

Thesis  
**Automatic Extraction and Application of Language  
Descriptions for Under-Resourced Languages**

Aditi Swanand Chaudhary

CMU-LTI-22-011

Language Technologies Institute  
School of Computer Science  
Carnegie Mellon University  
Pittsburgh, PA 15123

**Thesis Committee:**

Graham Neubig (Chair)	Carnegie Mellon University
Alan Black	Carnegie Mellon University
David R. Mortensen	Carnegie Mellon University
Antonios Anastasopoulos	George Mason University
Isabelle Augenstein	University of Copenhagen

*Submitted in partial fulfillment of the requirements  
for the degree of Doctor of Philosophy  
in Language and Information Technologies*

Copyright © 2022 Aditi Swanand Chaudhary



*This thesis is dedicated to my family, especially my parents Dr. Mrs. Shubhalakshmi and Dr. Swanand Chaudhary, for their support and encouragement have made everything possible!*



## Abstract

Languages of our world are amazingly diverse, consisting of varied and complex systems of word, phrase, sentence construction, and vocabulary, to name a few. Understanding these systems is critical not only for language communication, but they also drive the design and development of language technologies. Creating a language description that illustrates such salient points of a language is therefore one of the major endeavours undertaken by language experts, and in fact, forms an indispensable step for language documentation and preservation efforts (Himmelmann, 1998; Moline, 2020). Manually creating such detailed descriptions for several languages that are usable by humans and machines can be challenging; therefore, in this thesis we explore *whether we can automate some of the processes involved in the language description creation and create language descriptions in a format usable by both humans and machines.*

Thanks to advances in natural language processing (NLP) research, we can automate some *local* aspects of linguistic analysis, such as identifying the syntactic function of a word (POS tagging) or identifying grammatical relations (dependency parsing). We take advantage of such advances to extract and explain complex linguistic behaviors, covering aspects of morphology, syntax, and lexical semantics that apply to language in general. To achieve this goal, we develop a system AUTOLEX<sup>1</sup> which automatically extracts these linguistic insights in a human- and machine-readable format for several languages. In the first part of the thesis, we describe this general framework, which takes as input a text corpus of the language of interest and a linguistic question that we are interested in exploring. AUTOLEX converts this into an NLP prediction task and produces a concise description which answers that question. As part of this framework, we develop manual and automatic evaluation methods to evaluate the resulting descriptions. We further demonstrate the application of these language descriptions in real-world settings of language analysis and education.

In the second part of the thesis, we describe how to improve the NLP building blocks that inform AUTOLEX, particularly for under-resourced languages. Most state-of-the-art methods that are involved in the building blocks (e.g. performing local linguistic analysis like POS tagging) require an abundance of labeled data, which is often not readily available for many languages. Therefore, we focus on improving these methods for such under-resourced languages. Specifically, we explore: 1) *Cross-lingual Transfer Learning* (CLTL) (Zoph et al., 2016), which leverages existing labeled data and models from high-resource languages and, 2) *Active Learning* (Lewis and Gale, 1994; Settles and Craven, 2008) (AL) which helps train models by collecting labeled data in the under-resourced language while minimizing human annotation effort. We propose combining both in a unified framework where CLTL helps improve the performance of the AL learner.

---

<sup>1</sup><https://aditi138.github.io/auto-lex-learn/index.html>



## Acknowledgments

I would like to express my immense gratitude to my mentors, colleagues, collaborators, family and friends, without whom this thesis would not have been possible. First and foremost, I would like to thank my advisor, Graham Neubig, who taught me everything I know about doing research, right from formulating a research question to designing the modeling and evaluating strategies, and most importantly to think about the implications of the work beyond just a paper submission. I also learnt a lot from him about how to be a good researcher, how to present one's ideas clearly to different audiences, how to be proactive and establish collaborations even outside of the department, and more importantly to not be afraid to work on something different. I am really glad that he encouraged and supported my interest in building something tangible through my PhD which has real-world implications. An important quality that I have learnt from him is also how to promote our work and to reach out to experts in need. During any project, if there is someone who is more knowledgeable about one aspect, Graham will encourage us to collaborate and take advice from the expert in question. This has helped produce high-quality research in his lab. Graham's vast knowledge and his enthusiasm for this field made working with him super fun. I am also grateful to Jaime Carbonell, who was my advisor in the initial years of the PhD and during my MLT, for giving me the opportunity in the first place. He inspired me to work on under-resourced languages through the LORELEI program and his enthusiasm for building applications having real-world impact motivated me to build tangible applications in my PhD. His passion for the field and what we as researchers can do to help people with our research, continues to inspire me.

Next, I would also like to thank Alan Black, from whom I have learnt so much. I have been collaborating with him since the LORELEI project and the discussions I have had with him has helped shape my research agenda. I would also like to thank David Mortensen, he has been the go-to person for any question I had about language or linguistics in general. He has been a wonderful collaborator and I have learnt a lot from him, especially the importance of incorporating linguistic knowledge in the systems we build, and how best to make use of those insights. I would also like to thank Antonios Anastasopoulos, who has also been a co-advisor to me in my PhD, and a close collaborator. He is really fun to work with and his experience in working with endangered languages and especially working with the language communities, has helped me understand and learn how to establish and foster long-term collaborations with the communities, how to identify problems which are not only interesting from a research point of view but also have utility to the communities themselves who are ultimately the users of the research. This thesis would not be complete without thanking Isabelle Augenstein, for being part of my thesis committee. Her feedback on my thesis proposal right from the beginning has helped shape my work since then, and I am grateful to her especially for encouraging me and highlighting the importance of considering how AUTOLEX can be useful for different target audiences, which has in turn helped shaped my general research direction and overall thought process.

I am extremely grateful to all my collaborators and friends from whom I have learnt so much not only about the field but also about research and life in general. I would like to

especially thank Chunting Zhou and Sai Krishna Rallabandi, both of whom were mentors to me from early on. I had many wonderful discussions with them that helped shaped my thinking. CMU in general is a wonderful place to be, I feel fortunate to have been in the company of so many smart and kind people from whom I have learnt a lot, including Danish Pruthi, Uri Alon, Divyansh Kaushik, Sanket Vaibhav Mehta, Sachin Kumar, Adithya Pratapa, Zaid Sheikh, and many others. I would also like to express my gratitude to the CMU staff as well, who took care of so many administrative and logistical aspects of being a PhD student, which allowed me to completely focus on my research without worrying about any of the details. This thesis would not have been possible without the funding support from the Waibel Presidential Fellowship, Dr. Robert Sansom Fellowship, National Science Foundation grants 1761548. Also, I would like to thank the many human experts and annotators who took the time out of their busy schedules to help with manual evaluation, annotation, and even consultation, as none of the research would be meaningful or even possible without their participation.

As part of my PhD, I feel fortunate to have interned with Google Research twice and I would like to thank both my mentors, Kartik Raman and Partha Talukdar, for guiding me and for encouraging my interests to work on Indian languages.

Finally, I am extremely grateful to my entire family for their constant support and encouragement all throughout my life, especially, my parents, Dr. Shubhalakshmi and Dr. Swanand Chaudhary, who instilled in me the importance of higher education, hard work and perseverance. They have always been there for me, supported me emotionally when things felt difficult and I was about to give up, they gave me the self-confidence and boost that helped me believe in myself again. Despite not being familiar with the world of computer science, they even strived hard to understand my world, attended each of my talks, became my practice audience, and even brainstormed with me on how I could use my research to help language learning for Indian languages. I am also thankful to my brother, Ameya Chaudhary, and my sister-in-law, Ankita Wyawahare, for also supporting me throughout, helping me to think clearly and make right choices, and also for just being there for me. I would also like to express gratitude towards Dr. Sandhya and Narendra Atre, Dr. Jyoti and Dr. Ashok Bidwai, for being my local guardians during my time in Pittsburgh, having family support closeby is a true treasure. I would like to acknowledge my grandmother Dr. Mrs. Pratibha Mohgaonkar, who has been a guiding light for me, especially in the moments when I was about to give up. Her perseverance for education and her confidence in me since childhood, gave me the strength to achieve my goals. Whenever I was down and felt low in confidence, remembering her helped me regain my motivation. I would also like to acknowledge my grandfather, Dr. Prabhakar Mohgaonkar, Mr. Eknath Chaudhary and grandmother, Mrs. Mandakini Chaudhary, for supporting my dreams and showering me with their blessings.



### **For the Reader**

Dear reader, if you are interested in exploring AUTOLEX<sup>2</sup>, which is the tool we built to explore the salient properties of different languages, without going into the technical details of the underlying models, I suggest reading [Chapter 2](#) after the introduction ( [Chapter 1](#)). To understand the technical pieces that led to this tool, I have organized this thesis into two parts, where [Part I](#) describes how each component in AUTOLEX was designed and [Part II](#) describes how to improve those individual components for under-resourced languages.

---

<sup>2</sup><https://autolex.co>



# Contents

- 1 Introduction** **1**
  - 1.1 Research Goals and Scope . . . . . 2
  - 1.2 Thesis Outline . . . . . 5
  - 1.3 Contributions . . . . . 9
  
- 2 AUTOLEX: A Tool to Explore Language Descriptions** **11**
  - 2.1 Background . . . . . 11
  - 2.2 AUTOLEX . . . . . 14
  - 2.3 Statement of Limitations . . . . . 20
  
- I Extracting Language Descriptions of Natural Languages Automatically.** **23**
  
- 3 Automatically Extracting Linguistic Descriptions for Agreement** **25**
  - 3.1 Overview . . . . . 25
  - 3.2 Proposed Approach . . . . . 26
  - 3.3 Experimental Settings . . . . . 31
  - 3.4 Gold-Standard Experiments . . . . . 34
  - 3.5 Under-resource Experiments . . . . . 37
  - 3.6 Limitations . . . . . 40
  - 3.7 Conclusion . . . . . 41
  
- 4 A General Framework for Extracting Linguistic Descriptions** **43**
  - 4.1 Overview . . . . . 43
  - 4.2 Proposed Approach . . . . . 45
  - 4.3 Experimental Settings . . . . . 49
  - 4.4 Gold-Standard Experiments . . . . . 52
  - 4.5 Hmong Daw Study . . . . . 56
  - 4.6 Other Applications . . . . . 57
  - 4.7 Conclusion . . . . . 57
  
- 5 L2 Semantic Subdivisions** **59**
  - 5.1 Overview . . . . . 59
  - 5.2 Proposed Approach . . . . . 61

5.3	Experimental Settings . . . . .	65
5.4	Other Applications . . . . .	73
5.5	Conclusion . . . . .	73
<b>6</b>	<b>ASSIST-A-TEACHER: Teacher Perception of Automatically Extracted Grammar Concepts for Language Learning</b>	<b>75</b>
6.1	Overview . . . . .	75
6.2	Proposed Approach . . . . .	78
6.3	Experimental Setting . . . . .	83
6.4	Results . . . . .	84
6.5	Conclusion . . . . .	91
 <b>II Leveraging Existing and New Data for Improving NLP for Under-resourced Languages.</b>		<b>93</b>
<b>7</b>	<b>Adapting Word Representations to New Languages using Linguistically-Motivated Information</b>	<b>95</b>
7.1	Overview . . . . .	96
7.2	Background . . . . .	98
7.3	Proposed Approach . . . . .	100
7.4	Experimental Settings . . . . .	101
7.5	Experiments . . . . .	103
7.6	Conclusion . . . . .	105
<b>8</b>	<b>Bootstrapping Active Learning with Cross-Lingual Transfer Learning</b>	<b>107</b>
8.1	Overview . . . . .	107
8.2	Background . . . . .	108
8.3	Query Strategies . . . . .	109
8.4	Active Learning for NER . . . . .	111
8.5	Experimental Settings . . . . .	115
8.6	Simulation Experiments . . . . .	116
8.7	Human Annotation Experiments . . . . .	118
8.8	Conclusion . . . . .	120
<b>9</b>	<b>Confusion Reducing Active Learning</b>	<b>123</b>
9.1	Overview . . . . .	123
9.2	Background: Failings of Query Strategies . . . . .	125
9.3	Proposed Approach . . . . .	126
9.4	Experimental Settings . . . . .	129
9.5	Simulation Experiments . . . . .	131
9.6	Human Annotation Experiments . . . . .	135
9.7	Conclusion . . . . .	137

**10 Conclusion and Future Directions** **139**  
10.1 Summary of Contributions . . . . . 139  
10.2 Future Directions . . . . . 141

**Bibliography** **145**



# List of Figures

1.1	Highlighting the salient subject-verb word order patterns in Marathi language, along with the conditions which trigger these. The dominant order is SV i.e. subjects come before verbs but there are a significant number of instances where this order deviates. . .	3
1.2	Teaching a learner which Spanish words to use for the English word <i>wall</i> and when one of the Spanish words is preferred over the other. . . . .	3
1.3	AUTOLEX overview: Given a linguistic question and use-case by a user, we highlight the different steps in the pipeline, where we use automatic methods to extract language descriptions. In this framework, language users and experts not only benefit from the extracted descriptions but can also help annotate and evaluate the intermediate steps involved in the process. . . . .	6
1.4	Outline of the thesis. . . . .	8
2.1	Order of object and verb in English as extracted by AUTOLEX. . . . .	12
2.2	Homepage of AUTOLEX which describes the different linguistic phenomena available for different languages. . . . .	15
2.3	Visualize salient information about grammatical gender in Marathi. The top figure shows the gender distribution across each POS tag, and the bottom figure shows some illustrative examples (e.g. pronouns). . . . .	16
2.4	Gender agreement rules extracted by AUTOLEX for Greek. . . . .	18
2.5	Illustrative examples for the rule ‘Gender need not match when Noun is the modifier in Greek’ (Figure 2.4). Positive examples denote when examples that follow the rule and model prediction, whereas the negative examples show any exceptions. . . . .	19
2.6	Rules explaining when the nominative case is used for Turkish nouns. . . . .	20
2.7	Different types of suffix (inflections) added for Marathi words. For example, suffix ‘ne’ is used typically for subjects in accusative case. Another way of explaining the usage is through its English counterpart, for example, its usage is similar to the usage of ‘by’ in English. . . . .	20
2.8	Semantic subdivision for the concept ‘oil’ results in different lexical manifestations in Spanish: ‘petróleo’ for petroleum oil and ‘aceite’ for cooking oil whereas in English both are referred as ‘oil’. . . . .	21

3.1	An overview of our method’s workflow for gender agreement in Greek. The example sentence translates to “The port of Igoumenitsa is connected to many ports in Italy and Albania.” First, we dependency parse and morphologically analyze raw text to create training data for our binary agreement classification task. Next, we learn a decision tree to extract the rule set governing gender agreement, and label the extracted leaves as either representing required or chance agreement. Finally these rules are presented to a linguist for perusal. . . . .	27
3.2	Subject-verb number agreement is required in Spanish, as in example A.1, which renders example A.2 ungrammatical. Object-verb agreement is not required, so both B.1 and B.2 are grammatical. The object and the verb in B.1 only agree by chance. . . . .	28
3.3	Extracting gender agreement rules in Spanish. (a) A decision tree is learned over dependency link triples, inducing a distribution of agreement over examples in each leaf. However, simple majority voting leads to false positives: Leaf-1 includes more agreeing data points, but in reality this agreement is purely by chance. (b) With a statistically-inspired threshold to label the leaves, Leaf-1 gets correctly labeled as <i>chance-agreement</i> . (c) We merge leaves with the same label to get a concise representation. Every dependency link triple receives the label of the unique leaf it falls under. . . . .	30
3.4	Comparing the UD (a) tree with the SUD (b) tree for the German sentence “Ich werde lange Bücher lesen.”. . . . .	32
3.5	Annotation interface for evaluating number agreement in English . . . . .	33
3.6	Difference in the ARM scores of decision trees over gold-standard syntactic analysis with baseline trees where all leaves predict <i>chance-agreement</i> . . . . .	35
3.7	Correlation between size of the decision trees constructed by our framework and morphological complexity of languages. . . . .	36
3.8	Annotation accuracy for Greek, Russian and Catalan per each morphological feature. . .	36
3.9	Comparing the (avg.) ARM score for NUMBER agreement with and without cross-lingual transfer learning (transfer language in parenthesis). <i>x</i> -axis in log space. The higher the ARM the better. . . . .	38
3.10	In most cases our framework (shaded bars) extracts a good first-pass specification for <i>true</i> zero-shot settings. Solid bars indicate the baseline. . . . .	40
4.1	An overview of the AUTOLEX framework being applied for understanding word order, with Adj-N order in Spanish as an example. The example sentence translates to <i>Four books were bought by the small girl</i> . First, we formulate a linguistic question (e.g. regarding Adj-N order) as a binary classification task (e.g. “whether the Adj comes before/after the N”). Next, we perform syntactic analysis on the raw text, from which we extract syntactic, lexical, and semantic features to construct the training data. Finally, we learn an interpretable model from which we extract concise rules. . . . .	44
4.2	Illustrating the free word order in Hindi and how the grammatical role of subjects and objects is expressed through the post-position (-ne for ergative, -ko for accusative). . . .	45



4.3	Examples of case variation in Greek nouns. For the above sentence – “these <u>provisions</u> correspond to the standards”, the underlined <u>noun</u> takes the nominative case because it is the subject of the main verb. The pronoun also takes the nominative case because its the determiner of the noun which is the subject. A noun takes the accusative case when it is the object. . . . .	47
4.4	A rule extracted for Spanish adjective-noun word order. . . . .	49
4.5	Rule evaluation form presented to the language expert. . . . .	51
4.6	(left) Comparing the effect of different features on the word order and case marking. (right) Comparing the accuracy of the model across different treebanks of fr-gsd. . .	54
4.7	Evaluating rule correctness (left), prior knowledge (middle) and feature correctness (right). Top plot shows the results for English while the bottom plot shows for Greek. . . . .	55
5.1	Semantic subdivision for the concept ‘wall’ results in different lexical manifestations in Spanish: ‘muro’ for <i>outside wall</i> and ‘pared’ for <i>inside wall</i> whereas in English both are referred as ‘wall’. . . . .	60
5.2	Distribution of the number of lexical choices for each POS tag. . . . .	66
5.3	Learning interface used by Spanish learners. A learner is required to select the appropriate choice using the provided English context and mark how confident they are in their answer. . . . .	67
5.4	Learning Interface. Descriptions of rules (extracted from the lexical selection model) are provided to the learner before the start of the exercise. . . . .	68
5.5	Learning Interface. Rules for the correct answer are displayed to the learner after each question. Individual rules that apply to the given example are highlighted for the convenience of the learner. . . . .	69
5.6	Learner accuracy and confidence in correct answers with and without access to rules against the number of attempted examples ( $x$ -axis ). Learners achieve higher accuracy with increasing confidence with fewer examples when they have access to rules. . . . .	71
5.7	Rules help more for words where learners do worse. $x$ -axis is the (avg.) learner accuracy (without rules) for first 20 examples. . . . .	72
5.8	Rules help more for words where model performs well. $x$ -axis is model accuracy per word. 73	73
6.1	Marathi words organized by basic categories. Each word contains a link to illustrative examples with their English translations. . . . .	80
6.2	Marathi adjectives extracted by AUTOLEX. . . . .	81
7.1	Subword units of a word in Hindi . . . . .	97
7.2	Similarity between Hindi and Bengali words becomes more apparent as phonemes are able to capture the relatedness between similar languages, despite their orthographic differences. . . . .	99
8.1	An overview of the Active Learning process. . . . .	109

8.2	Our proposed recipe: cross-lingual transfer is used for projecting annotations from an English labeled dataset to the target language. Entity-targeted active learning is then used to select informative sub-spans which are likely entities for humans to annotate. Finally, the NER model is fine-tuned on this partially-labeled dataset. . . . .	112
8.3	Comparison of the NER performance trained with the FineTune scheme, across six datasets. Solid lines compare the different token-level strategies. Dashed lines show the ablation experiments. The x-axis denotes the total number of tokens annotated and the y-axis denotes the F1 score. . . . .	117
8.4	Examples from Hindi human annotation experiments for both ETAL and SAL. Square brackets denote the spans (for ETAL) or the entire sequence (for SAL) selected by the AL strategy. . . . .	120
8.5	Comparing the number of entities in the data selected by ETAL and SAL, as annotated by oracle. . . . .	120
8.6	Example of the human annotation process for Hindi. . . . .	121
9.1	Illustration of selecting representative token-tag combinations to reduce confusion between the output tags on the German token ‘die’ in an idealized scenario where we know true model confusion. . . . .	124
9.2	Comparing the difference in POS performance across the AL methods with BRNN/MLP architecture, averaged across 20 iterations. . . . .	131
9.3	Our method (CRAL) outperforms existing AL methods for all six languages. Y-axis is the difference in POS accuracy between CRAL and other AL methods, averaged across 20 iterations with batch size 50. . . . .	132
9.4	Confusion score measures the percentage of correct predictions in the first iteration which were incorrectly predicted in the second iterations. Lower values suggest that the selected annotations in the subsequent iterations cause less damage on the model trained on the existing annotations. . . . .	133
9.5	In the <i>oracle</i> setting, our method (CRAL-ORACLE) outperforms UNS-ORACLE and QBC-ORACLE in most cases, while the non-oracle CRAL matches the performance of its oracle counterpart. y-axis measures the difference in average accuracy across 20 iterations. . . .	134
9.6	We report the mean and median of $\mathbf{p}$ over all the 50 token-tag pairs selected by the first AL iteration of CRAL. We see that across all languages majority of the token-tag pairs satisfy the criteria of using weighted representations with centroid for token selection. .	135
10.1	Summary of the different approaches to support a new language. Inspired from Graham Neubig’s course CS 11-747 <a href="#">slides</a> . . . . .	140

# Chapter 1

## Introduction

While languages of our world are amazingly diverse, all languages obey a set of principles, also known as ‘grammar’, that provide a framework for meaningful communication. There are separate principles which govern the different systems in a language, such as the system of sounds, word formation, phrase and sentence construction, the system of assigning meanings, and so on. Creating ‘language descriptions’ that describe these systems in as natural a setting as possible (Harris, 1954) is, therefore, of great value for language understanding and communication. Such descriptions further form an indispensable component of *language documentation* which aims to create a lasting multi-purpose record of a language (Himmelmann, 1998). They, therefore, play a crucial role in the process of language preservation and revitalization of indigenous languages which are often endangered (Hale et al., 1992; Moseley, 2010). Saving indigenous languages is important not only for communities to preserve their cultural heritage, but also for preserving the deep historical knowledge carried by these languages throughout several generations (Nunn and Reid, 2016). Language descriptions also form the basis for the development of language technologies, which has also been cited as important for language survival. As noted by Williams (2019) “languages that miss the opportunity to adopt language technologies will be less and less used”.<sup>1</sup>

In this thesis, the term *language descriptions* refers to a set of concise text descriptions through which the salient linguistic properties of a language can be explained. For example, for a language, such a description can provide answers to linguistic questions such as ‘what are the nouns and verbs in that language’ or ‘how should the nouns be positioned with respect to the verbs’ or ‘which word should be used for *rice* in that language’, and so on. Manually creating such descriptions that cover the different linguistic behaviors in a format that can be consumed by both humans and machines is a challenging process, as this not only requires considerable human effort and time, but also such human experts might not be readily available. There are more than 7000 languages (Hammarström, 2015) in the world today, of which several of the languages that are on the verge of extinction often do not have easily accessible trained linguists or native speakers, making this a challenging task. Therefore, we explore the question of *how natural language processing (NLP) can help automate the process of a language description creation*.

The past two decades have seen significant advances in the fields of deep learning, machine learning, and natural language processing (NLP), and we can leverage these advances to automate some of the

---

<sup>1</sup>A cautionary point to keep in mind when building resources, descriptions, models for indigenous languages is that we should not bring in colonial bias but rather respect and consider viewpoints of the native language users while designing Williams (2019).

processes involved in creating language descriptions. For example, the popular NLP tasks of POS tagging (Toutanova and Manning, 2000), dependency parsing (Kiperwasser and Goldberg, 2016), machine translation (Koehn, 2020) can essentially provide answers to the questions of ‘what are the nouns and verbs’ or ‘what are the syntactic relations between words e.g. whether the noun is a subject or object of a verb’ or ‘which word to use for *rice*, say, in Marathi’. Interestingly, these same core tasks could also be leveraged to answer questions at the language level, as we show in this thesis. For example, a linguist interested in knowing about the typical word order of a language (e.g. whether it is subject-verb-object or subject-object-verb), require to know a) which word is the verb (i.e. perform POS tagging), b) which words are the subject and the object (i.e. perform dependency parsing), and c) combine the two pieces of information effectively to extract and explain the salient patterns. Thanks to the advances brought about by deep learning methods such as *neural networks* (Goldberg, 2017), which can automatically discover patterns and features in the underlying data, we have achieved notable gains in the accuracy of these core tasks (e.g. POS tagging (Ma and Hovy, 2016), dependency parsing (Dyer et al., 2016; Kulmizev et al., 2019), morphological analysis (Malaviya et al., 2018; Kondratyuk and Straka, 2019)). However, a major bottleneck in using these deep models is the availability of good quality and quantity of language resources required for training these models. Because of this, most NLP research has shown these notable gains for a subset of high-resourced languages such as English, which have these resources easily available and thus enable training the large and data hungry models (e.g. BERT (Devlin et al., 2019), ELMO (Peters et al., 2018)). With the development of multilingual datasets such as Universal Dependencies (UD) (Nivre et al., 2016), WikiAnn (NER) (Pan et al., 2017), XNLI (Conneau et al., 2018)), models (mBERT (Devlin et al., 2019)) and benchmarks (XTREME (Hu et al., 2020), Xglue (Liang et al., 2020)), these advancements are now increasingly being seen even for under-resourced languages, which lack sufficient resources for the task at hand, by leveraging the commonalities between languages.

## 1.1 Research Goals and Scope

The aim of this thesis is to *help automate the processes involved in the creation of language descriptions and visualize the extracted descriptions in a human- and machine-readable format*. Specifically, we propose the AUTOLEX framework which extracts and visualizes the salient language patterns, along with illustrative examples, from a text corpus of a language of interest. Since we are interested in extracting these descriptions for all languages of the world, many of which are under-resourced, this thesis also describes the steps taken for improving the syntactic analysis for such under-resourced languages.

Broadly, language descriptions provide guidelines to produce a grammatically correct and comprehensible sentence, for example, in this thesis we will present descriptions that can describe the relative position of words and phrases (*word order*), the syntax and semantics of the arguments of a predicate (*argument structure*), the patterns of word formation (*morphological agreement and inflection*) and the word meanings (*lexical semantics*). An example of such a language description extracted for Marathi is shown in Figure 1.1. This description highlights the salient word order patterns along with the conditions under which they are typically observed. According to Shieber (2003), the choice of metalanguage in any grammar formalism should be determined by the following criteria: *language felicity*, the extent to which the grammar descriptions can explain the linguistic phenomenon as a user wishes to see them, *expressiveness*, which informs the kind of phenomena that can be covered, and *computational effectiveness*, which checks

[Back to Marathi-ENMRLEM page](#)

Order of **subjects** with respect to the syntactic head **verb**

The dominant order in the corpus is **before**

Word Order
Generally the word order for <b>subject-verb is before</b> i.e. <b>subject before verb</b>
Some examples are: <a href="#">Examples</a>
subject is <b>after</b> verb when:
verb is also governing= काय (kaay) subject is nearby= compound subject is governed by a word with <a href="#">Aspect</a> = Simp ( <a href="#">Examples</a> ) <b>OR</b>
verb is also governing= काय (kaay) subject is a= proper noun subject is governed by a word with <a href="#">Tense</a> = Past subject is nearby= म्हण (mhan) ( <a href="#">Examples</a> ) <b>OR</b>

Figure 1.1: Highlighting the salient subject-verb word order patterns in Marathi language, along with the conditions which trigger these. The dominant order is SV i.e. subjects come before verbs but there are a significant number of instances where this order deviates.

L1: English	L2: Spanish	Rules	Examples
wall.NOUN	pared/paredón	<b>Short phrases:</b> (face, wall), (hang, wall), (picture, wall), (back, wall) (right, wall), (write, wall), (stand, wall)  <b>Words:</b> ear, hang, room, picture face, write, stand, back, four, hand	Examples
wall.NOUN	muralla/muro/muros	<b>Short phrases:</b> (climb, wall), (city, wall), (brick, wall), (jump, wall) (behind, wall), (outside, wall)  <b>Words:</b> break, climb, man, high within, jericho, garden, jump, stone, city, outside, build	Examples

Figure 1.2: Teaching a learner which Spanish words to use for the English word *wall* and when one of the Spanish words is preferred over the other.

whether descriptions can be interpreted by machines. Based on this, in AUTOLEX, descriptions can take different forms depending on the target audience. For example, for a linguist exploring a language, the answers to specific questions (e.g. word order) can be presented using a linguistic schema (e.g. UD annotation scheme) that contains detailed syntactic information (as shown in [Figure 1.1](#)). However, for a language learner (especially beginner learners), understanding a grammar concept through concrete examples without drowning them in too many linguistic details would be more beneficial ([Figure 1.2](#)).

Typically, language descriptions, as created by language experts, often span hundreds of pages.<sup>2</sup> Researchers or other stakeholders interested in using these descriptions often require them to manually parse all information, making it a tedious and time-consuming process not only for content curators, but also for the users. There have been efforts to digitize this information, for example, WALS ([Dryer and Haspelmath, 2013](#)) describes the structural properties of different languages as gathered from reference grammars, all these properties (grammatical, phonological, lexical) are described at a much coarse-grained level, but most of these properties, in fact, often vary significantly at the phrasal level. For example, WALS describes Marathi as having word order SOV, however, as we saw in [Figure 1.1](#), there are cases where this order shows deviation. Furthermore, WALS lacks descriptions of many grammar aspects (e.g. when does a noun show agreement with the verb and when is that not required). Additionally, the information in the description itself is often created manually by field linguists and native speakers, and can suffer from human bias. For example, *inter-linear glossed text* (IGT) is often one of the first resources created by field linguists in their analysis process. IGT provides brief linguistic information about morphosyntax structure, which is often accompanied by translations which are collected in resources such as ODIN ([Lewis and Xia, 2008](#)).<sup>3</sup> While a linguist carefully chooses examples to create the IGT corpus such that they are representative of the linguistic phenomena of interest, insights derived from IGT may suffer from this bias as the data does not encompass many naturally occurring examples (details in [Chapter 2](#)). Similarly, language learning books typically contain manually created simple examples which illustrate one grammar concept at a time, but real-life communication often contains varied linguistic phenomena. [Jones and Waller \(2015\)](#) note that most English textbook writers did not consult a corpus when writing them, but rather relied on their own intuition or followed other textbooks. But [Long \(2000\)](#) argues that explaining language *only* deductively can get overwhelming for learners, and does not expose learners to real-life usage. Furthermore, language varies considerably across different contexts (e.g. formal vs informal, spoken vs written, news vs social media, etc), and manually introspecting these differences or finding relevant illustrations, is a challenging task, even for trained linguists or other curators.

Through AUTOLEX, we hope to help ease this process by showing how to extract salient language patterns from any available text corpus, along with relevant examples which illustrate each pattern. The goal is not to replace human experts but rather to assist them in their process (e.g. language documentation, language teaching, etc.) by creating automatic tools. Within the different systems covered in a typical language description, we focus on the systems of morphology, syntax, and semantics, to some extent. Specifically, we select those linguistic phenomena to study that have been widely identified and studied by linguists and form a core part of language understanding and learning. More concretely, we

---

<sup>2</sup><https://linguistic-typology.org/grammarwatch/>

<sup>3</sup>Definition adapted from <http://linguistics-ontology.org/gold/2010/InterlinearGlossedText>

outline the research goals as follows:

- A framework to automatically discover and extract language descriptions from a text corpus, to answer questions about word order, argument structure, morphological agreement and inflection, lexical semantics.
- An online interface that allows users to visualize these descriptions along with illustrative examples, which are available in both human- and machine-readable formats for numerous languages.

The general workflow of AUTOLEX is as follows – (1) Given a linguistic question we want to answer for a language (e.g. how are subjects arranged with respect to the verbs in Marathi), we first formulate the question as a prediction task (e.g. predict whether the subject is before/after the verb). (2) From the text corpus of that language, identify and extract the features which we believe govern this phenomenon (e.g. POS tagging and dependency parsing to identify subjects, verbs, and other syntactic features) to construct the training data. (3) Learn a prediction model from which human- and machine-readable descriptions can be extracted. (4) Visualize the extracted description through an online interface. Within this framework, we also propose methods to perform an automatic evaluation when manual evaluation is unavailable or infeasible. We envision AUTOLEX to be a machine-in-the-loop system, where human experts can be both end users and input source (Figure 1.3). For example, as described in Chapter 2, some features (used in step (2)) such as POS tags, dependency parses, have been collected by language experts for many languages, and can be used directly as input for step (3). Given that these are created by human experts, for many languages, these annotations are very limited in size and linguistic variety. However, as we will show in Part II, there are NLP and machine learning innovations which we can leverage to extract these features for such under-resourced languages. For this, we explore approaches that leverage both existing data, which relies on commonalities between the languages, and by collecting new data in the under-resourced language for the task at hand.

## 1.2 Thesis Outline

In this section, we describe the outline of the thesis:

- Chapter 2 presents an overview of the different features supported in AUTOLEX, without going into the technical details. This chapter also describes relevant prior work which inspired the methods used in developing AUTOLEX.
- Part I describes the technical framework behind AUTOLEX including the design, evaluation, and its real-world applications. We demonstrate the usability of AUTOLEX for three target audiences, *linguists*, *language learners*, and *language teachers*. We first explain the AUTOLEX design in detail by extracting language descriptions for one language phenomenon, the morphological agreement process. Next, we show how to generalize this design to other language phenomena such as word order, case marking, lexical semantics.
  - Chapter 3 proposes a method for automatically extracting rules describing the morphological agreement process across several languages. We describe these rules over syntactic features and find that our framework is able to extract decent first-pass agreement rules even for under-resourced languages by leveraging existing syntactic features from related languages. We evaluate our extracted rules with the help of language experts and further propose an

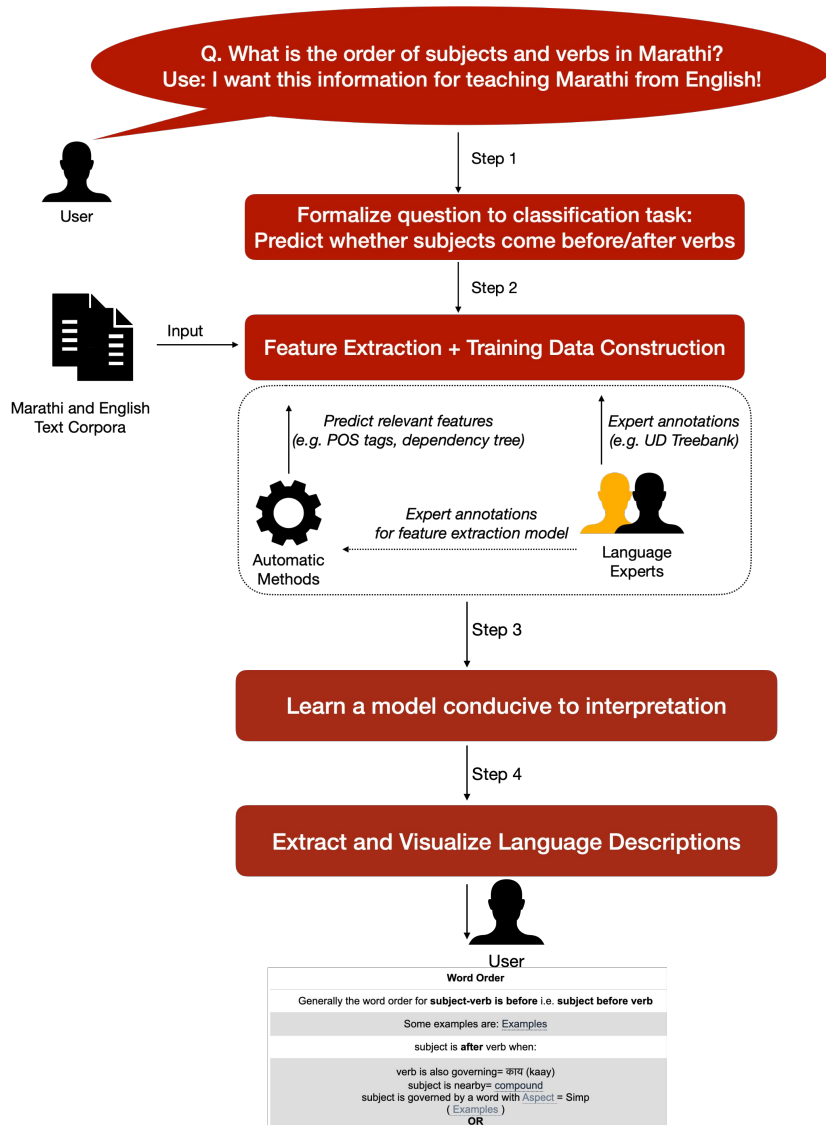


Figure 1.3: AUTOLEX overview: Given a linguistic question and use-case by a user, we highlight the different steps in the pipeline, where we use automatic methods to extract language descriptions. In this framework, language users and experts not only benefit from the extracted descriptions but can also help annotate and evaluate the intermediate steps involved in the process.



automated evaluation method which helps verify the rules in absence of language experts.

- [Chapter 4](#) builds on the framework used in [Chapter 3](#) and proposes a general framework to automatically extract rules that describe various linguistic phenomena such as case marking and word order, in addition to morphological agreement. In addition to automatically evaluating the quality of the extracted descriptions, we also conduct a user study with language experts to evaluate how correct, readable, and novel the descriptions are perceived. Finally, we apply this framework to an endangered language variety, Hmong Daw, to evaluate how well the framework extracts descriptions under true zero-resource conditions.
- [Chapter 5](#) proposes a method for automatically extracting rules describing fine-grained semantic distinctions that display different lexical manifestations in a second language. This has applications in language learning, particularly for second language vocabulary acquisition from a first or native language. We evaluate our automatically extracted rules with human learners and find our rules to facilitate both faster and more effective learning than without them.
- [Chapter 6](#) combines the approaches described in [Chapter 4](#) and [Chapter 5](#) to semi-automatically create a curriculum for second language learning. Specifically, we explore how much of the language material extracted by AUTOLEX is practically relevant and usable by language teachers who are actively involved in the teaching of Indian languages of Marathi and Kannada to English speakers. For this, we establish a collaboration with the Kannada Academy in North America, which aims at teaching Kannada (a Dravidian language), and with two Marathi schools (Marathi Vidyalay, New Jersey and Marathi Shala, Pittsburgh) who are teaching Marathi (a Indo-Aryan language) primarily to learners outside of India. This collaborative study is conducted to not only evaluate the quality of our extracted descriptions, but also to understand how these descriptions could assist the teachers in their teaching process. Overall, the teachers find our materials to be interesting as they cover the non-dominant linguistic behaviors or the exceptions to general rules, which is relevant to their teaching process. They especially like the illustrative examples shown for each grammar aspect, as a helpful reference material for their own lesson preparation or even for learner evaluation.
- [Part II](#) describes some of the building blocks used in [Part I](#). As shown in [Figure 1.3](#), step (2) entails *feature extraction* i.e. identifying and extracting features that govern the linguistic phenomenon of interest. We present techniques that can leverage both existing and new data to improve feature extraction for under-resourced languages. Specifically, we demonstrate these methods for two type of feature: *syntactic* (POS tagging) and *semantic* (Named Entity Recognition (NER)).
  - [Chapter 7](#) proposes a method for adapting continuous word representations using linguistically motivated subword units (phonemes, morphemes, and graphemes). We use this method for leveraging resources from higher-resourced related languages, thereby improving generalization to under-resourced languages. The main motivation of this approach is to map the word representations of the under-resourced and higher-resourced languages in the same space, which allows the neural models to better leverage the existing resources for improving task performance on the under-resourced languages.
  - [Chapter 8](#) proposes a framework for improving entity recognition for under-resourced lan-

<b>Background</b>			<b>Target audience:</b> Linguists, language educators, NLP researchers	Chapter 2
Language Descriptions	Morpho-Syntax	Agreement Word Order Case Marking	Linguists, NLP researchers	Chapter 3 Chapter 4 Chapter 4
	Lexical-Semantics	Semantic Subdivision	Language learners, teachers, NLP researchers	Chapter 5
	Applications	Language Education	Language learners, teachers	Chapter 6
NLP methods for Under-resourced Languages	Cross-lingual transfer learning		NLP researchers	Chapter 7
	Active learning		NLP researchers, language experts	Chapter 8 Chapter 9

Figure 1.4: Outline of the thesis.

guages. This framework proposes to leverage existing resources from related higher-resourced languages and collect new resources in the target language from native speakers. Within this framework, we propose a novel strategy for collecting entity annotations efficiently which aims to reduce the annotation effort without compromising on the task performance. We evaluate our framework under both simulated settings, where we simulate data collection using gold-labeled data and, true human annotation settings where we collect data from native speakers of the language.

- [Chapter 9](#) extends the framework proposed in [Chapter 8](#) for the POS tagging task. Like before, we leverage both existing data from related languages and also collect new POS annotations in the target under-resourced languages. We propose a novel strategy to collect POS annotations by reducing the confusion between possible POS tags. We show its effectiveness in the simulated and true human annotation settings, where we collect POS tags for an endangered language Griko.

## 1.3 Contributions

In [Figure 1.4](#) we outline the structure of our thesis, along with the expected target audience that would find that piece of work of interest or useful. For example, [Chapter 3](#) and [Chapter 4](#) introduce the general framework for extracting descriptions regarding morpho-syntax and we show how both linguists (for language exploration) and NLP researchers (for model evaluation) would benefit from these. [Chapter 5](#) and [Chapter 6](#) show applications of the framework for language education and thus language learners or teachers would find this more useful. [Chapter 7](#), [Chapter 8](#), [Chapter 9](#) describes cross-lingual transfer and active learning to improve NLP models for under-resourced methods. This would probably be more useful for researchers involved in improving NLP models generally, having applications to newer NLP tasks going beyond the ones covered in these works.



## Chapter 2

# AUTOLEX: A Tool to Explore Language Descriptions

In the previous chapter, we briefly introduced our language explorer tool AUTOLEX, which presents holistic human- and machine-readable descriptions extracted automatically across several languages. AUTOLEX is designed with the motivation to provide NLP researchers, language experts, learners, teachers, or just curious enthusiasts a platform to explore a plethora of languages in a consistent format. In this chapter, we present a general overview of the language aspects covered by AUTOLEX and other similar efforts, without going into the technical modeling details. We also discuss similar prior efforts undertaken for language documentation, learning, and teaching. The tool can be explored online<sup>1</sup>.

## 2.1 Background

As mentioned earlier, languages are amazingly diverse with complex systems governing syntax, morphology, semantics, pragmatics, and phonology, to name a few. Understanding these complex systems is crucial not only for language understanding and communication, but also to drive the design and development of several language technologies. This means that there is a need for language descriptions which are not only human-readable but also machine-readable. Below we present some prior efforts undertaken along this direction.

### 2.1.1 Linguistic Databases and Datasets

Linguists and researchers have undertaken initiatives to collect linguistic properties in a machine-readable format in several languages, WALS (Dryer and Haspelmath, 2013) being a standing example. WALS is a database describing structural properties (phonological, grammatical, lexical) of a language as gathered from reference grammars. For instance, it can tell us that English objects occur after verbs, or that Turkish pronouns have symmetrical case. Currently, WALS contains such properties for over 1000+ languages, however, because WALS presents these properties across many diverse languages, these properties are necessarily defined at a coarse-grained level and cannot capture language-specific nuances. WALS does not inform us of any exceptions to its general rules (e.g. the cases when English

---

<sup>1</sup><https://aditi138.github.io/auto-lex-learn/index.html>

objects come before verbs), and there are many aspects that are not even covered (e.g. when a Turkish pronoun takes the accusative marker and when the nominative). In AUTOLEX, we aim to cover such fine-grained properties, for example, Figure 2.1 is showing the cases where objects come before verbs in English. In Chapter 4 we present the methodology for extracting such fine-grained descriptions. PHOIBLE (Moran et al., 2014), Ethnologue (Hammarström, 2015) and Glottolog (Nordhoff and Hammarström, 2011; Hammarström et al., 2018) are similar such collection of linguistic properties in different formats. Although these databases cover 1000+ languages, many of the properties are missing or undocumented. Parallel efforts (Malaviya et al., 2017; Bjerva et al., 2020) have looked at methods to predict these missing properties.

In addition to documenting the coarse-grained properties, the research community has also led efforts to collect fine-grained properties such as POS tags, morphological features, dependency parses across several languages – Universal Dependencies (UD) is one such community-led project<sup>2</sup> (Nivre et al., 2006; Nivre et al., 2018), covering 200 treebanks over 100 languages. Most of these datasets are annotated by language experts and, therefore, are limited in size and domain coverage. However, advances in neural networks (Kondratyuk and Straka, 2019; Kulmizev et al., 2019; Nguyen et al., 2021) have made it possible, to some extent, to learn from this limited data and acquire more data automatically from raw text. In Chapter 7, Chapter 8, Chapter 9, we present some of these methods in more detail, where we specifically look at adapting and improving existing NLP methods for under-resourced languages.

### 2.1.2 Grammar Rule Extraction

We are not the first to look at answering linguistic questions about language automatically, there have been several threads of work. For instance, while documenting a language and its grammar, one of the first resources created by linguists is the inter-linear glossed text (IGT). IGT contains information about the morphosyntax structure such as POS, morphemes, different morphosyntactic features and values. These are also accompanied with word translations and sometimes with phonetic information as well.<sup>3</sup> ODIN corpus (Lewis and Xia, 2010; Xia et al., 2014) is a collection of IGT data curated from linguistic annotations in several languages. Prior work (Lewis and Xia, 2008; Hellan, 2010; Bender et al., 2013; Howell et al., 2017) has proposed methods to map the information present in IGT to existing grammar formalisms (e.g. head-phrase structure grammar (HPSG) (Pollard and Sag, 1994) or lexical-functional grammar (LFG) (Kaplan et al., 1981)) such that it is machine-readable. Lewis and Xia (2008) enrich the IGT data with syntactic structures to determine the canonical word order and case marking observed in the language. One drawback of using IGT as the starting point is that, while a linguist carefully chooses examples to create the IGT corpus such that they are representative of the linguistic phenomena of in-

Word Order
Generally the word order for <b>object-verb is after</b> i.e. <b>object after verb</b>
Some examples are: <u>Examples</u>
object is <b>before</b> verb when:
object has lemma= that ( <u>Examples</u> )
<b>OR</b>
object is governed by a word with <u>Tense</u> = Pres object has lemma= what ( <u>Examples</u> )

Figure 2.1: Order of object and verb in English as extracted by AUTOLEX.

<sup>2</sup><https://universaldependencies.org/>

<sup>3</sup><http://linguistics-ontology.org/gold/2010/InterlinearGlossedText>

terest, at the same time this may lead to confirmation bias, as the data may focus on phenomena that the linguist found particularly interesting and not encompass many of the naturally occurring examples. Furthermore, given that IGT is manually curated, it is limited in size, domain and the variety of linguistic phenomena covered. [Bender et al. \(2013\)](#) extract major constituent word order and case marking properties from the IGT for a diverse set of languages. Potentially, grammar rules can also be derived from existing projects such as the LinGO Grammar Matrix ([Bender et al., 2002](#)), ParGram ([Butt et al., 2002](#); [King et al., 2005](#)). These are grammar development tools designed to write and create grammar specifications that support a wide range of languages in a unified format. They focus on mapping simple descriptions of languages, obtained from existing IGT-annotated data or input from a linguist, to precision grammar fragments, grounded in a grammar formalism such as HPSG or LFG. Another thread of work focuses on answering specific questions about language from natural text, such as the analysis of word order ([Östling, 2015](#); [Wang and Eisner, 2017](#)). Our work differs from prior work in that we seek to discover and explain the linguistic behaviors of the language in a format understandable by both humans and machines. Currently, we extract these descriptions using the UD annotation schema ([McDonald et al., 2013](#)) as this schema offers a consistent annotation of grammar (POS tags, dependency parses, and morphological analyses), allowing us to also represent the descriptions in a consistent format across all languages. Additionally, AUTOLEX does not extract rules for an individual sentence in isolation, as some of the HPSG/LFG-based approaches do, but rather extracts patterns that generalize across the language as a whole. Most importantly, it discovers these behaviors from naturally occurring sentences, reflecting how the language is used in the world. Given that we use the UD formalism as our underlying schema, it provides us with the flexibility to extract and inspect patterns directly from the raw text, as UD has a relatively wide coverage of datasets and state-of-the-art models built using the same schema.

