. In *Proceedings of the NAACL HLT 2009 workshop on active learning for natural language processing*, pages 27–35. Association for Computational Linguistics, 2009.

Jan Hulstijn. Implicit and incidental language learning: Experiments in the processing natural and partly artificial input. In H. W. Dechert and M. Raupach, editors, *Interlingual processing*, pages 49–73. Gunter Narr, Tübingen, 1989.

Otto Jespersen. *Essentials of English grammar.* Routledge, 2013.

Panayiota Kendeou and Paul Van Den Broek. The effects of prior knowledge and text structure on comprehension processes during reading of scientific texts. *Memory & cognition*, 35(7):1567–1577, 2007.

Adam Kilgarriff. I don't believe in word senses. *Computers and the Humanities*, 31(2):91–113, 1997.

Adam Kilgarriff and Gregory Grefenstette. Introduction to the special issue on the web as corpus. *Computational linguistics*, 29(3):333–347, 2003.

Adam Kilgarriff and Joseph Rosenzweig. English senseval: Report and results. In *LREC*, volume 6, page 2, 2000.

J Peter Kincaid, Robert P Fishburne Jr, Richard L Rogers, and Brad S Chissom. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, Naval Technical Training Command Millington TN Research Branch, 1975.

Christian Kohlschütter, Peter Fankhauser, and Wolfgang Nejdl. Boilerplate detection using shallow text features. In *Proceedings of the third ACM international conference on Web search and data mining*, pages 441–450. ACM, 2010.

Stephen Krashen. Some issues relating to the monitor model. *On Tesol*, 77(144-158), 1977.

Stephen D Krashen. *Explorations in language acquisition and use.* Heinemann Portsmouth, NH, 2003.

Daniël Lakens. Performing high-powered studies efficiently with sequential analyses. *European Journal of Social Psychology*, 44(7):701–710, 2014.

Daniel Lakens. Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. *Social Psychological and Personality Science*, 2017.

Maria Lapata and Alex Lascarides. Learning sentence-internal temporal relations. *Journal of Artificial Intelligence Research*, 27:85–117, 2006.

James F. Lee and Bill VanPatten. *Making communicative language teaching happen*, volume 1: Directions for Language Learning and Teaching. ERIC, 1995. URL `https://doi.org/10.2307/328644`.

John Lee. Verb tense generation. *Procedia-Social and Behavioral Sciences*, 27: 122–130, 2011.

Sang-Ki Lee and Hung-Tzu Huang. Visual Input Enhancement and Grammar Learning: A meta-analytic review. *Studies in Second Language Acquisition*, 30: 307–331, 2008. URL `https://doi.org/10.1017/s0272263108080479`.

Jennifer Leeman, Igone Arteagoitia, Boris Fridman, and Catherine Doughty. Integrating attention to form with meaning: Focus on form in content-based spanish instruction. *Attention and awareness in foreign language learning*, pages 217–258, 1995.

George B Leonard. *Education and ecstasy.* ERIC, 1968.

Ronald P. Leow, Hui-Chen Hsieh, and Nina Moreno. Attention to form and meaning revisited. *Language Learning*, 58(3):665–695, 2008. URL `https://doi.org/10.1111/j.1467-9922.2008.00453.x`.

Mark R Lepper. Microcomputers in education: Motivational and social issues. *American Psychologist*, 40(1):1, 1985.

Dirk Lewandowski. The retrieval effectiveness of web search engines: considering results descriptions. *Journal of Documentation*, 64(6):915–937, 2008.

Ming Liu, Rafael A Calvo, and Vasile Rus. Automatic question generation for literature review writing support. In *International Conference on Intelligent Tutoring Systems*, pages 45–54. Springer, 2010.

Shawn Loewen. Incidental focus on form and second language learning. *Studies in Second Language Acquisition*, 27(3):361–386, 2005. doi: 10.1017/S0272263105050163.

Michael H Long. Focus on form: A design feature in language teaching methodology. *Foreign language research in cross-cultural perspective*, 2(1):39–52, 1991.

Wenting Ma, Olusola O Adesope, John C Nesbit, and Qing Liu. Intelligent tutoring systems and learning outcomes: A meta-analysis. *Journal of Educational Psychology*, 106(4):901, 2014.

Nitin Madnani, Jill Burstein, John Sabatini, Kietha Biggers, and Slava Andreyev. Language muse: Automated linguistic activity generation for english language learners. *Proceedings of ACL-2016 System Demonstrations*, pages 79–84, 2016.