### 2.1.3 Computer Assisted Tools

Computer-assisted tools have long been used for language documentation, understanding, and learning. We describe a subset of them.

**Annotation Tools** The first step in any language documentation process is data collection, which is made more efficient with annotation tools. For example, ELAN ([Sloetjes and Wittenburg, 2008](#)) is a popular annotation tool for audio and video recordings and has been used under varied contexts such as for language documentation of endangered languages such as Engdewu ([Vaa, 2013](#)), speech transcription of child bimodal corpora [Pichler et al. \(2010\)](#) and sign language transcription [Zahedi et al. \(2006\)](#). FLEx ([Butler and Van Volkinburg, 2007](#)) is another such tool, which is used for data management and analysis primarily by field linguists for documentation purposes. It has been used for dictionary collection for indigenous languages such as Choctaw ([Anumpa and Himona, 2016](#)) and Matsigenka ([Pereira et al., 2011](#)). BRAT ([Stenetorp et al., 2012](#)) and INCEpTION ([Klie et al., 2018](#)) have been widely used in the NLP community to collect both syntactic and semantic annotations. BRAT has also been used for data visualization in the UD project ([Nivre et al., 2016](#)).

**Learning Tools** In recent years, there has been an increasing interest in learning new languages for both personal and professional purposes. This has led to an influx of language learning toolkits such as

Rosetta Stone (Stone, 2010), Duolingo<sup>4</sup>, LingQ<sup>5</sup>, LearnALanguage<sup>6</sup>, Omniglot<sup>7</sup> and many more. Most of these tools curate learning content manually with the help of subject matter experts, which, however, makes it difficult to extend them to numerous languages. GrammarTagger (Hagiwara et al., 2021) is another resource developed for language education that identifies useful grammatical features for learning, currently supporting English and Chinese. Such computer-assisted language learning (CALL) systems have been increasingly using NLP techniques to create learning content. We discuss more about CALL systems in Chapter 5 and Chapter 6.

**Grammar Description Tools** Although the above language learning tools provide grammar resources for multiple languages, these resources are typically focused on language learning and therefore discuss only commonly used phrases and constructions. Such resources do not delve deep into grammatical aspects such as syntax, word order, morphology agreement, sentence construction, etc, which typically constitute a grammar description. Works aimed at providing grammar descriptions do so mostly at an individual language level, such as for Sanskrit<sup>8</sup> where a grammar guide is provided based on existing grammar books, and Russian<sup>9</sup> where a self-study guide including grammar summary tables and vocabulary lessons is provided. To the best of our knowledge, there is no one toolkit that presents the grammar descriptions for many languages in a unified format. Having descriptions in a unified format allows for easy comparison between languages and provides easy extensibility to incorporate new languages.

## 2.2 AUTOLEX

AUTOLEX is a tool for visualizing language descriptions, where a first-pass set of rules is extracted using an automated framework from raw text in a concise, human-and machine-readable format. As mentioned in the Introduction (Chapter 1) we extract descriptions of linguistic phenomena covering aspects of syntax, morphology, and lexical semantics. Typically, a grammar description starts with describing different word classes or part-of-speech (POS) in a given language, dedicating separate chapters or sections for each of them (e.g. in Lhomi (Vesalainen, 2014), North Tanna (Sverredal, 2018), Fuyug (Bradshaw, 2007)). Under each word class section, the morphological properties and examples that describe the function of the word class are described in detail. For example, a section for ‘nouns’ in Marathi would describe that nouns have three grammatical genders (feminine, masculine, and neuter), three number properties (singular, plural, dual), and grammatical case (accusative, nominative, dative, genitive, instrumental, locative, vocative, ablative). The section would further describe each morphological property in detail, including any inflection patterns and/or grammatical agreement observed. Any irregular forms or exceptions are also included in the same section. They also describe common derivational morphology patterns which help understand the process of word production. Any specific information pertaining to the given word class are also added as subsections. Next sections describe the sentence structure in detail, which includes the general word order and any exceptions to it. For example, in Hindi the unmarked

---

<sup>4</sup><https://www.duolingo.com/>

<sup>5</sup><https://www.lingq.com/en/grammar-resource/>

<sup>6</sup><https://www.learnalanguage.com/>

<sup>7</sup><https://www.omniglot.com/>

<sup>8</sup><https://www.learnanskrit.org/>

<sup>9</sup><http://www.russianforeveryone.com/>



### AutoLEX: An Automatic Framework for Linguistic Exploration

AutoLEX is a tool for exploring language structure and provides an automated framework for extracting a first-pass grammatical specification from raw text in a concise, human-and machine-readable format.

Along with the language structure, we also provide rules to help with vocabulary learning, which we also extract automatically.

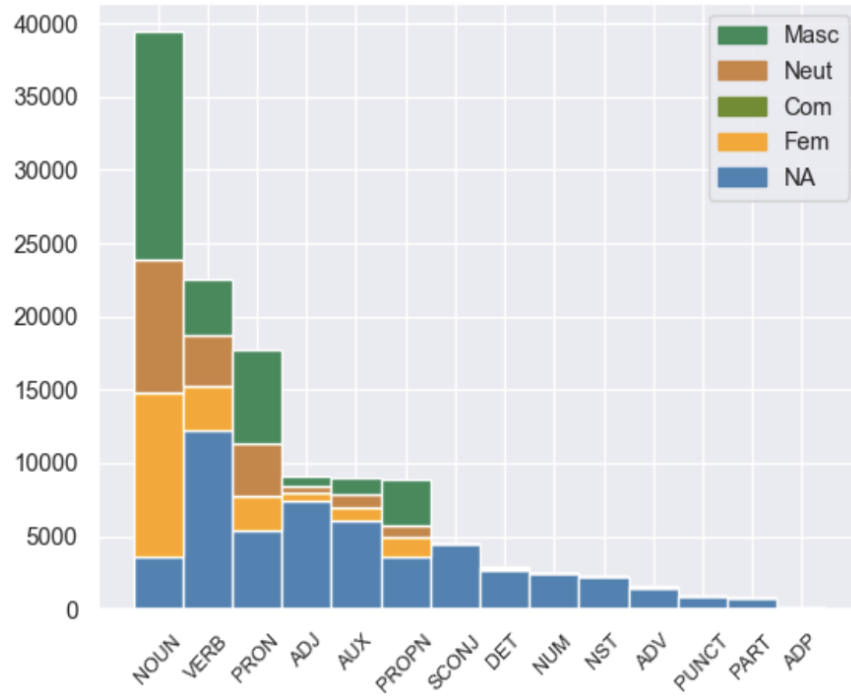
We apply our framework to all languages of the [Syntactic Universal Dependencies project](#).

Here are the languages (and treebanks) we currently support.

Search for language (e.g. English)

ISO	Language	Treebank	Linguistic Analysis
en	English	EWT	General Information Agreement WordOrder CaseMarking
el	Greek	GDT	General Information Agreement WordOrder CaseMarking Learn Vocab
es	Spanish	GSD	General Information Agreement WordOrder CaseMarking Learn Vocab
mr	Marathi	SAM-EN	General Information Learn Vocab WordOrder Suffix Usage Agreement

Figure 2.2: Homepage of AutoLEX which describes the different linguistic phenomena available for different languages.



Lemma	Morphosyntactic Attributes	Gender			
		Neut	NA	Fem	Masc
तो (to)	Acc;Plur	त्यांना (tyana)	-	-	त्यांना (tyana)
तो (to)	Acc;Sing	त्याच (tyaach)	-	त्याच (tyaach)	त्याला (tyaalaa)
ते (to)	Nom;Sing	ते (te)	-	-	ते (to)
तो (to)	Nom;Plur	-	-	त्या (tya)	ते (te)
तो (to)		-	तोही (tohi)	-	-

Figure 2.3: Visualize salient information about grammatical gender in Marathi. The top figure shows the gender distribution across each POS tag, and the bottom figure shows some illustrative examples (e.g. pronouns).

word order is SOV, however, when both locative and accusative cases are used in the same sentence, the word order is flexible since both case markers are lexically different, thus marking their role explicitly. This section is then followed by the phrase and clause structure sections. The noun phrase (NP) chapter, for instance, describes the constituent order, morphology, and modifiers accepted by the NPs. Different types of clauses (transitive and intransitive) are described in detail in a separate chapter. After the phrase and clause structure chapters, different types of sentence constructions are discussed, such as interrogatives, declaratives, yes/no questions, imperatives, negation, wh-questions. Word lists describing the basic concepts (body parts, verbs, natural phenomenon, etc) are often included in the grammar descriptions. Apart from these universal concepts, they could also include culture-specific words. Inspired by this structure, we design AUTOLEX to follow a similar structure and describe the following linguistic phenomena within these broad fields:

- **Word Order**, which describes the relative position of constituents in a sentence (Täckström et al., 2013) (Chapter 4).
- **Morphological Agreement**, wherein a word or morpheme selects morphemes in correspondence with another word or phrase in the sentence (Corbett, 2009) (Chapter 3, Chapter 4).
- **Case Marking**, which marks syntactic dependents for the type of grammatical relation they bear to their heads (Blake, 2009) (Chapter 4).
- **Morphology Inflection**, which describes the process of word formation, where the form of the lexeme changes based on different grammatical contexts (Lieber, 2009) (Chapter 6).
- **Semantic Subdivisions**, wherein the fine-grained semantic distinctions in one language displays different lexical manifestations in another language (Chapter 5).

An overview of the tool can be seen in Figure 2.2. In order to extract consistent descriptions catering to many different languages, we use multilingual resources which have data annotated consistently across the different languages such as UD/SUD project (Nivre et al., 2016; Gerdes et al., 2018, 2019), wherever possible. We describe the salient features of the AUTOLEX interface in the following sections.

### 2.2.1 General Information

For each language, we present salient information, which describes the different syntactic and morphological properties observed at a token level. For example, Figure 2.3 informs us whether Marathi exhibits any grammatical gender, if so, what are the different gender values and which syntactic categories typically exhibit them. This information is also visualized here.<sup>10</sup> We further extract examples of the tokens for each POS tag. We organize these examples by their lemmas, also showing other morphological values marked by the respective examples. As we can see, for some words there are blanks under the different gender columns. Blanks in some words indicate that a particular noun can be expressed in only a single gender but for some words they could also denote missing word forms.

### 2.2.2 Morph-Syntactic Information

We discover and present morpho-syntactic information such as word order, morphological agreement, inflection, case marking, for each language (wherever applicable). Specifically, we aim to answer a

---

<sup>10</sup>[https://aditi138.github.io/auto-lex-learn/mr\\_en/helper/Gender\\_PRON.html](https://aditi138.github.io/auto-lex-learn/mr_en/helper/Gender_PRON.html)

## Rules for Gender agreement for NOUN

The Gender values **should match** between the **NOUN** and its governor (i.e syntactic head) when **label = should-match**, else any observed agreement is purely by chance (**label = need-not-match**)

Agreement
Gender need not match between the NOUN and its governor or head when:
NOUN is governed by= conjunct (Examples) OR
NOUN is nearby= numeral (Examples) OR
NOUN is governed by= conjunct NOUN is nearby= και (Examples) OR
NOUN is the= modifier (Examples)
Generally Gender should match between the NOUN and its governor or head
Some examples are: Examples

Figure 2.4: Gender agreement rules extracted by AUTOLEX for Greek.

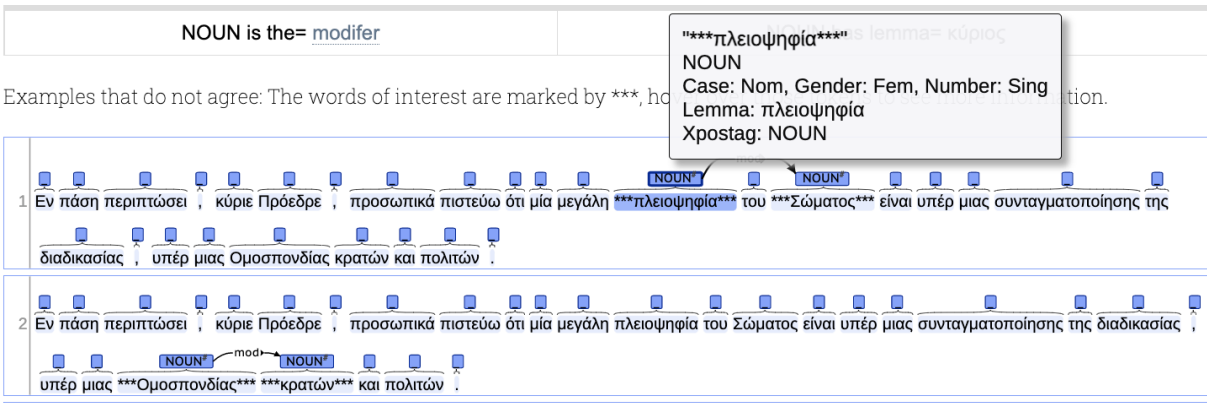
linguistic question with human-readable explanations, which are accompanied with automatically extracted illustrative examples.

**Morphological Agreement** The agreement process typically entails making values of certain morphological attributes (e.g. gender, number) agree or match between the words in the sentence. In AUTOLEX, we ask the question *when is agreement required between a head and its dependent for a morphological attribute  $m$* . We focus on the morphological attributes  $M = \{\text{gender, person, number}\}$ , which more often show agreement than other attributes (Corbett, 2009). Example gender agreement rules learnt for Greek nouns is shown in Figure 2.4.<sup>11</sup> We include illustrative examples with each rule to show the user how the rule is applied in natural language. We show both positive examples, which follow the said rule and the model’s predicted label and, negative examples which show any exceptions to that rule, as shown in Figure 2.5. The methodology used for automatic extraction is described in Chapter 3.

**Word Order** Word order describes the relative position of the syntactic elements, and is one of the major axes of linguistic description appearing in grammar sketches or databases such as WALS. We consider the following five relations: subject-verb, object-verb, adjective-noun, adposition-noun and numeral-noun. In contrast to WALS, which only provides a single canonical order for the entire language, we pose the linguistic question as determining *when does one word in such a relation appear before or after the other*. Figure 2.1 shows example word order rules extracted for English objects and verbs.<sup>12</sup> The methodology used is described in Chapter 4.

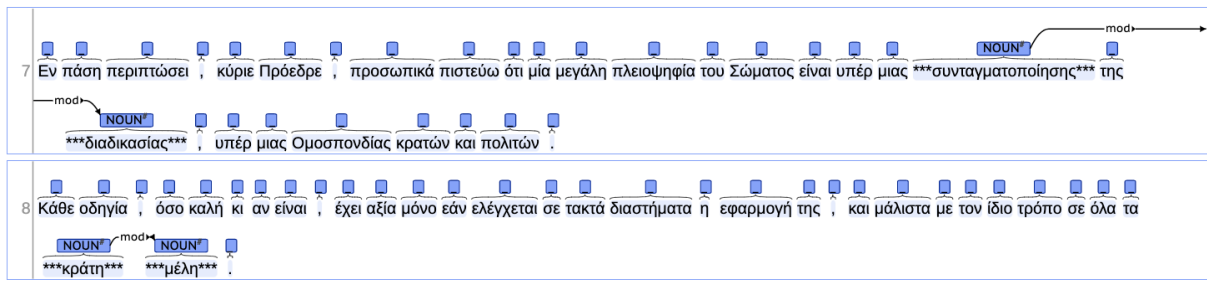
<sup>11</sup>[https://aditi138.github.io/auto-lex-learn/el\\_gdt/Agreement/Gender/NOUN/NOUN.html](https://aditi138.github.io/auto-lex-learn/el_gdt/Agreement/Gender/NOUN/NOUN.html)

<sup>12</sup>[https://aditi138.github.io/auto-lex-learn/en\\_ewt/WordOrder/object-verb/object-verb.html](https://aditi138.github.io/auto-lex-learn/en_ewt/WordOrder/object-verb/object-verb.html)



(a) Positive Examples

Examples that agree: The words of interest are denoted by \*\*\*



(b) Negative Examples

Figure 2.5: Illustrative examples for the rule ‘Gender need not match when Noun is the modifier in Greek’ (Figure 2.4). Positive examples denote when examples that follow the rule and model prediction, whereas the negative examples show any exceptions.

**Case Marking** Similar to morphological agreement and word order, we extract and visualize rules for understanding case marking. In our formulation, case marking entails when a given word class (e.g. nouns) marks a particular case value (e.g. nominative, ergative). We visualize the extracted rules using a table as shown in Figure 2.6 for Turkish.<sup>13</sup> The methodology used is discussed in Chapter 4.

**Morphology Inflection** Inflection is an important component of word formation, formally it is the process of changing the form of the lexemes such that they fit into the grammatical context (Lieber, 2021). Specifically, in AUTOLEX, we aim to understand these different forms of inflection and when should one form be used over the other. In Figure 2.7 we show the rules extracted for two types of Marathi suffixes.<sup>14</sup> Currently, this feature is supported for only two languages, Marathi and Kannada. The methodology used for automatic extraction is discussed in Chapter 6.

<sup>13</sup>[https://aditi138.github.io/auto-lex-learn/tr\\_imst/CaseMarking/NOUN/NOUN.html](https://aditi138.github.io/auto-lex-learn/tr_imst/CaseMarking/NOUN/NOUN.html)

<sup>14</sup>[https://aditi138.github.io/auto-lex-learn/mr\\_en/Suffix/NOUN/NOUN.html](https://aditi138.github.io/auto-lex-learn/mr_en/Suffix/NOUN/NOUN.html)

[Back to Turkish-IMST page](#)

The dominant case for NOUN in the corpus is **Nom**

Case Marking
Generally the case for <b>NOUN</b> is <b>Nom</b>
Some examples are: <a href="#">Examples</a>
case of NOUN is <b>Loc</b> when:
NOUN is the= <b>object</b> NOUN is governed by= <b>ki</b> ( <a href="#">Examples</a> )

Figure 2.6: Rules explaining when the nominative case is used for Turkish nouns.

ने (ne) suffix is used when:
in English you would use the following word= by current word with Case = Acc in English you would use the following word= government in English you would use the following word= court current word is the= <b>subject</b>
<a href="#">Examples</a>
वर (var) suffix is used when:
current word is in the neighborhood of the word= परिणाम (parinaam) current word is in the neighborhood of the word= रस्ता (rastaa) current word's lemma is= रस्ता (rastaa) current word is the= underspecified dependency in English you would use the following word= on
<a href="#">Examples</a>

Figure 2.7: Different types of suffix (inflections) added for Marathi words. For example, suffix ‘ne’ is used typically for subjects in accusative case. Another way of explaining the usage is through its English counterpart, for example, its usage is similar to the usage of ‘by’ in English.

### 2.2.3 Lexical Semantics

Along with extracting the syntactic descriptions, we also extract descriptions to understand the semantics, specifically the vocabulary of a new language. Specifically, we focus on explaining those words in a language of interest (from English) which show different lexical manifestations for a given English concept. We refer to these as semantic subdivisions, as the same concept in one language (e.g. English) is subdivided into fine-grained concepts in another language. An example of one such semantic subdivision in Spanish is shown in [Figure 2.8](#).<sup>15</sup> Currently, we support this feature for explaining Marathi, Greek, Spanish and Kannada words from English. Additionally, for Marathi and Kannada we also present definitions, examples, synonyms and antonyms for popular nouns and adjectives.<sup>16</sup>

## 2.3 Statement of Limitations

The primary advantage of using data and models from existing multilingual projects such as UD, SUD, Wikipedia, is that they are available for hundreds of languages and are annotated using a consistent annotation schema, which allows us to extract consistent descriptions across the different languages. Although, using a consistent and unified format for representing rules helps in easy extensibility of our approach to hundreds of languages, we concede that there are certain language-specific aspects which cannot be represented universally across all languages. One possible solution is to add specific sections

<sup>15</sup>[https://aditi138.github.io/auto-lex-learn/es\\_gsd/LearnVocab/English/English.html](https://aditi138.github.io/auto-lex-learn/es_gsd/LearnVocab/English/English.html)

<sup>16</sup>[https://aditi138.github.io/auto-lex-learn/mr\\_en/WordUsage/WordUsage.html](https://aditi138.github.io/auto-lex-learn/mr_en/WordUsage/WordUsage.html)

L1: English	L2: Spanish	Rules
oil.NOUN	óleo/petróleo/petrolera/petrolero	<b>Words:</b> well, price, tanker, man field, business, company, find, painting, think, strike, crude, much, deal
oil.NOUN	aceite	<b>Short phrases:</b> (boil, oil), (change, oil), (castor, oil), (check, oil) (olive, oil) <b>Words:</b> lamp, palm, -, vinegar little, pressure, put, castor, boil, water, change, check, olive

Figure 2.8: Semantic subdivision for the concept ‘oil’ results in different lexical manifestations in Spanish: ‘petróleo’ for petroleum oil and ‘aceite’ for cooking oil whereas in English both are referred as ‘oil’.

along with the general sections under each language to address some of the aspects. Another limitation is that the examples we use under each section might not be culturally-sensitive or representative as they are based on data resources which are available publicly and itself might not be representative enough.





## **Part I**

# **Extracting Language Descriptions of Natural Languages Automatically.**



## Chapter 3

# Automatically Extracting Linguistic Descriptions for Agreement

In [Chapter 2](#), we describe the different linguistic phenomena currently supported by AUTOLEX. In this chapter and the subsequent ones, we describe methods to extract language descriptions from raw text across several languages for each of these phenomena, and show applications in language documentation, learning, and teaching. We focus on aspects of morphosyntax and semantics which describe the rules governing the structure and meaning of a sentence in its “own terms” i.e. from the raw text of that language which is as natural as possible, as is done in the realm of descriptive linguistics ([Harris, 1954](#)). In this chapter, we explain the workflow in AUTOLEX by taking the example of *morphological agreement*.

Aditi Chaudhary, Antonios Anastasopoulos, Adithya Pratapa, David R. Mortensen, Zaid Sheikh, Yulia Tsvetkov, Graham Neubig. 2020. [Automatic Extraction of Rules Governing Morphological Agreement](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

### 3.1 Overview

*Morphological agreement* is an important aspect of morphosyntax that is prevalent in several languages. Agreement has been widely studied in documentary linguistics as it forms a core component of language production and understanding ([Keenan, 1974](#); [Corbett, 1979](#); [Lehmann, 1968](#)). There are multiple views on defining the agreement process, and a prominent definition is that agreement is the process wherein a word or morpheme selects morphemes in correspondence with another word or phrase in the sentence ([Steele, 1978](#); [Corbett, 1979](#)). In this definition, the element (e.g. word, phrase) that drives the agreement is called as *controller* and the element whose form is determined by this agreement is called *target*. *Morphological property or feature* (e.g. gender) is the grammatical category with respect to which agreement is occurring. To avoid overloading of the term ‘feature’, we refer to the ‘morphological features’ as grammatical categories going forward and the term ‘features’ is used to denote the syntactic or semantic features which govern the agreement process. This notion of *directionality* between the controller and the target roles where the former determines the form of the latter, however, can be difficult to apply in situations where the controller may be absent (pro-drop) ([Barlow and Ferguson, 1988](#); [Pollard and Sag, 1994](#)). These issues are handled well in the unification theories ([Barlow and Ferguson, 1988](#)),

for example, in agreement, these theories accumulate partial information from both the controller and the target, ignoring the directionality between them.

Many grammar formalisms including Lexical Functional Grammar (LFG), Generalized Phrase Structure Grammar (GPSG), Head-Driven Phrase Structure Grammar (HPSG) fall under *Unification Grammar*, where linguistic objects under study are represented by feature structures (Sag et al., 1986). These feature structures impose ‘constraints’ which could be universal or language specific on the grammatical information associated with each linguistic object, order-independent. In this work, we follow this and do not specify features for the controller or the target separately, rather present them as a unified set of features. The AUTOLEX formalism differs from these existing formalisms in that a linguistic phenomenon (e.g. agreement) is described not just at the individual surface level, rather common patterns that are generalizable across the language as a whole are derived. Furthermore, along with the common patterns, AUTOLEX aims to identify the deviations in these patterns and explain the conditions that trigger each one.

Understanding agreement is not only important for syntax and morphology, but has also seen application in language acquisition, psycholinguistics (Nichol, 1995; Vigliocco and Nicol, 1998; Vigliocco et al., 1996; Clahsen and Hansen, 1993). Considering such a widespread interest in understanding this process, we aim to extract rules describing this process concisely in both human- and machine-readable formats. Having rules in machine-readable format will further enable NLP applications such as identifying and mitigating gender stereotypes in morphologically rich languages (Zmigrod et al., 2019), designing metrics for evaluating natural language generation tasks (Pratapa et al., 2021a).

Our contributions are summarized as follows:

1. We introduce a framework to automatically extract agreement rules from raw text, and release these rules for 55 languages as part of the AUTOLEX interface<sup>1</sup> which visualizes the rules in detail along with examples and counter-examples. The interface is described in detail in Chapter 2.
2. We design a human evaluation interface to allow linguists to easily verify the extracted rules and also devise an automated metric to evaluate our framework for scenarios where human evaluation is infeasible.
3. We evaluate the quality of extracted rules under real zero-shot conditions (on Breton, Buryat, Faroese, Tagalog, and Welsh) as well as simulated low-resource conditions by varying the amount of syntactically analysed data. We find that using cross-lingual transfer learning helps bridge the data availability gap for the under-resourced settings.

## 3.2 Proposed Approach

As described in Chapter 1 (Figure 1.3), AUTOLEX comprises of four steps: *formalization*, where a linguistic question is formulated into a classification task, *feature extraction*, where relevant features known to govern the linguistic phenomenon are extracted and converted into training data, *model learning*, where a model conducive to human interpretation is learnt, and *rule extraction and visualization* wherein the rules are extracted in a human- and machine-readable format from the learnt model. One assumption we need to define for this problem is that any linguistic phenomena we want to explain should not be totally

---

<sup>1</sup><https://neulab.github.io/autolex/>

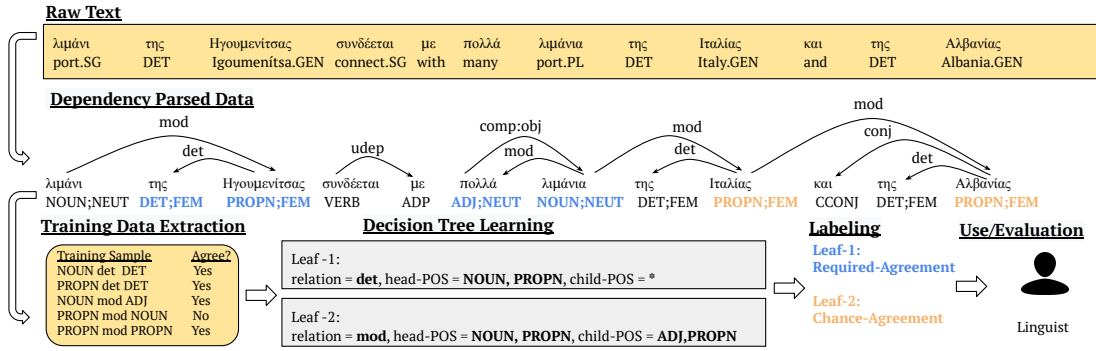


Figure 3.1: An overview of our method’s workflow for gender agreement in Greek. The example sentence translates to “The port of Igoumenitsa is connected to many ports in Italy and Albania.” First, we dependency parse and morphologically analyze raw text to create training data for our binary agreement classification task. Next, we learn a decision tree to extract the rule set governing gender agreement, and label the extracted leaves as either representing required or chance agreement. Finally these rules are presented to a linguist for perusal.

random, i.e. its mechanisms can be derived from some underlying criteria (e.g. syntactic) and therefore can generalize across most linguistic inputs. For example, syntax is largely driven by the syntactic heads of grammatical relations and their arguments, and, therefore we can use statistical models to learn some of the constraints that hold between the heads and their arguments. Manning (1994) describe some such linguistic phenomena which are sensitive to argument structure. However, there are exceptions to general syntax rules, for example, arising from subclasses of words, for which we then require to use lexical and semantic features.

More concretely, the process of agreement entails matching the value of grammatical categories (e.g. gender, person, number) between different words/phrases in the sentence. The extent of agreement displayed across these categories varies drastically both within a language and across different languages. For instance, finite verbs in Marathi usually agree with their subjects on gender, number, and person, whereas regular verbs in English agree only on person and number. In this work, we focus on the agreement observed in the following six grammatical categories, namely: gender, person, number, tense, mood, and case. which are known to display agreement extensively across multiple languages (Barlow and Ferguson, 1988; Corbett, 2003). Canonically, agreement is described in the syntactic environment of a language (Corbett, 2017), however semantic features also govern agreement in some situations (Puljum, 1984). For example, *United Nations is*, despite *United Nations* being plural it is treated as singular for purposes of agreement. Corbett (2003) distinguishes between the agreement governed by syntax as *syntactic agreement* and that governed by meaning as *semantic agreement*. In this work, we describe the agreement process using syntactic features i.e. the syntactic agreement using features derived from syntactic dependency, head and the dependent (Nichols, 1985) and present a framework that automatically extracts these rules from raw text (Figure 3.1). We now describe each of the four steps in AUTOLEX to understand the agreement process.

<p>A.1    Los            enigmas    son       fáciles  DET.PL    riddle.PL   be.PL    easy.PL</p> <p style="text-align: center;">↔req↔</p> <p>‘The riddles are easy.’</p> <hr/> <p>A.2    *Los            enigmas    es       fácil  DET.PL    riddle.PL   be.SG    easy.SG</p> <p style="text-align: center;">↔wrong↔</p> <hr/>	<p>B.1    Mi    hermano    tiene    un       perro  My    brother.SG   has.SG   ART.SG   dog.SG</p> <p style="text-align: center;">↔req↔    ↔chance↔</p> <p>‘My brother has a dog.’</p> <hr/> <p>B.2    Mi    hermano    tiene    muchos    perros  My    brother.SG   has.SG   many.PL   dog.PL</p> <p style="text-align: center;">↔req↔    ↔correct↔</p> <p>‘My brother has many dogs.’</p> <hr/>
--	---

Figure 3.2: Subject-verb number agreement is required in Spanish, as in example A.1, which renders example A.2 ungrammatical. Object-verb agreement is not required, so both B.1 and B.2 are grammatical. The object and the verb in B.1 only agree by chance.

### 3.2.1 Problem Formulation

In AUTOLEX, we first determine whether we can formulate a given linguistic phenomenon  $p$  as a prediction problem, where given an input set of features  $\mathbf{X}_p = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  we predict an output label  $\mathbf{Y}_p = y_1, y_2, \dots, y_n$  that indicates the linguistic phenomena. Next, we determine the set of features which we believe are known to govern the phenomena represented in the UD schema (McDonald et al., 2013). Below we describe how we define  $X, Y$  formally for agreement.

**Agreement** Formally, we pose the linguistic question of agreement as *under what conditions should two tokens in a sentence agree on some morphological property and when they need not*. The prediction problem of morphological agreement then becomes predicting whether the value of a morphological property matches between the head and the dependent token. However, not all observed agreement can be attributed to an underlying grammatical rule. For example, in Figure 3.2 the Spanish A.1 shows an example where the subject (*enigmas*) and the verb (*son*) need to agree on number property. We will refer to such rules as *required-agreement*. Such a required agreement rule dictates that an example like A.2 is ungrammatical and would not appear in well-formed Spanish sentences, since the subject and the verb do not have the same number marking. However, not all word pairs that agree do so because of some underlying rule, and we will refer to such cases as *chance-agreement*. For example, in Figure 3.2 the object (*perro*) and the verb (*tiene*) in B.1 only agree in number by chance, and example B.2 (where the object of a singular verb is plural) is perfectly acceptable. Therefore, we pose the problem of extracting agreement rules as identifying for which head-dependent pairs the language displays *required-agreement* and for which we will observe at most *chance-agreement*. We focus on the morphological attributes  $M = \{\text{gender, person, number, case, tense, aspect}\}$ , and train a separate model for each. Although, among these six morphological attributes, gender, person, and number often show agreement than other attributes (Corbett, 2009). The pair of head-dependent words which both mark the morphological property  $m$  form the input example  $x_i$  and the output labels ( $y_i$ ) are binary, denoting if agreement is observed or not between the pair.

### 3.2.2 Feature Extraction

After formulating the linguistic question into a prediction task, we design features to help predict its answer. We use linguistic knowledge to design features, but the feature extraction process itself is automatic. For a different question or language, a linguist can begin the process by using these initial features or even design new features as they deem fit. The chosen criteria and features in the framework can be further honed by using inputs from language experts or by consulting relevant literature. For example, there are several debates in the literature about what mechanisms govern agreement, is it fully syntactic, is it restricted to only some grammatical categories, is it defined over a local context, and so on. We design our general framework to allow a user to experiment with different combinations of features. In this chapter, we make the simplifying assumption that the head and dependent elements (tokens) are represented by only POS features, as we would like our extracted rules to be *concise* and easily interpretable downstream, although this can be extended further by adding more descriptive features (check [Chapter 4](#)). We refer to the words participating in an input  $x_i$  as *focus words*, in this case the head-dependent word pairs.

### 3.2.3 Training Data and Model Learning

**Training Data** To construct training data  $D_{\text{train}}^p$  for the agreement phenomenon  $p$ , we start with the raw text  $D$  of the language in question and perform syntactic analysis, producing POS tags, lemmas, morphological analysis and dependency trees for each sentence (shown in [Figure 3.2](#)). Using this analysis, we then identify the focus word(s) and extract features, forming the input example ( $\mathbf{x}_i = \{x_i^0, x_i^1, \dots, x_i^k\}$ ). Specifically, we convert each dependency relation into a triple  $\langle w_h, w_d, r \rangle$ , indicating the head token, dependent token, and dependency relation between  $w_h$  and  $w_d$  respectively. For the entire text, we now have input features for each morphological property of our interest  $m$  as  $X_{\text{agree}}^m = \{\langle w_h^{(1)}, w_d^{(1)}, r^{(1)} \rangle, \dots, \langle w_h^{(n)}, w_d^{(n)}, r^{(n)} \rangle\}$  and binary output labels  $Y = y_1, \dots, y_n$ , where if the head and the dependent token agree on property  $m$  (such that  $w_h^m = w_d^m$ ) we set  $y = 1$ , otherwise  $y = 0$ .

**Model Learning** Given that the learned model must be interpretable to linguists using the system, we opt to use decision trees ([Quinlan, 1986](#)), which split the data into leaves, where each leaf corresponds to a portion of the input examples following common syntactic patterns. We train decision trees using the CART algorithm ([Breiman et al., 1984](#)). A major advantage of decision trees is that they are easy to interpret and we can visualize the exact features used by the decision tree to split nodes. The decision tree induces a distribution of agreement over training samples in each leaf, e.g. 99% observed agreement, 1% not agreeing in Leaf-3 for gender agreement in Spanish ([Figure 3.3\(a\)](#)).

### 3.2.4 Rule Extraction and Visualization

The above step constructs a decision tree for each morphological property and language, where each tree leaf corresponds to a salient partition of the possible syntactic structures in the language. The next step is to identify which of these leaves correspond to a likely agreement rule, i.e. we need to label these leaves with either *required-agreement* or *chance-agreement* label. For this, we apply a threshold on the ratio of training samples which have matching values within a leaf – if the ratio exceeds a certain

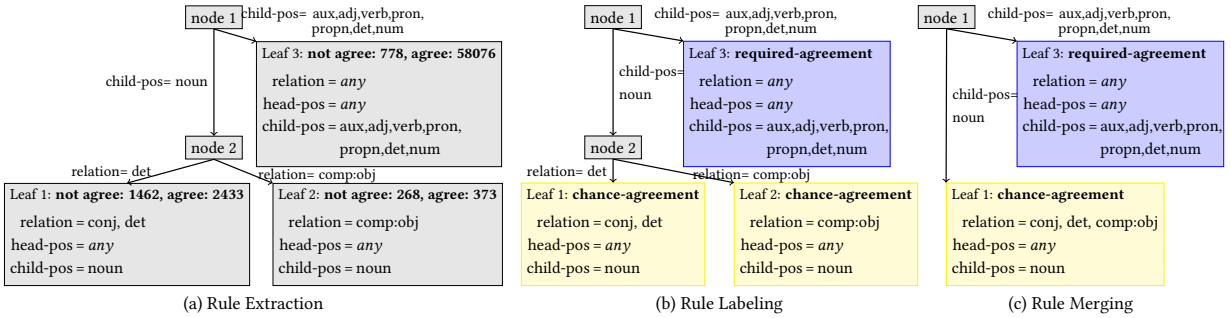


Figure 3.3: Extracting gender agreement rules in Spanish. (a) A decision tree is learned over dependency link triples, inducing a distribution of agreement over examples in each leaf. However, simple majority voting leads to false positives: Leaf-1 includes more agreeing data points, but in reality this agreement is purely by chance. (b) With a statistically-inspired threshold to label the leaves, Leaf-1 gets correctly labeled as *chance-agreement*. (c) We merge leaves with the same label to get a concise representation. Every dependency link triple receives the label of the unique leaf it falls under.

number, the leaf will be judged as *required-agreement*. We experiment with two types of thresholds, *hard threshold* and a *statistical threshold*.

**Hard Threshold** In this setting, a leaf having the number of agreeing examples more than 90% of all examples in that leaf are labeled as *required-agreement*. We set this threshold based on manually inspecting some resulting trees to find a threshold that limited the number of non-agreeing syntactic structures being labeled as *required-agreement*.<sup>2</sup>

**Statistical Threshold** A hard threshold alone is insufficient to capture probable agreement because leaves with very few examples may exceed the hard threshold purely by chance. Therefore, we use a statistical measure to better determine whether the agreements are due to a true pattern of required agreement. For all leaves displaying an agreement majority, we apply a chi-squared (Oakes, 1998) goodness of fit test to compare the observed output distribution with an expected probability distribution specified by a null hypothesis. Our null hypothesis  $H_0$  is that any agreement we observe is due to chance. If the null hypothesis is rejected, we conclude from the alternative hypothesis  $H_1$  that there exists a grammatical rule that requires agreement for this leaf.

For computing the expected probability distribution, we assume that the morphological properties of the head and the dependent token are independent and identically distributed discrete random variables following a categorical distribution *if* there is no rule requiring agreement. We compute the probability of chance agreement based on the number of values that the specific morphological property  $m$  can take. Since the category values are not equally probable, we use a probability proportional to the observed value counts. For a binary number property where 90% of all observed occurrences are singular and 10% are plural, the probability of chance agreement is equal to  $0.82 = 0.9 \times 0.9 + 0.1 \times 0.1$ , which gives the observed output distribution  $p = [0.18, 0.82]$ . Using  $p$  we compute the expected frequency count  $E_i = np_i$  where  $n$  is the total number of samples in the given leaf,  $i = [0, 1]$  is the output class of the

<sup>2</sup>Initial experiments with a majority voting strategy i.e. setting the threshold to-50% yielded much worse trees and hence we decided to use 90% as the hard threshold.



leaf, and  $p_i$  is the hypothesized proportion of observations for class  $i$ . The chi-squared test calculates the test statistic  $\chi^2$  as follows:

$$\chi^2 = \sum_{i \in [0,1]} \frac{(O_i - E_i)^2}{E_i} \quad (3.1)$$

where  $O_i$  is the observed frequency count in the given leaf. The test outputs a  $p$ -value, and if this  $p$ -value is smaller than a chosen significance level (we use 0.01) we reject the null hypothesis and label the leaf as *required-agreement*.

The chi-squared test especially helps to be cautious with leaves with very few examples. However, for leaves with larger number of examples, statistical significance alone is insufficient because there are a large number of cases where there are small but significant differences from the ratio of chance agreement. Therefore, in addition to comparing the  $p$ -value we also compute the *effect size* which provides a quantitative measure on the magnitude of an effect (Sullivan and Feinn, 2012). Cramér’s phi  $\phi_c$  (Cramér, 1946) is a commonly used method to measure the effect size:

$$\phi_c = \frac{\chi^2}{N(k-1)} \quad (3.2)$$

where  $\chi^2$  is the test statistic computed from the chi-squared test,  $N$  is the total number of samples within a leaf, and  $k$  is the degree of freedom (which in this case is 2 since we have two output classes). Therefore, a leaf is now labeled as *required-agreement* if the  $p$ -value is less than the significance value and the effect size is greater than 0.5.<sup>3</sup> Now Leaf-1 in Figure 3.3(b) is correctly identified as *chance-agreement*. One limitation of our formulation is that rules that show agreement *sometimes* get incorrectly labeled as *chance-agreement* or *required-agreement*. We do consider this in evaluation, although.

To obtain a concise set of rules, we merge sibling leaves with the same label as shown in Figure 3.3(c). Furthermore, we collapse tree nodes that have all leaves with the same label to reduce the apparent depth of the tree for easy visualization.

**Rule Visualization** After the leaves have been labeled and merged, each rule comprises of triples of head-POS tag, dependent-POS tag, and the dependency relation between them. For each such rule, we extract illustrative examples from the underlying corpus and visualize them in an interface (Figure 2.5 in Chapter 1). We select such examples that are short and consist of diverse word forms to illustrate the rule usage in different contexts. Along with examples which follow a rule, we also show examples which do not follow the rule, giving a softer, more nuanced view of the data. Specifically, to not overwhelm the user, we only present 10 examples for each type.

### 3.3 Experimental Settings

We evaluate our extracted rules using both an *automated evaluation* where we measure the accuracy against a test set and, *human evaluation* where we present rules to language experts for verification. We conduct two types of experiments: 1) *gold-standard experiments* where we use the gold-standard syntactic analyses to evaluate whether our proposed approach extracts linguistically plausible rules across

<sup>3</sup>This threshold is selected based on Cohen (2013) who provide rules of thumb to interpret the effect size.

a diverse set of languages (section 3.4) and, 2) *under-resourced experiments* to evaluate our system in the absence of gold-standard analyses where we use cross-lingual transfer to obtain noisy parses on the languages of interest (section 3.5). Experimenting with languages that have been already studied and have annotated treebanks is crucial for verifying the efficacy of our approach before applying it to other truly low- or zero-resource languages. Under this setting, we not only have clean and expert-annotated data, but we can also quickly compare the effect of data size on the system performance as different languages have treebanks of varying size.

**Data and Model** We use Surface-Syntactic Universal Dependencies (SUD) treebanks (Gerdes et al., 2018, 2019) as the gold-standard source of complete syntactic analysis. The SUD treebanks are derived from Universal Dependencies (UD) (Nivre et al., 2016; Nivre et al., 2018), but unlike the UD treebanks which favor content words as heads, SUD treebanks express the dependency structure using syntactic criteria, which is more conducive to our goal of learning syntactic rules. Figure 3.4 presents a comparison of UD and SUD-style trees for the German sentence, “Ich werde lange Bücher lesen.”. The SUD tree has the function word ‘werde’ as the syntactic head to the content word ‘lesen’. We use the tool Gerdes et al.

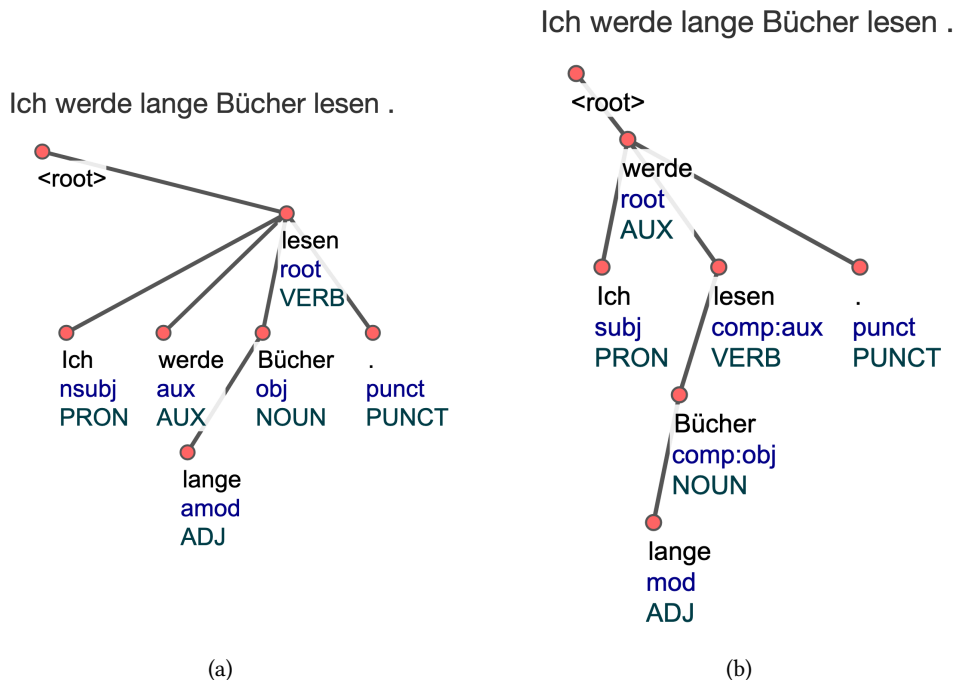


Figure 3.4: Comparing the UD (a) tree with the SUD (b) tree for the German sentence “Ich werde lange Bücher lesen.”.

(2019) to convert UD v.2.5 (Nivre et al., 2020) into SUD covering 55 languages in 91 treebanks, which are publicly available with annotations for POS tags, lemmas, dependency parses, and morphological analysis. We use their provided split of train/dev/test and learn the rules only on the training portion of the treebanks.

We use `sklearn`’s (Pedregosa et al., 2011) implementation of decision trees and train a separate model for each morphological property  $m$  for each treebank. As mentioned earlier, we experiment with

relation=subj, head=VERB, dependent=PRON  
 Almost Always Agree  Sometimes Agree  Need Not Agree  
[\[Examples\]](#)

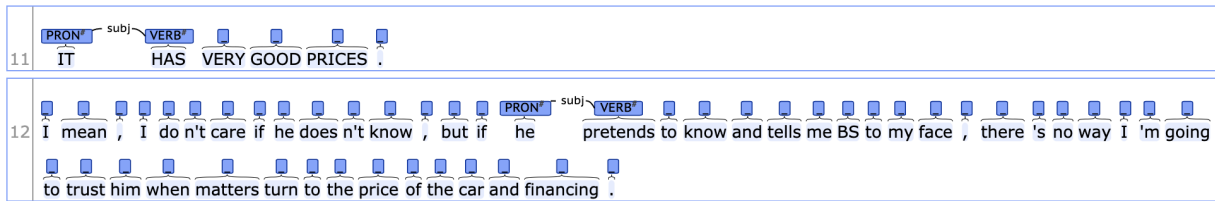


Figure 3.5: Annotation interface for evaluating number agreement in English

six grammatical categories (gender, person, number, mood, case, tense) which are also most frequently present across several languages. We experiment with *Statistical Threshold* and *Hard Threshold* and find that the former performs better on manual inspection, especially for under-resourced languages. One reason why *Statistical-Threshold* performs better for these languages is because there are more leaves with fewer samples overall, causing *Hard Threshold* to have more false positives. Whereas *Statistical Threshold* uses effect size with the significance test which takes into account the sample size within a leaf leading to better leaves. Therefore, we chose to use *Statistical-Threshold* for all our experiments. We perform a grid search over the following hyperparameters of the decision tree:

- `criterion = [gini, entropy]`
- `max depth = [6,15]`
- `min impurity decrease = 1e-3`

The best parameters are selected based on the performance of the validation set. For treebanks that have no validation split, we use the default cross-validation provided by `sklearn` (Buitinck et al., 2013). The average model runtime for a treebanks is 5-10 mins depending on the size of the treebank.

### 3.3.1 Human Evaluation

We evaluate our extracted head-relation-dependent triples for agreement with the help of language experts. Ideally, we want to collect these annotations for all triples in the treebank, but this would require annotating hundreds of triples across the six grammatical categories and languages, requiring a large time commitment from linguists evaluating the language. Instead, for a subset of languages (treebanks) we extract and evaluate the top 20 most frequent triples for the six grammatical categories, amounting to 120 sets of triples to be annotated.<sup>4</sup> The linguist is then asked to annotate whether there is a rule in this language governing agreement between the head-dependent pair for this relation. The allowed labels are: *Almost always agree* if the construction must almost always exhibit agreement on the given category; *Sometimes agree* if the linked arguments sometimes must agree, but sometimes do not have to; *Need not agree* if any agreement on the category is random. An example of the annotation interface is shown in Figure 3.5.

To calculate the accuracy of human annotations, for each annotated triple marking grammatical category  $f$ , we extract the label assigned to it by the learnt decision tree  $\mathcal{T}$ . We find the leaf to which

<sup>4</sup>The top 20 most frequent triples covered approximately 95% of the triples where this feature was active on average.

the given triple  $t$  belongs and assign the label of that leaf to the triple, referred to by  $l_{\text{tree},f,t}$ . We then compare this label with  $l_{\text{human},f,t}$  which is the label assigned to the triple  $t$  by the human annotator and average the accuracy across all annotated triples  $T_f$  to get the human evaluation metric (HRM) for feature  $f$  given by:

$$\text{HRM}_f = \frac{\sum_{t \in T_f} \mathbb{1}\{l_{\text{human},f,t} = l_{\text{tree},f,t}\}}{|T_f|} \quad (3.3)$$

where  $\mathbb{1}$  denotes an indicator function.

### 3.3.2 Automated Evaluation

Since it is not always feasible to conduct the human evaluation, we also present an automated evaluation method which acts as a proxy for the human evaluation. We propose an automated rule metric (ARM) that evaluates how well the rules extracted from the decision tree  $\mathcal{T}$  fit to an unseen gold-annotated test data. For each triple  $t$  marking the grammatical category  $f$ , we first retrieve all the examples from the test data corresponding to that triple. Next, we calculate the empirical agreement by counting the fraction of test samples that exhibit agreement, as referred to by  $q_{f,t}$ . For a *required-agreement* leaf, we expect most test samples satisfying that rule to show agreement.<sup>5</sup> To account for any exceptions to the rule and/or parsing-related errors, we use a threshold that acts as a proxy for evaluating whether the given triple denotes *required agreement*. We use a threshold of 0.95, and if  $q_{f,t} > 0.95$  then assign the test label  $l_{\text{test},f,t}$  for that triple as *required-agreement*, and otherwise choose *chance-agreement*.<sup>6</sup> Similar to the human evaluation, we compute a score for each triple  $t$  marking the category  $f$  and average across all triples annotated in  $T_f$  to obtain the ARM score as shown below.

$$\text{ARM}_f = \frac{\sum_{t \in T_f} \mathbb{1}\{l_{\text{test},f,t} = l_{\text{tree},f,t}\}}{|T_f|} \quad (3.4)$$

We compare our produced trees with the baseline trees that predict *chance-agreement* for all triples.

## 3.4 Gold-Standard Experiments

In this section, we evaluate our extracted rules in the setting where we have access to gold-standard syntactic analyses. We discuss the results of models trained on the SUD treebanks.

### 3.4.1 Automated Evaluation Results

We learn the decision trees on the training portion of each treebank and find that our extracted rules outperform the baseline trees by 7.4 ARM points, averaged across all treebanks.<sup>7</sup> In Figure 3.6, we show improvements over the baseline averaged across language families/genera. In families with extensive and well-documented agreement systems such as Indo-Aryan, Slavic, Baltic (Comrie, 1984; Crockett, 1976) our models clearly outperform the baseline discovering correct rules. For mood and tense, the *chance-agreement* baseline performs on par with our method. This is not surprising because little agreement

<sup>5</sup>There are exceptions: e.g. when the head of dependent is a multiword expression (MWE), in which case dependency parsers might miss or pick only one of its constituents as head/dependent, or if the MWE is syntactically idiosyncratic.

<sup>6</sup>We keep a 5% margin to account for any exceptions or parsing errors based on the feedback given by the annotators.

<sup>7</sup>Individual scores for each treebank are in the original paper.

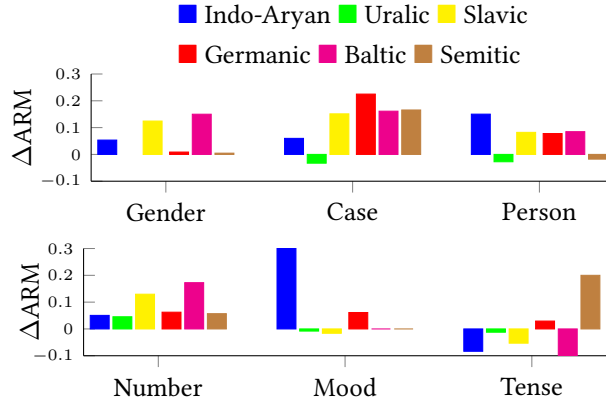


Figure 3.6: Difference in the ARM scores of decision trees over gold-standard syntactic analysis with baseline trees where all leaves predict *chance-agreement*.

is observed for these categories, given that only verbs and auxiliary verbs mark these categories. We find that for both tense and mood in the Indo-Aryan family, our model identifies *required-agreement* primarily for conjoined verbs, which must mostly agree only if they share the same subject. However, subsequent analysis revealed that in the treebanks, nearly 50% of the agreeing verbs do not share the same subject, but do agree by chance.

We further measure the conciseness of the constructed trees by plotting the correlation between the number of leaves and the morphological complexity of the languages in Figure 3.7. To compute the morphological complexity of a language, we use the word entropy measure proposed by Bentz et al. (2016) which measures the average information content of words and is computed as follows:

$$H(D) = - \sum_{i \in V} p(w_i) \log p(w_i) \quad (3.5)$$

where  $V$  is the vocabulary,  $D$  is the monolingual text extracted from the training portion of the respective treebank,  $p(w_i)$  is the word type frequency normalized by the total tokens. Since this entropy does not account for unseen word types, Bentz et al. (2016) use the *James-Stein shrinkage* estimator (Hausser and Strimmer, 2009) to calculate  $p(w_i)$ :

$$p(w_i) = \lambda p^{\text{target}}(w_i) + (1 - \lambda) p^{\text{ML}}(w_i) \quad (3.6)$$

where  $\lambda \in [0, 1]$ ,  $p^{\text{target}}$  denotes the maximum entropy case given by the uniform distribution  $\frac{1}{|V|}$  and  $p^{\text{ML}}$  is the maximum likelihood estimator given by the normalized word type frequency. Languages with a larger word entropy are considered to be morphologically rich as they pack more information into the words. In Figure 3.7 we plot the morphological richness with the average number of leaves across all grammatical categories and find them highly correlated.

### 3.4.2 Human Evaluation Results

Through the above experiments, we *automatically* evaluated that the extracted rules are predictive (to some extent) and applicable to the language in general. Now, we conduct a manual evaluation for three

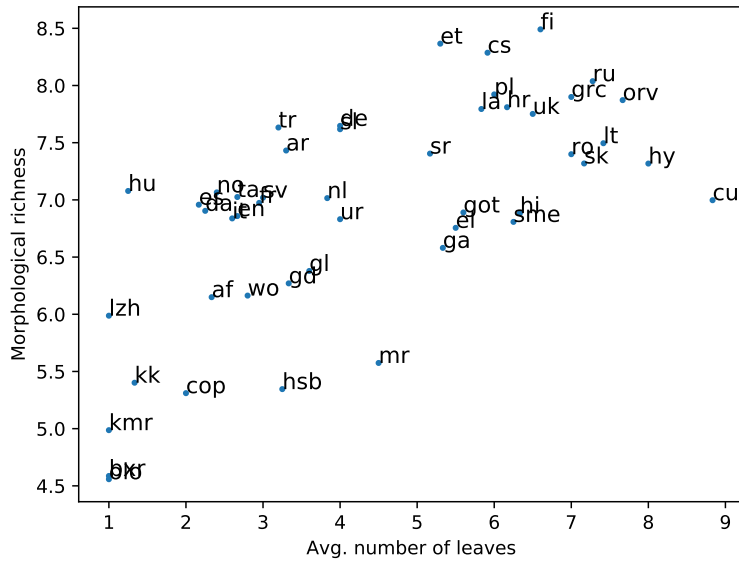


Figure 3.7: Correlation between size of the decision trees constructed by our framework and morphological complexity of languages.

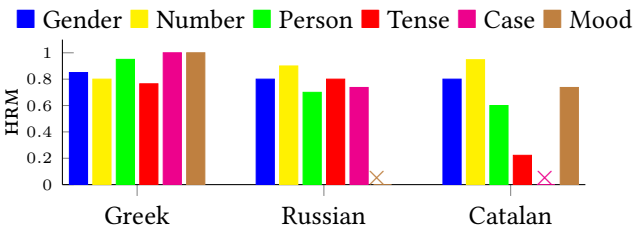


Figure 3.8: Annotation accuracy for Greek, Russian and Catalan per each morphological feature.

languages: Greek (el), Russian (ru) and Catalan (ca). As described above, we require language experts to annotate a head-relation-dependent triple with either *almost always agree*, *sometimes agree* and *need not agree*. For a strict setting, we consider *sometimes agree* and *need not agree* as *chance-agreement* and report the human evaluation metric (HRM) in [Figure 3.8](#). Overall, our method extracts first-pass grammar rules, achieving 89% accuracy for Greek, 78% for Russian, and 66% for Catalan.

We analyze some of the errors made by our model and find that in most error cases, like the person in Russian, our model produces incorrect *required-agreement* labels, which we can attribute to skewed data statistics in the treebanks. In Russian and Greek, for instance, conjoined verbs only need to agree in person and number if they share the same subject; however, in the treebanks we find them to *implicitly* agree because they both must agree with the same subject phrase. In treebanks, though, only 15% of the agreeing verbs do indeed share the same subject, the rest agree by chance. In a reverse example from Catalan, the overwhelming majority (92%) of 8650 tokens are in the third person, causing our model to label all leaves as chance agreement despite the fact that person/number agreement is required in such cases. Similarly for tense in Catalan, our framework predicts *chance-agreement* for auxiliary verbs with verbs as their dependent because of the overwhelming majority of disagreement examples. We believe this is because of both the annotation artifacts and the way past tense is realized. Agreement in TAM (tense, aspect, and modality) is not that common because frequently only one verb in relation is finite and for many languages TAM are optionally marked ([Gil, 2021](#)).

Since manual evaluation is not always feasible, we also conduct the automated evaluation whose results we discussed before. To assess how well automated evaluation correlates with the human evaluation protocol, we compute the Pearson’s correlation ( $r$ ) between ARM and HRM for each language under four model settings: *simulate-50*, *simulate-100*, *baseline* and *gold*. *simulate- $x$*  is a simulated low-resource setting where the model is trained using syntactically analyzed gold standard data  $x$ .<sup>8</sup> The *baseline* setting is the one where all leaves predict *chance-agreement* and under the *gold* setting we train using the entire gold-standard data. We compute the ARM and HRM scores for the rules learnt under each of the four settings and report the Pearson’s correlation, averaged across all categories. Overall, we observe a moderate correlation for all three languages, with  $r = 0.59$  for Greek,  $r = 0.41$  for Russian and  $r = 0.38$  for Catalan. The correlations are very strong for some categories such as gender ( $r_{el} = 0.97$ ,  $r_{ru} = 0.82$ ,  $r_{ca} = 0.98$ ) and number ( $r_{el} = 0.97$ ,  $r_{ru} = 0.69$ ,  $r_{ca} = 0.96$ ) where we expect to see extensive agreement.

### 3.5 Under-resource Experiments

The experiments on gold-standard syntactic analyses showed that our model extracts decent first-pass agreement rules. However, it is not always the case that we have access to a large quantity of gold-standard analyses. Therefore, to investigate how the quality of rules is affected by the quality of the analyses, we conduct simulation experiments by varying the amount of gold-standard syntactically analyzed training data. For each language, we sample  $x$  fully parsed sentences from the treebank of the available training sentences  $L$ . For the remaining  $L - x$  sentences, we use *silver* syntactic analysis i.e., we train a syntactic analysis model on  $x$  sentences and use the model predictions for the  $L - x$  sentences. We experiment with Spanish, Greek, Belarusian and Lithuanian. Data statistics and treebank details are

<sup>8</sup>More details on the experimental setup in [section 3.5](#).

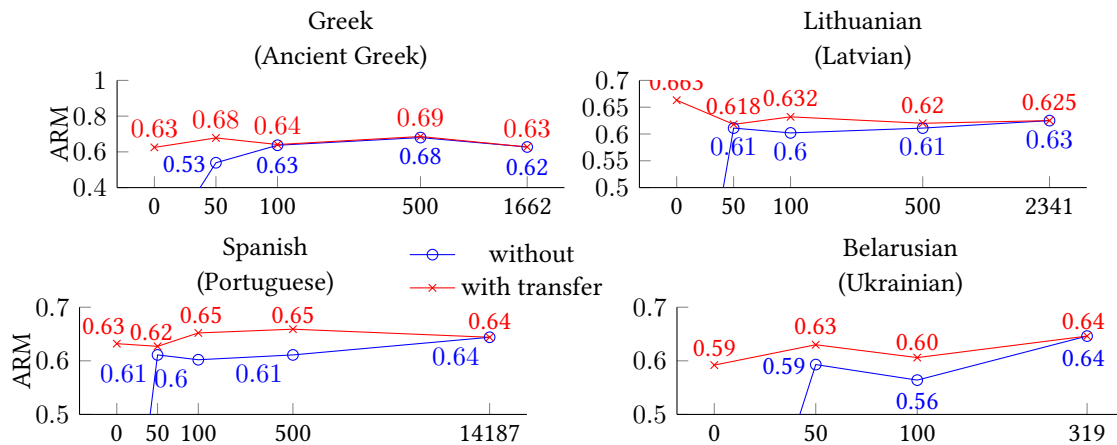


Figure 3.9: Comparing the (avg.) ARM score for Number agreement with and without cross-lingual transfer learning (transfer language in parenthesis).  $x$ -axis in log space. The higher the ARM the better.

presented in Table 3.1.

LANGUAGE	TRAIN/DEV/TEST	TRANSFER LANGUAGE
Spanish-GSD	14187 / 1400 / 426	Portuguese-Bosque
Greek-GDT	1662 / 403 / 456	Ancient Greek-PROIEL
Belarusian-HSE	319 / 65 / 253	Ukrainian-IU
Lithuanian-ALKSNIS	2341 / 617 / 684	Latvian-LVTB

Table 3.1: Dataset statistics. Train/Dev/Test denote the number of sentences in the respective treebank used for the target language.

We train `Udify` (Kondratyuk and Straka, 2019), a parser that jointly predicts the syntactic analysis (POS tags, morphological features, and dependency trees) using the  $x$  gold-standard sentences as our training data. We generate model predictions on the remaining  $L - x$  sentences. Finally, we concatenate the  $x$  gold data with the  $L - x$  automatically parsed data from which we extract the training data for learning the decision tree. We experiment with  $x = [50, 100, 500]$  gold-standard sentences. To account of sampling randomness, we repeat the process 5 times and report averages across runs. To further improve the quality of the automatically obtained syntactic analysis, we use cross-lingual transfer learning where we train the `Udify` model by concatenating  $x$  sentences of the target language with the entire treebank of the related language. We use Portuguese, Ancient Greek, Ukrainian and Latvian treebanks, respectively, as the transfer languages for Spanish, Greek, Belarusian and Lithuanian. We also conduct zero-shot experiments in this setting, where we directly use the `Udify` model trained only on the related language and get the model predictions on  $L$  sentences. As before, we train five decision trees for each  $x$  setting and report the average ARM on the test data.

**Results** In Figure 3.9, we report the results for the number agreement. Similar plots for other languages and grammatical categories can be found in the original paper. We observe that using cross-lingual



[Relation, Head, Dependent]	correct label	gold	zero-shot
det, NOUN, DET.	almost always	required	required
mod, NOUN, ADJ	almost always	required	required
flat, PROPN, PROPN	almost always	required	chance
mod, PROPN, PROPN	almost always	required	chance
appos, PROPN, PROPN	sometimes	required	chance
comp:aux@pass, AUX, VERB	need not	chance	required
conj, PROPN, PROPN	need not	required	chance
ARM score over the test set:		0.644	0.632

Table 3.2: The Spanish gender rules extracted in a zero-shot setting are generally similar to the ones extracted from the gold data (93%). We **highlight** the few mistakes that the zero-shot tree makes.

LANGUAGE	TRAIN / TEST
Breton-KEB	30000 / 888
Buryat-BXR	10000 / 908
Faroese-OFT	50000 / 1208
Tagalog-TRG	30000 / 55
Welsh-CCG	30000 / 956

Table 3.3: Dataset statistics. Training data is obtained by parsing the Leipzig corpora [Goldhahn et al. \(2012\)](#) and test data is obtained from the respective treebank. Each cell denotes the number of sentences in train/test.

transfer learning (CLTL) already leads to high scores across all languages even in zero-shot settings where we do not use any data from the gold-standard treebank. For example, Spanish zero-shot trees produce rules similar to those of the Spanish gold standard trees ([Table 3.2](#)), making a few mistakes as also reflected in the ARM score. Using CLTL, training with just 50 gold-standard target language sentences is almost equivalent to training with 100 or 500 gold-standard sentences. This is encouraging for language documentation of endangered or new languages, as with only 50 expertly-annotated syntactic analysis our framework can produce decent first-pass agreement rules using CLTL. The rules improve as we increase the number of gold-standard sentences, which is not surprising.

**True Zero-shot Results** We also evaluate our model in a *true* zero-shot setting such as for Breton, Buryat, Faroese, Tagalog, and Welsh which do not have gold-standard syntactic analyses available for training but have test data available in SUD. In such cases, we can still extract grammar rules with our framework using zero-shot dependency parsing. For these languages, we collect raw text from the Leipzig corpora ([Goldhahn et al., 2012](#)). Data statistics are listed in [Table 3.3](#).

To enable transfer, we use the Udify model that has been pre-trained on all UD treebanks, as released by [Kondratyuk and Straka \(2019\)](#), and predict the syntactic analysis on the above corpora. As

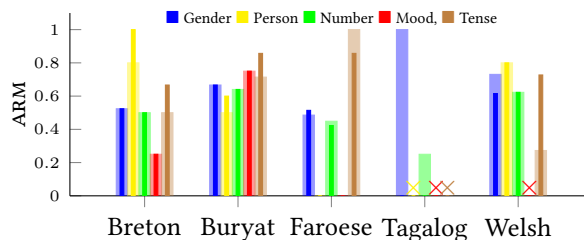


Figure 3.10: In most cases our framework (shaded bars) extracts a good first-pass specification for *true* zero-shot settings. Solid bars indicate the baseline.

before, we use these automatically parsed syntactic analyses to extract the rules which we evaluate with ARM over the gold-standard test data of the corresponding SUD treebanks in Figure 3.10. Tagalog and Buryat are the most distant languages that we test on (no Philippine and Mongolic language is present in our training data) and yet we observe our method being at par with the baseline and even outperforming in the case of Tagalog. Breton and Welsh, on the other hand, are an interesting test bed: Celtic languages are to some degree outliers among Indo-European languages (Borsley and Roberts, 2005), and we suspect that as a result the parser performs generally worse. Despite that, our approach has an ARM of 0.730 for Welsh gender agreement, as opposed to the mere 0.615 that the baseline achieves.

### 3.6 Limitations

While we demonstrate that describing agreement using head-relation-dependent triples achieves decent performance, a limitation of our approach is that it does not capture more complex phenomena that require a broader context or operate at the phrase level. For example, in this English example: “John and Mary love their dog”, under both the UD and SUD formalisms, the coordinating conjunction “and” is dependent, and hence the verb will not agree with either of the (singular) nouns (“John” or “Mary”). Furthermore, as mentioned in the introduction, certain types of agreement are driven semantics, and therefore the feature set needs to be expanded accordingly. Handling such phenomena requires incorporating more descriptive features in the model which, however, could make the tree more complex to comprehend and visualize. Also, in discussing with linguists, we find that annotating triples with exclusively one label is tricky because often there are sub-rules governing the agreement for the same triple specification. For example, for proper nouns in Russian the gender agreement also depends on phonotactics: *Пьер Морал* here the first name would be declined, but not the second, which means only the first name would have the morphological feature explicitly marked. This is an issue with the UD/SUD annotation scheme, which usually annotates morphological features for tokens with inflections in the form. Additionally, we can only capture agreement for tokens which have the morphological property annotated, this could result in ignoring tokens which although exhibit agreement, but under the UD scheme, have not been annotated.

## 3.7 Conclusion

In this chapter, we presented a framework for extracting and evaluating a first-pass set of language patterns from the raw text directly. We showed that the framework extracts decent descriptions under the gold setup where the syntactic analyses are of high quality and, in the under-resourced setting, how using cross-lingual transfer learning can help bridge the gap in performance when such high quality or quantity of data is not available.



## Chapter 4

# A General Framework for Extracting Linguistic Descriptions

In the previous [Chapter 3](#), we described the general framework of `AUTOLEX`, where we showed how language patterns for the morphological agreement process can be extracted automatically from the raw text directly. While we demonstrated the efficacy of our method in extracting a decent first-pass set of rules, we find that our underlying syntactic features are unable to capture more complex phenomena, including the semantic agreement process. One reason for this limitation is our restricted feature set, which is derived from only the syntactic dependency relation, head and dependent. Such restricted feature sets would similarly be insufficient to explain other complex linguistic phenomena such as case marking, and argument structure, which are governed by both syntactic and semantic features. Furthermore, since each linguistic phenomenon will be defined on a subset of features known to govern that particular phenomenon, having separate frameworks can get challenging to maintain or extend. We want researchers or linguists to be able to quickly add new features to improve existing models or even add new linguistic phenomena across numerous languages, wherever applicable. Therefore, in this chapter, we show how `AUTOLEX` can answer other linguistic questions such as word order and case marking, in addition to morphological agreement.

Aditi Chaudhary, Zaid Sheikh, David R. Mortensen, Antonios Anastasopoulos, Graham Neubig. 2022. [`AUTOLEX`: An Automatic Framework for Linguistic Exploration](#). On *arxiv*.

### 4.1 Overview

As mentioned in [Chapter 2](#), most of the grammar description focuses on aspects of syntax and morphology, of which *case*, *word order*, and *morphological agreement* are the most important.

*Case* is formally defined as the ‘system of marking dependents by the type of relation they bear to their syntactic heads’, for example, the nominative marking on the noun could inform that the noun is the grammatical subject of the verb ([Blake, 1994, 2001](#)). One reason to understand case assignment is for understanding the grammatical functions of words in a sentence ([VanPatten and Smith, 2019](#)). For example, in English the constituent order is fixed, i.e., typically subjects come before verbs, which in turn come before objects, while in many languages this order need not be fixed. For example, in Hindi,

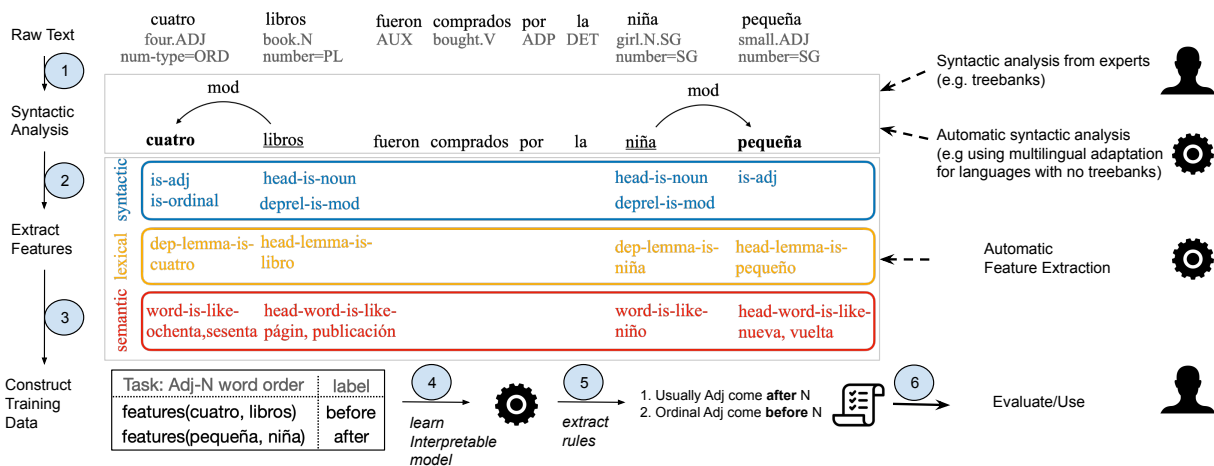


Figure 4.1: An overview of the AUTOLEX framework being applied for understanding word order, with Adj-N order in Spanish as an example. The example sentence translates to *Four books were bought by the small girl*. First, we formulate a linguistic question (e.g. regarding Adj-N order) as a binary classification task (e.g. “whether the Adj comes before/after the N”). Next, we perform syntactic analysis on the raw text, from which we extract syntactic, lexical, and semantic features to construct the training data. Finally, we learn an interpretable model from which we extract concise rules.

the subject and object phrases are free to move around, and in order for humans to understand which phrase is the subject and which the object, the grammatical function is encoded in terms of case marking (Figure 4.2).<sup>1</sup> Linguists have long debated on what exactly defines a case. In the literature, there are multiple viewpoints, but we focus on the viewpoint that there are two types of cases: *abstract case* and *morphological case* (Chomsky, 1993; Halle et al., 1993; Legate, 2008). Abstract case is a universal property, and the morphological case is the overt realization, which triggers under certain conditions and varies cross-linguistically. This morphological realization can occur through word order (English), inflection (Malayalam) or adpositions (Hindi, Marathi, Spanish) and are triggered by syntactic and/or semantic conditions. In AUTOLEX, we are more interested in understanding *abstract case* i.e. extracting the syntactic and/or semantic conditions which govern when some categories of words take the nominative case versus when they take the accusative case. In Chapter 6, we look at some examples of the morphological case where we focus on understanding the inflection, i.e. which suffix to use under what conditions.

*Word Order* describes the relative position of the syntactic elements (e.g. subject with respect to verbs, object with respect to verbs, etc) (Dryer, 2007), and is one of the major axes of linguistic description appearing in grammar sketches or databases such as WALS. In languages such as English, which have a relatively fixed word order, the position of the element conveys the grammatical role and helps reduce sentence ambiguity (e.g. in ‘Tom likes Anna’ it is clear that ‘Tom’ is the subject that likes ‘Anna’ the object, if the order of these elements is swapped then the meaning conveyed also changes.) While in Figure 4.2, we see how the order between elements in Hindi is not fixed and how case marking helps reduce the sentence ambiguity. Therefore, understanding the patterns of word order and when one

<sup>1</sup>Example and explanations inspired from [https://www.ling.upenn.edu/courses/Spring\\_2001/ling150/ch5.html](https://www.ling.upenn.edu/courses/Spring_2001/ling150/ch5.html)

A.1	Meera	<b>ko</b>	Ram	<b>ne</b>	bachaya
	Meera		Ram	by	saved
	ACC		ERG		
	‘Meera was saved by Ram’				
<hr style="border: 0.5px solid black;"/>					
A.2	Ram	<b>ne</b>	Meera	<b>ko</b>	bachaya
	Ram	by	Meera		saved
	ERG		ACC		
	‘Meera was saved by Ram’				

Figure 4.2: Illustrating the free word order in Hindi and how the grammatical role of subjects and objects is expressed through the post-position (-ne for ergative, -ko for accusative).

pattern is observed over another is important for language understanding. Patterns of word order have been widely studied in NLP, for example, Wang and Eisner (2017); Östling (2015) perform statistical analyses on a corpus to find different word order patterns for subjects, verbs, and objects, and conduct a cross-lingual comparison. In AUTOLEX, we are interested not only in extracting these patterns, but also in extracting the conditions under which one pattern is typically observed.

Case, word order, and agreement have their own linguistic purpose, but they often overlap and correlate. For example, word order conveys information about the structure of the sentence and can also be used to disambiguate subjects from objects, similar to case, and how the case manifests reflects in the morphology inflection leading to agreement (Malchukov, 2018).<sup>2</sup> In the previous chapter, we saw how to use NLP methods to derive linguistic insights about complex processes such as morphological agreement. For that, we followed a multi-step process of *formalization, feature extraction, model learning, and rule extraction and visualization*. In this chapter, we show how to adapt these steps to the linguistic questions of *case marking* and *word order*, an example of word order is shown in Figure 4.1. Like before, we experiment with several languages for which we design an automated evaluation protocol that informs us how successful our framework is in discovering valid grammar rules (subsection 4.4.1). We also conduct a user study with linguists to evaluate how correct, readable, and novel the rules are perceived to be (subsection 4.4.2). Finally, we apply this framework to a threatened language variety, Hmong Daw (mww), and evaluate how well our framework extracts rules under zero-resource conditions (section 4.5).

## 4.2 Proposed Approach

Similar to section 3.2, we formally define the problem formulation for each linguistic question.

### 4.2.1 Problem Formulation

We formulate a linguistic phenomenon  $p$  as a prediction problem, where given an input set of features  $\mathbf{X}_p = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  we predict an output label  $\mathbf{Y}_p = y_1, y_2, \dots, y_n$  indicating the linguistic phenomena. Next, we determine the set of features which we believe are known to govern the phenomena. For example, in previous Chapter 3 the prediction problem of morphological agreement was to predict whether the values of a morphological property  $m$  match between the head ( $w_h$ ) and the dependent

<sup>2</sup><http://serious-science.org/grammatical-case-morphology-syntax-and-word-order-9354>

token ( $w_d$ ). Below, we describe how we define  $Y$  formally for the other two phenomena of *case marking* and *word order*, and discuss how to construct  $X$  in the following section.

**Case Marking** Earlier, we discussed how there are two types of case (abstract and morphological), and in this work we consider modeling the abstract case. Since the abstract case or case is considered a universal property present inherently for all word classes, we formulate the explanation of case marking determining *when a word class (e.g. nouns) marks a particular case (e.g. nominative, etc.)*. We note that this is a simplifying assumption, as some linguists believe that case assignment is a relation between the head and its dependent, for example, a nominal has nominative case with respect to the verb (Chomsky, 2000). On the other hand, the Dependent Case Theory (DCT) proposes that case assignment is a function of a relation between two determiner phrases (DP’s) and not via a syntactic head (Marantz, 2000; Baker and Vinokurova, 2010), but Puškar and Müller (2018) argue for a unifying approach where case assignment can happen through agreement also. Therefore, to not completely abandon the role of the head in case assignment, we include syntactic features derived from the syntactic head, including agreement (details are discussed in the next subsection 4.2.2). Formally, for each POS tag  $t$  we learn a separate model, where the input examples  $x_i$  are the words that have the POS tag  $t$  with the case feature marked (e.g. Case=Nominative). The model is trained to predict an output label ( $y_i \in Y$ ), where  $Y$  is the label set of all observed case values for that language.

**Word Order** For word order, consider the following five WALS (Dryer and Haspelmath, 2013) relations  $R$ : subject-verb (82A), object-verb (83A), adjective-noun (87A), adposition-noun (85A) and, numeral-noun (89A), which are most popularly studied in literature. In contrast to WALS, which provides only a single canonical order for the entire language, we pose the linguistic question as determining *when does one word in such a relation appear before or after the other*. Formally, the pair of words involved in the syntactic relation  $\langle w_i^a, w_i^b \rangle \in r$  form the input example  $x_i$  and the output label  $y_i \in Y$  where  $Y = \{\text{before, after}\}$ .

#### 4.2.2 Feature Extraction

After formulating each linguistic question into a prediction task, we design features to help predict each question’s answer. In step-2 of Figure 4.1, we demonstrate example features extracted from a Spanish sentence to train the adjective-noun word order model. We refer to the words that participate in an input  $x_i$  as *focus words*. These include the words describing the relation itself (e.g. the adjective `cuatro` and its noun `libros`) and also their respective heads and dependents.

**Syntactic Features** Prior work (Blake, 2009; Kittilä et al., 2011; Corbett, 2003) has discussed the role of syntax and morphology being important in determining the case and agreement. Case is traditionally dependent-marking i.e. the grammatical markers of morphology and case are on the dependents, but these markers can also be found on the heads as well as some languages are head-marking or via agreement. Therefore, we derive features from both the dependents and the heads. In Figure 4.1, we show a subset of features extracted for some of the focus words. For example, for the adjective, we derive features from its POS tag (e.g. “is-adj”), all its morphological tags (e.g. “is-ordinal”) and the dependency relation in which it is involved in (e.g. “deprel-is-mod”). We extract similar features for the adjective’s



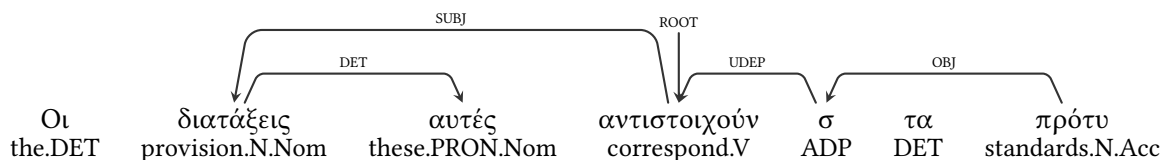


Figure 4.3: Examples of case variation in Greek nouns. For the above sentence – “these provisions correspond to the standards”, the underlined noun takes the nominative case because it is the subject of the main verb. The pronoun also takes the nominative case because its the determiner of the noun which is the subject. A noun takes the accusative case when it is the object.

head, which is *libros* (e.g. “head-is-noun”). In [Chapter 3](#), our set of features for agreement comprised of three features, POS tag of the head, dependent and, the dependency relation between them. Now, we also include these additional features. As motivated in the previous section, agreement itself can govern some phenomenon such as case marking; therefore, we construct a feature “is-agree” which checks if the a morphological property of interest (e.g. case) for a dependent token is having the same value as the syntactic head. Consider the Greek example in [Figure 4.3](#), which shows a type of agreement feature used for case marking. In Greek, typically nouns are in the nominative case when they are subjects and in the accusative case when they are objects. In the same example, we can see that the pronoun is in the nominative case. This is because the noun that the pronoun is modifying is the subject of the main verb, and we know from before that subjects take the nominative case. Therefore, the pronoun will get a feature (“is-agree”) denoting this agreement, which will help capture the rule that – ‘pronouns take the same case as their modifying noun’.

**Lexical Features** An influential family of linguistic theories such as lexical functional grammar ([Kaplan et al., 1981](#)), head-driven phrase structure grammar ([Pollard and Sag, 1994](#)), places most of the explanatory weight for morphosyntax in the lexicon: the properties of the head word (and other words) drive the realization of the rest of the phrase or sentence. Therefore, we add the lemma for the focus words (e.g. “dep-lemma-is-cuatro, head-lemma-is-libro”) as features.

**Semantic Features** There is a strong interaction between semantics and sentence structure. Some well-known examples are of *animacy* or semantic class of a word that determines the case marking ([Dahl and Fraurud, 1996](#)) and word order ([Thuilier et al., 2021](#)) for some languages. Animacy ([Yamamoto, 1999](#)) is the grammatical and semantic property that informs how salient or volitional the referent of a noun is. Grammatical animacy is still annotated as part of syntactic analysis in some languages such as Tamil, however, of the many languages that display semantic animacy, only a few high-resourced languages have publicly available datasets such as English ([Zaenen et al., 2004](#); [Moore et al., 2013](#)). Annotation initiatives have also begun to enrich existing UD treebanks with animacy categories such as for Hindi ([Jena et al., 2013](#)), Swedish ([Nivre et al., 2006](#)). There are also efforts to build automatic animacy classifiers such as for Norwegian ([Øvrelid, 2006](#)), Dutch ([Bloem and Bouma, 2013](#)), Swedish ([Øvrelid, 2009](#)), Japanese ([Baker and Brew, 2010](#)). However, we cannot directly use the available annotations or models to automatically annotate the remaining languages because the categories of animacy are not consistent across different languages, and some languages even have abstract nouns and objects that are animate

(Aissen, 1997; Quinn, 2001).

Massive effort from the community, as undertaken by researchers and linguists, to create Universal Dependencies (Nivre et al., 2016) for syntactic analysis is also required to annotate the animacy for the different languages. Until such a comprehensive resource is available, we use NLP tools to simulate animacy annotations *automatically* in several languages. Instead of annotating binary animacy labels (*animate* vs *inanimate*), we choose to categorize words using fine-grained labels (humans, machines, vehicles, etc.) as done by Zaenen et al. (2004). Therefore, we can formulate the problem of animacy as *identifying and categorizing words into semantic classes*.

Continuous word vectors provide an unsupervised way to achieve this, as these vectors (Mikolov et al., 2013c; Bojanowski et al., 2016) have been used to capture semantic (and syntactic) similarity across words. However, most vectors are high-dimensional and not easily interpretable, i.e. what semantic/syntactic property each individual vector value represents is not obvious. Since our primary goal is to extract comprehensible descriptions of linguistic phenomena, we first generate sparse non-negative vectors using Subramanian et al. (2018), such that each dimension has a higher level of interpretability. For each dimension, we extract the top- $k$  words having a high positive value, resulting in features like  $\text{dim-1}=\{\text{radio,nuclear}\}$ ,  $\text{dim-2}=\{\text{hotel,restaurante}\}$ . This helps us to interpret what properties each dimension is capturing; for example,  $\text{dim-1}$  refers to words about nuclear technology, while  $\text{dim-2}$  refers to accommodations. Now that we can interpret what each feature (dimension) corresponds to, we directly add these vector as features. In Figure 4.1, a semantic feature (e.g. “dep-word-is-like={ochenta,sesenta}”<sup>3</sup>) extracted for `cuatro` informs us that the adjective denotes a numeric quantity.

### 4.2.3 Training Data and Model Learning

**Training Data** Similar to subsection 3.2.3, we construct the training data  $D_{\text{train}}^p$  for each task  $p$  from the raw text  $D$  of the language by performing the complete syntactic analysis, producing POS tags, lemmas, morphological analysis, and dependency trees for each sentence. And, as we show in Chapter 3, such an analysis can also be automatically acquired using state-of-the-art parsers (Kondratyuk and Straka, 2019; Nguyen et al., 2021). Using this analysis, we then identify the focus word(s) and extract the different types of features, forming the input example ( $\mathbf{x}_i = \{x_i^0, x_i^1, \dots, x_i^k\}$ ).

**Model Training** We use decision trees (Quinlan, 1986), like subsection 3.2.3, which are human-interpretable and split the data into leaves, where each leaf corresponds to a portion of the input examples following common syntactic/semantic/lexical patterns.

### 4.2.4 Rule Extraction and Visualization

As we saw in Chapter 3, each leaf in the decision tree is assigned a label based on the distribution of examples within that leaf. However, a majority-based threshold alone is insufficient, as it does not account for leaves with very few examples, which may be based on spurious correlations or nonsensical feature divisions, as found in Chapter 3. Instead, we use the statistical threshold for leaf labeling, as outlined in subsection 3.2.4, performing a chi-squared test to first determine which leaves differ significantly from the base distribution. For this, we first define the null  $H_0$  and test  $H_1$  hypotheses, and in subsection 3.2.4

---

<sup>3</sup>This translates to {eight, sixty}

**adjective** is **before** its head **noun**

Features that make up this rule	
Active Features	Inactive Features
adjective with NumType= Ord	-

Examples that agree with label: **before**. The **adjective** is denoted by \*\*\*

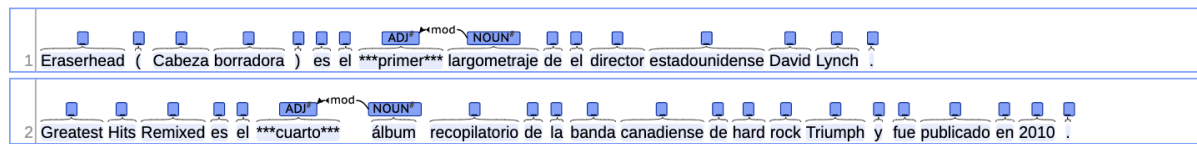


Figure 4.4: A rule extracted for Spanish adjective-noun word order.

we have already described how we designed such hypotheses for agreement. For word order we define a leaf:

- $H_0$  : takes either *before/after* label
- $H_1$  : takes the label dominant under that leaf

We design such  $H_0$  as the words participating in the relation can either be *before* or *after* the other. For case marking, we follow a similar approach to that explained for word order. We can design  $H_0$  as word order, because in the abstract case viewpoint, case is a universal property for each word. We then apply the chi-squared test and compute the p-value. Leaves which are not statistically significant are assigned the label of *cannot decide*, which informs the user that the model is uncertain about the label. Leaves that pass this test are then assigned the majority label and correspond to a rule that will be shown to linguists, where the “rule” is described by the syntactic/semantic/lexical features on the branch that lead to that leaf. After the leaf labeling step, we merge the sibling leaves to get a concise tree, as done in [subsection 3.2.4](#).

**Rule Visualization** For each rule, we extract illustrative examples from the underlying corpus and visualize them in an interface ([Figure 4.4](#)) Since we only show a small set of examples (10 positive and 10 negative, [subsection 3.2.4](#)), we select these examples to be concise and representative. We first group the examples under the rule with the lemmatized forms of the focus words. For example, under the Type-1 rule ([Table 4.1](#)) extracted for Spanish adjective-noun word order, the focus words are **adjective** ( $w_a$ ) and **noun** ( $w_b$ ). We group these examples by the lemmatized forms of the adjective and noun  $\langle l_a, l_b \rangle$ . The examples grouped under a lemmatized pair  $\langle l_a, l_b \rangle$  are then sorted by their lengths. For each lemmatized pair  $\langle l_a, l_b \rangle$ , we select the top-k shortest examples. Finally, all selected examples are shuffled, and we randomly select 10 examples.

### 4.3 Experimental Settings

Similar to our experimental setup in [section 3.3](#), we first experiment with gold-standard syntactic analysis ([subsection 4.4.1](#)) and then manually verify a subset of these extracted rules ([subsection 4.4.2](#)). There-

fore, we evaluate our extracted rules using both an *automated evaluation* where we measure accuracy against a test set and, *human evaluation* where we present rules to language experts for verification.

**Data and Model** Like Chapter 3, we use Syntactic Universal Dependencies v2.5 (SUD) (Gerdes et al., 2019) treebanks and experiment with treebanks for 61 languages, which are publicly available with annotations for POS tags, lemmas, dependency parses, and morphological analysis. Syntactic and lexical features are extracted directly from these gold syntactic analyses. Semantic features are derived from continuous word vectors: we start with 300-dim pre-trained fasttext word vectors (Bojanowski et al., 2017) which are transformed into sparse vectors using Subramanian et al. (2018)<sup>4</sup>. Last, we use the XGBoost (Chen and Guestrin, 2016) library to learn the decision tree. For each language, the running time of the model is approximately 2-5 mins. We perform a grid search over a set of hyperparameters and select the best-performing model based on the validation set performance. Here are the hyperparameters we use:

- `criterion`: {gini, entropy}
- `max-depth`: {3, 4, 5, 6, 7, 8, 9, 10, 15, 20}
- `n-estimators`: 1
- `learning-rate`: 0.1
- `objective`: multi:softprob

### 4.3.1 Automated Evaluation

We describe the automatic evaluation process for the linguistic phenomenon of word order and case marking. For agreement, we follow the same protocol as subsection 3.3.2 and compare the new model with the previous model, which used simple syntactic features such as POS of the head, the dependent, and the dependency relation between them.

**Case Marking** As noted earlier, we use the UD scheme to derive the training data. Under this scheme, not every word is labeled with *case*, restricting our training and evaluation to only such labeled examples. For simplicity, we consider *case* to be a universal property i.e. each word marks a particular *case* value and, we evaluate whether our model can correctly predict that value. Thus, we measure the accuracy on a test example  $\langle \mathbf{x}_i, y_i \rangle \in D_{\text{test}}^t$ , comparing the model's prediction  $\hat{y}_i$  with the observed case value  $y_i$ . We compare our model against a frequency-based baseline which assigns the most frequent case value in the training data to all input examples.

**Word Order** Similarly, we assume that every input example has a word order value, for example subjects will occur either *before* or *after* the verbs. Therefore, for an input example, we consider the observed order to be the ground truth and compute the accuracy by comparing it with the model's prediction. We compare against a frequency-baseline where the most frequent word order value is assigned to all input examples.

---

<sup>4</sup><https://github.com/harsh19/SPINE>

Q1. Looking at the examples below, is the rule

- precisely defining a linguistic distinction
- too specific
- too general
- not corresponding to a real linguistic distinction in the language
- cannot decide as the examples are incorrectly parsed

Q2. If you selected any of the first three options in Q1, does it match the rules you provided earlier? If you selected the fourth option in Q1, leave blank.

- Yes, precisely
- Yes, not exactly but somewhat
- No, but I was aware of such a construction
- No, I was not aware of this before

Q3. Do the features accurately describe the group of positive samples below? If this is a "default" rule, leave blank.

- Yes
- No
- Partially correct

If there's an alternative set of features that more accurately or concisely describe them, please briefly describe them in the comment box.

Other comments:

Figure 4.5: Rule evaluation form presented to the language expert.

Comparing the model’s prediction with the observed order is reasonable for languages which have a dominant word order. There is a considerable set of languages which have a freer order. WALS labels such relations as “no dominant order” (e.g. subject-verb order for Modern Greek). For such cases, considering accuracy alone might be insufficient as there is no ground truth. Therefore, we also report the entropy over the predicted distribution:

$$H_{\text{wo}}^r = - \sum_{k=\text{before, after}} p_k \log p_k$$

$$p_k = \frac{\sum_{\langle \mathbf{x}_i^r, y_i \rangle \in D_{\text{test}}^r} \mathbb{1} \left\{ \begin{array}{l} 1 \quad \hat{y}_i = k \\ 0 \quad \text{otherwise} \end{array} \right.}{|D_{\text{test}}^r|}$$

For languages with no dominant order, the model should be uncertain about the predicted order and we expect the model’s entropy to be high. The accuracy computed against the observed order is still useful, as despite there being “no dominant order”, speakers tend to prefer one order over the other. A high accuracy would entail that the model was successful in capturing this “preferred order.”

### 4.3.2 Human Evaluation

In [Chapter 3](#), we had verified the rules extracted for correctness with the help of language experts. We are also interested in checking if the rules are of assistance to the linguists and for that we evaluate *prior knowledge* and *feature correctness*. Before starting with the actual evaluation, we first ask the expert to provide answers regarding the linguistic questions we are evaluating. For example, we ask questions such as “when are subjects after verbs in Greek”, and they are required to provide a brief answer (e.g. “for questions or when giving emphasis to a subject”). We then direct them to our interface where we show the extracted features and a few examples for each rule, then ask questions regarding each of

Type	Rule Features	Examples	Label
Type-1 (valid)	Adj is a Ordinal	También se utilizaba en las <b>primeras</b> grabaciones y arreglos jazzísticos. <i>It was also used in <b>early</b> jazz <u>recordings and arrangements</u>.</i> Las <b>primeras</b> 24 <u>horas</u> son cruciales. <i>The <b>first</b> 24 <u>hours</u> are crucial.</i>	Before
Type-2 (valid, not informative)	Adj belongs to group: con,como,no,más,lo	Matisyahu piensa editar pronto un <b>nuevo</b> <u>disco</u> grabado en estudio. <i>Matisyahu plans to release a <b>new</b> <u>studio-recorded</u> album soon.</i> Es una experiencia <b>nueva</b> <u>estar</u> desempleado. <i>It's a <b>new</b> <u>experience</u> being unemployed</i>	Before
Type-3 (valid, too general)	Adj is NOT Ordinal	Además de una <b>gran</b> <u>variedad</u> de aplicaciones <i>In addition to a <b>great</b> <u>variety</u> of applications.</i> Una <u>unión</u> <b>solemnizada</b> en un país extranjero <i>An <u>union</u> <b>solemnized</b> in a foreign country</i>	After
Type-4 (valid, too specific)	Adj's lemma is numeroso	En África hay <b>numerosas</b> <u>lenguas</u> tonales <i>In Africa there are <b>numerous</b> <u>tonal</u> <u>languages</u></i> Ellas poseen <b>varios</b> <u>libros</u> <i>They own <b>several</b> <u>books</u></i>	Before
Type-5 (invalid)	Adj's head noun is a conjunct	Las consecuencias de cualquier (colapso) de divisa e <u>inflación</u> <b>masiva</b> . <i>The consequences expected from any currency collapse and <b>massive</b> <u>inflation</u>.</i> (Realizan) trabajos de alta calidad , muy <b>buenos</b> <u>profesionales</u> <i>They do high quality work, very <b>good</b> <u>professionals</u></i>	After

Table 4.1: Types of rules discovered by the model for Spanish adjective-noun word order. **Adjectives** are highlighted and the nouns they modify are underlined. Illustrative examples under each rule are also shown with their English translation in italics. Label denotes the predicted order.

the three parameters (Figure 4.5). Each rule consists of the features identified by the model and the set of illustrative examples.