Prashanth Mannem, Rashmi Prasad, and Aravind Joshi. Question generation from paragraphs at upenn: Qgstec system description. In *Proceedings of QG2010: The Third Workshop on Question Generation*, pages 84–91, 2010.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, 2014. URL `http://www.aclweb.org/anthology/P/P14/P14-5010`.

Emma Marsden and Hsin-Ying Chen. The roles of structured input activities in processing instruction and the kinds of knowledge they promote. *Language Learning*, 61(4):1058–1098, 2011.

Elaine Marsh and Dennis Perzanowski. Muc-7 evaluation of ie technology: Overview of results. In *Proceedings of the seventh message understanding conference (MUC-7)*, volume 20, 1998.

David Martínez, Eneko Agirre, and Lluís Màrquez. Syntactic features for high precision word sense disambiguation. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

Diana Maynard and Mark A Greenwood. Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In *Lrec*, pages 4238–4243, 2014.

Karen Mazidi and Rodney D Nielsen. Linguistic considerations in automatic question generation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 321–326, 2014.

David McClosky, Eugene Charniak, and Mark Johnson. Effective self-training for parsing. In *Proceedings of the main conference on human language technology conference of the North American Chapter of the Association of Computational Linguistics*, pages 152–159. Association for Computational Linguistics, 2006.

Detmar Meurers. Natural language processing and language learning. *The Encyclopedia of Applied Linguistics*, 2012.

Detmar Meurers. Learner corpora and natural language processing. In Sylviane Granger, Gaëtanelle Gilquin, and Fanny Meunier, editors, *The Cambridge Handbook of Learner Corpus Research*, pages 537–566. Cambridge University Press, 2015.

Detmar Meurers, Ramon Ziai, Luiz Amaral, Adriane Boyd, Aleksandar Dimitrov, Vanessa Metcalf, and Niels Ott. Enhancing authentic web pages for language learners. In *Proceedings of the NAACL HLT 2010 Fifth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 10–18. Association for Computational Linguistics, 2010.

Rada Mihalcea. Instance based learning with automatic feature selection applied to word sense disambiguation. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

Rada Mihalcea. Co-training and self-training for word sense disambiguation. In *Proceedings of the Eighth Conference on Computational Natural Language Learning (CoNLL-2004) at HLT-NAACL 2004*, 2004.

George Miller. Wordnet: a lexical database for English. *Communications of the ACM*, 38(11):39–41, November 1995. URL `http://aclweb.org/anthology/H94-1111`.

Ruslan Mitkov, Le An Ha, and Nikiforos Karamanis. A computer-aided environment for generating multiple-choice test items. *Natural Language Engineering*, 12(2):177–194, 2006. doi: 10.1017/S1351324906004177.

Kara Morgan-Short, Jeanne Heil, Andrea Botero-Moriarty, and Shane Ebert. Allocation of attention to second language form and meaning. *Studies in Second Language Acquisition*, 34(04):659–685, 2012. URL `https://doi.org/10.1017/s027226311200037x`.

Bruce Morrison. Using news broadcasts for authentic listening comprehension. *ELT journal*, 43(1):14–18, 1989.

J. Mostow, J. E. Beck, J. Bey, A. Cuneo, J. Sison, B. Tobin, and J. Valeri. Using automated questions to assess readingcomprehension, vocabulary, and effects of tutorial interventions. *Technology, Instruction, Cognition and Learning*, 2(1-2): 97–134, 2004.

Robert Munro, Steven Bethard, Victor Kuperman, Vicky Tzuyin Lai, Robin Melnick, Christopher Potts, Tyler Schnoebelen, and Harry Tily. Crowdsourcing and language studies: the new generation of linguistic data. In *Proceedings of the NAACL HLT 2010 workshop on creating speech and language data with Amazon's Mechanical Turk*, pages 122–130. Association for Computational Linguistics, 2010.

Raymond Murphy. *English grammar in use.* Ernst Klett Sprachen, 2012.

William E Nagy and Patricia A Herman. Incidental vs. instructional approaches to increasing reading vocabulary. *Educational perspectives*, 23(1):16–21, 1985.

Ramanathan Narayanan, Bing Liu, and Alok Choudhary. Sentiment analysis of conditional sentences. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 180–189. Association for Computational Linguistics, 2009.

Roberto Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10, 2009.