Regarding *correctness*, the expert is asked to annotate whether the illustrative examples, shown for that rule, are governed by some underlying grammar rule. If so, they are then required to judge how precise it is. Consider some rules extracted for Spanish adjective-noun order in Table 4.1. Looking at the examples and features for the Type-1 rule, it is evident that this rule *precisely defines the linguistic distinction*.<sup>5</sup> Some rules, although valid, may be too general (Type-3) or too specific (Type-4). The Type-3 rule is clearly *too general*, as there are considerable number of adjectives which come before nouns even when they are not ordinals. The Type-4 rule is *too specific* because although it is correct, it does not generalize to other similar examples. Finally, a rule *may not correspond to any underlying grammar rule*, like Type-5 where the model simply discovered a spurious correlation in the data. For *prior knowledge*, if an extracted rule was indeed a valid grammar rule, then we ask the expert if they were aware of such a rule. This will inform us how useful our framework is in discovering rules which a) align with the expert's prior knowledge and, b) are novel i.e. rules which the expert were not aware of apriori. Finally, for *feature correctness*, we ask whether the features selected by the model accurately describe said rule. For the Type-1 rule, the answer would be *yes*. But for rules like Type-2, the features are not informative even though the corresponding examples do follow a common pattern.

## 4.4 Gold-Standard Experiments

We discuss the results of models trained on the SUD treebanks.

<sup>5</sup><https://www.thoughtco.com/ordinal-numbers-in-spanish-3079591>

Linguistic Phenomena	Model	Gain
Word Order	adjective-noun	2.61
	subject-verb	6.95
	object-verb	10.78
	numeral-noun	9.88
	noun-adposition	2.31
Agreement	Gender	4.02
	Person	1.08
	Number	4.95
Case Marking	NOUN	30.03
	PRON	32.66
	DET	47.33
	PROPN	29.77
	ADJ	35.59
	VERB	18.76
	ADP	15.4
	NUM	25.81

Table 4.2: Breakdown of the performance gain (over the baseline) for each linguistic question. The performance of the agreement models is compared with the models trained over simple syntactic features in [Chapter 3](#).

#### 4.4.1 Automated Evaluation Results

We train models using syntactic features for all languages covered by SUD, wherever the linguistic question is applicable. We find that our models outperform the respective baselines by an (avg.) accuracy of +7.3 for word order, +28.1 for case marking, and +4.0 for agreement. We also experimented with Random forests and found decision trees to be slightly underperforming ((avg.) -0.12 acc). However, given that it is straightforward to extract interpretable rules from the latter, which is our primary goal, we use the decision trees for all experiments.

We also report the result breakdown under three resource settings, low, mid, and high, where low-resource refers to the treebanks with < 500 sentences, mid-resource has 500 – 5000 sentences and high-resource has > 5000 sentences. Across all three linguistic phenomena, the (avg.) model gains over the baseline are +3.19 for the low-resource, +10.7 for the mid-resource and +12.8 for the high-resource. The larger the treebank size, the larger the improvement of our model’s performance over the baseline. Even in low-resource settings, a gain over the baseline suggests that our approach is extracting valid rules, which is encouraging for language documentation efforts. We present the result breakdown of individual relations in [Table 4.2](#).

As motivated in [subsection 4.2.2](#), the conditions which govern a linguistic phenomenon vary considerably across languages, which is also reflected in our model’s performance. For example, the model trained on syntactic features alone is sufficient to reach a high accuracy (avg.94.2%) for predicting the adjective-noun order in Germanic languages. But for Romance languages, using only syntactic features leads to much lower performance (avg.74.6%). We experiment with different features and report results for a subset of languages in [Figure 4.6\(a\)](#). Observe that for Spanish adjective-noun order adding lexical

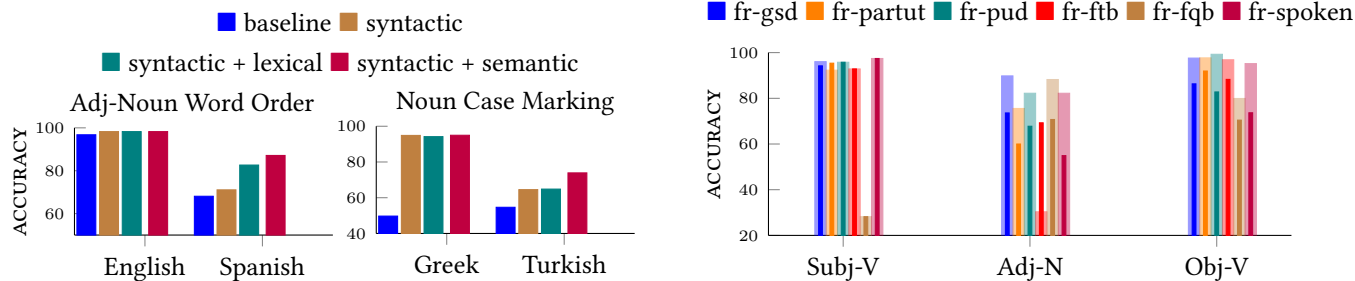


Figure 4.6: (left) Comparing the effect of different features on the word order and case marking. (right) Comparing the accuracy of the model across different treebanks of `fr-gsd`.

features improves the performance significantly (+11.57) over syntactic features, and semantic features provide an additional gain of +4.48. Studying the languages marked as having “no dominant order” in WALS, we find that our model shows a higher entropy. SUD contains 8 such languages for subject-verb order, and our model produces an (avg.) entropy of 1.09, as opposed to (avg.) 0.75 entropy for all other languages. For the noun case marking in Greek, syntactic features already bring the model performance to 94%. For Turkish, the addition of semantic features raises the model performance by +9.38. The model now precisely captures that nouns for locations like *ev*, *oda*, *kapı*, *dünya*<sup>6</sup> typically take the locative case. This is in line with [Bamyacı and von Heusinger \(2016\)](#) which outlines the importance of animacy in Turkish differential case marking.

To confirm that these *discovered conditions generalize to the language as a whole and not the specific dataset on which it was trained*, we train a model on one treebank of a language and apply the trained model directly on the test portions of other treebanks of the same language. There are 30 languages in the SUD which fit this requirement. Figure 4.6(b) demonstrates one of those settings to understand word order patterns in different French corpora, where the models have been trained on the largest treebank (`fr-gsd`). For subject-verb order, all treebanks except `fr-fqb` show similar high test performance (>90% acc.). Interestingly, the model severely underperforms (28% acc.) on `fr-fqb` which is a question-bank corpus comprising of only questions, and questions in French can have varying word order patterns.<sup>7</sup> The model fails to correctly predict the word order because in the training treebank only 1.7% of examples are questions making it challenging for the model to learn word order rules for different question types.

Through this tool, a linguist can potentially inspect and derive insights on how the patterns discovered for a linguistic question vary across different settings, both within a language and across different languages as well.

#### 4.4.2 Human Evaluation Results

Through the above experiments, we *automatically* evaluated that the extracted rules are predictive (to some extent) and applicable to the language in general. Before applying this framework on an endan-

<sup>6</sup>house, room, door, world

<sup>7</sup>In questions such as *Que signifie l'acronyme NASA?* (“What does the acronym NASA mean?”), the verb comes before its subject, but for questions such as *Qui produit le logiciel?* (“Who produces the software?”) the subject is before the verb.



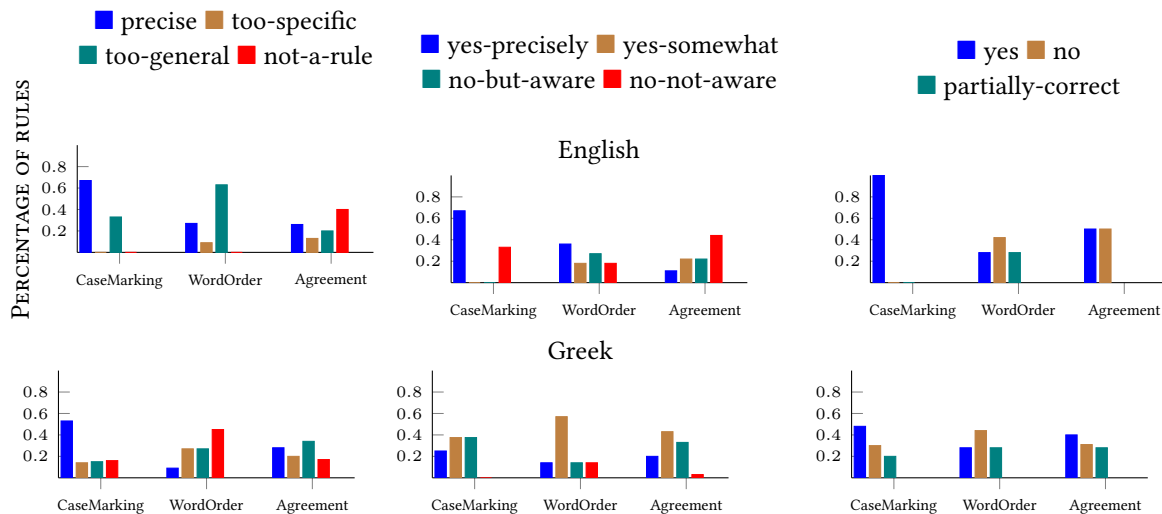


Figure 4.7: Evaluating rule correctness (left), prior knowledge (middle) and feature correctness (right). Top plot shows the results for English while the bottom plot shows for Greek.

Linguistic Phenomena	Rule	Examples	Label
Number Agreement	dependent’s head is a NOUN	<b>Kids</b> fun games are added to the building. <b>Nationalist</b> <u>groups</u> are coming to the conference.	Not-required-agreement
Object Case Marking	Pronoun is a oblique	Because Large Fries give <b>you</b> FOUR PIECES ! Give <b>him</b> a call tommorow	Accusative

Table 4.3: Some example of rules for agreement and case marking, which the expert annotator was not aware of. The **focus word** is highlighted, for agreement we also underline the head with which the dependent’s agreement is checked. The examples under number agreement demonstrate that when dependent’s head is a noun the **dependent** need not agree with its head. We show one example where the first example shows the dependent matches the number of the head, and the second example shows that it didn’t not match.

gered language we first perform a manual evaluation ourselves for English and Greek. We select these languages based on the availability of human annotators, using one expert each for English and Greek. First, we note that the total number of rules for English (29) are much less than that for Greek (161), the latter being more morphologically rich.

We find that 80% of the rules (across all phenomena) are valid grammar rules for both languages. A significant portion (40%) of the valid rules are either too specific or too general, which highlights that there is scope of improvement in the feature and/or model design. Interestingly, even for English, there were 7 rules which the expert was not aware of, as shown in Table 4.3. For example, the following rule for adjective-noun order – “when the nominal is a word like *something, nothing, anything*, the adjective can come after the noun.“. For Greek, almost all valid rules were known to the expert, except for one Gender agreement rule which was, “proper-nouns modifiers do not need to necessarily agree with their head nouns”. Regarding feature correctness, the Greek expert found 69% of the valid rules to be readable and informative, while the English expert found 58% of such rules. We show individual results in Figure 4.7.

These insights may have utility even for languages that already have automatic NLP tools for POS

tagging or dependency parsing, or even a treebank, as existing annotations do not exhaustively describe fine-grained or complex linguistic behaviors on a holistic level (e.g. deviation in word order patterns or explaining the process of agreement). From the user-study above, we do find that the approach discovered fine-grained behaviors for English and Greek, which the language experts were not aware of or could not think of readily. In addition, even if language documentation does exist for a language, this does not mean that it is readily available in a standardized machine-readable format, whereas the output of our method is.

## 4.5 Hmong Daw Study

To test the applicability of AUTOLEX in a language documentation situation, we experiment with Hmong Daw (mww), a threatened language variety, spoken by roughly 1M people across US, China, Laos, Vietnam and Thailand. This variety can be categorized as a low-resourced language with respect to computational resources and accessible and detailed machine-readable grammatical descriptions. Furthermore, this study presents a realistic setting for language analysis, as there is no expert-annotated syntactic analysis available.

One of co-authors of this work is a Hmong linguist who is in close collaboration and consultation with the community and is the expert who provided us with the Hmong data and helped evaluate the extracted grammar rules. We had access to 445k Hmong sentences, which were collected from the `soc.culture.hmong` Usenet group (Mortensen et al., forthcoming). Since the data was scraped from the Web, it was noisy and intermixed with English. Therefore, first we automatically clean the corpus using a character-level language model trained on English. This automatically filtered 61k sentences. Next, we automatically obtain syntactic analyses using `Udify` (Kondratyuk and Straka, 2019). We use training data from Vietnamese, Chinese and English treebanks and apply the resulting model to the Hmong text. We randomly split the parsed data into a train and test set (80:20) and apply our general framework to extract rules.

**Results** Hmong has no inflectional morphology, so we only train the model to answer word order questions. We conduct the expert evaluation on four relations where our model outperforms the baseline, albeit slightly (+4.08 for Adj-N, +0.12 for Subj-V, +0.52 for Adp-N, +0.72 for Num-N). For Obj-V relation, our model is on par with the baseline which could indicate that there were not many examples whose word order deviated from the dominant order or the model needs improvement. First, we ask the expert, a linguist who studies Hmong, to describe the rules (if any) for each relation. Compared to the rules provided by the expert, we find that the model is successful in discovering the dominant pattern for all relations. However, of the 30 rules (across all relations) presented to the expert for annotation, only 5 rules (1 rule for subject-verb, 4 rules for numeral-noun) were found to precisely describe the linguistic distinction. For instance, according to the expert, numerals cannot occur immediately before nouns, rather they occur before classifiers which then occur before nouns (“1 clf-1 noun-1”). Interestingly, one rule captured examples where the numerals occurred immediately before nouns without the classifiers (e.g. “1 noun-1, 2 noun-2”), which the expert was not aware of. On the one hand, this is promising as the model, despite being trained on noisy sentences and syntactic analyses, was able to discover instances of interesting linguistic behavior. However, the expert noted that a large portion of the rules were difficult

to evaluate, as these referred to examples which were incorrectly parsed, some of which even described the English portion of code-mixed data.

Despite showing the promise of automatically obtaining detailed descriptions on languages with good syntactic analyzers, we can see that it is still challenging to apply methods to such under-resourced languages. This poses a new challenge for zero-shot parsing; even the relatively strong model of (Konratyuk and Straka, 2019) resulted in a high enough error rate that it impacted the effectiveness of our method, and methods with higher accuracy may further improve the results of end-to-end grammar descriptions generation.

## 4.6 Other Applications

Along with helping linguists and researchers in language exploration efforts, these machine-readable rules can also be used in NLP applications, such as to evaluate natural language generation (NLG) outputs. Specifically, Pratapa et al. (2021a) propose the L'AMBRE metric to evaluate the morphosyntactic well-formedness of text by applying such automatically extracted grammar rules on machine outputs. As we saw in this and the previous chapter, such grammar rules can be extracted for many languages, which makes L'AMBRE multilingually applicable, also, thus providing a referenceless metric for advancing NLG in multiple languages. Natural language text from the machine output is first run through a parser to identify the syntactic/semantic/lexical information for each sentence. The rules are applied to each parsed sentence to check whether that sentence follows the rule. The metric can be explored [here](#).

## 4.7 Conclusion

In this chapter, we presented our general framework, which allows a linguist to ask questions about different linguistic behaviors. Each linguistic question is formulated as a prediction task from which we then extract and visualize concise human- and machine-readable rules. While the framework extracts decent quality rules for languages with high-quality syntactic analysis, we do find that for true under-resourced languages such as Hmong, the quality of rules depends on the quality of the underlying parses.



## Chapter 5

# L2 Semantic Subdivisions

In addition to understanding the syntax and morphology of a language, as we did in [Chapter 3](#) and [Chapter 4](#), the semantics of a language also forms a key component in language understanding and learning. In this chapter, we focus on one aspect of semantics – *lexical semantics* which addresses meaning at the level of words, and we explore it in the context of *second-language acquisition* (SLA) ([Settles et al., 2018](#)). SLA or L2 (language 2) acquisition refers to the process of learning a new language. A popular pedagogical technique for SLA is using associations with the *learner language*, also referred as L1 (language 1) ([Hulstijn et al., 1996](#); [Watanabe, 1997](#)). However, different languages carve their semantic space differently, for instance, ‘wall’ in English is called as ‘pared’ in Spanish when referring to an indoor wall and ‘muro’ when referring to an outdoor wall. Learning the usage of such fine-grained lexical distinctions might be challenging for L2 learners, more importantly, because these distinctions do not exist in the learner language (L1). Therefore, to aid L2 learners in their learning process, we follow the AUTOLEX framework to automatically identify such distinctions and extract human- and machine-readable rules to explain them.

Aditi Chaudhary, Kayo Yin, Antonios Anastasopoulos, Graham Neubig. 2021. [When is Wall a Pared and when a Muro?: Extracting Rules Governing Lexical Selection](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

### 5.1 Overview

With increasing globalization there is a widespread prevalence and need for creating good materials and tools to help people learn new languages. A recent report<sup>1</sup> by Duolingo<sup>2</sup>, a popular language learning application, shows that people are learning new languages for a variety of reasons, including help with school curriculum, professional work, tourism, culture and so on. In addition to individual motivations, communities and governments are taking initiatives to teach indigenous languages ([Moline, 2020](#)) for preserving cultural heritage and knowledge (e.g. [Ullrich et al. \(2020\)](#) for Owóksape; [Longenecker et al. \(2019\)](#) for Kala, [Yotsumoto \(2020\)](#) for Ainu). Furthermore, the demand for digital and online learning applications has seen a substantial increase due to the COVID-19 pandemic, which forced several peo-

---

<sup>1</sup><https://blog.duolingo.com/global-language-report-2020/>

<sup>2</sup><https://www.duolingo.com/>

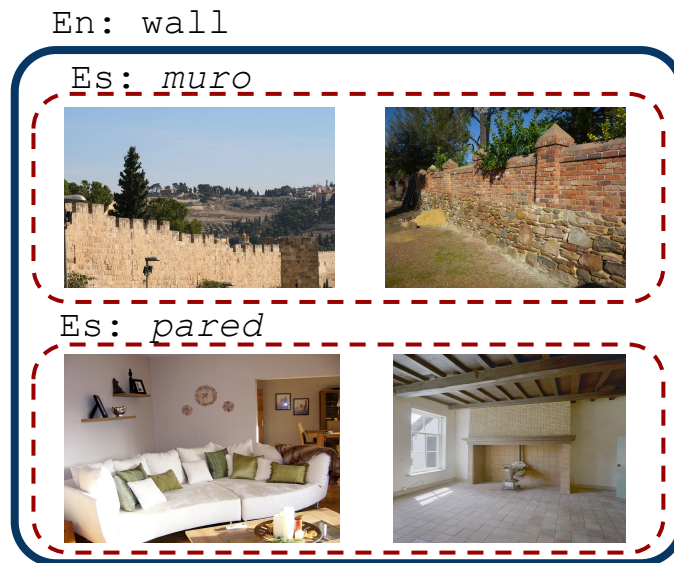


Figure 5.1: Semantic subdivision for the concept ‘wall’ results in different lexical manifestations in Spanish: ‘muro’ for *outside wall* and ‘pared’ for *inside wall* whereas in English both are referred as ‘wall’.

ple to be in lockdown across the world (Li and Lalani, 2020). Curating such learning content manually can be time-consuming and expensive, but more importantly having access to such language experts can become challenging, especially for languages where the experts are remote and inaccessible. Therefore, in this chapter, we present a method for content creation for learning words in a new language automatically.

*Vocabulary acquisition*, the process of learning new words, is a key step in language acquisition. As mentioned above, a popular approach to L2 acquisition is by associating words or forms with a language familiar to the learner, typically referred to as L1 (Watanabe, 1997; Jiang, 2002). In fact, Jiang (2002) suggest that learners map L2 lexical forms to L1 semantic space instead of mapping them to their own new semantic specifications. However, semantic structures vary across languages leading to *semantic subdivisions* where lexical distinctions made in one language are not present in the second (Bowerman and Choi, 2001). An example of this can be seen in Figure 5.1 where “wall” in English manifests itself as “pared” and “muro” depending on the location of the wall. Learning such fine-grained lexical distinctions might not be obvious to a learner, and having a skilled teacher or a comprehensive learning resource may be able to provide explanations to aid in L2 learning.

Some early examples of such resources are: GLOSSER (Nerbonne et al., 1998) which helps Dutch speakers learn French by describing morphology, word usage in context; CAVOCA (Groot, 2000) where a learner is taken through different phases of the vocabulary acquisition process, including word definitions, examples of word usage, etc. More recently, Revita (Katinskaia et al., 2017) supports endangered language learning using exercises for grammar and vocabulary practice. Similarly, SMILLE (Zilio et al., 2017) is a reading assistant that helps users understand linguistic structures while reading text in a target language. Robertson (2020) present word-definitions in context to Finnish learners while browsing Finnish text on the web. Duolingo, Rosetta Stone (Stone, 2010) are some popular language learning applications that are available online. However, most of the above listed works rely on language content

curated manually by language experts, which makes it difficult to scale to numerous languages.

In this work, we propose a method to *automatically* discover learning content explaining fine-grained lexical distinctions and present L2 learners with concise explanations in an interactive framework. Research in L2 vocabulary acquisition (Groot, 2000; Ortega, 2015; Godwin-Jones, 2018) has shown that it is effective to combine strategies using explicit definitions and examples in context. Therefore, we present these concise rules to learners along with illustrative examples of the word in context. Like previous chapters, we follow the four steps of AUTOLEX i.e. formalization, feature extraction, model learning, rule extraction and visualization. In Chapter 3 and Chapter 4 we used monolingual data of the target language as our starting point, in this case, we i) use a parallel corpus to identify words in L1 which show different lexical manifestations owing to a semantic subdivision in L2, and ii) create human- and machine-readable rules by training a prediction model that distinguishes between the lexical choices, which allow for easier interpretation of each lexical distinction. These rules can be used as-is (as done by us in this chapter), or could be used a starting point for further curation by educators (discussed in next Chapter 6).

Since the primary motivation of this work is to help with language learning, we confirm the quality of the extracted rules by conducting an interactive study in which we use the rules to teach Spanish words and Greek words from English, focusing on the words arising from semantic subdivisions. We make this study interactive by presenting the learning content in the form of *cloze* tests (Taylor, 1953) where the lexical distinctions in Spanish or Greek to be taught are presented to the learner in an English context together with concise rules. Concretely, the learner is presented with an English sentence containing the word (e.g. “wall”) which shows different lexical distinctions in the target language and is required to select the most appropriate lexical choice ( e.g. “pared” vs “muro” for Spanish) from the given set. We conduct a parallel study with a control group where we do not show learners the extracted rules and instead they are required to learn distinctions and answer using only the English context. Like before, we also confirm the quality of the model through automated evaluation before proceeding with the human evaluation. Our contributions are summarized below:

1. We present an *automatic* framework to identify semantic subdivisions in L2 from L1. Our approach is able to identify 407 such words in Spanish and 707 words in Greek, across different word classes.
2. We create an interactive learning exercise and experiments with 7 Spanish learners and 9 Greek learners show that they learn faster when given access to the rules; for example they achieve an (avg.) accuracy of 81% in roughly 20 questions as opposed to 40 questions required by control-group learners to achieve the same accuracy.

## 5.2 Proposed Approach

We formally define the problem of lexical choice selection and describe the procedure for rule extraction.

### 5.2.1 Problem Formulation

In the context of this work, we define cross-lingual lexical selection (Lefever and Hoste, 2010) or substitution (Mihalcea et al., 2010) as the task of selecting contextually appropriate words in one language given a word in context in another language.

Formally, given a sentence in the source language (SL)  $\mathbf{x} = x_1, x_2, \dots, x_{|\mathbf{x}|}$ ,  $\text{trans}(x_i) \subseteq V_y$  denotes the set of “possible” target translations for the source word  $x_i$ , i.e. words in the target language (TL) to which the focus word  $x_i$  could be translated (concrete methods to define this set are explained later). We denote  $\mathbf{y} = y_1, y_2, \dots, y_{|\mathbf{y}|}$  as the translation of  $\mathbf{x}$  in the target language (TL) and  $V_x$  and  $V_y$  are the source and target vocabulary, respectively. The task of cross-lingual lexical selection involves choosing the most appropriate translation  $y_i \in \text{trans}(x_i)$ , which can be performed by machines or humans.<sup>3</sup> In this work, we focus on *machine-learned methods to help humans learn lexical selection*, extracting lexical selection models that are not only usable by machines but also interpretable by humans in order to aid the process of learning a new language.

In the next paragraph, we first present a method to identify L1 words that show semantic subdivisions in L2 using a parallel corpus. Next, we train a lexical selection model which allows extraction of human- and machine-readable rules (subsection 5.2.3). Finally, we present our evaluation framework where we help to teach L2 learners these words using our extracted rules (section 5.3).

**Identifying Semantic Subdivisions** In this section, we describe in detail the steps to identify L1 words that show different lexical manifestations in L2 owing to the semantic subdivision. Going forward, we refer to the L1 word in question as *focus word* and the corresponding L2 distinctions as *lexical choices*. We extract these focus word-lexical choice pairs from a parallel corpus  $D = \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_{|D|}, \mathbf{y}_{|D|})\}$  where  $(\mathbf{x}_m, \mathbf{y}_m)$  denotes the source and target sentence pairs. As a preliminary step, we automatically extract word alignments using a word aligner that produces sets of pairs of source and target words  $A_m = \{\langle x_i, y_j \rangle : x_i \in \mathbf{x}_m, y_j \in \mathbf{y}_m\}$ . Since our aim is to discover semantic subdivisions as opposed to morphological variations, we normalize the parallel corpus by lemmatizing all words. Thus,  $V_x$  and  $V_y$  refer to the lemmatized vocabulary of the source and target language. All references to words refer to their respective lemmatized forms going forward. Furthermore, we perform automatic part-of-speech (POS) tagging, dependency parsing, and word sense disambiguation (WSD) on the source side data, resulting in a POS tag and word sense associated with each source word,  $\text{tag}(x_i) \in T_x$  and  $\text{sense}(x_i) \in S_x$  where  $T_x$  is the set of POS tags in the source language and  $S_x$  is the word sense vocabulary in the source language. Using the automatic analysis described above, we extract the L1 word types that display distinct L2 lexical choices. We use the POS information to further filter the L1 word types, giving us tuples of the form  $\langle v_x, t_x \rangle$ . This ensures that we do not conflate meanings across POS tags, because in many languages the semantics of a word can vary widely across its different POS tags.<sup>4</sup> We now describe the steps to extract semantic subdivisions which result in L1 word-POS tuples with their corresponding L2 lexical choices.

1. **Extract translations:** For each aligned word pair  $\langle x_i, y_j \rangle$ , we compute the number of times the lemmatized source word type ( $v_x = \text{lemma}(x_i)$ ) along with its POS tag ( $t_x = \text{tag}(x_i)$ ) is aligned to the lemmatized target word type ( $v_y = \text{lemma}(y_j)$ ) across the whole corpus in  $c(v_x, t_x, v_y)$ . Also, store the number of times the word sense of  $x_i$  ( $s_x = \text{sense}(x_i)$ ) appears with the source word type, source POS tag and the translation word type in  $g(v_x, t_x, s_x, v_y)$ .
2. **Filter on frequency:** Extract tuples of source types and POS tags  $\langle v_x, t_x \rangle$  that have been aligned to at least two target words at least 50 times ( $\{v_y : |c(v_x, t_x, v_y)| \geq 50\} \geq 2$ ), to account for

<sup>3</sup>The notation here refers to single-word translations which are the focus of this work.

<sup>4</sup>“Brown” as a verb (as in “brown the meat”) is treated differently from the adjective sense (as in “brown hair”).



alignment errors. To avoid ambiguity on the target side, translations aligned to words other than the word  $v_x$  in question (at least 3 times) are excluded.

3. **Filter on entropy:** Remove source tuples that have an entropy  $H(v_x, t_x)$  less than a pre-selected threshold. The entropy is computed using the conditional probability of a target translation given the source type and POS tag:

$$p := p(v_y | v_x, t_x) = \frac{c(v_x, t_x, v_y)}{c(v_x, t_x)} \quad (5.1)$$

$$H(v_x, t_x) = \sum_{v_y \in \text{trans}(v_x, t_x)} -p \log_e p \quad (5.2)$$

where  $\text{trans}(v_x, t_x)$  is the set of target translations for the source tuple  $\langle v_x, t_x \rangle$  and  $p(v_y | v_x, t_x)$  is the conditional probability of the target translation for the source word type  $v_x$  and its POS tag  $t_x$ . A high entropy suggests that the word is ambiguous, with fine-grained distinctions that likely require context to resolve, and thus this word is one that we can focus on.<sup>5</sup>

4. **Filter on word sense:** Remove source word-POS tuples whose target translations have distinct source word senses. For some words, the differences between target translations can be straightforwardly explained by the different source word senses. For example, *banco* in Spanish refers to the financial institution, given by the WordNet (Miller, 1995) sense ‘bank.n.02’ while *orilla* refers to the edge of a river, outright matched to ‘bank.n.01’. For this study, we are interested in finding those focus words where the word sense information alone is insufficient to distinguish between the lexical choices and are hence likely to be hard for human learners. For a source tuple, use the highest occurring word sense for a given target translation  $v_y$  computed as:

$$Q(v_y) = \operatorname{argmax}_{s_x \in S_x} g(v_x, t_x, s_x, v_y) \quad (5.3)$$

Finally, retain the source word-POS tuples whose target translations all have the same sense.

$$W_{\text{ambig}} = \{ \langle v_x, t_x \rangle; Q(v_{y_0}) = \dots = Q(v_{y_{|\text{trans}(v_x, t_x)|}}) \} \quad (5.4)$$

## 5.2.2 Feature Extraction

The linguistic question, therefore, becomes ‘given an L1 word in context and a set of L2 lexical choices, which L2 choice is the most appropriate’. Therefore, the input to the model is the source sentences containing the focus L1 word  $\mathbf{x}_{\langle v_x, t_x \rangle} \in D_{\langle v_x, t_x \rangle}$  and the model is trained to predict the contextually correct target translation  $v_y$  from a set of possible  $k$  choices  $L(v_x, t_x) = v_{y_1}, v_{y_2}, \dots, v_{y_k}$ . For designing features, we take inspiration from prior work which uses extracted contextual information to improve cross-lingual sense disambiguation in machine translation systems (Garcia-Varea et al., 2001; Carpuat and Wu, 2007b,a). We focus on features extracted only from the current source sentence, although the framework can be easily extended to include features from the target sentence as well. From each source sentence, we extract the following three kinds of features:

<sup>5</sup>To handle lemmatization errors, edit-distance based post-processing is used to group separate lexical choices into a single choice. Hence, an additional filtration step is used to remove source tuples where one choice accounts for 90% of all cases. A heavily imbalanced dataset is undesirable since it might prevent the lexical selection model from extracting informative rules for the minority classes.

Lexical Choice	feature $\rightarrow$ $\langle$ rule name $\rangle$ $\langle$ feature value $\rangle$
muro	Bigram $\rightarrow$ Short phrases: ('climb', 'wall'), ('city', 'wall'), ('brick', 'wall') Lemma $\rightarrow$ Words: break, climb WSD $\rightarrow$ Concepts: 'city' as in a large and densely populated urban area (city.n.01)
pared	Bigram $\rightarrow$ Short phrases: ('face', 'wall'), ('hang', 'wall'), ('picture', 'wall') Lemma $\rightarrow$ Words: ear, hang, room

Table 5.1: Human-readable rules extracted for the ambiguous word *wall* (top-6 rules per lexical choice).

- **Lemma:** lemma of all words within a fixed window of the focus word.
- **WSD:** word sense of all words within a fixed window of the focus word.
- **Bigrams:** bigrams constructed from lemmatized words present within a fixed window of the focus word. We exclude punctuation and stop words from within this window.

### 5.2.3 Training Data and Model Learning

After extracting the L1-L2 tuples, and the features, we train a linear prediction model which allows us to extract human and machine readable rules to explain the selection of lexical choices. We train a prediction model parameterized by  $\theta_{\langle v_x, t_x \rangle}$  for each focus word  $\langle v_x, t_x \rangle$ . As rules, we extract the features which govern the lexical selection for each choice  $v_y \in L(v_x, t_x)$ . These rules are defined over a set of lexical and semantic features extracted from the source sentences in  $D_{\langle v_x, t_x \rangle}$ , as shown above.

**Model Learning** To allow the extraction of interpretable rules, we use a model that is conducive to interpretation: the linear SVM (LinearSVM; Cortes and Vapnik, 1995), which gives us feature weights  $\theta_{\langle v_x, t_x \rangle}$  that can be easily interpreted as the importance of each feature in making the decision.<sup>6</sup> Since there can be  $n$ -ary lexical choices for a given focus word, we train using the one-vs-rest (OvR) method which trains one model per each lexical choice  $v_{y_k}$ , where data from  $v_{y_k}$  are treated as positive examples and data from all other choices as negative, allowing us to extract feature weights for each decision.

### 5.2.4 Rule Extraction and Visualization

We get one model per each choice  $v_{y_k}$  from which we can then extract the top- $N$  features having the highest weight coefficients for each choice. To present these rules in a human-readable form, we create concise rule templates as shown in Table 5.1 for the word “wall”. The bigram features are represented as ‘Short phrases’, lemmas are represented as ‘Words’ and the WSD senses form the ‘Concepts’.

<sup>6</sup>We also examined other interpretable models such as gradient boosted decision trees (Friedman, 2001), which gave less intuitive results. Further, we could use state-of-the-art neural translation models and model interpretation techniques, but we leave this as an interesting challenge for future work.

## 5.3 Experimental Settings

We present two approaches to evaluate our framework: 1) *automated evaluation*, a preliminary validation where we evaluate how well our interpretable model performs in cross-lingual lexical selection (subsection 5.3.1), and 2) *human evaluation*, which answers our main question of whether it can teach human learners the usage of L2 words (subsection 5.3.2).

**Data** We experiment with two L2 languages: Spanish and Greek. These languages were chosen because of (1) the availability of parallel corpora with which to train models and (2) the availability of linguists and annotators to verify and analyze the data used in our experimental setting. For Spanish we use 10 million English-Spanish parallel sentences from OpenSubtitles (Lison and Tiedemann, 2016), Tatoeba, TED (Tiedemann, 2012), and Europarl (Koehn, 2005).<sup>7</sup> For Greek, we use 31 million English-Greek parallel sentences extracted from OpenSubtitles. For word alignment we use the AWESOME aligner (Dou and Neubig, 2021), for lemmatization we use spaCy (Honnibal et al., 2020), for POS tagging and dependency parsing we use Stanza (Qi et al., 2020), and for English WSD we use EWISER (Bevilacqua and Navigli, 2020).<sup>8</sup>

**Model** We implement the LinearSVM model using `sklearn` (Pedregosa et al., 2011). We train one model per each focus word and divide the extracted parallel sentences for that word into a balanced train/test split with 80-20 ratio per lexical choice. We perform 5-fold cross-validation to select the best model hyperparameters from which we then extract the top-20 features for each lexical choice to form our rule set. We clean the data to remove punctuation and extract features within a 3-word window of the focus word. We perform a grid search over the following hyperparameters for training the LinearSVM model. The hyperparameters are: `C:[0.001, 0.01]`, `class_weight:['balanced', None]`.

### 5.3.1 Automated Evaluation

We verify whether our interpretable lexical selection model is able to learn cross-lingual lexical selection at all by measuring its performance compared to selecting the most frequently occurring translation in the corpus for a given focus word (“Frequency”). We also compare with another alternative interpretable model, decision trees (DTree) trained using the same features as LinearSVM, to validate the choice of SVMs as an interpretable model over other alternatives. Further, we check how our interpretable linear SVM model compares with a “performance skyline”; a less interpretable BERT-based neural model (Devlin et al., 2019) that extracts representations of the source sentence from BERT and trains a classifier to predict the correct lexical choice. We train all models for each identified focus word and measure the accuracy on respective portion of the data reserved for evaluation.

Using our automatic pipeline, we identify 407 English words that show distinct lexical manifestations in Spanish, and 707 such words for Greek, the distribution of which is shown in Figure 5.2. A manual inspection by a Greek-English bilingual speaker revealed that most of the automatically created lexical choices (> 90% of 100 words) were correct. In just a couple of cases, lemmatizer errors lead to two

---

<sup>7</sup>We use only 1 million sentences from Europarl because we found sentences from Europarl to contain fewer semantic subdivisions owing to the very specific domain of the dataset.

<sup>8</sup>POS tagging, dependency parsing and WSD is required *only* for the source language, here English.

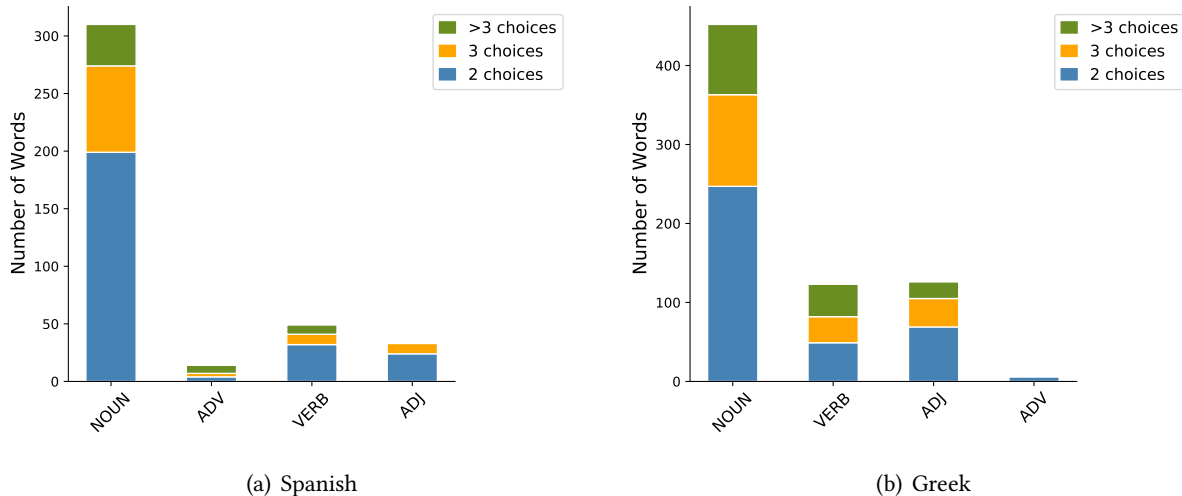


Figure 5.2: Distribution of the number of lexical choices for each POS tag.

choices corresponding to the same actual lemma (which were manually corrected for user studies). We can see that, along with the nouns that account for a major portion of the data, in other words, the classes also display  $\geq 2$  lexical choices. Prior efforts such as ContraWSD (Rios et al., 2018), SemEval tasks (Lefever and Hoste, 2013) have released datasets for cross-lingual lexical selection; however, they use manual word curation with the help of language experts, which only covers a small subset of nouns.

**Results** Table 5.2 shows the test accuracy averaged across all focus words for both Spanish and Greek. We first find that LinearSVM significantly outperforms both Frequency and DTree by a significant margin, indicating that it is both learning to perform lexical selection to a significant degree, and outperforming other reasonable alternatives for interpretable models. This gives us confidence in using it in our following human learning experiments. Interestingly, our interpretable LinearSVM model is within 97% relative accuracy of the skyline BERT model (just 2.09 percentage points behind). The fact that the more complicated but less inherently interpretable BERT model is overall better paves the way for future work to apply model interpretation techniques (Abnar and Zuidema, 2020, *inter alia*) to extract human-interpretable rules for lexical selection, although this is beyond the scope of the current paper.<sup>9</sup> We find that lexical selection accuracy varies by part of speech; all models perform poorly on adverbs with (avg.) gain of only +0.97 points over the baseline (c.f. with gains of +8.04 for nouns, +5.16 for verbs, +6.24 for adjectives).

### 5.3.2 Human Evaluation

Our preliminary automatic evaluation shows that our interpretable lexical model performs decently on cross-lingual lexical selection. Therefore, we conduct our main evaluation where we examine how effective are the extracted rules in helping human learners understand distinctions in L2 words.

<sup>9</sup>Overall accuracy is low, with even BERT getting 70%, possibly due to lack of sufficient source-side context. OpenSubtitles comprises of movie dialogues where the sufficient context could span more than a single sentence.

Lang.	Model	Test Accuracy				
		All	nouns	verbs	adj.	adv.
Spanish	Frequency (Baseline)	59.43	59.36	60.17	60.67	53.03
	DTree	62.40	62.45	61.57	65.22	54.82
	LinearSVM	<b>66.87</b>	<b>67.41</b>	<b>65.34</b>	<b>66.91</b>	<b>56.29</b>
	BERT	<u>70.72</u>	<u>71.75</u>	<u>69.04</u>	<u>67.31</u>	<u>54.07</u>
Greek	Baseline	58.56	59.48	53.04	60.48	61.82
	DTree	63.79	64.49	59.74	65.39	61.13
	LinearSVM	<b>66.46</b>	<b>67.09</b>	<b>63.30</b>	<b>67.51</b>	<b>64.98</b>
	BERT	<u>71.74</u>	<u>70.91</u>	<u>78.14</u>	<u>68.86</u>	<u>62.76</u>

Table 5.2: Among interpretable models, LinearSVM wins and is almost on par with a BERT, which is not that interpretable skyline.

Winning Streak: If you get 10 answers correct in a row for each label you are done with this word! Yay!

muralla/muro/muros : 1      pared/paredón : 0

Source sentence:

it is a magic rope with it we can scale the highest **walls**

Target words:

muralla/muro/muros

pared/paredón

How confident are you?

Figure 5.3: Learning interface used by Spanish learners. A learner is required to select the appropriate choice using the provided English context and mark how confident they are in their answer.

We take inspiration from existing research on second language acquisition (SLA) to design our evaluation. For example, Groot (2000) highlights the different learning strategies based on the generally accepted language acquisition theories (Nation, 2005; Richards et al., 1999), suggesting that the learner must go through different levels of language processing to effectively learn vocabulary. In particular, Groot (2000) empirically show that some of these levels can be accelerated with appropriate design of language tasks by combining strategies that use both examples in context and definitions.

We conduct an interactive learning exercise in the form of a cloze test where we present the human learner with the English focus word in context along with the set of possible L2 lexical choices. Our cloze-style tasks are essentially examples in context showing word usage in a given context, and the extracted rules are a proxy for human-provided definitions. The learner then selects the most appropriate L2 translation and mark how confident (“Not at all”, “Slightly”, “Somewhat”, “Quite” or “Very”) they are in their answer. After selecting the answer to each question, the correct answer is immediately told. For each focus word, we ask the learner to answer up to  $N$  multiple choice questions in sequence, which contain roughly the same number of questions for each lexical choice. We perform this study in two

**Task**

Target Word: **muralla/muro/muros**

If the source sentence contains the following:

Short phrases: ('climb', 'wall'), ('city', 'wall'), ('brick', 'wall'), ('jump', 'wall'), ('behind', 'wall'), ('outside', 'wall')

Words: break, climb, man, high, within, jericho, garden, jump, stone, city, outside, build

Concepts: 'city' as in a large and densely populated urban area; may include several independent administrative districts, 'will' as in determine by choice

---

Target Word: **pared/paredón**

If the source sentence contains the following:

Words: ear, hang, room, picture, face, write, stand, back, four, hand

Short phrases: ('face', 'wall'), ('hang', 'wall'), ('picture', 'wall'), ('back', 'wall'), ('right', 'wall'), ('write', 'wall'), ('stand', 'wall')

Concepts: 'wholly' as in to a complete degree or to the full or entire extent ('whole' is often used informally for 'wholly'), 'paint' as in apply paint to; coat with paint, 'room' as in space for movement

[Continue To Tasks](#)

Figure 5.4: Learning Interface. Descriptions of rules (extracted from the lexical selection model) are provided to the learner before the start of the exercise.

setups, the baseline setup without the rules and the use of the proposed system with rules.

**Baseline Setup** In the control setup, the learner has no access to any rules and immediately starts answering the questions. As mentioned above, the learner is shown the correct answer immediately after attempting the question. We expect learners to begin with a chance accuracy (50% for two choices), but as they are provided feedback, they may be able to grasp the patterns under which one particular translation or another is used and gradually rise above chance accuracy. The interface to answer the questions is shown in Figure 5.3.


**Proposed Setup** In the proposed setup, before starting the task, the learner is shown brief descriptions or “rules” as we will call them going forward, on when to use each possible lexical choice  $v_{y_k} \in \text{trans}(v_x, t_x)$ , constructed from the rule-set  $R_{\langle v_x, t_x, v_{y_k} \rangle}$ . They take as much time as they want to review these rules and then move to answering the questions (Figure 5.4). The interface to answer the questions is the same as the baseline (Figure 5.3). When selecting a choice, the learner is shown the correct answer accompanied by its corresponding human-readable rules of *only* the correct answer. Furthermore, we highlight the individual rules that helped decide the correct answer (Figure 5.5) for the convenience of the learner. By highlighting it in the two bottom panes, we hope to draw the learner’s attention to these hints and thus strengthen the understanding of the underlying concept. In this setting, we expect the learners to start with a non-chance accuracy and improve as they attempt more questions. The accuracy will likely further increase as they practice and become familiar with actual examples and how the extracted features apply to them.

## Experimental Settings

We select 7 Spanish learners and 9 Greek learners for the study.<sup>10</sup> Each learner is presented with the same set of words, half of which are to be annotated in the baseline setup and the other half in the

<sup>10</sup>We allow participants who know languages other than the target language or any other language that belongs to the language family to which Spanish or Greek belongs.

Winning Streak: If you get 10 answers correct in a row for each label you are done with this word! Yay!

muralla/muro/muros : 0      pared/paredón : 2 

Source sentence:

there's a stone **wall** runs clear around

Target words:

muralla/muro/muros

pared/paredón

How confident are you?

**Correct Answer: muralla/muro/muros**

Rules active in this example are highlighted:

**Words:** - break, fall, high, man, climb, within, jericho, jump, garden, city, **stone**, outside, build,

**Short phrases:** - ('city', 'wall'), ('brick', 'wall'), ('jump', 'wall'), ('behind', 'wall'), ('outside', 'wall'),

**Concepts:** - 'will' as in determine by choice, 'rampart' as in an embankment built around a space for defensive purposes,

Tokens with the respective rule highlighted:

there's a **stone** wall runs clear around

Continue

Figure 5.5: Learning Interface. Rules for the correct answer are displayed to the learner after each question. Individual rules that apply to the given example are highlighted for the convenience of the learner.

proposed setup. To ensure an unbiased setup, we randomize whether each focus word uses rules or not, while ensuring that at least half the annotators see the proposed setup and the other half perform the same task in the baseline setup for each word. We further shuffle the order in which the words are presented. For each English word, we select up to 40 examples each for the respective lexical choices. As an incentive, we end the exercise for a word early if the learner gets 10 correct answers in a row for each lexical choice. Below, we describe the selection criteria of the words presented to the learners.

**Word Selection** Ideally, we would like to conduct this study for all identified words in our automatic pipeline, however, this would require a large time commitment and cost. Instead, we shortlist a handful of words using the following automated procedure: First, for a given L2 study, we sort all focus words by the number of available data points ( $D_{\langle v_x, t_x \rangle}$ ). Next, from the trained lexical selection model  $\theta_{\langle v_x, t_x \rangle}$  we compute an F1-score for each lexical choice and filter focus words where the model gets an  $F1 > 0.5$  for each lexical choice. Finally, we select upto 10 focus words with the most data points that fit the above condition. For each word ( $\langle v_x, t_x \rangle$ ), we then select 40 *representative* examples for each lexical choice (see paragraph below). Details on the shortlisted words can be found in [Table 5.3](#).

**Representative Example Selection** Not all sentences in the parallel corpus contain sufficient context to select the appropriate lexical choice. For instance, the OpenSubtitles parallel corpus used in this study contains movie dialogues such as “this is the wall” which requires context spanning across multiple previous dialogues. Since in this study we extract features from context comprising of single sentences, in order to facilitate an effective learning process, we present examples to the learner that have the sufficient source-side context contained within a single sentence, required for correctly identifying the target-side lexical choice. To get such meaningful examples, we present bilingual English-Spanish and English-Greek speakers with the English sentence containing the focus word and the set of possible

Spanish		Greek	
(en) word	(es) lexical choices	(en) word	(el) lexical choices
wall.N	muralla/muro/muros: 33, pared/paredón: 60	bill.N	χαρτονόμισμα: 40, λογαριασμός: 40, νόμος/νομοσχέδιο: 40 (chartonómisma, logariasmós, nómos/nomoschédio)
farmer.N	agricultor: 29, granjero: 48	tour.N	δητεία:23, περιοδεία: 29, ξενάγηση: 33 (thitía, periodeía, xenágisi)
figure.N	cifra/cifras: 87, figura: 85	break.JJ	σπάω: 40, ράγομαι: 40, ξεσπάω: 40, διαρρηγνύω: 40 (spáō, rágoimai, ksespáō, diarrignýō)
vote.N	votemos/voto: 77, votación: 75	turn.JJ	στρίβω: 40, χαμηλώνω: 40, απορρίπτω: 40, καταδίδω: 40, σβήνω: 34 (strívo, chamilóno, aporrípto, katadído, svíno)
oil.N	aceite: 81, óleo/petróleo/petrolera/petrolero: 74	roof.N	ταράτσα: 40, οροφή: 40, στέγη: 39 (tarátsa, orofi, stégi)
wave.N	onda: 55, ola: 40, <i>oleado: 0</i>	wheel.N	τροχός: 40, ρόδα: 40, τιμόνι: 40 (trohós, róda, timóni)
pill.N	pastilla: 41, somnifero: 27, <i>pildora: 3</i>	old.JJ	αρχαίος: 40, κλασικ: 21, έτος: 40, ηλιωμένος: 40, παραδοσιακός: 36 (archaios, klasikos, etos, elikiomenos, paradosiakos)
language.N	idioma: 52, lenguaje: 68	turn.JJ	στρίβω: 40, χαμηλώνω: 40, απορρίπτω: 40, καταδίδω: 40, σβήνω: 34 (strívo, chamilóno, aporrípto, katadído, svíno)
ticket.N	multa: 24, boleto: 23, <i>pasaje: 0</i>	effect.N	παρενέργεια: 40, επίδραση: 40, εφέ: 40 (parenérgeia, epídrasi, efé)
<i>servant.N</i>	sirvienta/sirviente: 39, <i>servidor/servidora: 8, siervo/siervos: 10</i>	bone.N	μυελός: 40, οστό: 40, Μπούου: 40 (myelós, ostó, bone)

Table 5.3: Example tasks with their lexical choices selected for Spanish and Greek learning setup. Words/-choices marked in *red* are discarded from the language learning setup as they have  $\leq 10$  filtered examples from the representative example selection step.

lexical choices in Spanish and Greek, respectively. They then select the word which best suits the given context and mark their confidence in the selection. The interface for the selection of examples is the same as in Figure 5.3. We collect these annotations from multiple native speakers and only keep those sentences on which all native speakers agree. We enlist 3 Spanish native speakers who each annotate roughly 200 examples each for 10 English focus words. The inter-annotator agreement for Spanish, computed using Fleiss’ kappa is 0.77. For Greek, we use 2 native speakers to annotate 10 English words. For 7 out of 10 words we did not always have access to 2 native speakers so we relied on a single expert annotator. The (avg.) inter-annotator agreement for the remaining 3 words (*tour*, *tie*, *bill*) between the two annotators is 0.83. Of the 10 selected words, we discard words/lexical choices which have  $< 10$  examples on which all native speakers agree, giving us 9 English words for the Spanish study and 10 English for the Greek study.

## Results and Discussion

We report results centered around answering the following questions:

**Do rules result in increased learning accuracy and confidence?** We calculate the learning accuracy for all learners and tasks (words) for each language. If learners achieve higher accuracy with fewer attempted examples when having access to the rules, then the extracted rules could be considered effective in the learning process. However, in a human-based study we cannot directly use the accuracy as computed from comparison with the gold label. This is because there are other sources of variability such as (a) underlying learner ability, as some learners may be more proficient than others, (b) underlying task



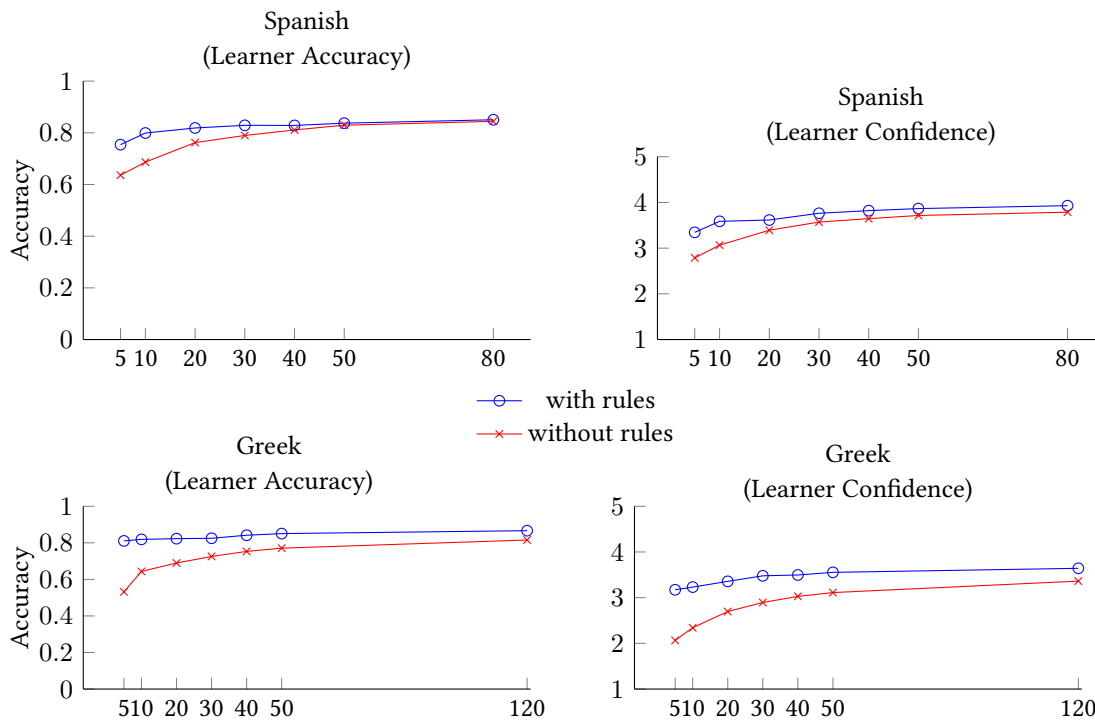


Figure 5.6: Learner accuracy and confidence in correct answers with and without access to rules against the number of attempted examples ( $x$ -axis). Learners achieve higher accuracy with increasing confidence with fewer examples when they have access to rules.

difficulty, as some words may be harder to disambiguate than others, or (c) word ordering, as learners may become proficient as they do more tasks. Therefore, we use a mixed-effects model (McLean et al., 1991), which models *random effects* and *fixed effects* to account for such random variability. Random effects are variables responsible for random variation such as task-identity, task-order and the learner, while fixed effects such as the presence of explanation are the variables of interest for determining the response variable, i.e. learner accuracy. A linear mixed-effect model (LME) is defined as follows:

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{u} + \epsilon \quad (5.5)$$

where  $\mathbf{y}$  is the learner accuracy,  $\beta$  and  $\mathbf{u}$  are the fixed-effect and random-effect regression coefficients with  $\mathbf{X}$  and  $\mathbf{Z}$  being the respective design matrices, and  $\epsilon$  denoting noise.

We fit LME models to our data by varying the number of attempted examples  $n = [5, 10, 20, 30, 40, 50, \text{all}]$ . Each fitted LME model gives us an intercept ( $\mathbf{Z}\mathbf{u} + \epsilon$ ) which informs us of the learner accuracy in the absence of explanations, and the fixed-effect coefficient  $\beta$  which informs us of the gain with explanation. As shown in Figure 5.6, it is clear that learners who have access to our automatically extracted rules achieve higher accuracy with fewer examples compared to without. As expected, with increasing number of attempted examples, the gap between the two settings reduces.

We fit similar LME models to estimate the effect of the presence of rules on learner confidence and find that the confidence in the correct answer increases more when rules are provided (Figure 5.6). This suggests that with our rules, learners require fewer examples to infer the patterns governing each lexical choice and get more confident in their understanding. This is encouraging as in true settings, the learning

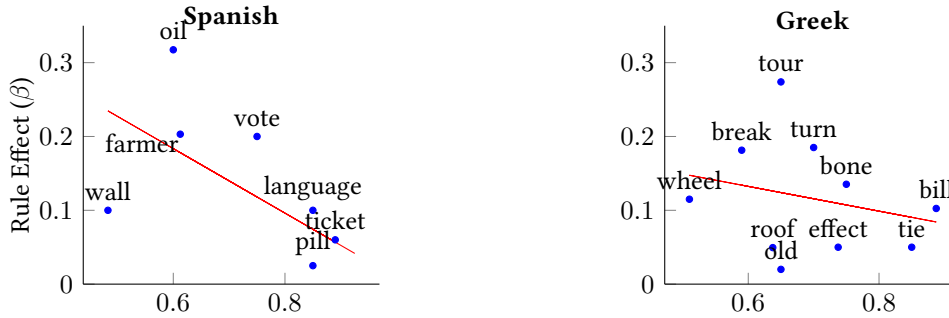


Figure 5.7: Rules help more for words where learners do worse. x-axis is the (avg.) learner accuracy (without rules) for first 20 examples.

exercise would be conducted for *every* focus word that the learner is attempting to learn, and because this process will have to be repeated many times, making it more efficient is of significant value. In Table 5.4 we report the  $p$ -value for the fitted LME models, which shows that the positive gains from the presence of rules are most significant for  $\leq 20$  examples for Spanish and for all examples for Greek.

Number	Fixed-effect coefficient ( $\beta$ )	Spanish p-value	Greek p-value
5	0.118	0.013**	$4.50e^{-09}$ ***
10	0.112	0.009***	$1.64e^{-07}$ ***
20	0.056	0.070*	$1.32e^{-06}$ ***
30	0.039	0.131	$4.23e^{-05}$ ***
40	0.017	0.462	$7.22e^{-05}$ ***
50	0.007	0.718	0.00015***
All	0.006	0.739	0.00173**

Table 5.4:  $p$ -value tests show that the fixed-effect of presence of rules for predicting learner accuracy is statistical significant up to first 20 attempted examples for Spanish and up to all examples for Greek. Significance codes: ‘\*\*\*’: 0.01, ‘\*\*’: 0.05, ‘\*’: 0.1.

Overall, we find that our extracted rules help Spanish and Greek learners in their learning process. We note that the results on Greek are promising as it does not enjoy the same luxuries as Spanish in having a high-quality lemmatizer or word aligner. This is encouraging especially for researchers involved in the revival efforts of endangered languages.

**Do the extracted rules help some words *more* over others?** Since the words vary on their difficult levels, we check if our extracted rules are more effective for some words over others. So, we fit a LME model on each focus word and compute the  $\beta$  coefficient to measure the effect of rules on learner accuracy after 20 attempted examples.<sup>11</sup> We plot the  $\beta$  coefficient with the accuracy (averaged across all learners) for each focus word when they did not have access to the rules in Figure 5.7 and find that words on which the learners performed the worst such as *wall*, *oil*, *farmer*, and *vote*, benefit most by our explanations. Some of these words, in fact, indeed have finer semantic subdivisions than the rest. For instance, the choices for *farmer*: *agricultor* refers exclusively to the one who works the land, harvests, sows, etc.,

<sup>11</sup>Because analysis revealed that rules are more effective earlier in the learning process.

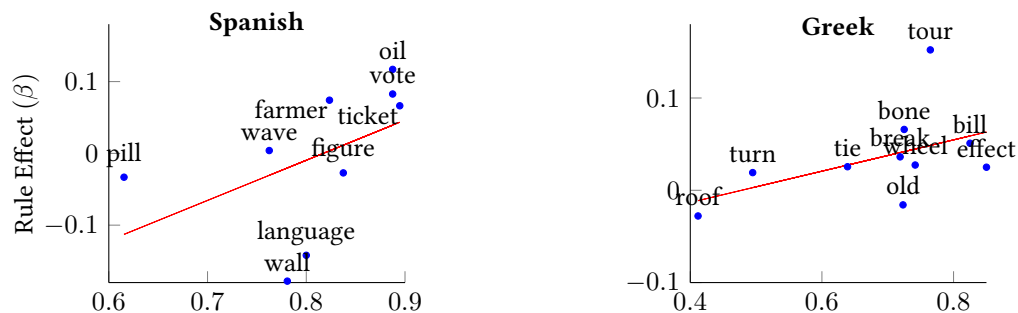


Figure 5.8: Rules help more for words where model performs well. x-axis is model accuracy per word.

while *granjero* is less formal, referring to the one who manages a farm or works or lives on it. Similar observations are seen for Greek where learners are benefited more for words (*break*, *wheel*, *tour*, *old*, *roof*) on which they performed the worst. Some of these words, in fact, indeed have finer semantic subdivisions than the rest. This analysis shows that, encouragingly, our explanations are especially helping learners with more difficult words. We also plot the  $\beta$  coefficient with the lexical model accuracy (Figure 5.8) and find a positive correlation, meaning that explanations help more for words where the model performs well. This suggests that if we can develop more accurate models with an equal level of interpretability, the learning effect might become even stronger.

## 5.4 Other Applications

In addition to helping with the language learning process, these semantic subdivisions have also been used to evaluate machine translation (MT) models. Because the semantic subdivisions we extract exhibit fine-grained distinctions, Yin et al. (2021) use them to evaluate whether the contextual MT models are indeed leveraging the required context for translation. In particular, they focus on English-French translation, where semantic subdivisions are first identified using the procedure outlined in section 5.2. A human study is then conducted to identify which context words (in both the source and target languages) are useful for disambiguating these semantic divisions. For a given focus word in English, human translators are required to select the correct lexical choice and additionally mark which words in the source and target context helped them in their decision. Next, the MT model’s attention is compared with the human-attention to evaluate how well the attention distribution of the model is similar to the human translations. This reveals that the overall model’s source-side attention is similar to human attention, but has poor alignment with the target-side context. Therefore, Yin et al. (2021) uses human attention to better guide the model’s attention by regularizing the model attention to human attention. We direct the reader to the respective paper for more details.

## 5.5 Conclusion

Through automated and human learning experiments, we demonstrated the effectiveness of our proposed approach in aiding the L2 learning process. In this process, we also collect sentence pairs automatically that show fine-grained semantic subdivisions and could potentially be used for evaluating cross-lingual word sense disambiguation models.



## Chapter 6

# ASSIST-A-TEACHER: Teacher Perception of Automatically Extracted Grammar Concepts for Language Learning

Up until now, we saw how to adapt the AUTOLEX framework for extracting language descriptions for individual questions about a language’s morpho-syntax (word order, agreement, case marking) and vocabulary (L2 semantic subdivisions), which were evaluated based on their intrinsic properties. In this chapter, we combine all these approaches to extract teaching material for two Indian languages, Marathi and Kannada. In addition to the linguistic questions covered in [Chapter 3](#), [Chapter 4](#) and [Chapter 5](#), we apply AUTOLEX to also explain morphology inflection, specifically suffix usage. To evaluate how usable the extracted material is, we conduct a user study with in-service teachers who teach these languages in North America.

Aditi Chaudhary, Arun Sampath, Ashwin Sheshadri, Antonios Anastasopoulos, Graham Neubig. 2022. [Teacher Perception of Automatically Extracted Grammar Concepts for L2 Language Learning](#) . On *arxiv*.

### 6.1 Overview

Similar to how grammar descriptions form a crucial component of language documentation, teachers often summarize the different systems of syntax, semantics, and phonology in a meaningful way for the consumption of language learners. Creating good quality pedagogical resources that explain and illustrate these complex concepts is the key to effective learning. As we also discussed in [Chapter 5](#), computer-assisted language learning (CALL) systems have broadened the outreach of language education by enabling both self-learning (e.g. Duolingo) and online-learning (e.g. Rosetta Stone which also provided online instruction), especially in the COVID-19 pandemic when in-person instruction was not possible, leading to the need for user-friendly and easily accessible applications for teachers and learners ([Li and Lalani, 2020](#)). Even for people learning in traditional classroom settings, the use of CALL systems in tandem has shown to be effective in the learning process ([Macaruso and Rodman, 2009](#); [Barrow et al., 2009](#)) And as we discussed in [Chapter 5](#), indigenous language communities are increasingly taking sim-

ilar initiatives to create learning resources as part of their language preservation efforts (Moline, 2020). Typically, these materials must be carefully designed and customized to address the needs of the community; for example, Network (2001) note that language educators must make effective use of locally relevant expertise and materials when designing the curriculum, as they embody the cultural heritage of the region. Because these materials are curated manually by subject experts, this, however, makes the curriculum design process a challenging and time-consuming process, especially for languages where subject experts or relevant resources are not easily accessible.

A good curriculum design requires significant human time and effort, as this process entails many steps, from designing material for different learning levels, covering different vocabulary and grammar aspects, finding relevant examples for each grammar concept, and even creating exercises for evaluating learners, to name a few. Additionally, for second language (L2) learning, it is not straightforward to reuse an existing curriculum even in the same language, as the background and requirements of L2 learners could be vastly different from the traditional L1 setting (Munby, 1981). With higher processing power and speed, modern corpus-based methods or *corpus technology* (Yoon, 2005) can analyze large text corpora in seconds and find language patterns that can accelerate some of these steps. Corpora technology has been widely advocated for language teaching (Bennett, 2010; Flowerdew, 2011; Farr, 2010; Reppen, 2010; Davies, 2008), as it exposes learners to “real” language usage, which is important for integrating learners with the community. Because such corpora comprise “natural text” i.e. text collected in natural settings with minimal experimental interference, corpus-based methods have been widely used for language learning; for example, they have helped students learn vocabulary (Ackerley, 2017; Lee and Liou, 2003), collocations (Chan and Liou, 2005; Du et al., 2022; Kanglong and Afzaal, 2020), grammar (Lin and Lee, 2015), L2 writing (Yoon and Jo, 2014; Crosthwaite, 2020). Research has shown the effectiveness of incorporating corpus-based methods or data-driven learning (DDL; John (1991)) in language teaching: Boontam and Phoocharoensil (2016) taught English prepositions to Thai learners using concordance lines which are “lists of all contexts in which the given word occurs in a particular text” (Lindquist, 2018). Similar positive results are also shown by Celik and Elkatmis (2013) in teaching the usage of Turkish punctuation. Mukherjee (2004) further revealed that few language instructors are aware of corpus technology, but after conducting a workshop demonstrating the utility of such technology, these instructors realized its potential in teaching. Language instructors can use corpus-derived materials (Bennett, 2010; Leńko-Szymańska, 2017; Ma et al., 2022) to present pre-prepared concordance lines to learners or provide direct access to learners to explore themselves (Chambers, 2010) or to mainly search common word patterns, keywords, in popular corpora (e.g. Davies (2008), Cobb (2002), SketchEngine, Skell) to supplement their teaching. But most of these corpus-based methods use the text corpora in the form of ‘Key-Word-In-Context’ (KWIC) concordance (O’keeffe et al., 2007), where the contexts with the searched ‘keyword’ are displayed, with the hope that it can enhance the learner’s lexical and grammar knowledge from the relevant context. Now, with advances in natural language processing (NLP) methods, we can extract instructional material for more complex linguistic use cases (e.g. word order), as shown in previous chapters.

Inspired by our findings in previous chapters, we use AUTOLEX to aid in language instruction by automatically extracting learning material for “teachable grammar points” covering different aspects of grammar, directly from the text corpora of the language of interest. We define teachable grammar points as individual syntactic or semantic concepts that can be taught to a learner. For instance, with respect

Grammar Aspect	Teachable Grammar Points
General Information	What Gender types are in Marathi? (e.g. masculine, feminine, neuter) Which type of words show Gender? (e.g. nouns, verbs) What are some example words and how are they used in real-world?
Vocabulary	What words to use for popular categories ( e.g. food, animals, etc.) What are some adjectives, their synonyms and antonyms? Which word to use when?
Word Order	Are subjects before or after verbs in Marathi? If both, when is subject before and when is it after the verb?
Suffix Usage	What are the common suffixes for Marathi nouns? When should a particular suffix (e.g. -‘laa’) be used?
Agreement	Do some words need to agree on Gender with each other? If so, when should they necessarily agree and when they need not?

Table 6.1: Example teachable grammar points covered in our language material.

to the grammatical aspect of *word order*, a “teachable grammar point” could be to understand “how adjectives are positioned with respect to nouns in this language”. In addition to syntactic points, we also include concepts covering lexical semantics in our materials, as vocabulary forms a crucial component of language learning. This learning material comprises human-readable explanations of the different linguistic behaviors for each concept (e.g. “most objects occur after the verbs except for interrogatives), along with illustrative examples. To our knowledge, the use of such linguistic insights has not yet been investigated for L2 language teaching. In particular, we test this framework for teaching the two Indian languages of Kannada and Marathi, to English speakers who reside outside India. We particularly select these languages for our study as these languages fulfilled certain desiderata i.e. i) these languages have far fewer pedagogical resources as well as NLP resources than English making them under-resourced and, ii) access to in-service Kannada and Marathi teachers, allowing us to evaluate first-hand how in-service instructors find our *automatically* extracted language learning material useful for their teaching process. Specifically, we aim to answer the following research questions:

- How can we most effectively extract “teachable grammar points” and the corresponding learning material?
- How many of these extracted grammar points are relevant to the language-learning curriculum?
- How many of these extracted grammar points are practically usable for language educators to further develop or improve their existing curriculum and if so, in what ways?

## 6.2 Proposed Approach

### 6.2.1 Why Marathi and Kannada?

To investigate the utility of automatically derived corpus-based language material in teaching, we work with in-service teachers of both Marathi and Kannada, specifically those involved in teaching these languages outside of India. Although Marathi and Kannada are spoken primarily in India, a small but significant populace has emigrated outside of India for personal and/or professional reasons. Therefore, the primary objective of these teachers is to provide instruction for spoken and written forms of the language to a) preserve and promote the language and culture, and b) help non-native speakers to communicate with their elders and community. Because of these specific objectives, existing Kannada or Marathi textbooks from Indian schools cannot be used as is, as they are based on more *traditional L1 teaching approach* (Selvi and Shehadeh, 2018), where the language is taught from the ground up, from introducing the alphabet, to its pronunciation, to other subsequent vocabulary and grammar aspects. Teachers have instead adapted the existing material and continue to design new material to suit their requirements. Therefore, this setting provides a good test bed to evaluate our extracted learning materials, as our primary objective is not to replace teachers but rather to assist them in their teaching process. By providing first-pass learning material, teachers could use it as is or supplement it with their existing material. Additionally, in comparison to English, these languages have far fewer CALL systems or resources that are freely or easily available. For example, the survey results (subsection 6.4.2) revealed that currently teachers refer to online resources like YouTube videos for reference materials, or some online dictionaries (e.g. Shabdakosh<sup>1</sup>), while for languages like English there are a plethora of online resources and tools (e.g. Rosetta Stone (Stone, 2010), Duolingo<sup>2</sup>, Cambridge learning<sup>3</sup>, ESL<sup>4</sup>, etc.), but for most of the world’s 7000+ languages, it is a struggle to find even a sufficiently large and good quality text corpus (Kreutzer et al., 2021), let alone teaching material. Currently, both Kannada and Marathi are not part of any popular learning applications (e.g. Duolingo or Rosetta Stone). For Marathi, there is an online learning tool Barakhadi<sup>5</sup>, however it is not free of cost. Therefore, these languages are under-resourced with respect to such online resources, and will likely benefit from this exercise.

### 6.2.2 Teachable Grammar Points

Although, language education has been widely studied in literature, there is no one ‘right’ method of teaching a language. Several teaching methods have been proposed and implemented, and we take inspiration from these existing methods to design the content of the materials. For example, Doggett (1986) discuss eight popular teaching methods, some of which, such as the *Grammar-Translation* method, require learners to translate grammar rules between their L1 and L2 languages, while methods such as *Direct Method*, *Suggestopedia*, *Community Language Learning* encourage learning in the L2 language itself. In *Communicative Approach*, learning through functions (e.g. self-introduction, identification of relationships, things, etc.) over grammar forms is given more importance. Jeyasala (2014) also note

---

<sup>1</sup><https://www.shabdkosh.com/>

<sup>2</sup><https://www.duolingo.com/>

<sup>3</sup><https://www.cambridgeenglish.org/learning-english/>

<sup>4</sup><http://a4esl.org/>

<sup>5</sup><https://barakhadi.com/>



that exposing learners to more in-language input in different communicative contexts will improve L2 communication. We build our materials around these principles.

To extract learning material, we choose AUTOLEX because it addresses the requirements quite well, that is, a) to discover the salient language patterns from authentic natural text in the required language and b) to provide means to present and explain the extracted patterns in a human-understandable format. Importantly, for each pattern, illustrative examples and examples of exceptions are extracted from the underlying corpora. We apply this framework to Kannada and Marathi and extend it to extract learning material for different grammar concepts. The first step in applying AUTOLEX is to identify a large text corpus in language of interest. Next, we make a list of “teachable grammar points”, which, as we defined earlier, are individual points that can be taught to a learner and are typically included in a curriculum. AUTOLEX already covers aspects of word order and agreement. In addition to those, we include more grammar points based on the material shared by the Kannada experts. We inspected three out of the eight Kannada textbooks shared by the experts and identified common grammar points such as identification of syntax categories (e.g. nouns, verbs, etc.), vocabulary, and suffix usage. In [Table 6.1](#), we show examples of grammar points that we ask. Next, we formulate each linguistic question into an NLP classification task and then construct training data from the underlying corpus. Finally, from the learned model, we extract concise explanations. We briefly describe the procedure for each concept and the motivation for choosing it.

**Word Order and Agreement** Both Marathi and Kannada predominantly follow an SOV word order, i.e. subject-object-verb, but because syntactic roles are often expressed through morphology rather than word order alone, there are often significant deviations from this dominant order. Because of richer morphology, these languages are highly inflected for gender, person, number, and morphological agreement between words is also frequently observed. Therefore, L2 learners must understand both the rules of word order and agreement to produce grammatically correct language. We follow the same problem formulation as AUTOLEX for the word order and agreement questions, as described in detail in [Chapter 4](#).

**Suffix Usage** Along with understanding sentence structure, it is equally important to understand how inflection works at word level, given that these languages are highly inflected. We first identify the common suffixes for each word type (e.g. nouns) and then ask ‘which suffix to use when’.<sup>6</sup> Similarly to word order and agreement, we identify the POS tags and morphological analysis for each word in a sentence. To identify the suffix, we then train a model that takes as input a word with its morphological analysis (e.g. ‘deshaala,N,Acc,Masc,Sing’) and outputs the decomposition (e.g. ‘desh + laa’). Next, a classification model is trained for each such suffix (e.g. ‘-laa’) to extract the conditions under which one suffix is typically used over the other. An example of Marathi suffixes was shown earlier in [Chapter 2](#) ([Figure 2.7](#)).

**Vocabulary** Vocabulary is probably one of the most important components of language learning ([Nation, 2021](#)). There are several debates on which is the best strategy to teach vocabulary; specifically, we organize vocabulary material around three questions, as shown in [Table 6.1](#). To extract vocabulary which shows fine-grained distinctions, along with explanations on when to use one word over the other, we

---

<sup>6</sup>We focus only on suffixes as typically both these languages show inflections via suffixes.

Vocabulary covering **nouns (n)**, **verbs (v)**, **adjectives (a)**, **adverbs (r)**

Search for a word (e.g. rice)	
Type	
food (n)	chocolate -- चॉकलेट (chocolate), <a href="#">Examples</a> pepper -- मीठ (meeth), <a href="#">Examples</a> sugar -- साखर (saakhar), <a href="#">Examples</a> fodder -- चारा (chaara), <a href="#">Examples</a> food -- अन्न (ann), <a href="#">Examples</a> rice -- तांदूळ (tandul), <a href="#">Examples</a> nutrient -- अस (as), <a href="#">Examples</a> liquor -- दारू (daaru), <a href="#">Examples</a> stock -- शेअर (share), <a href="#">Examples</a> flour -- पीठ (peeth), <a href="#">Examples</a> lemon -- लिंब (limb), <a href="#">Examples</a> chop -- चिर (chir), <a href="#">Examples</a> produce -- निर्मिती (nirmiti), <a href="#">Examples</a> vegetable -- भाजी (bhaaji), <a href="#">Examples</a>
relationships (n)	sibling -- आहे (aahe), <a href="#">Examples</a> brother -- भाऊ (bhaau), <a href="#">Examples</a> dad -- बाबा (baba), <a href="#">Examples</a>

Figure 6.1: Marathi words organized by basic categories. Each word contains a link to illustrative examples with their English translations.

follow the procedure described in [Chapter 5](#) and use parallel data between English and L2 language as a starting point. Similarly to AUTOLEX, explanations are then extracted to understand the L2 usage. As we saw before, *Communicative Approach* ([Johnson and Brumfit, 1979](#)) focuses on teaching through functions over grammar forms, therefore, we also organize vocabulary around popular semantic categories (e.g. words for food, relationships, etc.). We run a word-sense disambiguation (WSD) model ([Pasini et al., 2021](#)) on English sentences, which helps us to identify the word sense for each word in context (e.g. the word sense ‘bank.n.02’ refers to a financial institution while ‘bank.n.01’ refers to a river edge). Since the word senses are hierarchical in nature, we can traverse the ancestors of each word sense to find whether it belongs to any of the pre-defined categories (e.g. food items, relationships, animals, fruits, colors, time, action verbs, body parts, vehicle, elements, furniture, clothing). An example of such words extracted for Marathi is shown in [Figure 6.1](#). In addition to basic words, we also identify popular adjectives, their synonyms and antonyms, and present them in a similar format to the users, as shown in [Figure 6.2](#). To identify adjectives, we use the POS tags (‘ADJ’) that were automatically extracted from the complete syntactic analysis of the underlying corpus. Definitions, synonyms, and antonyms are automatically identified first in English using the WordNet ([Miller, 1995](#)) resource, and the respective L2 translations are obtained from word alignments. For each word, we also present the accompanying examples that illustrate its usage in context, along with its English translations. For the benefit of users who are not familiar with the script of the L2 languages, we automatically transliterate into Roman script using [Bhat et al. \(2015\)](#).<sup>7</sup>

<sup>7</sup><https://github.com/libindic/indic-trans>

English Word	Definition	Marathi Word	Synonyms	Antonyms
first	the first or highest in an ordering or series	पहिला (pahila), <a href="#">Examples</a>		
social	a party of people assembled to promote sociability and communal activity	सोशल (soshal), <a href="#">Examples</a>		
new	not of long duration; having just (or relatively recently) come into being or been made or acquired or discovered	नवा (nava), <a href="#">Examples</a>	ताजा (taaja), <a href="#">Examples</a>	
important	of great significance or value	महत्त्व (mahatva), <a href="#">Examples</a>	मोठा (mothaa), <a href="#">Examples</a>	
private	an enlisted man of the lowest rank in the Army or Marines	खासगी (khaasgi), <a href="#">Examples</a>		सार्वजनिक (saarvajnik), <a href="#">Examples</a>

Figure 6.2: Marathi adjectives extracted by AUTOLEX.

**General Information** In addition to these specific morpho-syntax and semantic patterns, we also present salient morphology properties at the language level. Specifically, from the syntactically parsed corpus of the target language, we answer basic questions such as ‘what morphological properties (e.g. gender, person, number, tense, case) does this language have’, ‘for a given property (e.g. gender) what are the types of values (e.g. masculine, feminine, neuter) and which words typically show which value’ and so on. These questions were inspired from Kannada textbooks shared by experts, where the textbooks introduce a learner to basic syntax and morphology such as identifying action verbs, adjectives, pronoun types across gender, person, number, and so on. For each question, we organize the information by frequency, as frequency acts as a proxy for popularity, for example, textbooks for language teaching comprise of common and frequently used examples (Dash, 2008).

In addition to content, the format in which the material is presented is equally important. Smith Jr (1981) outline four fundamental steps involved in language teaching: *presentation* of material to learners, *explanation* of material, *repetition* of material until it is learned, and *transfer* of materials in different contexts, together called PERT. They further mention that there is no fixed order of these steps. For example, some teachers prefer presentation of content (e.g. reading material, examples in context etc) first followed by explanation (e.g. grammar rules), while Smith Jr (1981) argue that for above-average learners, explanation followed by presentation works better. In AUTOLEX, we provide both (i.e. rules and examples) without any specific ordering, with the purpose that educators can decide based on their experience and objectives. By providing illustrative examples from the underlying text at each step, we hope to address the *transfer* step, where learners are exposed to real situations of language use.

### 6.2.3 Evaluating Learning Materials

Before presenting the content to volunteer teachers, we first conduct a limited study to evaluate the sanity of the material presented.

**Quality Study** In Chapter 3 and Chapter 4, we had conducted a quality study with language experts in multiple languages (English, Greek, Russian, Catalan) that revealed that the rules extracted are decent.<sup>8</sup> Therefore, to ensure that we are also achieving a minimal level of quality in this work, we conducted

<sup>8</sup>80% rules extracted for agreement, word order, and case marking were deemed valid for English and Greek, for Russian and Catalan only agreement was evaluated and were deemed 78% and 66% valid respectively.

Type	Question	Answer Choices
Relevance	1. What percentage of the materials presented in the tool cover existing curriculum requirements?	0-100%
Utility	2. How likely are you personally inclined to use this tool in your lesson planning or teaching?  2.1. If likely, for what purpose do you foresee this being used? (multiple answers can be selected):  2.2. If likely, what aspects would you use: (multiple answers can be selected):  2.3. if NOT likely, why? (multiple answers can be selected):	3: Highly likely 2: Likely 1: Not likely  a. For lesson preparation, knowledge b. For evaluating students c. Present this to the students for self-exploration d. Other (please specify the reason)  a. The general concept introduced by the material b. The rules which are described in the table c. Illustrative examples that accompany the rule d. Other (please specify the reason)  a. material outside the scope of current curriculum b. material was unclear and needs improvement c. material is already covered by existing curriculum d. Other (please specify the reason)
Presentation	3. How did you find the tool?	3. Very easy to use and navigate 2. Somewhat easy to use, but took some time to get used to 1. Difficult to use
Feedback	4.1 What did you like about the tool or the learning materials? 4.2 What did you not like about the tool? 4.3 What would you like to improve in the tool?	

Table 6.2: Perception study: Questions posed to the in-service teachers for evaluating the learning materials on relevance, utility and presentation. This set of questions is asked for each grammar concept.

a similar study only for some Kannada materials with two experts. We ask the experts to evaluate the materials for word order, word usage and suffix usage. Specifically, for word order and suffix usage, we ask two questions namely 1) whether the rules along with accompanying examples demonstrate the shown concept correctly, and 2) if so, whether this material is already covered in their existing material. For word usage, we present them with the extract word pairs between English and Kannada and ask them how many of the extracted word pairs are correct. In [Chapter 5](#), we have already evaluated the efficacy of the rules extracted in a true learning setup for Greek and Spanish, and therefore we focus only on the evaluation of word pairs.

**Perception Study** To check whether the materials are practically usable and, if so, in what aspects, we conduct a broader set of teachers' *perception* of the presented materials, with Kannada and Marathi teachers. Through this study, specifically, we hope to understand 1) *relevance* of the curriculum materials, 2) *utility* of the teaching materials, and 3) *presentation* of the materials, with the help of in-service teachers. This study was conducted in three parts; this involved a 30–60 minute introductory meeting with teachers, in which we introduced the tool, all types of grammar concepts covered by the tool, and how to navigate the online interface. This introductory meeting was held via video call due to pandemic

restrictions. In the next part of the study, teachers were given one week to explore the materials. Finally, teachers received a questionnaire that required them to assess the relevance, utility, and presentation of the tool. We ask this set of questions for each grammar concept (i.e. general information, vocabulary, suffix usage, word order, and agreement), as shown in Table 6.2. In addition to evaluating the materials, we also ask for their general feedback on the materials, which is more open-ended.

### 6.3 Experimental Setting

In this section, we describe the details of the data and models used to extract the learning materials.

**Data** Since our goal is to create teaching material for learners, most of whom are based outside India and have English as L1, we use the parallel corpus of Kannada-English and Marathi-English from SAMANANTAR dataset (Ramesh et al., 2022). This consists of 4 million Kannada sentences and 4 million Marathi sentences with their respective English translations, and covers text from a variety of domains such as news, Wikipedia, talks, religious text, movies. Of these genres the underlying corpus has a particularly large amount of newspapers and legal proceedings, and thus consists of more formal and traditional language than typically appears in textbooks.

**Model** As mentioned in subsection 6.2.2, the first step in extracting materials for the different grammar concepts is to parse sentences for POS tags, morphological analysis, and dependency parsing. To obtain this analysis for our corpus, we use an automatic parser UDIFY (Kondratyuk and Straka, 2019). To train a parser for Marathi and Kannada, we used the training data collected by IIIT-Hyderabad<sup>9</sup>, which is annotated in the Paninian Grammar Framework (Bhat et al., 2017). However, the UDIFY model requires training data in the Universal Dependencies annotation scheme (McDonald et al., 2013), so we followed Tandon et al. (2016) to convert between the two formats to obtain POS tags, lemmatization, and morphological analysis. Another challenge in using this converted data is that it did not have dependency information. To obtain dependency data, we first train the UDIFY model in a related language (Hindi) and apply it directly to the converted data above, giving us dependency parses for Marathi and Kannada.<sup>10</sup> We then train a new model on this converted data and augment it with the Hindi training data as well, and apply the resulting model on the 4 million Marathi and Kannada raw sentences. The performance of the resulting parser is seen in Table 6.3. Similar to previous chapters, we use the SUD annotation format to represent the syntactic information and follow the same modeling setup as Chapter 4 and Chapter 5 to extract the patterns, explanations and accompanying examples. For suffix usage, we additionally train a morphology decomposition model which break a word into its lemma and affixes, for which we use the model from Ruzsics et al. (2021).

**Participants** For Kannada, we work with teachers from the Kannada Academy<sup>11</sup> (KA), which is one of the largest organizations of free Kannada teaching schools in the world, with more than 70 learning

---

<sup>9</sup><https://ltrc.iiit.ac.in/showfile.php?filename=downloads/kolhi/>

<sup>10</sup>The data is publicly released <https://github.com/Aditi138/auto-lex-learn/tree/master/data>

<sup>11</sup><https://www.kannadaacademy.com/>

Language	POS	Morphological Analysis	Lemmatization	Dependency Parse (UAS)
Marathi (PAN)	85.9	70.1	82.2	-
Marathi (UD)	63.3	22.1	50.5	60.4
Kannada (PAN)	90.3	79.3	90.6	-

Table 6.3: Parser performance on the respective test sets. PAN refers to the Paninian treebanks and UD refers to the test set from the UD treebank, which is available only for Marathi.

centers in the United States, Europe, Australia and Asia. The initial quality study of the Kannada learning material was carried out by two teachers who are on the academy board. We chose these teachers, whom we also refer to as Kannada experts, for the quality study, as they are actively involved in training other teachers of the academy and designing the lesson material. To answer the primary research questions about the teachers’ perception of the materials, we conduct a wider-range study by recruiting volunteer teachers. KA has 800 volunteer teachers, of which 12 participated in this study. For Marathi, there is no one central organization as for Kannada, rather there are many independent schools in the North America. We reached out to two such schools, namely, the Marathi Vidyalay<sup>12</sup>, a school in New Jersey, USA, which was established 40 years ago and teaches Marathi to learners in the age group of 6-15, and Marathi Shala in Pittsburgh, USA<sup>13</sup>. Marathi Vidyalay is a small school consisting of 7 volunteer teachers, of whom 4 agreed to participate in the study, while the Marathi Shala only has one teacher. All of the participants are volunteer teachers; i.e. teaching is not their primary profession, rather they perform teaching as a volunteer service.

## 6.4 Results

In this section, we present the results of the quality and perception study. In addition to human evaluation, in Chapter 4 we outlined a strategy to automatically evaluate the quality of extracted materials. Specifically, the learnt model is applied on a held-out set of sentences and the accuracy metric on that held-out set is compared with a baseline. We apply the same evaluation protocol for word order, suffix usage, agreement, and vocabulary, and report the results in Table 6.4. We can see that in most cases (except for Marathi agreement), the rules extracted by the model outperform the respective baselines, which suggests that the model is able to extract decent first-pass rules with 98% prediction accuracy for Kannada word order, 48% for agreement, 85% for suffix usage, 68% for vocabulary, 98% for Marathi word order, 61% for agreement, 85% for suffix usage and 70% for vocabulary. For Marathi, the model for agreement does not outperform the baseline because the overwhelming majority of head-dependent pairs show agreement for gender, which causes the statistical threshold, used for deciding the ground truth label (subsection 3.3.2), to not be exceeded.

<sup>12</sup><https://marathivishwa.org/marathi-shala/>

<sup>13</sup><https://www.mmpgh.org/MarathiShala.shtml>

Grammar Concept	Type	Kannada		Marathi	
		AUTOLEX	baseline	AUTOLEX	baseline
Word Order	subject-verb	<b>97.02</b>	96.97	<b>97.8</b>	97.7
	object-verb	<b>99.11</b>	99.06	<b>97.89</b>	96.78
	numeral-noun	<b>98.63</b>	98.36	99.54	99.54
	adjective-noun	99.92	99.92	-	-
	noun-adposition	99.14	99.14	-	-
Agreement	Gender	<b>71.87</b>	65.69	61.11*	<b>81.44*</b>
	Person	24.73	<b>25.16</b>	-	-
Suffix Usage	NST	<b>91.58</b>	50	<b>90.44</b>	93.77
	NUM	<b>85.2</b>	82.63	85.91	<b>93.61</b>
	NOUN	<b>78.61</b>	39	<b>70.23</b>	67.8
	PRON	<b>87.13</b>	58.03	<b>75.66</b>	65.07
	PART	<b>94.73</b>	89.35	<b>90.58</b>	76.77
	ADJ	<b>87.74</b>	66.82	<b>87.55</b>	83.83
	VERB	<b>63.19</b>	30.52	<b>78.44</b>	65.87
	PROPN	<b>74.57</b>	46.68	65.6	<b>71.19</b>
	SCONJ	<b>96.85</b>	64.6	<b>97.59</b>	86.9
	DET	<b>99.53</b>	61.83	<b>83.91</b>	81.71
	AUX	<b>76.92</b>	38.46	<b>92.8</b>	81.57
	ADV	<b>75.19</b>	37.27	<b>86.84</b>	65.89
	ADP	<b>93.55</b>	76.43	<b>97.12</b>	67.63
Vocabulary	Semantic Subdivisions	<b>68.68</b>	58.48	<b>70.58</b>	56.26

Table 6.4: Automated evaluation results for learning materials extracted for each grammar concept. \* denotes that the model was learnt on a subset of data (200k sentences) because of computational issues.

### 6.4.1 Quality Study Results

We first conduct a limited study in Kannada for a sanity check, asking them to evaluate the materials for the order of subject-verb and object-verb words, the usage of noun and verb suffixes and the vocabulary words under word usage. We present the results of that study below:

**Vocabulary** A total of 385 semantic subdivisions were identified for Kannada, of which we presented both experts with 100 word pairs for evaluation.<sup>14</sup> These translations were extracted so that they show fine-grained semantic differences in their usage. In general, both experts found 80% of the word pairs to be valid, that is, the set of translations for a given English word showed interesting and different usages. For example, for ‘doctor’, the model discovered four unique translations, namely ‘vaidya, vaidyaro, daktor, vaidyaru’ in Kannada which the expert found interesting for teaching as they demonstrated fine-grained distinctions, both semantically and syntactically. For instance, ‘vadiya’ is the direct translation

<sup>14</sup>A total of 285 such word pairs for extracted for Marathi.

of ‘doctor’, whereas ‘daktor’ is the English word used as-is, ‘vaidyaro’ is the plural form of doctors and ‘vaidayaru’ is also a formal way of saying a doctor. Regarding usability, the experts mentioned that currently in their curriculum there is no good way of handling synonyms or such fine-grained distinctions, specifically, they said –

*“Given that these word pairs have been extracted from natural text, its interesting to see that there are certain word senses which are so frequently used in the real world which currently we haven’t covered in our lesson but are we are now thinking of adding them. For example, words like ‘igaagale’ which means ‘already’, is such a simple word that we have missed and should have been added. Often teachers struggle to come up with different examples to illustrate word usages, so the accompanying examples you presented are extremely useful.”*

**Word Order** We follow the same evaluation protocol as outlined in [subsection 4.3.2 \(Figure 4.5\)](#). For subject-verb, 6 grammar rules explaining the different word order patterns were extracted (4 explaining when the subject can occur both before and after the verb, 2 rules informing when subjects occur after the verb, and 1 showing the default order before). Of the six rules presented, experts found three valid linguistic patterns, of which 2 were too specific and 1 precisely captured the distinction. For object-verb, the model also identified six rules (4 explaining the flexible word order patterns and 2 showing the non-dominant pattern that objects come after the verb). Of these 6 rules, 2 rules precisely captured the word order patterns and 1 rule was too specific. Interestingly, all these rules which were deemed valid were the ones which showed non-dominant patterns. The experts also note that this existing material for subject-verb is not covered in any textbook they use, as their school’s primary focus is on beginner learners, but they did mention that this is suitable for textbooks for advanced learners, specifically the textbook-3 and 4 in their curriculum. Along with the rules, the material also presents illustrative examples that demonstrate these rules in real-world contexts, and the experts found this to be the most beneficial. In the words of the expert –

*“the examples could become exercise material to evaluate learners. They could also be used as inspiration to create simpler examples, as some examples involved pro-drop, where some words are omitted for brevity”.*

Some of the invalid rules incorrectly identified the subjects in the sentence. Such syntactic errors are expected given that there is not sufficient quantity and quality of expertly annotated Kannada syntactic analysis available to train a high-quality parser. As we note in [Chapter 4](#), the quality of the extracted patterns is highly dependent on the quality of the syntactic parser, and improvements in the underlying parser will improve the quality of the extracted rules. For both the subject-verb and the object-verb, the model slightly outperformed the baseline (+0.05). Another challenge that experts pointed out is that most volunteer teachers are not trained in formal linguistics, meaning that some might not use such terminology of ‘subject, object’ in classroom teaching. Despite these issues, experts did mention that the material could be beneficial for teachers to know more about the structure of Kannada. As some of these non-dominant word order usages are interesting and often learners do ask questions in the classroom about such exceptions.

**Suffix Usage** We extract the different suffixes used for each word type (e.g. nouns, verbs, adjectives, etc.) but in the interest of time ask the experts to evaluate only the suffixes extracted for nouns and verbs.