Roberto Navigli, Kenneth C Litkowski, and Orin Hargraves. Semeval-2007 task 07: Coarse-grained english all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 30–35. Association for Computational Linguistics, 2007.

Madhu Neupane. Processing instruction: An input based approach for teaching grammar. *Journal of NELTA*, 14(1):111–118, 2009.

Hwee Tou Ng. Getting serious about word sense disambiguation. 1997.

Hwee Tou Ng and Hian Beng Lee. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 40–47. Association for Computational Linguistics, 1996.

Hwee Tou Ng, Bin Wang, and Yee Seng Chan. Exploiting parallel texts for word sense disambiguation: An empirical study. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 455–462. Association for Computational Linguistics, 2003.

Bikram Nobal Niraula and Vasile Rus. *Proceedings of the Tenth Workshop on Innovative Use of NLP for Building Educational Applications*, chapter Judging the Quality of Automatically Generated Gap-fill Question using Active Learning, pages 196–206. Association for Computational Linguistics, 2015. doi: 10.3115/v1/W15-0623. URL `http://aclweb.org/anthology/W15-0623`.

John M. Norris and Lourdes Ortega. Effectiveness of L2 instruction: A research synthesis and quantitative meta-analysis. *Language Learning*, 50(3):417–528, September 2000. doi: 10.1111/0023-8333.00136.

Michael O'Malley and Anna Chamot. *Learning Strategies in Second Language Acquisition*. Cambridge University Press, New York, 1990.

Niels Ott and Detmar Meurers. Information retrieval for education: Making search engines language aware. *Themes in Science and Technology Education*, 3(1-2): pp–9, 2011.

Yasuhiro Ozuru, Kyle Dempsey, and Danielle S McNamara. Prior knowledge, reading skill, and text cohesion in the comprehension of science texts. *Learning and instruction*, 19(3):228–242, 2009.

Martha Palmer, Hoa Trang Dang, and Christiane Fellbaum. Making fine-grained and coarse-grained sense distinctions, both manually and automatically. *Natural Language Engineering*, 13(2):137–163, 2007.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

Gabriel Parent and Maxine Eskenazi. Clustering dictionary definitions using amazon mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 21–29. Association for Computational Linguistics, 2010.

Matthew Peacock. The effect of authentic materials on the motivation of efl learners. *ELT journal*, 51(2):144–156, 1997.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos,

D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Wenzhe Pei, Tao Ge, and Baobao Chang. An effective neural network model for graph-based dependency parsing. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 313–322, 2015.

Ildikó Pilán, Elena Volodina, and Richard Johansson. Automatic selection of suitable sentences for language learning exercises. In L. Bradley and S. Thouësny, editors, *20 Years of EUROCALL: Learning from the Past, Looking to the Future. Proceedings of the 2013 EUROCALL Conference*, pages 218–225, 2013. URL `https://doi.org/10.14705/rpnet.2013.000164`.

Juan Pino and Maxine Eskenazi. Semi-automatic generation of cloze question distractors effect of students' l1. In *SLaTE*, pages 65–68, 2009.

Juan Pino, Michael Heilman, and Maxine Eskenazi. A selection strategy to improve cloze question quality. In *Proceedings of the Workshop on Intelligent Tutoring Systems for Ill-Defined Domains. 9th International Conference on Intelligent Tutoring Systems*, pages 22–34, Montreal, Canada, 2008.

Sameer S Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. Semeval-2007 task 17: English lexical sample, srl and all words. In *Proceedings of the 4th International Workshop on Semantic Evaluations*, pages 87–92. Association for Computational Linguistics, 2007.

Cristina Puente and José A Olivas. Analysis, detection and classification of certain conditional sentences in text documents. In *Proceedings of the 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, IPMU*, volume 8, pages 1097–1104, 2008.

R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2008. URL `http://www.R-project.org`. ISBN 3-900051-07-0.

Altaf Rahman and Vincent Ng. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language*

Processing: Volume 2-Volume 2, pages 968–977. Association for Computational Linguistics, 2009.

Janice Rattray and Martyn C Jones. Essential elements of questionnaire design and development. *Journal of clinical nursing*, 16(2):234–243, 2007.

Roi Reichart and Ari Rappoport. Tense sense disambiguation: a new syntactic polysemy task. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 325–334. Association for Computational Linguistics, 2010. URL http://aclweb.org/anthology/D10-1032.