Grammar Concept	Relevance		Utility		Presentation	
	% of relevant curriculum covered	% of teachers likely to use	% of teachers that would use for		% of teachers found this _____ to navigate	
General Information	62.1%	highly likely: 8.3% <b>likely: 83.3%</b> not likely: 8.3%	<b>lesson prep: 81.8%</b> student exploration: 54.5% student evaluation: 10%		very easy: 33.3% <b>somewhat easy: 58.3%</b> difficult: 8.3%	
Vocabulary	67.5%	highly likely: 33.3% <b>likely: 58.3%</b> not likely: 8.3%	<b>lesson prep: 72.7%</b> student exploration: 72.7% student evaluation: 45.5%		very easy: 36.3% <b>somewhat easy: 58.3%</b> difficult: 8.3%	
Suffix Usage	52.5%	highly likely: 9.1% <b>likely: 72.7%</b> not likely: 18.2%	<b>lesson prep: 77.8%</b> student exploration: 55.6% student evaluation: 33.3%		very easy: 36.4% <b>somewhat easy: 63.6%</b> difficult: 0%	
Word Order	66%	highly likely: 10% <b>likely: 70%</b> not likely: 20%	<b>lesson prep: 88.9%</b> student exploration: 44.2% student evaluation: 22.2%		very easy: 27.3% <b>somewhat easy: 72.7%</b> difficult: 0%	
Agreement	53.75%	highly likely: 20% <b>likely: 60%</b> not likely: 20%	<b>lesson prep: 77.8%</b> student exploration: 44.4% student evaluation: 22.4%		very easy: 36.4% <b>somewhat easy: 45.5%</b> difficult: 18.2%	

Table 6.5: Perception study results for Kannada. 12 teachers participated in this study

Of the 18 noun suffixes, 7 were marked as valid, 2 suffixes were not suffixes in traditional terms but arise due to ‘sandhi’ i.e. transformation in the characters when two words are joined together. Similarly, for verb suffixes, 53% (7/13) were marked as valid. The experts mentioned that understanding suffix usage is particularly important in Kannada as it is an agglutinative language with different affixes for different grammar categories. They identified that some suffixes (e.g. -ga.Lu, -i.su) although are covered by their existing textbooks 1 and 3, but the examples shown in the materials will still be helpful in the teaching process. They mentioned that –

*“All variations of the suffix cannot be easily understood by students, given that Kannada is highly inflected, for instance the model captured the one variation of the suffix -i.su, but this suffix additionally changes based on gender, person, number. If a large corpus of material is available, these complicated usages can be picked up as an example”.*

#### 6.4.2 Perception Study Results

After the quality study, we conduct the wider-range study with Marathi and Kannada teachers to assess their perception of the extracted materials, and below we present the results of that study for both languages.

#### Kannada Results and Discussion

12 teachers having varying levels of teaching experience, participated in this study, of which 9 identified themselves as female and 3 as male. Three teachers have less than three years of experience, four teachers have between 3-10 years, and the remaining four have 10+ years of experience. Three teachers teach only beginners, while others have experience teaching higher levels as well. All teachers have used some online tools, but mostly for creating assignments and quizzes for the learners (e.g. Google Classroom,

Kahoot<sup>15</sup>, Quizlet<sup>16</sup>), or conducting classes (e.g. Zoom). Some teachers have also referred to YouTube videos and online dictionaries such as Shabdkosh<sup>17</sup> as reference materials. However, they have not used online tools such as AUTOLEX, which in addition to vocabulary explains the concepts of syntax and morphology with illustrative examples. As described in subsection 6.2.3, we ask questions centered on the utility, relevance and presentation of the tool, for each grammar point covered by the tool. We report individual results in Table 6.5.

**Relevance** In general, we see that teachers, on average, find 45–60% of material presented as relevant to their existing curriculum, which is notable given that the underlying text corpus is not specifically curated for language teaching. In fact, the underlying corpus has been extracted from mostly newspapers and legal proceedings and thus consists of more formal and traditional Kannada. The teachers note that especially for beginners they prefer starting with simpler and more conversational language style, but for advanced learners this would be very helpful, in their own words–

*“The examples are well written, however, for the beginners and intermediates, this might be too detailed information. The corpus could be from a wider data source. The use of legal and court related terms are less commonly used in day-to-day life. Advanced learners will certainly benefit from this.”*

**Utility** We find that for all grammar concepts, most teachers expressed that they were likely to use the materials for lesson preparation. Some teachers also mentioned that they could present the material to students for self-exploration, and about 70% teachers voted that it would be especially helpful for vocabulary learning. When asked what aspects of the presented material would they consider using, all teachers said that they would in particular use the illustrative examples for all sections except for the word order and agreement sections. For agreement and word order sections, although they liked the general concepts presented in the material (for example, the non-dominant patterns shown under each section), 88% of the teachers felt that the material covered advanced topics outside the current scope. Although quality evaluation of the rules was not part of this study, teachers did note that if the accuracy of the rules, particularly for suffix usage, could be improved further, they could foresee this tool being used in classroom teaching, as suffixes are essential in Kannada.

**Presentation** In terms of presentation of the materials, we can see from the Table 6.5, all teachers found them easy to navigate through, although it took some getting used to. This is expected given that the teachers spent only a few hours (5-6) over the course of one week to explore all the materials. Additionally, the meta-language used to describe the materials consisted of formal linguistic jargon (for example, most teachers were unfamiliar with the term ‘lemma’) and some teachers noted that:

*“Tool is great and provides clues and ideas for teaching. This is a very vast material; unless you know what exactly you want to look up to, it is a maze where you can easily get lost. The idea of pattern recognition is a very natural way of learning for children who relate to audio and visual patterns to grasp concepts. So this tool helps a lot.”*

---

<sup>15</sup><https://kahoot.com/>

<sup>16</sup><https://quizlet.com/>

<sup>17</sup><https://www.shabdkosh.com/dictionary/english-kannada/>

Grammar Concept	Relevance		Utility		Presentation	
	% of relevant curriculum covered	% of teachers likely to use	% of teachers that would use for		% of teachers found this _____ to navigate	
General Information	15%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -		very easy: - somewhat easy: 80% difficult: 20%	
Vocabulary	16%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: 50%		very easy: - somewhat easy: 100% difficult: -	
Suffix Usage	9%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -		very easy: - somewhat easy: 100% difficult: -	
Word Order	8%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -		very easy: - somewhat easy: 80% difficult: 20%	
Agreement	5%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -		very easy: - somewhat easy: 80% difficult: 20%	

Table 6.6: Perception study results for Marathi. 4 teachers participated in this study

### Marathi Results and Discussion

Compared to the Kannada study, only 5 teachers participated for the Marathi study, all of whom identified themselves as female. These teachers volunteer at small schools that teach mainly at the beginner level with a few intermediate learners. We report the individual results in Table 6.7.

**Relevance** We see that teachers find only 10–15% of the presented materials are relevant to their existing curriculum. This is much less than what the Kannada teachers reported, probably because the Marathi schools’ primary focus is teaching beginners. For beginners, teachers begin with introducing alphabets, simple vocabulary and sentences. In our tool, currently we do not curate the material according to learner age/experience and we have extracted the learning materials from a publicly available text corpus which comprises of news articles, that are not beginner-oriented, as the teachers quote –

*“We focus on varnamala i.e. letters, need to figure out how to use material for 6-7 years old students as the basics are there, but many of the words that are here are from core Marathi newspaper based language, which is very difficult for kids to grasp. They need simpler words and sentences to effectively understand the words and build vocabulary. Also if possible there should be some age and language skill based approach to this learning.”*

**Utility** Similar to the Kannada findings, all teachers noted that they would likely use the materials for lesson preparation. Some teachers also said that they could provide the materials to the advanced students for their self-exploration, to encourage them to explore the materials on their own and ask questions. Similar to the Kannada study, the teachers found the illustrative examples to be of the most utility as they demonstrate a variety of usage. However, they did note that they because the underlying corpus was too restricted in genre, they would benefit more from applying this tool to their curated set of stories, which are written in age-appropriate language as sometimes the example sentences felt a little

Grammar Concept	Relevance % of relevant curriculum covered	% of teachers likely to use	Utility % of teachers that would use for	Presentation % of teachers found this _____ to navigate
General Information	15%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -	very easy: - somewhat easy: 80% difficult: 20%
Vocabulary	16%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: 50%	very easy: - somewhat easy: 100% difficult: -
Suffix Usage	9%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -	very easy: - somewhat easy: 100% difficult: -
Word Order	8%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -	very easy: - somewhat easy: 80% difficult: 20%
Agreement	5%	highly likely: - likely: 40% not likely: 60%	lesson prep: 100% student exploration: 50% student evaluation: -	very easy: - somewhat easy: 80% difficult: 20%

Table 6.7: Perception study results for Marathi. 4 teachers participated in this study

too long and difficult to grasp for students.

**Presentation** All teachers found the materials somewhat easy to navigate and similar to the Kannada teachers they mentioned that it did require some time to understand the format. Some teachers gave a feedback that currently the material is too content heavy and not visually engaging, if the presentation could be improved along those aspects it would make the tool more inviting. Another common feedback about the presentation was regarding organization of the content – the teachers felt it would be more helpful if the content could be broken into sub-sections with not all technical details added in a single layer. For instance, they said –

*“There is a lot of technical details with various parts of speech, subject, object in one go. Perhaps it could be broken down into sub-sections for someone new to the language, for example, it would be helpful to have more clarity on the rules in simpler way followed by simpler examples.”*

In general, we can see from the above discussion that both the Marathi and Kannada findings have some common themes, such as teachers finding the selected grammar concepts relevant to their teaching, but all note that in the current state of the tool, the presented content is more suitable for advanced learners. Among the different features, teachers find the illustrative examples to be most useful, especially for understanding the non-dominant linguistic behaviors or the exceptions to general rules. Currently, a major limitation of the tool, as noted by the teachers, is that the underlying corpus is not curated for learning, and therefore the rules or examples derived from them are not oriented towards different learning levels. However, teachers find this overall effort promising, as this tool can be applied to a corpus of their choice, which is more suited for the learning experience or requirements. Teachers typically base the lessons on interesting stories that are not only engaging, but also have a simpler language. And, especially for beginner learners, the language properties are built through these stories with little use of formal grammar terms. The tool design is promising in that it is corpus-agnostic i.e. each component of

the tool can be directly applied to a teacher-selected text corpus to extract relevant material

## **6.5 Conclusion**

In this chapter, we demonstrated how `AUTOLEX` can be applied for the real-world application of language education, by collaborating with Marathi and Kannada language teachers. Overall, teachers note that the grammar points covered by the tool are interesting, as they also cover non-dominant linguistic patterns or exceptions to general rules, which are important for learners to know. Teachers especially liked the illustrative examples shown under each section, as a helpful reference material for their own lesson preparation or even for learner evaluation. They also note that in the tool's current state, it is more suited for the advanced learners' requirement, but we know that it can be easily adapted to another level (e.g. beginners) by applying the tool on a learner-appropriate text corpora, as selected by the teachers. A next step would be to involve teachers in this extraction process to organize the content by each level, taking the learner incrementally through the complexities of language.



## **Part II**

# **Leveraging Existing and New Data for Improving NLP for Under-resourced Languages.**





## Chapter 7

# Adapting Word Representations to New Languages using Linguistically-Motivated Information

In [Chapter 3](#), [Chapter 4](#), [Chapter 5](#), we demonstrated how, using NLP methods we could extract decent first-pass answers to different linguistic questions for multiple languages. As we discussed earlier in [Chapter 2](#), labeled data for some languages and tasks has been created by human experts (e.g. UD/SUD treebanks ([Nivre et al., 2006](#); [Nivre et al., 2018](#); [Gerdes et al., 2018](#))) and can be used directly for extracting language descriptions, as we did in [Chapter 3](#) and [Chapter 4](#). But, often these annotations are limited in size and variety and we can instead learn automatic models (e.g. syntactic parsers) to get such annotations automatically for a larger variety of data and languages, as we did in [Chapter 6](#). These state-of-the-art automatic methods (e.g. UDIFY ([Kondratyuk and Straka, 2019](#)) for syntactic parsing or AWESOME ([Dou and Neubig, 2021](#)) for acquiring word alignments), although have shown significant performance improvements across many languages and have even allowed their application to languages which have no labeled data, do not generalize well on many under-resourced languages. And as we motivated earlier, the primary goal of this thesis is to automatically create language descriptions for *all* languages, many of which are under-resourced with respect to the quality and quantity of resources required to train high quality models. We also saw earlier how the quality of these extracted language descriptions is directly dependent on the underlying quality of the parsers, translations, etc, and how that directly reflects in the utility of these descriptions ([Chapter 6](#)). Due to lack of sufficient labeled data in the under-resourced languages, the state-of-the-art tools, which are mostly neural network-based, require good quality and quantity of labeled data and thus do not perform equally well on all languages. Therefore, in the next few chapters, we will look at how to quickly add support for a new language to train high-quality models, by leveraging commonalities between languages, as well as by collecting labeled data efficiently.

Specifically, in this chapter, we explore methods to adapt existing models that are trained on high-resource languages onto the under-resourced languages via continuous word representations. Word representations help capture properties of the language, which have led to significant performance improvements in several NLP tasks, such as named entity recognition (NER; [Ma and Hovy \(2016\)](#)), machine

reading (Tan et al., 2017), sentiment analysis (Tang et al., 2016; Yu et al., 2018), etc. These vector representations have especially been instrumental in improving performance in under-resourced languages through cross-lingual transfer learning that allow models to benefit from related languages that have higher resources (Mikolov et al., 2013b; Xing et al., 2015; Devlin et al., 2019). Although learning these word representations requires no supervised data, the quality of the representations is highly contingent upon the availability of the unlabeled data in the required languages. Therefore, in this chapter, we present two approaches to learn better cross-lingual word representations for under-resourced languages by leveraging linguistically-inspired units such as morphemes, graphemes and phonemes.

Aditi Chaudhary, Chunting Zhou, Lori Levin, Graham Neubig, David R. Mortensen, Jaime G. Carbonell. 2018. [Adapting Word Embeddings to New Languages with Morphological and Phonological Subword Representations](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

## 7.1 Overview

Despite there being more than 7000 languages (Hammarström, 2015) in our world, languages are often related to each other, sometimes because of geographical proximity to the place where they are spoken or even because of the common parentage where they are descendants of the same language (Rowe and Levine, 2018). This relatedness can be used to transfer the knowledge learned by the model in one language to a related language through *transfer learning*. Transfer learning (Bozinovski and Fulgosi, 1976) is a family of machine learning methods in which a model learned on one task can be used as a starting point to learn a model on a second task (Tan et al., 2018). These methods can be adapted to leverage the relatedness between languages and are commonly referred to as *cross-lingual transfer learning* methods. This relatedness can manifest itself in terms of vocabulary overlap or even overlap between grammar properties of languages (e.g. syntax, morphology, phonology, etc.), which in turn can help neural network models learn from related languages that have large training data and generalize to those that do not (Mikolov et al., 2013b; Xing et al., 2015; Devlin et al., 2019; Cotterell and Heigold, 2017). Word representations or embeddings (Mikolov et al., 2013a; Bojanowski et al., 2016; Pennington et al., 2014a) have shown great potential for cross-lingual transfer learning (CLTL) which has thus enabled NLP models to leverage data and resources from higher-resourced languages to improve performance on the under-resourced languages (Ammar et al., 2016b; Bharadwaj et al., 2016; Gouws and Søgaard, 2015; Ruder et al., 2019). Recent work (Peters et al., 2018; Devlin et al., 2019) have proposed using contextual word representations instead, in which a given word has different representations based on the context in which they occur. In this chapter, we focus on improving non-contextual word representations for under-resourced languages. This work is still relevant to the present day since non-contextual representations are easy and computationally fast to learn, allowing them to be more accessible across research communities, especially communities with computational constraints. Additionally, as we saw in [Chapter 4](#), non-contextual word embeddings can be easily transformed to derive interpretable semantic features paving way for models trained on these features to also be interpretable, and in turn providing a way to get human-readable language descriptions.

Popular approaches to learn these cross-lingual word embeddings either perform joint training on

graphemes	मूर्तियाँ
phonemes	/mu:rtijã:/
morphemes	/mu:rti-jã:/
lemma+tag	mu:rti + Noun + Nom + Fem + 3PL
gloss	'statues'

Figure 7.1: Subword units of a word in Hindi

the concatenated corpora (Zhang et al., 2017; Conneau et al., 2017; Devlin et al., 2019) or extend monolingual objective functions to align monolingual pre-trained representations in the same space (Mikolov et al., 2013b; Faruqui and Dyer, 2014; Ammar et al., 2016b). Such alignment objectives often require some form of parallel data which is typically of limited quantity and quality. Another dimension on which the existing approaches vary is the lexical unit used during the training. Mikolov et al. (2013b) use entire word as the lexical unit while Bojanowski et al. (2016) propose using subword units in the form of character n-grams to train word embeddings. Several works (Ling et al., 2015; Sennrich et al., 2016), including more recent work (Devlin et al., 2019), have demonstrated the effectiveness of modeling subword units in better cross-lingual learning.

In this chapter, we propose two approaches for enabling CLTL on under-resourced languages by training word representations to effectively leverage resources from higher-resourced language. Both approaches are aimed at mapping the representations of the transfer language and the under-resourced language in the same space. We use linguistically-motivated subword information which cover aspects of word structure (graphemes), inflection (morphemes) and phonological (phonemes) properties, as shown for a Hindi word in Figure 7.1. The key hypothesis of both approaches is that languages are related to each other along multiple dimensions, such as phonology and morphology, which enables effective transfer. We evaluate our approach empirically on the downstream task of NER, because word vectors have a direct impact on the NER model performance— as suggested by Ruder et al. (2019) and also observed by us in Table 7.3, where the model without any pre-trained embeddings scores an average of 18 F1 points less. It thus provides a transparent way to measure the effectiveness of different subword units. Our contributions are summarized below:

1. We show that embeddings trained on subword representations yield better performance on the task than those trained only on whole words, especially in the cross-lingual transfer setting. We further show that embeddings trained on morphological representations often outperform those trained only on whole words.
2. We demonstrate that training embeddings on character-based phonemic representations presents substantial performance advantages over training on orthographic characters in some transfer settings, e.g. when there are script differences across languages.
3. We produce and release continuous representations for each subword unit, giving researchers the ability to use them in their own tasks as they see fit.

## 7.2 Background

In this section, we provide a brief background on the *skip-gram* objective function, a popular objective used for training word embeddings in [subsection 7.2.1](#) and, the different subword units we explore in [Section subsection 7.2.2](#).

### 7.2.1 Skip-Gram Objective

[Mikolov et al. \(2013c\)](#) proposed two objective functions: skip-gram and continuous bag of words (CBOW), to learn word embeddings from a monolingual corpus. Both objectives aim to exploit the dependence of words on the surrounding context in which they occur ([Harris, 1954](#)). While CBOW predicts the word given its context, skip-gram predicts the surrounding context given the word. We use the skip-gram objective function in our approach as it is known to give a better representation for infrequent words<sup>1</sup>, which is crucial for the low resource setting.

We first present the skip-gram objective formulation. More formally, given a sequence of  $T$  words  $w_1, \dots, w_T$ , the skip-gram model maximizes the following log-likelihood:

$$\begin{aligned} p(v|w_i) &= \frac{e^{s(v,w_i)}}{\sum_{j=1}^W e^{s(w_i,j)}} \\ \text{obj} &= \sum_{i=1}^T \sum_{v \in C_i} \log_e p(v|w_i) \end{aligned} \tag{7.1}$$

where  $C_i$  are the context tokens within a specified window of the focus word  $w_i$  and  $p(v|w_i)$  is the probability of observing context word  $v$  given the focus word  $w_i$ .  $s$  is a scoring function mapping the context word and focus word to  $\mathbb{R}$ . The summation in the denominator is over the entire vocabulary  $V$  which makes this formulation computationally inefficient, as the cost of computing the gradient is proportional to  $V$  which is quite large ( $\sim 10^5$ ). [Mikolov et al. \(2013c\)](#) employ negative sampling in order to make this computation tractable resulting in the following log-likelihood:

$$\sum_{i=1}^T \left( \sum_{w_c \in C_i} l(s(w_i, w_c)) + \sum_{w_n \in N_i} l(-s(w_i, w_n)) \right) \tag{7.2}$$

where  $N_i$  are the negative words sampled randomly from vocabulary and  $l$  is the log-sigmoid function. The scoring function  $s$  is a dot product similarity function given by  $s(w_i, w_c) = \mathbf{u}_{w_i}^\top \mathbf{v}_{w_c}$  where  $\mathbf{u}_{w_i}$  and  $\mathbf{v}_{w_c}$  are the embeddings of the focus word and its context word respectively. We use this modified objective function in our approach.

### 7.2.2 Subword Units

A major limitation of [Mikolov et al. \(2013c\)](#) is that they use whole words as their lexical unit which means that these approaches fail to represent new words effectively. [Bojanowski et al. \(2016\)](#) thus proposed to represent individual words using character-level information which helps alleviate the problem

---

<sup>1</sup><https://code.google.com/archive/p/word2vec/>

English	Hindi	Bengali
Pratibha	प्रतिभा <b>prathibha</b>	প্রতিভা <b>perētibā</b>
first	प्रथम <b>prəthmə</b>	প্রথম <b>perəthəm</b>
India	भारत <b>ba:rtə</b>	ভারত <b>barətə</b>
born	जन्मी <b>dʒənmi:</b>	জন্মগ্রহণ <b>dʒənəməgərəhəne</b>
fly	उड़ने <b>ur̥ne</b>	উড়ে <b>ur̥e</b>

Figure 7.2: Similarity between Hindi and Bengali words becomes more apparent as phonemes are able to capture the relatedness between similar languages, despite their orthographic differences.

of representing out-of-vocabulary words. Character-level modeling helps capture information about the internal structure of the word which allows words such as ‘run’ and ‘running’ to be closer in the representation space. Given a sufficiently large monolingual corpus, we expect most or all morphological forms of a lexeme (of which there may be many) to have similar vector representations, however the amount of available data in most languages appears to be the bottleneck. This becomes even more problematic for morphological rich languages such as Hindi, Turkish, Russian where words have several morphologically forms.

We thus explore using three types of linguistically-inspired subword units for training word representations: 1) Orthographic units, 2) Morphological units, and, 3) Phonological units.

*Orthographic units* essentially capture the internal structure of a word by leveraging character-level information. For instance, [Bojanowski et al. \(2016\)](#) represent the focus word  $w_i$  as a set of its character n-grams, denoted by  $\mathbf{u}_{w_i} = \frac{1}{|G|} \sum_{g \in G} \mathbf{x}_g$ , where  $G$  is the set of character ngrams and  $\mathbf{x}_g$  is the vector representation of ngram  $g$ . Such representations capture morphological information in a brute-force but principled fashion—words that share the same morpheme are more likely to share the same character n-grams than words that do not.

*Morphological Units* capture relationships between words more directly. This information is carried in both lemmas (stems or citation forms) and morphological properties (the sets of which are sometimes called “tags”). Lemmas capture information about the lexical identity of a word and are closely correlated with the semantics of a word; tags capture information about the syntactic context of a word. Prior work ([Botha and Blunsom, 2014](#); [Cotterell and Schütze, 2015](#)) have learnt embeddings on morphological representations as well for improving downstream tasks on morphologically rich languages. [Avraham and Goldberg \(2017\)](#) extend [Bojanowski et al. \(2016\)](#) to show the relationship between semantics and morphology by explicitly modeling the lemma and morphological tags. They found these to boost performance in different tasks, with lemmas contributing most to lexical similarity tasks and tags contributing most to morphological similarity tasks.

*Phonological units* capture the phonemic similarity between languages. These are especially useful when closely-related languages share no orthography in which case the above subword units other than morphological tags will likely be of no use. In [Figure 7.2](#), we can see that for the related languages Hindi and Bengali, the similarity between these languages becomes quite obvious when words are represented in their phonemic form. One popular approach to get phonemes is to convert text from its surface (or-

thographic) form into a phonemic representation, stated in terms of the International Phonetic Alphabet (IPA), and train embeddings on this representation. This means that, roughly speaking, morphemes that sound the same will be represented in the same way. [Tsvetkov and Dyer \(2016\)](#); [Bharadwaj et al. \(2016\)](#) have demonstrated the effectiveness of projecting words from orthographic space to phonemic space on downstream tasks (NER, MT).

## 7.3 Proposed Approach

Our proposed approach comprises of: 1) a word embedding training objective function which leverages subword information, and, 2) a training regimen to enable cross-lingual transfer learning.

### 7.3.1 Objective Function

We base our approach on the skip-gram objective function. More formally, let  $P_w$  be the set of linguistic properties of a word consisting of the phoneme n-grams, lemma and individual morphological tags. The focus word is then represented as the average sum of its linguistically motivated subword units:

$$\mathbf{v}_{w_c} = \frac{1}{|P_{w_c}|} \sum_{p \in P_{w_c}} \mathbf{x}_p \quad (7.3)$$

where  $\mathbf{x}_p$  is the vector representation of subword unit  $p$  of word  $w_c$ . For instance, the Hindi word in [Figure 7.1](#) is represented using its phoneme n-grams, lemma and morphological tags as follows:

$$\begin{aligned} &x_{\langle \text{mu} \rangle} + x_{\text{mur}} + x_{\text{murti}} + \dots + x_{\text{murtijā}} \rangle \\ &x_{\text{murti}} + x_{\text{Noun}} + x_{\text{Nom}} + x_{\text{Fem}} + x_{\text{3PL}} \end{aligned} \quad (7.4)$$

The average operation is important to remove any bias towards words having too many or too few subword units. Our objective function differs from [Avraham and Goldberg \(2017\)](#) in that they encode the different morphological inflections as one tag, so that  $\text{Noun}+\text{Nom}+\text{Fem}+\text{3PL}$  would be encoded as  $x_{\text{Noun}+\text{Nom}+\text{Fem}+\text{3PL}}$  instead of encoding each tag separately as proposed by us. We encode each property in a tag separately to avoid data sparsity issues and empirically find this approach to perform better.

### 7.3.2 Training Regimes for Cross-Lingual Transfer Learning

To learn cross-lingual word embeddings, we present two training regimes namely **CT-JOINT** and **CT-FINETUNE** to map representations from the languages into the same space. We hypothesize that having word representations of both languages lying in a similar space will aid the under-resourced language in leveraging resources from the higher resourced language, including annotations for a downstream task.

In the CT-JOINT setting, we learn word representations by applying the above proposed objective function on the concatenated corpora of the higher-resourced and under-resourced language. By virtue of the shared subword units between the languages, the model captures the morphological and phonological similarity between them. [Duong et al. \(2016\)](#) and [Gouws et al. \(2015\)](#) have previously shown the advantages of joint training and we observe that to be true in our case as well.

While CT-JOINT explicitly maps the word representations into the same space through joint training, CT-FINETUNE achieves this implicitly. In this setting, the model takes the learned continuous representations of the high resource subword units, and uses them to initialize the model for the under-resourced language. First, the model is trained using the proposed objective function on the higher-resourced language. Next, the learned representations are then used for initializing the subword units for the under-resourced language. This idea of transferring parameters from high resource language has been previously explored by Zoph et al. (2016) and showed considerable improvement for low resource neural machine translation.

## 7.4 Experimental Settings

We evaluate our word embeddings primarily on the NER task and also show some results on the MT task. We conduct two types of experiments for each task: 1) *cross-lingual transfer experiments* on the low-resource languages—Uyghur and Bengali—using Turkish and Hindi as the high-resource languages respectively, and, 2) *monolingual experiments* on all four languages: Uyghur, Turkish, Bengali and Hindi. In this section, we first describe the experimental setup and data used for training the word embeddings. These language pairs were chosen partly out of convenience—the data were available to us as part of the DARPA LORELEI program—and partly because they satisfied certain deeper desiderata. Turkish and Uyghur are fairly closely related to one another, as are Hindi and Bengali. Despite this relationship, the members of both pairs are written in different scripts (Roman and Perso-Arabic; Devanagari and Bengali). Finally, all four languages are morphologically rich, especially Turkish and Uyghur. These qualities allow us to showcase the value of embeddings with subword units.

**Data** We use data, comprised of unlabeled corpora, English bilingual dictionaries, annotations, from the Linguistic Data Consortium (LDC) language packs—Turkish and Hindi<sup>2</sup>, Bengali<sup>3</sup>, from which we generate train-dev-test splits. Uyghur data was released as part of LoReHLT16 task, organized by NIST<sup>4</sup> under the aegis of DARPA, and training annotations were acquired using native speakers as part of the task. For Uyghur we evaluate on an unsequestered set consisting of 199 annotated evaluation documents, released by NIST. For Turkish, Hindi and Bengali, we create our own train-dev-test splits (Table 7.1). The Uyghur corpus has 27 million tokens and the Turkish corpus has about 40 million tokens. Although Bengali is widely-spoken and the unlabeled corpus contains more than 140 million tokens, there are very few named entity annotations available, making it a low-resource language for the purposes of this exercise. To have a fair experimental setup across language pairs, we sub-sample the Bengali and Hindi corpora to have comparable corpus sizes with Uyghur and Turkish respectively. We also up-sample the low-resource data for both unlabeled corpora and NER annotations, so the model doesn't become biased towards the high-resource language.

**Training Objective Setup** We base the implementation of our training objective function on the C++ implementation of *fasttext*<sup>5</sup> (Bojanowski et al., 2016). For each word in the training corpus, we retrieve

<sup>2</sup>LDC2014E115,LDC2017E62,[http://www.cfilt.iitb.ac.in/iitb\\_parallel/](http://www.cfilt.iitb.ac.in/iitb_parallel/)

<sup>3</sup>LDC2017E60, LDC2015E13

<sup>4</sup><https://www.nist.gov/>

<sup>5</sup><https://github.com/facebookresearch/fastText/>

LANG.	TRAIN	DEV	TEST
Turkish	3376	1126	1126
Uyghur	1822	240	2448*
Hindi	3974	497	497
Bengali	1908	53	7012

Table 7.1: Sentences in train/dev/test set for NER. (\*Unsequestered set)

its phonemes using Epitran (Mortensen et al., 2018) which represents a word using IPA. We consider phoneme and grapheme n-grams ranging from 3-grams to 6-grams and append a special start symbol  $<$  and end symbol  $>$  to the word. The lemmas and morphological tags for a word in context are obtained using a rule-based morphological analyzer in such a fashion as to produce tags similar to the high resource language. For Turkish we use the morphological disambiguator developed by Shen et al. (2016), while for Uyghur, Hindi and Bengali, we developed our own analyzers using a stemmer-like framework<sup>6</sup> over a span of few weeks (2-3).

**Hyperparameters** For training the word embeddings, we consider context tokens within a window size 3 of the focus word and we sample 5 negative examples from the vocabulary. Subword units are initialized with uniform samples from  $[\frac{-1}{dim}, \frac{1}{dim}]$  where  $dim = 100$ . We use the same training regime as Bojanowski et al. (2016). For CT-FINETUNE, instead of uniform samples we initialize the subword units of the low resource language with the representations learnt on a related high resource language.

**Baselines** We compare our cross-lingual word embeddings with two baselines:

- We compare with MULTICCA (Ammar et al., 2016b) which trains multilingual embeddings by projecting multiple languages in the same shared space of one language (English) using canonical correlation analysis (CCA). These projections are learnt using bilingual lexicons. For a fair comparison, we run MULTICCA on embeddings learnt first learnt on monolingual data trained with different subword units.
- We also compare with Bharadwaj et al. (2016) and Mayhew et al. (2017) both of which report results on the same NER datasets. While Bharadwaj et al. (2016) use a neural attention model over phonological features and report the best performance for Turkish using transfer from Uzbek and Uyghur, Mayhew et al. (2017) use some cheap translation methods such as edit distance with related language and report best NER results for Uyghur.

For our monolingual experiments, we compare our proposed approach with models using subword representations—Bojanowski et al. (2016) and Avraham and Goldberg (2017).

<sup>6</sup><https://github.com/dmort27/mstem>



MODEL	SUBWORD UNITS	UYGHUR	BENGALI
CT-JOINT	phoneme-ngrams + lemma + morph	55.00	<b>60.33</b>
	phoneme-ngrams + lemma	<b>56.20</b>	59.63
	phoneme-ngrams	54.90	58.50
	phoneme	51.30	53.75
	char-ngrams + lemma + morph	50.20	55.10
	char-ngrams + lemma	48.20	53.83
	char-ngrams	49.60	52.77
	word	51.80	53.69
CT-FINETUNE	phoneme-ngrams + lemma + morph	48.60	56.19
	lemma + morph	52.80	57.72
	phoneme-ngrams + lemma	51.00	56.83
	phoneme-ngrams	50.50	57.69
	phoneme	49.20	59.86
MULTICCA (BASELINE)	char-ngrams + lemma + morph	41.00	50.63
	char-ngrams + lemma	43.10	50.63
	char-ngrams	45.80	38.06
	word	42.70	45.86

Table 7.2: Transfer experiments on NER. Metric F1 (out of 100%). Uyghur transfer is from Turkish; Bengali transfer is from Hindi. For CT-FINETUNE, **SUBWORD UNITS** refers to the subword units used for pre-training on the high resource language which were then used to initialize the respective subword representations for the low resource language.

## 7.5 Experiments

Our main experiments are focused on improving named entity recognition (NER) on under-resourced languages. NER is the task of identifying named entities such as persons, locations, organizations, geopolitical entities from raw text (Nadeau and Sekine, 2007). We use a hierarchical neural conditional random field (CRF) model proposed by Ma and Hovy (2016) as the base model.

**NER Model Setup** For the cross-lingual transfer experiments we combine the training data from the related languages and train a model over the concatenated training data. We use 100-dimensional word embeddings, pre-trained using the proposed strategies, and use hidden dimension of size 100 for each direction of the BiLSTM. SGD is used as optimizer with a learning rate of 0.015. Dropout of 0.5 is used in the LSTM layer to prevent over-fitting. Uyghur and Turkish were trained for 100 epochs, Bengali and Hindi converged after 70 epochs.

### 7.5.1 Main Results

**Cross-Lingual Experiment Results** Table 7.2 shows the results of the cross-lingual transfer experiments. We experiment with embeddings learnt using different combinations of subword units and find

MODEL	SUBWORD UNITS	TURKISH	UYGHUR	HINDI	BENGALI
Ours	char-ngrams + lemma + morph	68.06	<b>52.50</b>	73.15	<b>52.77</b>
	char-ngrams + lemma	<b>68.61</b>	52.40	73.37	52.09
	char-ngrams + morph	67.97	47.80	<b>73.46</b>	52.06
prop2vec	word + lemma	66.52	46.00	71.82	50.03
	word + morph	64.45	46.00	71.52	49.27
	word + lemma + morph	68.46	47.70	70.51	48.16
fastText	char-ngrams	66.81	50.80	72.67	52.10
word2vec	word	62.85	46.80	72.04	49.83
Random	No embedding	58.94	31.30	59.89	21.25

Table 7.3: NER results for monolingual experiments. Metric F1 (out of 100%)

MODEL	UYGHUR* (UNSEQ.)	UYGHUR*	TURKISH	BENGALI
Ours	<b>56.20</b>	<b>56.00</b>	<b>68.61</b>	<b>60.33</b>
<a href="#">Bharadwaj et al. (2016)</a>	–	51.2	66.47	–
<a href="#">Mayhew et al. (2017)</a>	51.32	55.6	53.44	45.70

Table 7.4: Comparison with previous work using data released by DARPA LORELEI. Metric F1 (out of 100%) \*Official NIST scores.

that the use of both morphological and phonological properties perform the best among all. Our proposed approaches outperform the baseline MULTICCA by a significant margin probably because of the latter strongly depends on bilingual dictionaries which in our low resource setting are not of high quality. Within the proposed approaches, we find CT-JOINT to be consistently better performing than CT-FINETUNE. Interestingly, the performance of CT-FINETUNE model converges to the monolingual performance (Table 7.3). We hypothesize that the model forgets the pre-trained subword units as training progresses, also known as *catastrophic forgetting* (Kirkpatrick et al., 2017), a phenomenon common in neural network models.

**Monolingual Experiment Results** Table 7.3 shows the results of the monolingual experiments. Similar to above, we experiment with different combinations of subword units with the combination of character-ngrams, lemma and morphological properties giving the best performance for Uyghur and Bengali. For Turkish, LEMMA performs better than LEMMA+MORPH, perhaps because the morphological analyzer outputs so many redundant properties which reduce the distance between words that are not particularly similar. In contrast, MORPH helps and LEMMA hurts in Hindi, perhaps because the morph analyzer outputs only a small number of highly informative properties, but is a poor general-purpose lemmatizer. In Table 7.4, we compare our NER performance (official NIST scores) with the then best results reports by prior work on the same unseen Uyghur test data and find our models outperform existing work.

Model	subword units	Uyghur	Bengali
Ours	char-ngrams + lemma + morph	23.59	<b>7.96</b>
	char-ngrams + lemma	<b>23.91</b>	7.77
	char-ngrams + morph	23.27	7.88
fastText	Char-ngrams	23.24	7.91
word2vec	Word	23.31	6.64
Random	No embedding	23.51	6.23

Table 7.5: MT results for monolingual experiments. Metric: BLEU

## 7.5.2 Auxiliary Results

An advantage of our approach is that it can directly be used as-is on any downstream task without requiring task-specific modifications. In this section, we present auxiliary results on a separate task of machine translation (MT) where we apply the (select) learnt embeddings directly to translate text from the low-resource language to English. We use the XNMT toolkit (Neubig et al., 2018) for this purpose.

In Table 7.5 we report the results of the monolingual experiments where we use the embeddings trained using our method for the low-resource language. We observe that the combination of CHARACTER-NGRAMS and LEMMA performs the best for Uyghur and the combination of CHARACTER-NGRAMS+ LEMMA+ MORPH gives the best performance for Bengali over the *word* baseline. This demonstrates the importance of subword units for low-resource MT as well. One likely reason that the combination of character-ngrams and lemmas consistently show the best performance is that, together, they capture lexical similarity, which is more important to translation than the syntactic information captured by morphological inflection (MORPH). However, cross-lingual transfer experiments do not follow the same trend as that of NER probably because the MT models were trained on a training set that did not have translation pairs from the high resource language. As Qi et al. (2018) note, when training MT systems on a single language pair, it is less necessary for the embeddings to be coordinated across the languages.

## 7.6 Conclusion

Empirically experiments show that linguistically-inspired subword-level modeling helps train better word representations overall. Incorporating phonemes and morphemes help bridge the gap between languages in a cross-lingual transfer setting, especially when related languages do not share any orthography. Though our proposed approaches require morphological analyzers, we find that even a morphological analyzer built in 2-3 weeks can boost performance and is a worthwhile investment of resources. Although we do not investigate the utility of such linguistic properties in methods which rely on contextual representations (e.g. Peters et al. (2018); Devlin et al. (2019)), but recent works (Leong and Whitenack, 2022; Nzeyimana and Niyongabo Rubungo, 2022) which do, have also shown the benefits of leveraging morphological and phonological properties in contextual models for under-resourced languages.



## Chapter 8

# Bootstrapping Active Learning with Cross-Lingual Transfer Learning

In the previous chapter, we explored methods for leveraging existing data from related high-resource languages in order to adapt state-of-the-art NLP methods for under-resourced languages. These set of methods are particularly useful when we do not have access to language experts or native speakers who can annotate datasets in the under-resourced language of interest. In situations where we have access to such language experts or native speakers, it is crucial to have methods to collect high quality data in the language of interest for training high-performing NLP models on the task at hand. However, collecting such high quality labeled data across multiple languages requires human effort which is both time-consuming and costly, even more so when the human annotators are not native speakers of the language in question. Therefore, in this chapter we explore methods to collect labeled data in the language of interest requiring minimal annotation effort and time. For this, we use *active learning* (Lewis and Gale, 1994) which uses a data selection algorithm to select useful training samples while minimizing annotation cost. To further improve the model performance on under-resourced languages, we combine benefits of cross-lingual transfer learning with active learning in a single unified framework. In this chapter and next, we present our proposed unified framework on two sequence-labeling tasks: named entity recognition (NER) and part-of-speech (POS) tagging ( Chapter 9), and propose novel active learning strategies in the process.

Aditi Chaudhary, Jiateng Xie, Zaid Sheikh, Graham Neubig, Jaime Carbonell. 2019. [A Little Annotation does a Lot of Good: A Study in Bootstrapping Low-resource Named Entity Recognizers](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*.

### 8.1 Overview

Supervised learning systems often require hundreds or thousands of training samples to perform well depending on the task and language at hand. In most cases, acquiring such labeled training samples is time-consuming and/or expensive, even more so for some languages where native speakers are not easily accessible. Active learning (AL) (Lewis and Gale, 1994; Lewis, 1995; Settles and Craven, 2008) is a sub-field of machine learning which aims to train effective models with less human effort and cost by

selecting such a subset of data that maximizes the end model performance. The key idea being that not all training samples are necessary to train a good model as some samples may offer more information than others for the end model to perform sufficiently well.

In this chapter, we attempt to answer the question *how can we efficiently bootstrap a high-quality named entity recognizer for an under-resourced language with only a small amount of human effort?* We leverage advances in data-efficient learning for under-resourced languages, proposing the following “recipe” for bootstrapping: First, we use cross-lingual transfer learning (CLTL) (Yarowsky et al., 2001; Ammar et al., 2016a) trained on related higher-resourced languages to provide a good preliminary model. Next, on this transferred model we employ AL which helps improve annotation efficiency by using model predictions to select informative, rather than random, data for human annotators. Finally, the model is fine-tuned on data obtained using AL to improve accuracy in the target language. Within this recipe, the choice of the specific method or strategy for choosing and annotating data within AL is highly important to minimize human effort. Furthermore, this strategy needs to be carefully designed according to the task at hand. For instance, in POS tagging each token in a sequence is assigned a POS tag whereas in the case of NER only a single entity within the sentence may be of interest, it can still be tedious and wasteful to annotate full sequences when only a small portion of the sentence is of interest (Neubig et al., 2011; Sperber et al., 2014). Therefore, for the NER task we propose an entity-targeted AL strategy considering the fact that named entities are both important and sparse and select uncertain subspans of tokens within a sequence that are most likely named entities. This way, the annotators only need to assign types to the chosen subspans without having to read and annotate the full sequence.

We evaluate our proposed methods in both a simulated experimental setup and in a human-annotation setup which presents a more practical setting. Experiments across multiple languages: Spanish, Indonesian, Hindi, show that under all settings our proposed strategies outperform existing AL strategies. Our contributions are summarized below:

1. We present a bootstrapping recipe combining AL with cross-lingual transfer learning for improving NER on under-resourced languages. We find that cross-lingual transfer is a powerful tool, outperforming the un-transferred systems with just one-tenth tokens annotated. The code is made publicly available here.<sup>1</sup>
2. We empirically demonstrate the efficacy of our approach across multiple languages through simulation experiments. Human annotation experiments show that annotators are more accurate in annotating entities when using the proposed entity-targeted strategy as opposed to full sequence annotation. Moreover, this strategy minimizes annotator effort by requiring them to label fewer tokens than the full-sequence annotation.

## 8.2 Background

An AL system consists of a *learning algorithm* which poses *queries* in the form of unlabeled data instances to an *oracle* who performs the *data labeling*. A learning algorithm or a learner is a machine learning model which applies different query strategies to select the unlabeled instances for labeling by an oracle which is usually the human annotator. AL is typically an iterative process where the labeled data is then used to update the learning algorithm which poses new queries to the learner and this cycle continues

---

<sup>1</sup><https://github.com/Aditi138/EntityTargetedActiveLearning>

until either all unlabeled instances are labeled or a stopping criterion is reached. This stopping criterion is usually decided by an evaluation metric such as accuracy of the end model on the given task i.e.. if the end model achieves an acceptable level of accuracy on the task then the AL process is stopped. This process is illustrated in [Figure 8.1](#).

In the classical setting, a single unlabeled instance is selected by the learning algorithm and presented to the learner for labeling. This setting is usually applied in situations where unlabeled instances are available to the learning algorithm as a continuous stream one instance at a time and the learner decides whether to query this instance or discard it. This setting is commonly referred to as *stream-based AL* ([Atlas et al., 1990](#)). Stream-based AL has been applied to several NLP tasks such as POS tagging ([Argamon-Engelson and Dagan, 1999](#)), information retrieval from databases ([Yu, 2005](#)), classification ([Žliobaitė et al., 2011](#)).

In this work, we assume that there is a large pool of unlabeled data available from which instances are then selected by the learning algorithm, referred as *pool-based AL*. This setting has also been applied to several applications such as text classification ([Lewis and Gale, 1994](#); [McCallumzy and Nigamy, 1998](#)), information extraction ([Thompson et al., 1999](#); [Settles and Craven, 2008](#)), image retrieval and classification ([Tong and Chang, 2001](#); [Sener and Savarese, 2018](#)). Most prior works select a batch of examples to be labeled in a single iteration instead of the single instance labeling as in the classical setting, since most learning algorithms train over a batch of examples for reducing training overhead. Going forward, all discussion pertaining to AL refers to the pool-based AL setting implemented in a batched fashion unless otherwise mentioned.

### 8.3 Query Strategies

In order to decide what queries should be presented to an oracle, a learning algorithm implements a *query strategy* on the unlabeled pool of data. As mentioned earlier, not all labeled samples are equally important for training, some samples offer more information, which is sufficient to train a model which is competent with a model having access to all training data. There are two schools of thought that inspire the different query strategies: 1) *informativeness* and 2) *representativeness*. Informativeness represents the ability of the selected data to reduce the model uncertainty on its predictions. Example strategies are: query-by-committee ([Dagan and Engelson, 1995](#); [Seung et al., 1992](#)) and uncertainty-sampling ([Lewis and Gale, 1994](#); [Balcan et al., 2007](#); [Tong and Chang, 2001](#); [Fang and Cohn, 2017](#)). A major drawback is that informativeness-only approaches could focus easily on a small subset of samples such as outliers leading to sample bias. Representativeness measures how well the selected data represent the entire unlabeled data. Example strategies explore clustering of unlabeled data ([Dasgupta and Hsu, 2008](#); [Nguyen and Smeulders, 2004](#)), however, the performance of these methods heavily depend on the quality of clustering

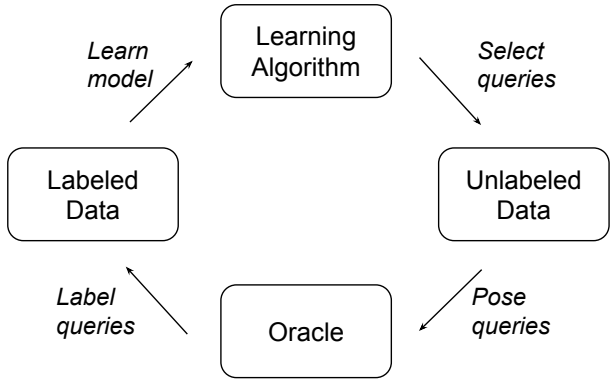


Figure 8.1: An overview of the Active Learning process.

algorithms. Several approaches have thus attempted to combine the benefits of both informativeness and representativeness in a single criteria (Donmez et al., 2007; Xu et al., 2003; Fang and Cohn, 2017). While some approaches such as Donmez et al. (2007) dynamically weigh the mixture of uncertainty-sampling and density-sampling, approaches such as Fang and Cohn (2017) combine these two criteria more seamlessly. For instance Fang and Cohn (2017) attempt to include the representativeness criterion by combining uncertainty sampling with a bias towards high frequency instances for POS tagging. We now present some of the popular query strategies, also used as baselines in this chapter. First, we formally define the problem.

### 8.3.1 Problem Formulation

Given an unlabeled pool of text sequences  $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  in a given language having vocabulary  $V$  and a learner  $\theta$ , an AL query strategy selects a batch  $b$  of unlabeled instances from  $D$  to be annotated by an annotator giving labeled data  $L = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\}$ .

A sequence-labeling task takes an input sequence  $\mathbf{x}_i = \{x_{i,0}, x_{i,1}, \dots, x_{i,|x|}\}$  and produces an output label sequence  $\mathbf{y}_i = \{y_{i,0}, y_{i,1}, \dots, y_{i,|x|}\}$  where each token in the input  $x_{i,t}$  receives as output label  $y_{i,t}$  from a set of possible labels  $\mathcal{J}$ . In the below sections, we describe the different query strategies as applied to sequence-labeling tasks.

Depending on the query strategy and the task at hand, an unlabeled instance can be either the entire sequence  $\mathbf{x}$  or subspans or single tokens within the sequence  $\mathbf{x}$ . In the following sections, we use  $S(\cdot)$  to denote a scoring function used by the different query strategies to score each unlabeled instance on the basis of which the learner then selects the unlabeled instances  $X_{\text{LABEL}}$  to annotate.

### 8.3.2 Uncertainty-Sampling (UNS)

Uncertainty-based sampling strategies (Lewis and Gale, 1994) are the most popular and commonly used query strategies in the AL framework. The key hypothesis is that a learner selects those unlabeled instances about which it is most *uncertain*. Here we assume that a learner is a probabilistic model and has access to the posterior probabilities to enable calculation of an uncertainty measure.