Stephen E Robertson and Steve Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241. Springer-Verlag New York, Inc., 1994.

Peter Robinson. *Cognition and second language instruction*. Ernst Klett Sprachen, 2001.

Peter Robinson. Attention and memory during sla. *The handbook of second language acquisition*, pages 631–678, 2003.

Björn Rudzewitz, Ramon Ziai, Kordula De Kuthy, and Detmar Meurers. Developing a web-based workbook for english supporting the interaction of students and teachers. In *Proceedings of the Joint 6th Workshop on NLP for Computer Assisted Language Learning and 2nd Workshop on NLP for Research on Language Acquisition at NoDaLiDa, Gothenburg, 22nd May 2017*, number 134, pages 36–46. Linköping University Electronic Press, 2017.

Marta Sabou, Kalina Bontcheva, Leon Derczynski, and Arno Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC*, pages 859–866, 2014.

M Rafael Salaberry and Yasuhiro Shirai. *The L2 Acquisition of Tense Aspect Morphology*, volume 27. John Benjamins Publishing, 2002.

Beatrice Santorini. Part-of-speech tagging guidelines for the penn treebank project (3rd revision). *Technical Reports (CIS)*, page 570, 1990.

R. Schmidt. The role of consciousness in second language learning. *Applied Linguistics*, 11:206–226, 1990.

Donald J Schuirmann. A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability. *Journal of Pharmacokinetics and Pharmacodynamics*, 15(6):657–680, 1987.

Paul Seedhouse. Combining form and meaning. *ELT journal*, 51(4):336–344, 1997.

RJ Senter and Edgar A Smith. Automated readability index. Technical report, CINCINNATI UNIV OH, 1967.

Michael Sharwood Smith. Input enhancement in instructed SLA: Theoretical bases. *Studies in Second Language Acquisition*, 15:165–179, 1993. URL `https://doi.org/10.1017/s0272263100011943`.

M Kathleen Sheehan, Irene Kostin, and Yoko Futagi. Sourcefinder: a construct-driven approach for locating appropriately targeted reading comprehension source texts. In *SLaTE*, pages 80–83. Citeseer, 2007.

Luo Si and Jamie Callan. A statistical model for scientific readability. In *Proceedings of the tenth international conference on Information and knowledge management*, pages 574–576. ACM, 2001.

Amit Singhal, Chris Buckley, and Mandar Mitra. Pivoted document length normalization. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–29. ACM, 1996.

D. I. Slobin, editor. *The crosslinguistic study of language acquisition.* L. Erlbaum Associates, Hillsdale, NJ, 1985.

Edgar A Smith and RJ Senter. Automated readability index. *AMRL-TR. Aerospace Medical Research Laboratories (6570th)*, page 1, 1967.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics, 2008.

Richard Socher, John Bauer, Christopher D Manning, et al. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 455–465, 2013.

Neil Stewart, Jesse Chandler, and Gabriele Paolacci. Crowdsourcing samples in cognitive science. *Trends in cognitive sciences*, 2017.

Seymour Sudman and Norman M Bradburn. *Asking questions: A practical guide to questionnaire design.* San Francisco Calif. Jossey-Bass Publishers 1983., 1982.

Janet K Swaffar. Reading authentic texts in a foreign language: A cognitive model. *The Modern Language Journal*, 69(1):15–34, 1985.

Merrill Swain. Communicative competence: Some roles of comprehensible input and comprehensible output in its development. *Input in second language acquisition*, 15:165–179, 1985.

Martha Trahey and Lydia White. Positive evidence and preemption in the second language classroom. *Studies in second language acquisition*, 15(02):181–204, 1993.

Victoria Tsiriga and Maria Virvou. Evaluating the intelligent features of a web-based intelligent computer assisted language learning system. *International journal on artificial intelligence tools*, 13(02):411–425, 2004.

Alexandra L Uitdenbogerd. Using the web as a source of graded reading material for language acquisition. In *Advances in Web-Based Learning-ICWL 2003*, pages 423–432. Springer, 2003.

Sowmya Vajjala and Detmar Meurers. On improving the accuracy of readability classification using insights from second language acquisition. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 163–173. Association for Computational Linguistics, 2012.

Sowmya Vajjala and Detmar Meurers. On the applicability of readability models to web texts. In *Proceedings of the Second Workshop on Predicting and Improving Text Readability for Target Reader Populations*, pages 59–68, 2013.

Sowmya Vajjala and Detmar Meurers. Assessing the relative reading level of sentence pairs for text simplification. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL-14)*, Gothenburg, Sweden, 2014. Association for Computational Linguistics. URL `http://purl.org/dm/papers/Vajjala.Meurers-14-eacl.html`.

C. J. van Rijsbergen. *Information Retrieval.* Butterworth-Heinemann, Newton, MA, USA, 2nd edition edition, 1979.

Bill VanPatten. Attending to form and content in the input. *Studies in second language acquisition*, 12(03):287–301, 1990. URL https://doi.org/10.1017/s0272263100009177.

Bill VanPatten. Processing instruction: An update. *Language learning*, 52(4): 755–803, 2002.

Bill VanPatten. *Processing instruction: Theory, research, and commentary.* Routledge, 2004. URL https://doi.org/10.4324/9781410610195.

Bill VanPatten and Teresa Cadierno. Explicit instruction and input processing. *Studies in Second Language Acquisition*, 15(02):225–243, 1993. doi: 10.1017/S0272263100011979. URL http://dx.doi.org/10.1017/S0272263100011979.

Bill VanPatten and Soile Oikkenon. Explanation versus structured input in processing instruction. *Studies in Second Language Acquisition*, 18(04):495–510, 1996. URL https://doi.org/10.1017/s0272263100015394.

Mark Warschauer and Deborah Healey. Computers and language learning: An overview. *Language teaching*, 31(02):57–71, 1998.

Thomas Wasow. Gradient data and gradient grammars. In *Proceedings from the Annual Meeting of the Chicago Linguistic Society*, volume 43, pages 255–271. Chicago Linguistic Society, 2007. URL https://web.stanford.edu/~wasow/Wasow_CLS.pdf.

Lydia White, Nina Spada, Patsy M Lightbown, and Leila Ranta. Input enhancement and l2 question formation. *Applied linguistics*, 12(4):416–432, 1991.

Jessica Williams and Jacqueline Evans. What kind of focus and on which forms. *Focus on form in classroom second language acquisition*, 139, 1998.

John H Wolfe. Automatic question generation from text-an aid to independent study. *ACM SIGCSE Bulletin*, 8(1):104–112, 1976.

Wynne Wong. Modality and attention to meaning and form in the input. *Studies in Second Language Acquisition*, 23(03):345–368, 2001. URL https://doi.org/10.1017/s0272263101003023.

Wynne Wong. Processing instruction in French: The roles of explicit information and structured input. *Processing instruction: Theory, research, and commentary*, pages 187–205, 2004. URL https://doi.org/10.4324/9781410610195.

Richard W Woodcock et al. *Woodcock reading mastery tests, revised.* American Guidance Service Circle Pines, MN, 1987.

Graham Workman. *Concept questions and time lines.* Gem Publishing, 2008.

Shaoqun Wu, Margaret Franken, and Ian H Witten. Refining the use of the web (and web search) as a language teaching and learning resource. *Computer Assisted Language Learning*, 22(3):249–268, 2009.

Xuchen Yao and Yi Zhang. Question generation with minimal recursion semantics. In *Proceedings of QG2010: The Third Workshop on Question Generation*, pages 68–75. Citeseer, 2010.

David Yarowsky. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196. Association for Computational Linguistics, 1995.

David Yarowsky and Radu Florian. Evaluating sense disambiguation across diverse parameter spaces. *Natural Language Engineering*, 8(4):293–310, 2002.

Omar F Zaidan and Chris Callison-Burch. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1220–1229. Association for Computational Linguistics, 2011.

Lishan Zhang and Kurt VanLehn. How do machine-generated questions compare to human-generated questions? *Research and Practice in Technology Enhanced Learning*, 11(1):7, 2016.

Ramon Ziai, Niels Ott, and Detmar Meurers. Short answer assessment: Establishing links between research strands. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pages 190–200. Association for Computational Linguistics, 2012.

Nicole Ziegler, Detmar Meurers, Patrick Rebuschat, Simon Ruiz, José L Moreno-Vega, Maria Chinkina, Wenjing Li, and Sarah Grey. Interdisciplinary research at the intersection of call, nlp, and sla: Methodological implications from an input enhancement project. *Language Learning*, 67(S1):209–231, 2017.

Justin Zobel and Alistair Moffat. Exploring the similarity space. In *ACM SIGIR Forum*, volume 32, pages 18–34. ACM, 1998.