A simple uncertainty-measure is computed by selecting unlabeled instances about which the model is *least confident* given by:

$$S_{\text{LC}}(\mathbf{x}_i) = 1 - P_{\theta}(\hat{y}_i | \mathbf{x}_i) \quad (8.1)$$

where  $\hat{y}_i = \operatorname{argmax}_y P_{\theta}(y_i | \mathbf{x}_i)$  is the prediction of the model, for example  $\mathbf{x}_i$  having the highest posterior probability  $P_{\theta}(\hat{y}_i | \mathbf{x}_i)$  under the model  $\theta$ . Culotta and McCallum (2005) and Settles and Craven (2008) employ this strategy for a variety of tasks including information extraction tasks and sequence labeling tasks (NER). One problem with the least-confident method is that it ignores the label distribution and focuses only on the most probable output class. To remedy this, prior work has proposed using entropy (Shannon, 2001) which measures the amount of information or uncertainty encoded in a variable's possible outcomes where a high entropy suggests a high uncertainty. It is computed as follows:

$$S_{\text{ENT}}(\mathbf{x}_i) = - \sum_{y_i \in Y_i} P_{\theta}(y_i | \mathbf{x}_i) \log_e P_{\theta}(y_i | \mathbf{x}_i) \quad (8.2)$$



where  $\mathbf{Y}_i$  refers to all possible label sequences for  $\mathbf{x}_i$ . For a full-sequence annotation, this could get computationally expensive to compute since the number of label sequences might grow exponentially with increasing sequence length of  $\mathbf{x}$ . Similar is the case for a span-level annotation strategy, however we provide a computationally tractable approach to compute the entropy in [subsection 8.4.2](#). This strategy is relatively easy to implement for a token-level annotation strategy. [Fang and Cohn \(2017\)](#) employ this entropy-based strategy for the POS tagging task where they perform token-level annotations. They calculate the token entropy  $H(x_{i,t}; \theta)$  for each unlabeled sequence  $\mathbf{x}_i$  under model  $\theta$ , defined as

$$H(x_{i,t}; \theta) = - \sum_{j \in \mathcal{J}} P_\theta(y_{i,t} = j | \mathbf{x}_i) \log_e P_\theta(y_{i,t} = j | \mathbf{x}_i) \quad (8.3)$$

where  $P_\theta(y_{i,t} = j | \mathbf{x}_i)$  is the posterior probability of the output class  $j$  for token  $x_{i,t}$  in input sequence  $\mathbf{x}_i$ . This entropy is aggregated across all token occurrences  $D$  to get the uncertainty score for each word type  $v \in V$ :

$$S_{\text{AGG-ENT}}(v) = \sum_{\mathbf{x}_i \in D} \sum_{x_{i,t}=v} H(x_{i,t}; \theta) \quad (8.4)$$

### 8.3.3 Query-by-committee (QBC)

Following the theoretical work on the QBC paradigm ([Seung et al., 1992](#); [Freund et al., 1997](#)), [Dagan and Engelson \(1995\)](#) propose a committee-based selection strategy where a learner selects the tokens having the highest disagreement between a committee of models  $C = \{\theta_1, \theta_2, \theta_3, \dots\}$ . For a token-level annotation strategy disagreement scores are defined as:

$$S_{\text{DIS}}(x_{i,t}) = |C| - \max_{y \in [\hat{y}_{i,t}^{\theta_1}, \hat{y}_{i,t}^{\theta_2}, \dots, \hat{y}_{i,t}^{\theta_c}]} L(y), \quad (8.5)$$

where  $L(y)$  is number of “votes” received for the token label  $y$ .  $\hat{y}_{i,t}^{\theta_c}$  is the prediction with the highest score according to model  $\theta_c$  for the token  $x_{i,t}$ . These disagreement scores are then aggregated over word types:

$$S_{\text{AGG-QBC}}(v) = \sum_{\mathbf{x}_i \in D} \sum_{x_{i,t}=v} S_{\text{QBC}}(x_{i,t}) \quad (8.6)$$

[Dagan and Engelson \(1995\)](#) and [Settles and Craven \(2008\)](#) apply this strategy to a full-sequence annotation scenario on information extraction tasks.

Finally, regardless of whether we use an uncertainty-based or a QBC-based score, the top  $b$  word types with the highest score are then selected as the to-label set

$$X_{\text{LABEL}} = \text{b-argmax}_v S(v), \quad (8.7)$$

where  $\text{b-argmax}$  selects top  $b$  instances having the highest score  $S(v)$ ,  $v$  refers to the annotation unit which can be full sequences, subspans or single tokens/types.

## 8.4 Active Learning for NER

We apply AL to collect labeled data to improve NER in under-resourced languages. To further improve both the active learner and the underlying NER model, we use cross-lingual transfer learning.

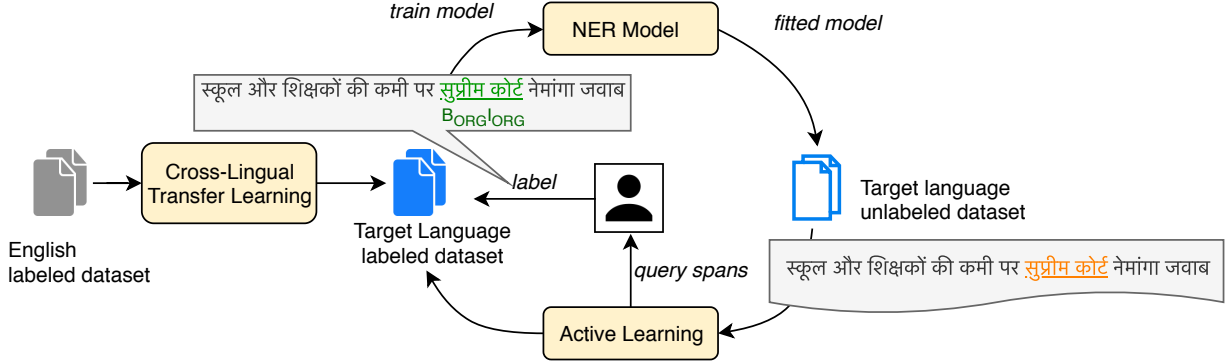


Figure 8.2: Our proposed recipe: cross-lingual transfer is used for projecting annotations from an English labeled dataset to the target language. Entity-targeted active learning is then used to select informative sub-spans which are likely entities for humans to annotate. Finally, the NER model is fine-tuned on this partially-labeled dataset.

### 8.4.1 Task Description

Named entity recognition (NER) is the task of detecting and classifying named entities in text into a fixed set of pre-defined categories such as person, location, organization, etc. We follow the BIOESx labeling scheme (Tjong Kim Sang and Veenstra, 1999) where beginning of an entity is marked with the prefix ‘B-’, middle of an entity by ‘I-’ and not an entity is denoted by ‘O’.

### 8.4.2 Proposed Approach

As mentioned in the introduction, our bootstrapping recipe consists of three components (1) cross-lingual transfer learning, (2) AL to select relevant parts of the data to annotate, and (3) fine-tuning of the model on these annotated segments. The overview of the system is shown in Figure 8.2.

**Cross-Lingual Transfer Learning** Cross-lingual transfer learning (CLTL) is a popular method used for training models for under-resourced languages. In the previous chapter, we had leveraged word embeddings for CLTL. In this case, CLTL projects annotations from a high-resource language (English) into the target language. For this, we follow the approach of Xie et al. (2018) as detailed below.

To begin with, we assume access to two sets of pretrained monolingual word embeddings in the source and target languages,  $X$  and  $Z$ , one small bilingual lexicon, either provided or obtained in an unsupervised manner (Artetxe et al., 2017; Conneau et al., 2017), and labeled training data in the source language. Using these resources, we train bilingual word embeddings (BWE) to create a word-to-word translation dictionary, and finally use this dictionary to translate the source training data into the target language, which we use to train an NER model. To learn BWE, we first obtain a linear mapping  $W$  by solving the following objective:

$$W^* = \underset{W}{\operatorname{argmin}} |WX_D - Z_D|_F \text{ s.t. } WW^\top = I, \quad (8.8)$$

where  $X_D$  and  $Z_D$  correspond to the aligned word embeddings from the bilingual lexicon.  $F$  denotes the Frobenius norm. We can first compute the singular value decomposition  $Z_D^\top X_D = U \Sigma V^\top$ , and

solve the objective by taking  $W^* = UV^\top$ . We obtain BWE by linearly transforming the source and target monolingual word embeddings with  $U$  and  $V$ , namely  $XU$  and  $ZV$ .

After obtaining the BWE, we find the nearest neighbor target word for every source word in the BWE space using the cross-domain similarity local scaling (CSLS) metric (Conneau et al., 2017), which produces a word-to-word translation dictionary. We use this dictionary to translate the source training data into the target language, and simply copy the label for each word, which yields transferred training data in the target language. We train an NER model on this transferred data as our preliminary model. Going forward, in this section we refer to the use of cross-lingual transferred data as CT.

**Entity-Targeted Active Learning** Using the above obtained projected annotations, we train an initial NER model  $\theta$  on the target language, which we use as the active learner to select queries for manual annotation. One relatively standard method used in previous work on NER is to select full sequences based on a criterion for the uncertainty of the entities recognized therein (Culotta and McCallum, 2005). However, as it is often the case that entities are sparse i.e. only a single entity within the sentence may be of interest, it can still be tedious and wasteful to annotate full sequences when only a small portion of the sentence is of interest (Neubig et al., 2011; Sperber et al., 2014). Inspired by this finding and considering the fact that named entities are both important and sparse, we propose an entity-targeted strategy to save annotator effort. Specifically, we select uncertain subspans of tokens within a sequence that are most likely named entities. This way, the annotators only need to assign types to the chosen subspans without having to read and annotate the full sequence. To cope with the resulting partial annotation of sequences, we apply a constrained version of conditional random fields (CRFs) during training that only learn from the annotated subspans (Tsuboi et al., 2008; Wanvarie et al., 2011).

Therefore, after training a model using CLTL, we start the AL process based on this model’s outputs. We begin by training a NER model  $\theta$  using the above model’s outputs as training data. Using this trained model, our proposed entity-targeted AL strategy, referred as ETAL, then selects the most informative spans from a corpus  $D$  of unlabeled sequences. Given an unlabeled sequence  $\mathbf{x}_i \in D$ , ETAL first selects a span of tokens  $x_{i,(a,b)} = x_{i,a} \cdots x_{i,b}$  such that  $x_{i,(a,b)}$  is a likely named entity in sequence  $\mathbf{x}_i$ , where  $a, b \in [0, |\mathbf{x}_i|]$ . Then, in order to obtain highly informative spans across the unlabeled pool  $D$ , ETAL computes the entropy  $H$  for each occurrence of the span  $x_{i,(a,b)}$  and then aggregates them over the entire corpus  $D$ , given by:

$$S_{\text{ETAL}}(v) = \sum_{\mathbf{x}_i \in D} \sum_{x_{i,(a,b)}=v} H(x_{i,(a,b)}; \theta) \quad (8.9)$$

where  $v$  here denotes a unique span of tokens in  $D$ .

We now describe the procedure for calculating  $H(x_{i,(a,b)})$ , which is the entropy of a span  $x_{i,(a,b)}$  being a likely entity. Given the unlabeled sequence  $\mathbf{x}_i$ , the trained NER model  $\theta$  is used for computing the marginal probabilities  $P_\theta(y_{i,t} = j \mid \mathbf{x}_i)$  for each token  $x_{i,t}$  across all possible labels  $j \in \mathcal{J}$  using the forward-backward algorithm (Rabiner, 1989), where  $\mathcal{J}$  is the set of all labels. Using these marginals we calculate the entropy of a given span  $x_{i,(a,b)}$  being an entity as shown in Alg. 1.

Let  $B$  denote the set of labels indicating beginning of an entity,  $I$  the set of labels indicating inside of an entity and  $O$  denoting outside of an entity. First, we compute the probability of a span  $x_{i,(a,b)}$  being an entity, starting with the token  $x_{i,a}$ , by marginalizing  $P_\theta(y_{i,a} \mid \mathbf{x}_i)$  over all labels in  $B$ , denoted as  $p_{\text{SPAN}}^{(a,b)}$ . Since an entity can span multiple tokens, for each subsequent token  $x_{i,b}$  being part of that

---

**Algorithm 1: Entity-Targeted Active Learning**

---

```
1  $B \leftarrow$  label-set denoting beginning of an entity
2  $I \leftarrow$  label-set denoting inside of an entity
3  $O \leftarrow$  outside of an entity span
4  $p_{i,t,j} := P_{\theta}(y_{i,t} = j \mid \mathbf{x}_i) \leftarrow$  marginal probability of token  $x_{i,t}$  taking output label  $j$ 
5 for  $a \leftarrow 1 \dots |\mathbf{x}_i|, b = 1$  do
6    $p_{\text{SPAN}}^{(a,b)} = \sum_{j \in B} p_{i,a,j}$ 
7   for  $b \leftarrow a + 1 \dots |\mathbf{x}_i|$  do
8      $p_{\text{ENTITY}}^{(a,b)} \leftarrow p_{\text{SPAN}}^{(a,b)} * p_{i,b,O}$ 
9      $H = -p_{\text{ENTITY}}^{(a,b)} \log_e p_{\text{ENTITY}}^{(a,b)}$ 
10    if  $H >$  threshold then
11       $v \leftarrow x_{i,(a,b)}$ 
12       $S_{\text{ETAL}}(v) \leftarrow S_{\text{ETAL}}(v) + H$ 
13    end
14     $p_{\text{SPAN}}^{(a,b)} \leftarrow p_{\text{SPAN}}^{(a,b)} * \sum_{j \in I} p_{i,b,j}$ 
15  end
16 end
```

---

entity, we marginalize  $P_{\theta}(y_{i,b} \mid \mathbf{x}_i)$  over all labels in  $I$  and combine it with  $p_{\text{SPAN}}^{(a,b)}$ . Finally, we compute  $p_{\text{ENTITY}}$  by multiplying  $p_{\text{SPAN}}^{(a,b)}$  with  $P_{\theta}(y_{i,b} = O \mid \mathbf{x}_i)$ , which denotes end of a likely entity. Since we use the marginal probability for computing  $p_{\text{ENTITY}}$ , it already factors in the transition probability between tags. Thus, any invalid sequences such as  $B_{\text{PER}}I_{\text{ORG}}$  have low scores. Further, since contiguous spans have overlapping tokens, we use dynamic programming (DP) to compute  $p_{\text{SPAN}}^{(a,b)}$  which avoids an exponential computation when considering all possible spans and labels in a sequence. Using  $p_{\text{ENTITY}}$ , we compute the entropy  $H$  and only consider the spans having  $H$  higher than a pre-defined threshold *threshold*. The reason for this thresholding is purely for computational purposes as it allows us to discard all spans that have a very low probability of being an entity, keeping the number of spans actually stored in memory low. As mentioned above, we aggregate the entropy of spans  $S_{\text{ETAL}}$  over the entire unlabeled set, thus combining uncertainty sampling with a bias towards high frequency entities, following [Fang and Cohn \(2017\)](#). Using this strategy, we select subspans in each sequence for annotation. The annotator only needs to assign named entity types to the chosen subspans, adjust the span boundary if needed, and ignore the rest of the sequence, saving much effort.

**Training Model and Regimen** With the newly obtained training data from AL, we attempt to improve the original transferred model. In this section, we first describe our model architecture, and try to address: 1) how to train the NER model effectively with partially annotated sequences? 2) what training scheme is best suited to improve the transferred model?

Our NER model is a BiLSTM-CNN conditional random field (CRF) model based on [Ma and Hovy \(2016\)](#) consisting of: a character-level CNN, that allows the model to capture subword information; a word-level BiLSTM, that consumes word embeddings and produces context sensitive hidden represen-

tations; and a linear-chain CRF layer that models the dependency between labels for inference. We use the above model for training the initial NER model on the transferred data as well as for re-training the model on the data acquired from AL.

AL with span-based strategies such as ETAL, produces a training dataset of partially labeled sequences. To train the NER model on these partially labeled sequences, we take inspiration from [Bellare and McCallum \(2007\)](#); [Tsuboi et al. \(2008\)](#) and use a constrained CRF decoder. Normally, CRF computes the likelihood of a label sequence  $\mathbf{y}$  given a sequence  $\mathbf{x}$  as follows:

$$p_{\theta}(\mathbf{y}|\mathbf{x}) = \frac{\prod_{t=1}^T \psi_i(y_{t-1}, y_t, \mathbf{x}, t)}{Z(\mathbf{x})} \quad (8.10)$$

$$Z(\mathbf{x}) = \sum_{\mathbf{y} \in \mathbf{Y}(T)} \prod_{t=1}^T \psi_i(y_{t-1}, y_t, \mathbf{x}, t) \quad (8.11)$$

where  $T$  is the length of the sequence,  $\mathbf{Y}(T)$  denotes the set of all possible label sequences with length  $T$ , and  $\psi_i(y_{t-1}, y_t, \mathbf{x}) = \exp(\mathbf{W}_{y_{t-1}, y_t}^T \mathbf{x}_i + \mathbf{b}_{y_{t-1}, y_t})$  is the energy function. To compute the likelihood of a sequence where some labels are unknown, we use a constrained CRF which marginalizes out the un-annotated tokens. Specifically, let  $\mathbf{Y}_L$  denote the set of all possible sequences that include the partial annotations (for unannotated tokens, all labels are possible), and we compute the likelihood as:

$$p_{\theta}(\mathbf{Y}_L|\mathbf{x}) = \sum_{\mathbf{y} \in \mathbf{Y}_L} p_{\theta}(\mathbf{y}|\mathbf{x}) \quad (8.12)$$

We refer to the use of a constrained CRF as PARTIAL-CRF.

To improve our model with the newly labeled data, we directly fine-tune the initial model, trained on the transferred data, on the data acquired through active learning, referred as FINETUNE. Each token-level run produces more labeled data, for which this training procedure is repeated again. We also compare the NER performance using two other training schemes: CORPUSAUG, where we train the model on the concatenated corpus of transferred data and the newly acquired data, and CORPUSAUG+FINETUNE, where we additionally fine-tune the model trained using CORPUSAUG on just the newly acquired data.

## 8.5 Experimental Settings

We evaluate our proposed strategy in both simulated and human-annotation experiments.

**Data** The first evaluation set includes the benchmark CoNLL 2002 and 2003 NER datasets ([Tjong Kim Sang, 2002](#); [Tjong Kim Sang and De Meulder, 2003](#)) for Spanish (from the Romance family), Dutch and German (like English, from the Germanic family). We use the standard corpus splits for train/dev/test. The second evaluation set is for the low-resource setting where we use the Indonesian (from the Austronesian family), Hindi (from the Indo-Aryan family) and Spanish datasets released by the Linguistic Data Consortium (LDC).<sup>2</sup> We generate the train/dev/test split by random sampling. Details of the corpus statistics are in [Table 8.1](#).

For extracting the English-transferred Data, we use the same experimental settings and resources as described in [Xie et al. \(2018\)](#) to get the translations of the English training data for each target language.

---

<sup>2</sup>LDC2017E62,LDC2016E97,LDC2017E66

SOURCE	DATASET	TRAIN / DEV / TEST	TOTAL TOKENS
		# SENTENCES	IN TRAIN
LDC	Hindi	2570 / 809 / 1592	48604
	Indonesian	3181 / 1001 / 1991	55270
	Spanish	1398 / 465 / 928	31799
CoNLL	Dutch	13274 / 2307 / 4227	200059
	German	12067 / 2849 / 2984	206846
	Spanish	8357 / 1915 / 1517	264715

Table 8.1: Corpus Statistics.

**Active Learning Setup** As described in [subsection 8.4.2](#), a DP-based algorithm is employed to select the uncertain entity spans which runs for all n-grams having length  $\leq 5$ . This length was approximated by computing the 90th percentile on the length of entities in the English training data. We set the entropy threshold for filtering individual spans to  $1e^{-8}$ .

**Model Setup** For each language, we train the model with 100d pre-trained GloVe ([Pennington et al., 2014b](#)) word embeddings trained on Wikipedia and the monolingual text extracted from the train set. We use hidden size of 200 for each direction of the LSTM and a dropout of 0.5. SGD is used as the optimizer with a learning rate of 0.015. During fine-tuning, the NER model is first trained on the transferred data with the above settings. For the first token-level run, the model is fine-tuned on the target language with a lower learning rate of  $1e^{-5}$  and for each subsequent run, this rate is increased to 0.015.

**Baselines** We use cross-lingual transfer (CT) to train our initial NER model and test on the target language. This is the same setting as [Xie et al. \(2018\)](#) and serves as our baseline. We also use existing AL strategies to select data for manual annotation using this trained NER model. We compare our proposed ETAL strategy with the following baseline strategies:

- **SAL** Select whole sequences for which the model has least confidence in the most likely labeling ([Culotta and McCallum, 2005](#)). Refer to the  $S_{LC}$  calculation in [section 8.3](#) for more details.
- **CFEAL** Select least confident spans within a sequence using the confidence field estimation method ([Culotta and McCallum, 2004](#)). They propose a computationally tractable approach to measure the confidence of subspans within a sequence which we use as an uncertainty-measure.
- **RAND** Select spans randomly from the unlabeled set for annotation.

## 8.6 Simulation Experiments

In the simulated experimental setting, we simulate manual annotation by using gold labels for the data selected by token-level. At each subsequent run, we annotate 200 tokens and fine-tune the NER model on all the data acquired so far, which is then used to select data for the next run of annotation.

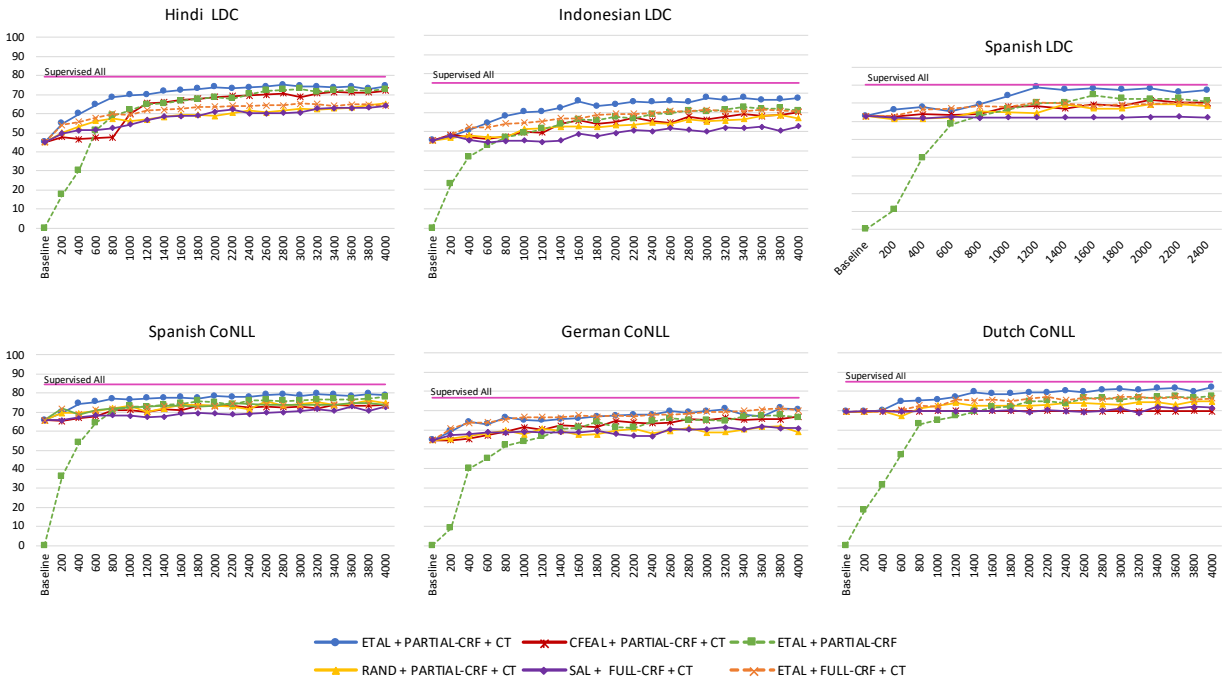


Figure 8.3: Comparison of the NER performance trained with the FineTune scheme, across six datasets. Solid lines compare the different token-level strategies. Dashed lines show the ablation experiments. The x-axis denotes the total number of tokens annotated and the y-axis denotes the F1 score.

**Results** Figure 8.3 summarizes the results for all datasets across the different experimental settings. Each data-point on the x-axis corresponds to the NER performance after annotating 200 additional tokens. CT denotes using cross-lingual transferred data to train the initial NER model for both kick-starting the token-level process and also for fine-tuning the NER model on the newly-acquired data. PARTIAL-CRF/FULL-CRF denote the type of CRF decoder used in the NER model. Throughout this section, we report results averaged across all token-level runs unless otherwise noted. Individual scores can be found in the original paper (Chaudhary et al., 2019).

As can be seen in the figure, our proposed approach, denoted by ETAL+PARTIAL-CRF+CT, outperforms the previous token-level baselines for all the datasets. Holding the other two components of CT and PARTIAL-CRF constant, we conduct experiments to compare the different token-level strategies, which are denoted by the solid lines in Figure 8.3. We see that ETAL outperforms the other strategies by a significant margin for both the CoNLL datasets and the LDC datasets at the end of all runs. Although CFEAL also selects informative spans, ETAL outperforms it because ETAL is optimized to select likely entities, causing more entities to be annotated for almost all datasets. Despite fully labeled data being added in SAL, ETAL outperforms it because SAL selects longer sentences with fewer entities. Furthermore, we find that even with just one-tenth annotated tokens, the proposed recipe is only (avg.) -5.2 F1 behind the model trained using all labeled data, denoted by SUPERVISED ALL.

We observe that the transferred data from English provides a good start to the NER model. As expected, cross-lingual transfer helps more for the languages closely related to English which are Dutch, German, Spanish. In our first ablation study, we train a ETAL+PARTIAL-CRF where no transferred data is used. We observe that as more in-domain data is acquired, the un-transferred setting soon approaches the transferred setting ETAL+PARTIAL-CRF+CT suggesting that an efficient annotation strategy can help close the gap between these two systems with as few as  $\sim 1000$  tokens (avg.).

In our second ablation study, we study the effect of using the original CRF (FULL-CRF) instead of the PARTIAL-CRF for training with partially labeled data. Since the former requires fully labeled sequences, the un-annotated tokens in a sequence are labeled with the model predictions. We see from Figure 8.3 that the FULL-CRF performs worse (avg. -4.1 F1) than when PARTIAL-CRF is used because FULL-CRF significantly hurts the recall for all datasets. We also experiment with different NER training regimes (described in section 8.4) for ETAL and observe that, generally, fine-tuning not only speeds up the training but also gives better performance than the other strategies. Therefore, for human annotation experiments, we use the FINETUNE strategy.

## 8.7 Human Annotation Experiments

We conduct human annotation experiments for Hindi, Indonesian and Spanish to understand whether ETAL helps reduce the annotation effort and improve annotation quality in practical settings. We compare ETAL with the full sequence strategy (SAL).

**Setup** We use six native speakers, two for each language, with different levels of familiarity with the NER task. Each annotator was provided with practice sessions to gain familiarity with the annotation guidelines and the user interface. The annotators annotated for 20 mins time for each strategy. For ETAL, the annotator was required to annotate single spans i.e. each sequence contained one span of



	ANNOTATOR PERFORMANCE		TEST PERFORMANCE (# ANNOTATED TOKENS)		
	ETAL	SAL	ETAL	SAL	SAL-Full
HI-1	<b>78.8</b>	63.7	<b>50.4</b> (326)	44.2 (326)	53.3 (1894)
HI-2	<b>82.7</b>	72.2	<b>49.1</b> (234)	45.9 (234)	55.6 (2242)
ID-1	66.1	<b>77.8</b>	<b>50.4</b> (425)	45.8 (425)	51.3 (3232)
ID-2	73.0	<b>79.5</b>	<b>51.2</b> (251)	46.5 (251)	54.0 (2874)
ES-1	<b>79.7</b>	75.0	<b>63.7</b> (204)	62.2 (204)	64.6 (2134)
ES-2	<b>83.1</b>	70.4	<b>63.8</b> (199)	62.2 (199)	62.6 (2134)

Table 8.2: Annotator performance measures F1 of each annotator with respect to the oracle annotator which is the gold data. Test Performance measures the NER F1 scores using the annotations as training data. The number in the brackets denote the number of annotated tokens used for training the NER model. ES:Spanish, HI:Hindi, ID: Indonesian.

tokens. This involved assigning the correct label and adjusting the span boundary if required. For SAL, the annotator was required to annotate all possible entities in the sequence. We randomized the order in which the annotators had to annotate using the ETAL and SAL strategy. Figure 8.6 illustrates the human annotation process for the ETAL strategy in the annotation interface.

**Results** Table 8.2 records the results of human annotation experiments. On comparing each annotator’s annotation quality with respect to the oracle, denoted by *Annotator Performance*, we find that both Hindi and Spanish annotators have higher annotation quality using ETAL. We believe this is because by selecting possible entity spans, ETAL not only saves effort on searching the entities in a sequence but also allows the annotators to read less overall and concentrate more on the things that they do read, as seen in Figure 8.4. However, for SAL we see that the annotator missed a likely entity because they focused on the other more salient entities in the sequence.<sup>3</sup>

On comparing the *Test Performance* of the NER models trained on these annotations in Table 8.2, we find that for the same number of tokens annotated (denoted by the number mentioned in brackets) ETAL outperforms SAL similar to the simulation results. SAL-FULL denotes the results of the strategy when trained on all the annotations acquired in the stipulated time. We do observe that SAL-FULL has a larger number of annotated tokens than ETAL. Upon analysis, we find that most sequences selected by SAL-FULL did not have any entities. Since “not-an-entity” is the default label in the annotation interface, no operation is required for annotating these, allowing for more tokens to be annotated per unit times. When we count the number of entities present in the data selected by the two strategies, we see in Figure 8.5 that data selected by ETAL has a significantly larger number of entities than SAL, across all the human annotation experiments. We note that when we consider all the annotated tokens, SAL-FULL has slightly better results. However, despite having six times fewer annotated tokens, the difference between ETAL and SAL-FULL is (avg.) 2.1 F1. This suggests that ETAL can achieve competitive performance with

<sup>3</sup>For Indonesian, we see an opposite trend due to several inconsistencies in the gold labels.

ETAL	Sentence: स्कूल और शिक्षकों की कमी पर [सुप्रीम कोर्ट] ने मांगा जवाब School and teachers 's lack of Supreme Court asks answer
	Gold: B <sub>ORG</sub> I <sub>ORG</sub> Human: B <sub>ORG</sub> I <sub>ORG</sub>
ETAL	Sentence: विराट [कोहली] को आईसीसी की टेस्ट टीम में जगह नहीं Virat Kohli has ICC 's Test Team in place no
	Gold: B <sub>PER</sub> I <sub>PER</sub> Human: B <sub>PER</sub> I <sub>PER</sub>
SAL	Sentence: [मिस्र के 21 ईसाइयों बंधकों का IS ने किया सिर कलम] Egypt 's 21 Christian brothers 's IS made head lines
	Gold: B <sub>GPE</sub> O O B <sub>ORG</sub> O O B <sub>ORG</sub> O O O O Human: B <sub>GPE</sub> O O O O O B <sub>ORG</sub> O O O O

Figure 8.4: Examples from Hindi human annotation experiments for both ETAL and SAL. Square brackets denote the spans (for ETAL) or the entire sequence (for SAL) selected by the AL strategy.

fewer annotations.

## 8.8 Conclusion

We propose a bootstrapping recipe for improving entity recognition in under-resourced languages using a combination of both cross-lingual transfer learning and active learning. From both the simulation and human experiments, we show that a targeted annotation strategy such as ETAL achieves competitive performance with less manual effort while maintaining high annotation quality. Given that ETAL can help find twice as many entities as SAL, a potential application of ETAL can also be for creating a high-quality entity gazetteer under a short time budget.

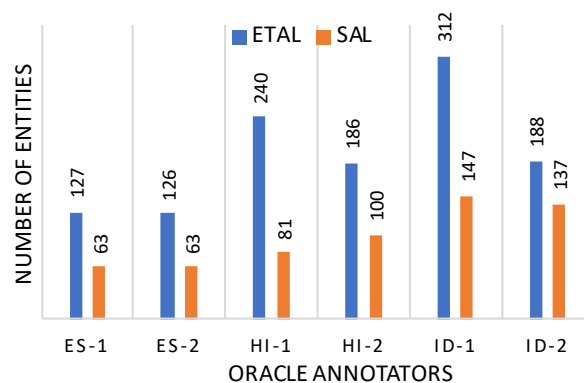


Figure 8.5: Comparing the number of entities in the data selected by ETAL and SAL, as annotated by oracle.

Original वरिष्ठ पत्रकार सुनीता ऐरॉन के मुताबिक ये साफ़ हो गया है कि अखिलेश यादव और राहुल गांधी के मुकाबले " नरेंद्र मोदी बड़े ' यूथ आइकन ' हैं " .

Original मायावती ने प्रेस काँग्रेस में आरोप लगाया है कि ईवीएम में गड़बड़ी थी और किसी भी बटन पर वोट देने से वोट भाजपा को मिल रहे थे .

(a) Selected spans using ETAL strategy are highlighted for the human annotator to annotate.

Original वरिष्ठ पत्रकार सुनीता ऐरॉन के मुताबिक ये साफ़ हो गया है कि अखिलेश यादव और राहुल गांधी के मुकाबले " नरेंद्र मोदी बड़े ' यूथ आइकन ' हैं " .

**Named Entities**

Person (p)     Organization (o)     Geopolitical Entity (g)     Location (l)

Unknown (u)     Not An Entity (n)

(b) Human annotator correcting the span boundary and assigning the correct entity type.

Original मायावती ने प्रेस काँग्रेस में आरोप लगाया है कि ईवीएम में गड़बड़ी थी और किसी भी बटन पर वोट देने से वोट भाजपा को मिल रहे थे .

**Named Entities**

Person (p)     Organization (o)     Geopolitical Entity (g)     Location (l)

Unknown (u)     Not An Entity (n)

(c) Human annotator assigning the correct entity type only since selected span boundary is correct.

Original वरिष्ठ पत्रकार सुनीता ऐरॉन के मुताबिक ये साफ़ हो गया है कि अखिलेश यादव और राहुल गांधी के मुकाबले " नरेंद्र मोदी बड़े ' यूथ आइकन ' हैं " .

Original मायावती ने प्रेस काँग्रेस में आरोप लगाया है कि ईवीएम में गड़बड़ी थी और किसी भी बटन पर वोट देने से वोट भाजपा को मिल रहे थे .

(d) Partially-annotated sequences after being annotated by the human annotator.

Figure 8.6: Example of the human annotation process for Hindi.



## Chapter 9

# Confusion Reducing Active Learning

In the previous chapter, we presented a unified framework which combines the benefits of cross-lingual transfer learning with active learning for improving entity recognition in under-resourced languages. In this chapter, we apply this unified framework on POS tagging, a core task in language documentation and understanding. As we also saw in [Chapter 3](#), [Chapter 4](#) and [Chapter 5](#), POS tagging is one of the first steps for syntactic parsing, from which we then derive language descriptions. In applying Active Learning (AL) to the POS tagging task, we find a surprising result that even in an *oracle* scenario where we know the true uncertainty of the predictions, these current query strategies are far from optimal. Based on this analysis, we pose the problem of AL for POS tagging as selecting instances which *maximally reduce the confusion between particular pairs of output tags*.

Aditi Chaudhary, Antonios Anastasopoulos, Zaid Sheikh, Graham Neubig. 2020. [Reducing Confusion in Active Learning for Part-Of-Speech Tagging](#). In *Transactions of the Association for Computational Linguistics 2020*.

### 9.1 Overview

Part-Of-Speech (POS) tagging is an important component of NLP systems such as named entity recognition (NER; [Ankita and Nazeer \(2018\)](#)), machine translation (MT; [Feng et al. \(2019\)](#)), question answering (QA; [Wang et al. \(2018\)](#)). It is also one of the first steps used by linguists who try to answer linguistic questions or document under-resourced languages ([Anastasopoulos et al., 2018](#)). The development of high-quality POS taggers ([Huang et al., 2015](#); [Bohnet et al., 2018](#)) often requires large amounts of labeled data that are not readily available for most languages. Therefore, to collect high-quality labeled data from human experts while minimizing annotation effort and cost, we use the AL framework described in detail in [Chapter 8](#).

While many query strategies have been proposed in the past ([Dagan and Engelson, 1995](#); [Settles and Craven, 2008](#); [Marcheggiani and Artières, 2014](#); [Fang and Cohn, 2017](#)), in this work we find that within the same task setup (POS tagging) these strategies perform inconsistently across different languages. We believe this inconsistent performance is because existing methods consider only uncertainty in predictions without considering the *direction* of the uncertainty with respect to the output labels. For instance, in [Figure 9.1](#) we consider the German token “die,” which may be either a pronoun (PRO) or determiner

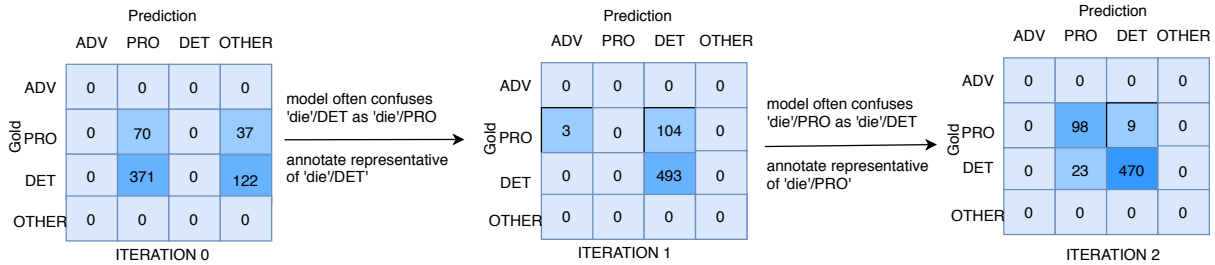


Figure 9.1: Illustration of selecting representative token-tag combinations to reduce confusion between the output tags on the German token ‘die’ in an idealized scenario where we know true model confusion.

(DET). According to the initial model (iteration 0), “die” was labeled as PRO majority of the time, but a significant amount of probability mass was also assigned to other output tags (OTHER) for many examples. Based on this, existing AL algorithms that select uncertain tokens will likely select “die” because it is frequent and its predictions are not certain, but they may select an instance of “die” with *either* a gold label of PRO or DET. Intuitively, because we would like to correct errors where tokens with true labels of DET are mis-labeled by the model as PRO, asking the human annotator to tag an instance with a true label of PRO, even if it is uncertain, is not likely to be of much benefit.

To remedy this problem, we pose the problem of AL for POS tagging as selecting tokens which maximally *reduce the confusion* between the output tags. For instance, in the above example we would attempt to pick a token-tag pair “die/DET” to reduce potential errors of the model over-predicting PRO. The task of POS tagging is likely to benefit more from addressing this issue because of the *syncretism* phenomenon observed in several languages. Syncretism is a linguistic phenomenon where distinctions required by syntax are not realized by morphology, meaning a word type can have multiple POS tags based on the context in which the word occurs. We evaluate our proposed AL strategy by running simulation experiments on six diverse languages namely German, Swedish, Galician, North Sami, Persian, and Ukrainian followed by human annotation experiments on Griko, an endangered language that truly lacks significant resources. Following the setup used in the previous chapter for NER, we bootstrap the AL strategy with cross-lingual transfer learning (CLTL) by transferring a POS tagger learnt on a set of related languages (Cotterell and Heigold, 2017) on the target language. Our contributions are summarized as follows:

1. We empirically demonstrate the shortcomings of existing AL methods under conventional as well as “oracle” settings where the true model confusions are known as in Figure 9.1. Extensive analysis across six diverse languages shows that the selected data using our proposed AL method closely matches the oracle (gold) data distribution. The code is publicly released here.<sup>1</sup>
2. We further present auxiliary results demonstrating the importance of model calibration, the accuracy of the model’s probability estimates themselves (Nixon et al., 2019), and show that cross-view training (Clark et al., 2018) is an effective way to improve calibration.
3. Finally, through the human annotation experiments on an endangered language, Griko, we collect 300 new token-level annotations which will help further Griko NLP systems.

<sup>1</sup><https://github.com/Aditi138/CRAL>

	QBC-ORACLE	UNS-ORACLE
<b>ITERATION-1</b>	PART=1	ADP=1
<b>ITERATION-2</b>	PART=1,ADP=1	ADP=2
<b>ITERATION-3</b>	ADV=1,PART=1,ADP=1	ADP=2
<b>ITERATION-4</b>	ADV=1,PART=1,ADP=2	ADP=3

Table 9.1: Each cell is the tag selected for German token ‘zu’ at each iteration. Gold output tag distribution for ‘zu’ is ADP=194, PART=103, ADV=5, PROPN=5, ADJ=1.

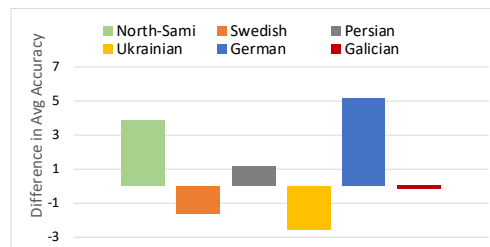


Table 9.2: Illustrating the inconsistent performance of UNS-ORACLE and QBC-ORACLE methods. y-axis is difference in the (avg.) POS accuracy for these two methods across 20 iterations.

## 9.2 Background: Failings of Query Strategies

Most existing AL query strategies proposed for sequence-labeling tasks such as NER, POS tagging, use some form of uncertainty-measure to select the informative data for labelling (Marcheggiani and Artières, 2014; Settles, 2009; Ringger et al., 2007; Fang and Cohn, 2017). They experiment with different variants of the entropy measure across both token- and sequence-level annotation schemes for sequence-labeling tasks. However, to the best of our knowledge, none of the existing works are targeted at reducing confusion within the output classes. Some of the most widely used query strategies are the *uncertainty-sampling* (UNS) (subsection 8.3.2) and the *query-by-committee* (QBC) (subsection 8.3.3) methods. While UNS selects the most uncertain word types in the unlabeled corpus for annotation, QBC selects the tokens having the highest disagreement between a committee of models. Similar to Chapter 8, we adopt a token-level annotation scheme as opposed to a full-sequence annotation which is time-consuming and requires more effort. We refer the reader to the equations of UNS in Equation 8.3, Equation 8.4 and, for QBC in Equation 8.5, Equation 8.6, both of which produce aggregated scores  $S_{\text{AGG-ENT}}(v)$  and  $S_{\text{AGG-QBC}}(v)$  respectively for each word type  $v$ .

In a preliminary empirical study, we find these existing methods are less-than optimal, and fail to bring consistent gains across multiple settings (languages). Ideally, having a single strategy that performs consistently across diverse languages is desirable for easy extensibility to new languages. Furthermore, to test the effectiveness of an AL strategy, it is often advisable to conduct multiple AL iterations. However, experimenting with different strategies across multiple iterations with human annotation is costly and thus having a single strategy known a-priori will reduce both time and human annotation effort. Specifically, we demonstrate this problem of inconsistency through a set of *oracle* experiments, where the data selection algorithm has access to the true labels. More details on the setup are in section 9.5. These experiments hope to serve as an upper-bound for their non-oracle counterparts, so if existing methods do not achieve gains even in this case, they will certainly be even less promising when true labels are not available at data selection time, as is the case in standard AL. Concretely, as an oracle *uncertainty sampling* method UNS-ORACLE, we select word types with the highest negative log likelihood of their true label. As an oracle *query-by-committee* method QBC-ORACLE, we select word types having the largest number of incorrect predictions. We find two key observations:

1. Between the oracle methods (Table 9.2), no method consistently performs the best across all six languages.
2. Simply relying on an uncertainty measure without considering the output class distribution leads to unbalanced selection of the resulting tags. This is demonstrated in Table 9.1 where the output tags selected for the German token ‘zu’ are shown across multiple iterations. While UNS-ORACLE selects the most frequent output tag, it fails to select tokens from other output tags. Interestingly, QBC-ORACLE selects tokens across multiple tags, however the distribution is not in proportion with the true tag distribution.

Our hypothesis is that this inconsistent performance occurs because none of the methods consider the confusion between output tags while selecting data. As mentioned earlier, this is especially important for POS tagging because we find that the existing methods tend to select highly syncretic word types.<sup>2</sup>

### 9.3 Proposed Approach

In order to address the above limitations, we propose a novel AL query strategy which aims at reducing confusion between the output tags, hereby referred as CRAL. We follow a similar bootstrapping approach as done for NER (in Chapter 8) where cross-lingual transfer learning (CLTL) seeds the active learner. We first present our proposed active learning strategy followed by the model and training regimen.

#### 9.3.1 Query Strategy: CRAL

To recap, given an unlabeled pool of input text sequences  $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$  in a given language having vocabulary  $V$  and a learner  $\theta$ , an active learning query strategy selects a batch  $b$  of unlabeled instances from  $D$  to be annotated by an annotator giving labeled data  $L = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\}$ . An output label sequence is denoted by  $\mathbf{y}_i = \{y_{i,0}, y_{i,1}, \dots, y_{i,|x|}\}$  where each token in the input  $x_{i,t}$  receives as output label  $y_{i,t}$  from a set of possible labels  $\mathcal{J}$ , in this case POS tags. Our proposed algorithm consists of two main steps. First, we select the word types about which the model is *most confused*, and second, we find the *most representative* token instance for each selected type to be presented to the annotator.

#### Selecting the most confusing word types.

The goal of this step is to find  $b$  word types which would *maximally reduce* the model confusion within the output tags. For each token  $x_{i,t}$  in the unlabeled sequence  $\mathbf{x}_i \in D$ , we first define the confusion as the sum of posterior probability  $P_\theta(y_{i,t} = j \mid \mathbf{x}_i)$  of all output tags  $\mathcal{J}$  other than the highest probability output tag  $\hat{y}_{i,t}$ :

$$S_{\text{CONF}}(x_{i,t}) = 1 - P_\theta(y_{i,t} = \hat{y}_{i,t} \mid \mathbf{x}_i), \quad (9.1)$$

$$S_{\text{CRAL}}(v) = \sum_{\mathbf{x}_i \in D} \sum_{x_{i,t}=v} S_{\text{CONF}}(x_{i,t}). \quad (9.2)$$

A high  $S_{\text{CONF}}(x_{i,t})$  indicates that the model is less confident on the most probable tag and thus more confused between the output tags. The model confusion is further aggregated over all token occurrences to get the type-level confusion score  $S_{\text{CRAL}}(v)$ . Next, we select the top  $b$  word types having the highest

---

<sup>2</sup>Details can be found in Table 9.4.



---

**Algorithm 2: Confusion-Reducing Active Learning**

---

```
1  $D \leftarrow$  unlabeled set of sequences
2  $V \leftarrow$  vocabulary
3  $\mathcal{J} \leftarrow$  output tag-set
4  $b \leftarrow$  active learning batch size
5  $P_\theta(y_{i,t} = j \mid \mathbf{x}_i) \leftarrow$  marginal probability
6  $p_{i,t,j} := P_\theta(y_{i,t} = j \mid \mathbf{x}_i)$ 
7 for  $\mathbf{x}_i \in D$  do
8   for  $(x_{i,t}) \in \mathbf{x}_i$  do
9      $v \leftarrow x_{i,t}$ 
10     $S_{\text{CRAL}}(v) \leftarrow S_{\text{CRAL}}(v) + (1 - p_{i,t,\hat{y}_{i,t}})$ 
11     $\hat{j} \leftarrow \operatorname{argmax}_{j \in \mathcal{J} \setminus \{\hat{y}_{i,t}\}} p_{i,t,j}$ 
12     $O_{\text{CRAL}}(v, \hat{j}) \leftarrow O_{\text{CRAL}}(v, \hat{j}) + 1$ 
13  end
14 end
15  $X_{\text{INIT}} \leftarrow b\text{-argmax}_{v \in V} S_{\text{CRAL}}(v)$ 
16 for  $v_k \in X_{\text{INIT}}$  do
17    $j_k \leftarrow \operatorname{argmax}_{j \in \mathcal{J}} O_{\text{CRAL}}(v_k, j)$ 
18   for  $x_{i,t} \in D$  s.t.  $x_{i,t} = v_k$  do
19      $\mathbf{c}_{x_{i,t}} \leftarrow \operatorname{enc}(x_{i,t})$ 
20      $W_{x_{i,t}} = p_{i,t,j_k} * \mathbf{c}_{x_{i,t}}$ 
21   end
22    $X_{\text{LABEL}}(v_k) = \operatorname{CENTROID}\{W_{x_{i,t}=v_k}\}$ 
23 end
```

---

aggregated confusion score (given by  $b\text{-argmax}$ ) which gives us the most confusing word types. For each token, we also store the output tag that is the second most probable tag (i.e. the tag with the second highest posterior probability) which we refer to as the “most confusing output tag” for a particular  $x_{i,t}$  in  $O(x_{i,t}, j)$ :

$$O(x_{i,t}, j) = \begin{cases} 1 & \text{if } j = \operatorname{argmax}_{j \in \mathcal{J} \setminus \{\hat{y}_{i,t}\}} p_{i,t,j} \\ 0 & \text{otherwise.} \end{cases} \quad (9.3)$$

For each word type  $v$ , we aggregate the frequency of the most confusing output tag across all token occurrences and compute the output tag with the highest frequency as the most confusing output tag for type  $v$ . Finally, for each of the top  $b$  most confusing word types, we retrieve its most confusing output tag resulting in type-tag pairs given by  $X_{\text{INIT}} = \{\langle v_1, j_1 \rangle, \dots, \langle v_b, j_b \rangle\}$ . This process is illustrated in steps 7–14 in [Alg. 2](#).

### Select the most representative token instances.

Now that we have the most confusing type-tag pairs  $X_{\text{INIT}}$ , our final step is selecting the most representative token instances for annotation. For each type-tag tuple  $\langle v_k, j_k \rangle \in X_{\text{INIT}}$ , we first retrieve

contextualized representations for all token occurrences ( $x_{i,t} = v_k$ ) of the word-type  $v_k$  from the encoder of the POS model. We express this in shorthand as  $\mathbf{c}_{i,t} := \text{enc}(x_{i,t})$ . Since the true labels are unknown, there is no certain way of knowing which tokens have the “most confusing output tag” as the true label. Therefore, each token representation  $\mathbf{c}_{i,t}$  is weighted with the model confidence of the most confusing tag  $j_k$  given by step 19–20 in [Alg. 2](#). Finally, the token instance that is closest to the centroid of this weighted token set becomes the most representative instance for annotation. Going forward, we also refer to the most representative token instance as the centroid for simplicity. This process is repeated for each of the word-types  $v_k$  resulting in the to-label set  $X_{\text{LABEL}}$ . We take inspiration from [Sener and Savarese \(2018\)](#) in selecting the centroid as a good approximation of representativeness. They pose AL as a core-set selection problem where a core set is the subset of data on which the model is trained closely matches the performance of the model trained on the entire dataset. They show that finding the core set is equivalent to choosing  $b$  center points such that the largest distance between a data point and its nearest center is minimized.

Similar to [Fang and Cohn \(2017\)](#) and [Chaudhary et al. \(2019\)](#), the selected representative tokens are presented in context for manual annotation.

### 9.3.2 Training Model and Regimen

In this section, we present the POS model architecture and the training algorithm. As mentioned before, we use cross-lingual transfer learning to improve the POS model on under-resourced languages.

#### Model Architecture

The POS model is a hierarchical neural conditional random field (CRF) tagger ([Ma and Hovy, 2016](#); [Lample et al., 2016](#); [Yang et al., 2017](#)) where each token  $(\mathbf{x}, t)$  from the input sequence  $\mathbf{x}$  is first passed through a character-level BiLSTM, followed by a self-attention layer ([Vaswani et al., 2017a](#)). On top of the self-attention layer, another BiLSTM is used to capture information about subword structure of the words. Finally, these character-level representations are fed into a token-level BiLSTM in order to create contextual representations  $\mathbf{c}_t = \vec{h}_t : \overleftarrow{h}_t$ , where  $\vec{h}_t$  and  $\overleftarrow{h}_t$  are the representations from the forward and backward LSTMs, and “:” denotes the concatenation operation. The encoded representations are then used by the CRF decoder to produce the output sequence.

Similar to the entity-targeted active learning ETAL strategy proposed in our previous chapter ([subsection 8.4.2](#)), we collect token-level annotations and thus cannot directly use the traditional CRF which expects a fully labeled sequence. Instead, we use a constrained CRF ([Bellare and McCallum, 2007](#)) which computes the loss only for annotated tokens by marginalizing the un-annotated tokens.

#### Cross-View Training Regimen

In order to further improve the above model, we apply cross-view training (CVT), a semi-supervised learning method ([Clark et al., 2018](#)). The key hypothesis of CVT is that it leverages both unlabeled and labeled data for training a robust model. On unlabeled examples, CVT uses a self-training algorithm ([Yarowsky, 1995](#)) where it trains auxiliary prediction modules. These auxiliary modules look at restricted “views” of the input sequence and attempt to match the prediction from the full view. By forcing the auxiliary modules to match the full-view module, CVT improves the model’s representation learning.

Not only does it help in improving the downstream performance under low-resource conditions, but also improves the model calibration overall (details in [section 9.5](#)). Having a well-calibrated model is quite useful for AL, as a well-calibrated model tends to assign lower probabilities to “true” incorrect predictions which allows the AL measure to select these incorrect tokens for annotation.

CVT is comprised of four auxiliary prediction modules, namely: the forward module  $\theta_{fwd}$  which makes predictions without looking at the right of the current token, the backward module  $\theta_{bwd}$  which makes predictions without looking at the left of the current token, the future module  $\theta_{fut}$  which does not look at either the right context or the current token and, the past module  $\theta_{pst}$  which does not look at either the left context or the current token. The token representations  $\mathbf{c}_t$  for each module can be seen as follows:

$$\mathbf{c}_t^{\text{fwd}} = \overrightarrow{\mathbf{h}}_t, \quad \mathbf{c}_t^{\text{bwd}} = \overleftarrow{\mathbf{h}}_t, \quad \mathbf{c}_t^{\text{full}} = \overrightarrow{\mathbf{h}}_t : \overleftarrow{\mathbf{h}}_t, \quad \mathbf{c}_t^{\text{fut}} = \overrightarrow{\mathbf{h}}_{t-1}, \quad \mathbf{c}_t^{\text{pst}} = \overleftarrow{\mathbf{h}}_{t+1} \quad (9.4)$$

For an unlabeled input sequence  $\mathbf{x}$ , the full-view model  $\theta_{full}$  first produces soft targets  $p_\theta(y|\mathbf{x})$  upon inference and then CVT matches the soft predictions from  $M$  auxiliary modules by minimizing their KL-divergence. Although CRF produces a probability distribution over all possible output sequences, for computational feasibility we compute the token-level KL-divergence using the posterior probability distribution  $P_\theta(y_t|\mathbf{x})$  over all output tags  $\mathcal{J}$ . The CVT loss function is given as:

$$l_{\text{CVT}} = \frac{1}{|D|} \sum_{\mathbf{x}_i \in D} \sum_{x_{i,t} \in \mathbf{x}_i} \sum_{m=1}^M KL(p_\theta^{full} || p_\theta^m) \quad (9.5)$$

where  $p_\theta^{full} := P_\theta^{full}(y_{i,t} = j | \mathbf{x}_i)$  and  $p_\theta^v := P_\theta^v(y_{i,t} = j | \mathbf{x}_i)$ .  $|D|$  is the total unlabeled examples in  $D$ . The loss obtained from CVT above is then interpolated with the supervised loss function.

### Cross-Lingual Transfer Learning

As mentioned in the introduction, we use cross-lingual transfer learning (CLTL) to bootstrap the active learning model. In the previous chapter ([Chapter 8](#)), we used annotation projection ([Xie et al., 2018](#); [Mayhew et al., 2017](#)) to transfer annotations from English onto the target under-resourced language using bilingual dictionaries. In this work our primary focus is on designing an active learning method, so we simply pre-train a POS model on a group of related high-resource languages ([Cotterell and Heigold, 2017](#)) which is a computationally cheap solution, a crucial requirement for running multiple AL iterations.

Therefore, using the architecture described above, for any given target language we first train a POS model on a group of related high-resource languages and then *fine-tune* this pre-trained model on the newly acquired annotations. In order to select a set of related higher-resourced languages, we first run the automated tool provided by [Lin et al. \(2019\)](#), which leverages features such as phylogenetic similarity, typology, lexical overlap, and size of available data, in order to predict a list of optimal transfer languages. This list is then refined using the experimenter’s intuition. Finally, a POS model is trained on the concatenated corpora of the related languages.

## 9.4 Experimental Settings

We evaluate our proposed approach using both simulation experiments, where we use the gold labels to simulate an annotator, and human annotation experiments where we ask linguists to perform the

TARGET LANGUAGE	TRANSFER LANGUAGES (TREEBANK)
German (de-gsd)	English (en-ewt) + Dutch (nl-alpino)
Swedish (sv-lines)	Norwegian (no-nynorsk) + Danish (da-ddt)
North Sami (sme-giella)	Finnish (fi-ftb)
Persian (fa-seraji)	Urdu (ur-udtb) + Russian (ru-gsd)
Galician (gl-treegal)	Spanish (es-gsd) + Portuguese (pt-gsd)
Ukrainian (uk-iu)	Russian (ru-gsd)
Griko	Greek (el-gdt) + Italian (it-postwita)

Table 9.3: Dataset details describing the group of related languages over which the model was pre-trained for a given target language.

manual annotation.

**Data** For the simulation experiments, we evaluate on six diverse languages: German, Swedish, North Sami, Persian, Ukrainian and Galician. We use data from the Universal Dependencies (UD) v2.3 (Nivre et al., 2016; Nivre et al., 2018; Kirov et al., 2018) project with the same train/dev/test split as proposed in McCarthy et al. (2018).<sup>3</sup> For each target language, the set of related languages used for pre-training is listed in Table 9.3. Persian and Urdu datasets being in the Perso-Arabic script, there is no orthography overlap along the transfer and the target languages. Therefore, we use uroman,<sup>4</sup> a publicly available tool for romanization. Details on the Griko data are discussed in section 9.6.

**Model Setup** We use a hidden size of 25 for the character BiLSTM, 100 for the modeling layer and 200 for the token-level BiLSTM. Character embeddings are 30-dimensional and are randomly initialized. We apply a dropout of 0.3 to the character embeddings before inputting to the BiLSTM. A further 0.5 dropout is applied to the output vectors of all BiLSTMs. The model is trained using the SGD optimizer with learning rate of 0.015. The model is trained till convergence over a validation set.

**Active Learning Setup** For all AL methods, we acquire annotations in batches of 50 and run multiple iterations for each method. We pre-train the model using the above parameters and after acquiring annotations, we fine-tune it with a learning rate proportional to the number of sentences in the labeled data  $lr = 2.5e^{-5}|X_{\text{LABEL}}|$ .

**Baselines** We compare our proposed method (CRAL) with the following baselines:

- **UNS** Select word types about which the model is most uncertain by aggregating entropy scores across all token occurrences for a given type.
- **QBC** Select word types on which a committee of models most disagree on their predictions. For each word type, the disagreement scores are aggregated across all token occurrences. We use the

<sup>3</sup><https://github.com/sigmorphon/2019/tree/master/task2>

<sup>4</sup><https://www.isi.edu/~ulf/uroman.html>

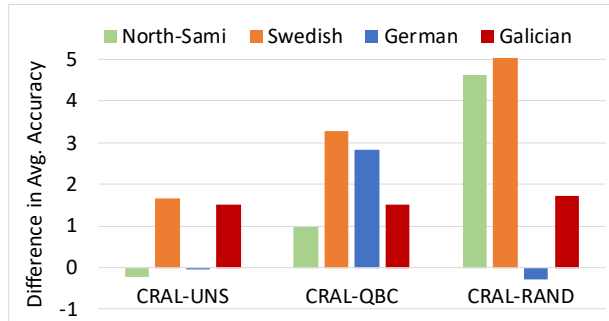


Figure 9.2: Comparing the difference in POS performance across the AL methods with BRNN/MLP architecture, averaged across 20 iterations.

following committee of models  $C = \{\theta_{fwd}, \theta_{bwd}, \theta_{full}\}$ , where  $\theta_i$  are the CVT views (section 9.3). We do not include the  $\theta_{fut}$  and  $\theta_{pst}$  as they are much weaker in comparison to the other views.<sup>5</sup>

- **RAND** Selects tokens randomly from the unlabeled data  $D$ .

For CRAL, UNS and RAND, we use the full model view.

## 9.5 Simulation Experiments

In this setting, we simulate the manual annotation by using gold labels for the data selected by AL. We conduct 20 AL iterations for each method in batches of 50, resulting in 1000 annotated tokens for each language.

### 9.5.1 Results

Figure 9.3 compares our proposed CRAL strategy with the existing baselines. Y-axis represents the difference in POS tagging performance between two AL methods and is measured by accuracy. The accuracy is averaged across 20 iterations. Across all six languages, we find that our proposed method CRAL shows significant performance gains over the other methods. In order to check how the performance of the AL methods is affected by the underlying POS tagger architecture, we conduct additional experiments with a different architecture. We replace the CRF layer with a linear layer and use token level softmax to predict the tags, keeping the encoder as before. We present the results for four (North Sami, Swedish, German, Galician) of the six languages in Figure 9.2. Our proposed method CRAL still always outperforms QBC. We observe that only for North Sami, UNS outperforms CRAL, which is similar to the results obtained from BRNN/CRF architecture where the CRAL performs at par with UNS. Next, we perform intrinsic evaluation to compare the quality of the selected data on two aspects:

**How similar are the selected and the true data distributions?** To measure this similarity, we compare the output tag distribution for each word type in the selected data with the tag distribution

<sup>5</sup>We chose CVT views for QBC over the ensemble for computational reasons. Training 3 models independently would require three times the computation. Given that for each language we run 20 experiments amounting to a total of 120 experiments, reducing the computational burden was preferred.

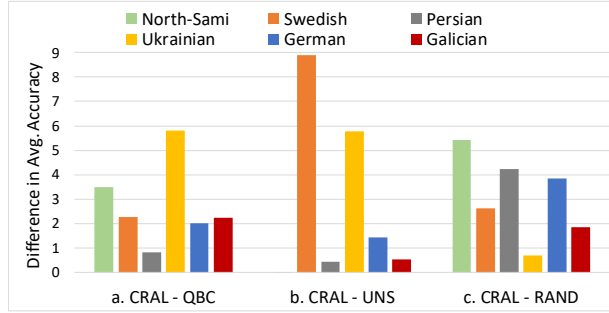


Figure 9.3: Our method (CRAL) outperforms existing AL methods for all six languages. Y-axis is the difference in POS accuracy between CRAL and other AL methods, averaged across 20 iterations with batch size 50.

TARGET LANGUAGE	UNS	QBC	CRAL
German	74 %	76 %	82%
Swedish	56 %	54 %	62 %
North-Sami	10 %	12 %	14 %
Persian	50 %	46 %	46 %
Galician	40 %	42 %	44 %
Ukrainian	38 %	48 %	48 %

Table 9.4: Percentage of syncretic word types in the first iteration of active learning (consisting of 50 types).

TARGET LANGUAGE	CRAL	UNS	QBC
German	<b>0.0465</b>	0.0801	0.0849
Swedish	<b>0.0811</b>	0.1196	0.1013
North Sami	<b>0.0270</b>	0.0328	0.0346
Persian	<b>0.0384</b>	0.0583	0.0444
Galician	0.0722	0.0953	<b>0.0674</b>
Ukrainian	0.0770	0.1067	<b>0.0665</b>

Table 9.5: Wasserstein distance between the output tag distributions of the selected data and the gold data, lower the better. The above results are after 200 annotated tokens.

in the gold data. This evaluation is necessary because there are significant number of syncretic word types in the selected data as seen in Table 9.4. To recap, *syncretic* word types are word types that can have multiple POS tags based on context. We compute the Wasserstein distance (a metric to compute distance between two probability distributions) between the annotated tag distribution and the true tag distribution for each word type  $v$ .

$$WD(v) = \sum_{j \in \mathcal{J}_v} p_j^{\text{AL}}(v) - p_j^*(v), \quad (9.6)$$

where  $\mathcal{J}_v$  is the set of output tags for a word type  $v$  in the selected active learning data.  $p_j^{\text{AL}}(v)$  denotes the proportion of tokens annotated with tag  $j$  in the selected data and  $p_j^*$  is the proportion of tokens having tag  $j$  in the entire gold data. Lower Wasserstein distance suggests high similarity between the selected tag distribution and output tag distribution. Given that each iteration selects unique tokens, this distance is computed after  $n = 4$  iterations. Table 9.5 shows that our proposed strategy CRAL selects data which closely matches the gold data distribution for four out of the six languages.

**How effective is the AL method in reducing confusion across iterations?** Across iterations, as more data is acquired we expect the incorrect predictions from the previous iterations to be rectified in the subsequent iterations, ideally without damaging the accuracy of existing predictions. However, as

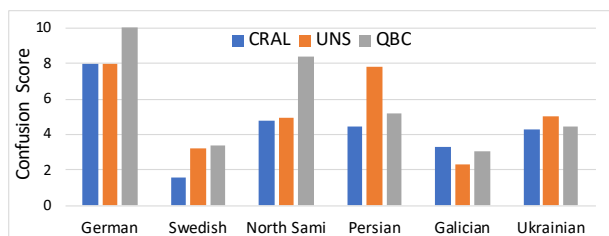


Figure 9.4: Confusion score measures the percentage of correct predictions in the first iteration which were incorrectly predicted in the second iterations. Lower values suggest that the selected annotations in the subsequent iterations cause less damage on the model trained on the existing annotations.

seen in Table 9.4, the AL methods have a tendency to select syncretic word types, suggesting that across multiple iterations the same word types could get selected albeit under a different context. This could lead to more confusion thereby damaging the existing accuracy if the selected type is not a good representative of its annotated tag. Therefore, we calculate the number of existing correct predictions which were incorrectly predicted in the subsequent iteration, and present the results in Figure 9.4. A lower value suggests that the AL method was effective in improving overall accuracy without damaging the accuracy from existing annotations, and thereby was successful in reducing confusion. From Figure 9.4, the proposed strategy CRAL is clearly more effective than the others in most cases in reducing confusion across iterations.

**Oracle Results** As mentioned in the introduction, we compare the AL methods under “oracle” settings as well, where we have access to the gold labels during data selection. This comparison is important because if the AL methods perform inconsistently even with access to true labels then they are likely to perform inconsistently in practical settings as well where they don’t have access to the true labels. The oracle versions of existing methods UNS-ORACLE and QBC-ORACLE are already described in section 9.2. For our proposed method CRAL, we construct the oracle version as follows: Select the word types having the highest number of incorrect predictions. Within each type, select that output tag which is the most incorrectly predicted. This gives the most confusing output tag for a given word type. From the tokens having the most confusing output tag, select the token representative by taking the centroid of their respective contextualized representations.

Figure 9.5 compares the performance gain of the POS model trained using CRAL-ORACLE over UNS-ORACLE and QBC-ORACLE (Figure 9.5.a, Figure 9.5.b). We find that our proposed method performs consistently better across all languages, except Ukrainian, unlike the existing methods as seen in Table 9.2. We hypothesize that this inconsistency is due to noisy annotations in Ukrainian. On analysis we found that the oracle method predicts numerals as NUM but in the gold data some of them are annotated as ADJ. We also find several tokens to have punctuations and numbers mixed with the letters.<sup>6</sup> Further, we find that CRAL closely matches the performance of its corresponding oracle CRAL-ORACLE (Figure 9.5.c) which suggests that the proposed method is close to an optimal AL method.

In order to verify whether CRAL is accurately selecting data at near-oracle levels, we analyze the

<sup>6</sup>This is also noted in the UD page: [https://universaldependencies.org/treebanks/uk\\_iu/index.html](https://universaldependencies.org/treebanks/uk_iu/index.html)

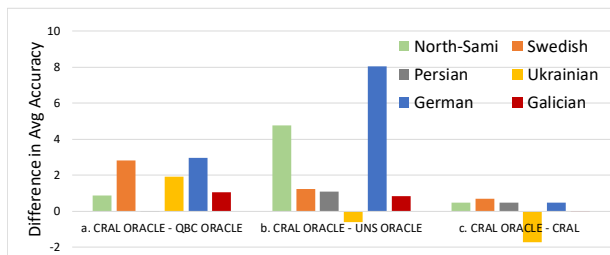


Figure 9.5: In the *oracle* setting, our method (CRAL-ORACLE) outperforms UNS-ORACLE and QBC-ORACLE in most cases, while the non-oracle CRAL matches the performance of its oracle counterpart. y-axis measures the difference in average accuracy across 20 iterations.

intermediate steps leading to the data selection. For each selected word type  $z \in X_{\text{LABEL}}$ , we analyze how well our proposed method of weighting encoder representations with the model confidence of the most confused tag and taking the centroid actually succeeds at “representative” token selection. If this is indeed the case, tokens in the vicinity of the centroid should also have the same “most confused tag” as their predicted label and thereby be mis-classified instances. To verify this hypothesis we compare how many of the 100 tokens closest to the centroid (in the representation space) ( $X_{\text{NN}}(z)$ ) are truly mis-classified. This score is given by  $p(z)$  for each selected word-type  $z$ :

$$X_{\text{NN}}(z) = \underset{x_{i,t}=z \in D}{\text{b- argmin}} |c_{i,t} - c_z|$$

$$p(z) = \frac{|\hat{y}_{i,t} \neq y_{i,t}^*|}{|X_{\text{NN}}(z)|}$$

where  $b = 100$ .  $c_z$  is the contextualized representation of the representative instance for the word-type  $z$  i.e. the centroid and  $c_{i,t}$  is the contextualized representation of  $z$ 's token instance  $x_{i,t}$ .  $y_{i,t}^*$  and  $\hat{y}_{i,t}$  are the true and predicted labels of  $x_{i,t}$ . We report the average and median of  $\mathbf{p}$  across all the selected tokens of the first AL iteration in Figure 9.6. We see that for all languages the median is high (i.e.  $> 0.8$ ) which suggests that the majority of the token-tag pairs satisfy this criteria, thus supporting the step of weighting the token representations and choosing the centroid for annotation.

We also compare the percent of token-tag overlap between the data selected from CRAL with its oracle counterpart: CRAL-ORACLE. For the first AL iteration, the proposed method CRAL has more than 50% overlap with the oracle method for all languages, providing some evidence as to why CRAL is matching the oracle performance.

### 9.5.2 Auxiliary Results

In this section, we present auxiliary results which show that cross-view training (CVT) not only helps improve our POS model overall but also helps in model calibration which can be important for active learning. A model is well-calibrated when a model’s predicted probabilities over the outcomes reflects the true probabilities over these outcomes (Nixon et al., 2019). We use Static Calibration Error (SCE), a metric proposed by Nixon et al. (2019) to measure the model calibration. SCE bins the model predictions separately for each output tag probability and computes the calibration error within each bin which is averaged across all the bins to produce a single score. For each output tag, bins are created by sorting the



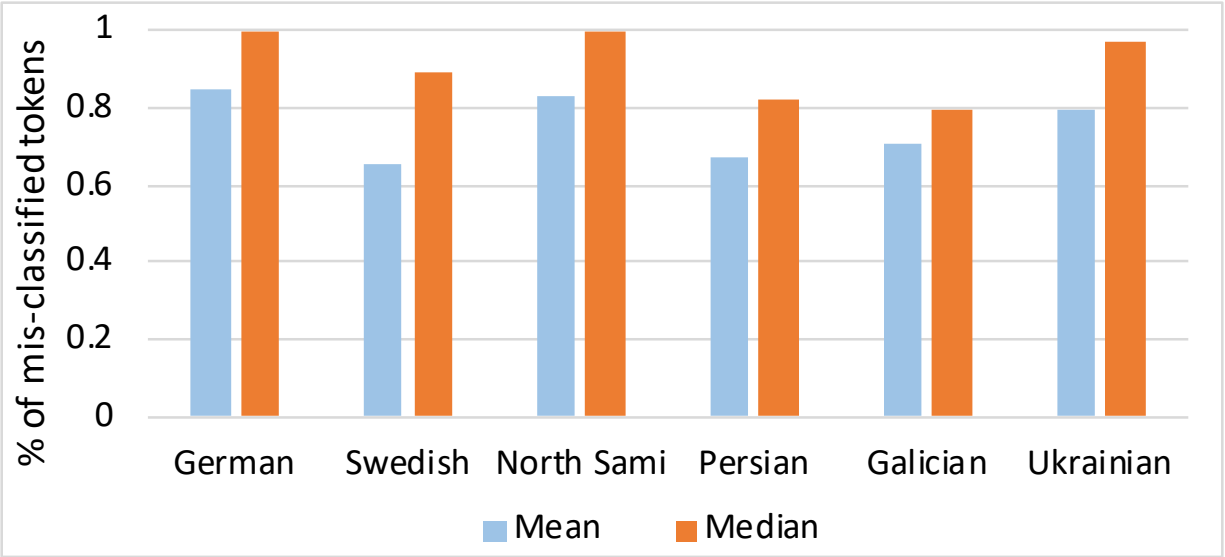


Figure 9.6: We report the mean and median of  $p$  over all the 50 token-tag pairs selected by the first AL iteration of CRAL. We see that across all languages majority of the token-tag pairs satisfy the criteria of using weighted representations with centroid for token selection.

EXPERIMENT SETTING	CVT	SCE	ACCURACY
EN + NO → EN	-	0.0190	95.53
	+	<b>0.0174</b>	<b>95.58</b>
EN + NO + DE-200 → DE	-	0.1658	69.90
	+	<b>0.1391</b>	<b>74.61</b>

Table 9.6: Evaluating the effect of CVT across two settings. EN: English, NO: Norwegian, DE-200: 200 German annotations. Left of ‘→’ are the pre-training languages and the on the right is the language on which this model is evaluated. Accuracy measures the POS model performance (higher is better) and SCE measures the model calibration (lower is better).

predictions based on the output class probability. Hence, the first 10% are placed in bin 1, the next 10% in bin 2, and so on. We conduct two ablation experiments to measure the effect of CVT. First, we train a joint POS model on English and Norwegian datasets using all available training data, and evaluate on the English test set. Second, we use this pre-trained model and fine-tune on 200 randomly sampled German data and evaluate on German test data. We train models with and without CVT, denoted by +/- in Table 9.6. We find that with CVT results both in higher accuracy as well as lower calibration error (SCE). This effect of CVT is much more pronounced in the second experiment, which presents a low-resource scenario and is common in an active learning framework.

## 9.6 Human Annotation Experiments

We conduct human annotation experiments for Griko, an endangered language, spoken by around 20 thousand people in southern Italy, in the Grecia Salentina area southeast of Lecce. The only available online Griko corpus, referred to as UoI (Lekakou, Marika and Baldissera, Valeria and Anastasopoulos,

	AL	ITERATION-0	ITERATION-1	ITERATION-2	ITERATION-3	IA Agr.	WD
Linguist-1	CRAL	52.93	<b>63.42</b> (10)	<b>69.07</b> (10)	65.16 (16)	0.58	0.281
	QBC	52.93	55.82 (15)	62.03 (17)	<b>66.51</b> (15)	0.68	0.243
	UNS	52.93	56.14 (15)	57.04 (15)	65.73 (11)	0.58	0.379
Linguist-2	CRAL	52.93	<b>61.24</b> (15)	<b>67.24</b> (20)	<b>67.05</b> (18)	0.70	0.346
	QBC	52.93	56.52 (20)	65.96 (20)	66.71 (17)	0.72	0.245
	UNS	52.93	55.45 (17)	58.80 (17)	65.73 (20)	0.70	0.363
Linguist-3 (Expert)	CRAL	52.93	<b>65.63</b>	<b>69.17</b>	<b>68.09</b>	-	0.159
	QBC	52.93	60.50	65.69	56.20	-	0.170
	UNS	52.93	58.51	64.26	65.93	-	0.125

Table 9.7: POS accuracy on Griko test set after each AL iteration, which consists of 50 token-level annotations. Number in the parentheses denotes the time in minutes required for annotation. IA AGR. reports the inter-annotator agreement against the expert linguist for the first iteration. WD is the Wasserstein distance between the selected tokens and the test distribution.

[Antonios, 2013](#)),<sup>7</sup> consists of 330 utterances by nine native speakers having POS annotations. Additionally, [Anastasopoulos et al. \(2018\)](#) collected, processed and released 114 stories, of which only the first 10 stories were annotated by experts and have gold-standard annotations.<sup>8</sup> We conduct human annotation experiments on the remaining un-annotated stories in order to compare the different AL methods.

### 9.6.1 Setup

We use two linguists, familiar with Modern Greek and somewhat with Griko. Using the same interface as [Chapter 8](#), tokens that need to be annotated are highlighted and presented with their surrounding context. The linguist then simply selects the appropriate POS tag for each highlighted token. Since we do not have gold annotations for these experiments, we obtain annotations from a third linguist who is more familiar with Griko. To familiarize the linguists with the annotation interface, a practice session was conducted in Modern Greek. We compare with UNS and QBC by conducting three AL iterations, where each iteration selects roughly 50 tokens for annotation. We use Modern Greek and Italian as the two related languages to train our initial POS model.<sup>9</sup> To further improve the model, we fine-tune on the UoI corpus which consists of 360 labeled sentences. We evaluate the AL performance on the 10 gold-labelled stories from UoI, of which the first two stories, comprising of 143 labeled sentences, are used as the validation set and the remaining 800 labeled sentences form the test set.

### 9.6.2 Results

[Table 9.7](#) records the result of the human annotation experiments. We find that our proposed method CRAL outperforms other methods in most cases. For Linguist-1, we observe a decrease in performance in Iteration-3 which we attribute to their poor annotation quality. This is also reflected in their low

<sup>7</sup><http://griko.project.uoi.gr>

<sup>8</sup><https://bitbucket.org/antonis/grikoresource/src/master/>

<sup>9</sup>With Italian being the dominant language in the region, code switching phenomena appear in the Griko corpora.

inter-annotator agreement scores (IA AGR) calculated against the expert annotator i.e. Linguist-3. We observe a slight decrease for other linguists which we hypothesize is due to domain mismatch between the annotated data and the test data. In fact, the test set stories and the unlabeled ones originate from different time periods spanning a century, which can lead to slight differences in orthography and usage. For instance, after three AL iterations, the token ‘i’ had been annotated as CONJ twice and DET once, whereas in the test data all instances of ‘i’ are annotated as DET.

We also compute the inter-annotator agreement at Iteration-1 with the expert (Linguist-3) (Table 9.7). We find that the agreement scores are lower than one would expect (c.f. the annotation test run on Modern Greek, for which we have gold annotations, yielded much higher inter-annotator agreement scores over 90%). The justification probably lies with our annotators having limited knowledge of Griko grammar, while our AL methods require annotations for ambiguous and “hard” tokens. However, this is a common scenario in language documentation where often linguists are required to annotate in a language they are not very familiar with, which makes this task even more challenging. We also recorded the annotation time needed by each linguist for each iteration in Table 9.7. Compared to the UNS method, the linguists annotated (avg.) 2.5 minutes faster using our proposed method which suggests that UNS tends to select harder data instances for annotation.

## 9.7 Conclusion

Extensive experimentation across six languages demonstrate the importance of considering confusion between the output tags for active learning. We test our approach under a true setting where we ask linguists to document POS information for an endangered language, Griko. Despite being unfamiliar with the language, our proposed method achieves performance gains over the other methods, in most iterations.



## Chapter 10

# Conclusion and Future Directions

In this thesis, we have looked at automated methods for extracting and visualizing different levels of language descriptions, from word-level insights such as POS tags and word usage, to language-level insights such as word order, agreement, and case marking. We designed these language descriptions in both human- and machine-readable formats and showed how they can be used for both human- and machine-centric applications. Since we want these language descriptions to answer questions about different languages, most of which are under-resourced, this thesis also discussed methods on how the language description extraction can be improved for such under-resourced languages. Below we summarize our main contributions.

### 10.1 Summary of Contributions

**AUTOLEX** This thesis presents AUTOLEX, an automatic framework that describes the process of extracting and visualizing language descriptions. Specifically, within this thesis, these language descriptions are intended to answer specific linguistic questions. In [Chapter 2](#), we show the different linguistic questions covered in this thesis, with their first-pass answers. Each linguistic question is posed as a classification task for which training data is constructed from the raw text of the language of interest. In [Chapter 3](#), [Chapter 4](#), and [Chapter 5](#), we show how to adapt this framework to answer different questions about word order, agreement, case marking, and word usage. Where possible, we verify the extracted descriptions with the help of language experts, but since manual verification is not always possible, as part of the framework, we provide automated methods for evaluation. Potentially, for a new question or language, a user can follow the above framework to similarly extract, visualize, and evaluate the first-pass answers. The descriptions we extract are hosted on <https://autolex.co>.

**AUTOLEX Applications** The primary motivation behind AUTOLEX is to provide an interface to explore a language. To understand how practically usable such an interface is in the real world, in [Chapter 5](#) and [Chapter 6](#) we apply AUTOLEX to language education. We find that, for both learners and teachers, AUTOLEX is of utility. Similarly, in [Chapter 4](#) we find that even language experts find AUTOLEX useful, as it is able to identify interesting linguistic behaviors which the experts were not aware of. Given that the same language often varies considerably across different settings (e.g. formality, regions, communities, etc.), we can potentially apply AUTOLEX to text collected from these different settings and compare how

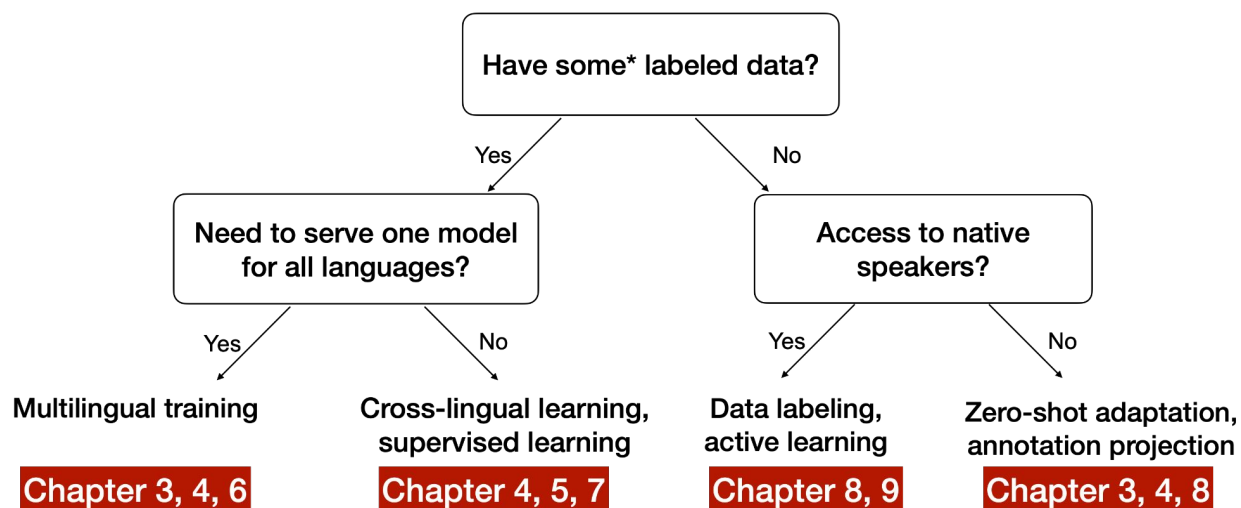


Figure 10.1: Summary of the different approaches to support a new language. Inspired from Graham Neubig’s course CS 11-747 [slides](#).

the salient grammar properties vary. In addition to such human-centric applications, we extract language descriptions in a machine-readable format, which have been used by researchers (Pratapa et al., 2021b; Yin et al., 2021) to evaluate and inform natural language generation systems in various languages.

**NLP for under-resourced languages** Despite the many applications of language descriptions, we do note that the quality of the extracted descriptions depends highly on the underlying NLP tools (e.g. parsers, word aligners, translators, etc.) used in the process. In Chapter 7, Chapter 8, Chapter 9, we discuss methods to improve the quality of such NLP tools, with a special focus on under-resourced languages, which do not have sufficient labeled data for training these models. Figure 10.1 outlines the different approaches that can be taken to add support for a new language. For example, if we have some labeled data in the required language and are not limited by computational resources, in that case we can use supervised learning with cross-lingual transfer learning, where we leverage resources from related high-resource languages. In Chapter 7, we presented one such approach that leverages word embeddings to learn from related high-resource languages, and found that using linguistically-motivated word embeddings can even help models to learn from languages which do not share orthography. Furthermore, if our focus is on a specific language, in that case, this approach is recommended, as research (Conneau and Lample, 2019; Siddhant et al., 2020) has found that adding multiple languages during training is of little help, compared to adding only a few selected related languages. However, if we are computationally limited and do not want to train one model for each language, training one model for all languages offers many advantages. Along with computational benefits, multilingual models can enable applying a model on a new language previously unseen by the model and can produce decent outputs in lieu of the fact that languages are related and can often benefit from each other. In Chapter 3, Chapter 4, we have therefore used a multilingual parser as a starting point for even zero-resource settings, where we do not have any labeled data. But what if we have access to some native speakers of the required language? In that case, we could request them to label a subset of data to train a high-quality model. However, labeling data manually is not only time-consuming but also requires effort, but more importantly, as we discussed

in the previous chapters, for many tasks and languages, finding such experts itself might be challenging. Therefore, we explore efficient data labeling techniques (*active learning*) which help us to automatically select such a subset of data for manual annotation that can result in the best model performance with that limited data. Specifically, we propose to combine cross-lingual approaches with active learning to make use of existing resources (both in the required language and related languages) as well as by collecting new data in the required language wherever possible, as shown in [Chapter 8](#) and [Chapter 9](#). In [Chapter 8](#), we found how the addition of linguistic properties (e.g. graphemes, phonemes, morphemes) was effective in learning better cross-lingual word representations which in turn led to an improved downstream performance on the under-resourced languages. Although, these findings were shown over models that use simpler architectures (e.g. LSTMs ([Hochreiter and Schmidhuber, 1997](#))), we believe these insights are also relevant in the current context which has seen more complex architectures such as transformers ([Vaswani et al., 2017b](#)) that are capable of modeling and discovering even long-term and more complex patterns from the data and have shown tremendous performance gains across several downstream tasks ([Devlin et al., 2019](#); [Conneau and Lample, 2019](#); [Shoeybi et al., 2019](#); [Hu et al., 2020](#); [Siddhant et al., 2020](#)). In fact, recent efforts have shown the benefits of using such explicit linguistic signals, even in these modern architectures, for improving performance on under-resourced languages – for example, [Leong and Whitenack \(2022\)](#) convert both text and audio input for a language into a single phonetic representation, and train a model over this common representation. This allows them to leverage resources from different modalities (e.g. text, speech) which is especially useful for languages which do not have either resource in large quantities. Similarly, [Nzeyimana and Niyongabo Rubungo \(2022\)](#) explicitly incorporate morphological signals by training a language model on meaningful subword units. Typically, language models use statistical tokenization techniques (e.g. BPE ([Sennrich et al., 2016](#); [Provilkov et al., 2020](#)), WordPiece ([Schuster and Nakajima, 2012](#)), etc.) to segment text into smaller units which are meaningful in capturing both semantics and syntactic information, even across languages. However, works ([Klein and Tsarfaty, 2020](#); [Wang et al., 2021](#)) have shown that often such statistical approaches often lead to incorrect text segmentation, especially for under-resourced languages, leading to poor downstream performance. [Nzeyimana and Niyongabo Rubungo \(2022\)](#) use a morphological analyzer to segment words into its stem and affixes, which is used to learn morphologically-aware contextual representations. They find, doing so, leads to an improved downstream performance for NER and news classification tasks for Kinyarwanda language, which is under-resourced. Such efforts highlight the importance of leveraging morphological and phonological properties even in more recent models, especially for under-resourced languages. Similarly, the active learning strategies we developed in [Chapter 8](#) and [Chapter 9](#) are also relevant today, as these are efficient data selection strategies which are not dependent on underlying model architecture. As we saw in [Chapter 9](#), even when we changed the underlying POS tagger, the proposed query selection strategy outperformed the existing baseline strategies, suggesting the utility of proposed strategies beyond the model architectures studied in this thesis.

## 10.2 Future Directions

**NLP for Under-resourced languages** In [Chapter 4](#) and [Chapter 5](#), we briefly discussed how extracting the language descriptions in machine-readable formats can help with the model evaluation and design. Similarly, these language descriptions could also be used to create synthetic training data, espe-

cially to help with the under-resourced languages. Recently Wang et al. (2022) demonstrated the utility of word dictionaries or bilingual lexicons to create synthetic training data which resulted in performance gains for several under-resourced languages. Prior works (Upadhyay et al., 2016; Chaudhary et al., 2020; Dufter, 2021) have also shown how even naively using the lexicons, i.e. by substituting a word in a sentence with its cross-lingual lexicon entry, to create synthetic code-mixed sentences can similarly help in a performance improvements. However, these works do note that although such a substitution provides useful signals to the models, the generated synthetic data often is ungrammatical as it could violate the substituted languages' word order or that the substituted word may not be appropriately inflected and so on. A next step would be to use our generated grammar descriptions to better inform this substitution by producing grammatically correct synthetic data, which will more likely produce even more gains for the under-resourced languages. This is even relevant to the mid-to-high resource settings, such as the code-mixing setting (Bokamba, 1989; Muysken et al., 2000) wherein people jump from one language to another. Code-mixing is interesting not just from a research perspective but also has practical importance given its high prevalence in the society (Ndebele, 2012; Kachru, 1978; Derrick, 2015). What makes it interesting is that there are several debates surrounding the grammar underlying code-mixing, for example, is there a specific ordering on how the languages can combine (Poplack, 2001; Johns et al., 2019). Possibly by automatically extracting the grammar patterns of code-mixed data observed in different contexts (e.g. social-media, movies, etc.), it can help us understand how code-mixing works and subsequently improve the NLP models.

**Interactive Environment** In addition to improving the underlying NLP tools, which will improve the extraction of language descriptions for potentially \*all\* languages, a possible next step would be to make AUTOLEX interactive. This will allow language experts to make edits to the rules, both for correcting any incorrect parses, and also to fill-in gaps in the descriptions. For example, a popular feedback from several teachers involved in the user study presented in Chapter 6 was that some examples were too advanced for their learners, instead they would prefer the examples to be presented in an incremental fashion, where first the sentence is introduced with its basic elements (e.g. subject, verb, object) and then step-by-step the learner is introduced to additional elements in the same example (e.g. addition of a prepositional phrase, adjectives, and so on). If a teacher is provided with an edit access, they could modify the examples accordingly. Additionally, we can leverage active learning principles, where the expert assisted by an automatic model can improve the underlying model (e.g. syntactic parsing), where based on the experts' input the model is re-trained to extract rules using the improved analyzes.

**Cultural Inclusivity** In Chapter 5, we saw how there are semantic divergences between different languages. Capturing these divergences is critical to building language technologies which are inclusive, for example, if we consider the application of machine translation, we want the machine to translate culturally appropriate, sensitive, and inclusive translations. Leaving a handful of languages such as English where there are several datasets/benchmarks available for training/evaluating NLP models that have been curated manually, which cover a variety of domains, most languages do not enjoy such luxury. Many datasets have, in fact, simply been translated from their English counterparts (e.g. XNLI Conneau et al. (2018)), and does not cover language-specific or cultural-specific nuances. However, more recent initiatives such as MaRVL (Liu et al., 2021) have taken a different approach, in which native speakers are



encouraged to drive the data curation process, with the goal of capturing more languages and cultures. [Liu et al. \(2021\)](#)'s design of such annotation is heavily influenced by existing ethnographic studies, which highlight the need to combine knowledge from fields such as linguistics, anthropology, and cognitive science. Similarly, we can take advantage of some of the approaches we presented in this thesis to predict linguistic insights to bring to surface culturally relevant phenomena. For example, in [Chapter 5](#) we automatically identified concepts that do not have exact one-to-one equivalence across languages. We would also want to identify things that do not exist in another culture or society, e.g. 'ushta' in Marathi refers to a thing that has been sipped/eaten/used/touched by another person with their mouth, in English there is no equivalent concept. In order to build models which cater to all people of the world, we need to include a) more languages, b) different viewpoints, and c) culture-inclusive topics. This is important to ensure that the tools, which are often based on such datasets/models, are also inclusive of the different cultures and languages. [Hershcovich et al. \(2022\)](#) have concretely outlined the different cultural dimensions that NLP researchers must keep in mind when designing technologies. Inclusion of these cultural aspects is equally important for language education as preserving and promoting culture is often one of its primary objectives. A next step in that direction would be applying AUTOLEX on a corpus carefully selected by the educators themselves to ensure that the derived linguistic insights are culturally appropriate, representative and unbiased towards any minorities.



# Bibliography

- Samira Abnar and Willem Zuidema. 2020. [Quantifying attention flow in transformers](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, Online. Association for Computational Linguistics. [5.3.1](#)
- Katherine Ackerley. 2017. Effects of corpus-based instruction on phraseology in learner english. *Language Learning & Technology*, 21(3):195–216. [6.1](#)
- J. Aissen. 1997. On the syntax of obviation. *Language*. [4.2.2](#)
- Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A Smith. 2016a. Many languages, one parser. In *TACL*, volume 4, pages 431–444. MIT Press. [8.1](#)
- Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A Smith. 2016b. Massively multilingual word embeddings. In *arXiv*. [7.1](#), [7.4](#)
- Antonios Anastasopoulos, Marika Lekakou, Josep Quer, Eleni Zimianiti, Justin DeBenedetto, and David Chiang. 2018. Part-of-Speech Tagging on an Endangered Language: a Parallel Griko-Italian Resource. In *International Conference on Computational Linguistics*. [9.1](#), [9.6](#)
- Ankita and K. A. Abdul Nazeer. 2018. Part-of-Speech Tagging and Named Entity Recognition using Improved Hidden Markov Model and Bloom Filter. In *GUCON*. [9.1](#)
- Chahta Anumpa and Tosholi Himona. 2016. New choctaw dictionary. In *The Choctaw Nation of Oklahoma Dictionary Committee*. [2.1.3](#)
- Shlomo Argamon-Engelson and Ido Dagan. 1999. Committee-based sample selection for probabilistic classifiers. In *Artificial Intelligence Research*. [8.2](#)
- Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *ACL*. [8.4.2](#)
- Les E Atlas, David A Cohn, and Richard E Ladner. 1990. Training connectionist networks with queries and selective sampling. In *NeurIPS*. [8.2](#)
- Oded Avraham and Yoav Goldberg. 2017. The interplay of semantics and morphology in word embeddings. In *CoRR*. [7.2.2](#), [7.3.1](#), [7.4](#)
- Kirk Baker and Chris Brew. 2010. *Multilingual animacy classification by sparse logistic regression*. Ohio State University. Department of Linguistics. [4.2.2](#)
- Mark C Baker and Nadya Vinokurova. 2010. Two modalities of case assignment: Case in sakha. *Natural Language & Linguistic Theory*, 28(3):593–642. [4.2.1](#)

- Maria-Florina Balcan, Andrei Broder, and Tong Zhang. 2007. Margin based active learning. In *International Conference on Computational Learning Theory*. 8.3
- Elif Bamyacı and Klaus von Heusinger. 2016. Animacy effects on differential object marking in turkish. *Poster presentation at linguistic evidence*. 4.4.1
- Michael Barlow and Charles A Ferguson. 1988. *Agreement in natural language*. Center for the Study of Language (CSLI). 3.1, 3.2
- Lisa Barrow, Lisa Markman, and Cecilia Elena Rouse. 2009. Technology’s edge: The educational benefits of computer-aided instruction. *American Economic Journal: Economic Policy*, 1(1):52–74. 6.1
- Kedar Bellare and Andrew McCallum. 2007. Learning extractors from unlabeled text using relevant databases. In *International Workshop on Information Integration on the Web*. 16, 9.3.2
- Emily M. Bender, Dan Flickinger, and Stephan Oepen. 2002. [The grammar matrix: An open-source starter-kit for the rapid development of cross-linguistically consistent broad-coverage precision grammars](#). In *COLING-02: Grammar Engineering and Evaluation*. 2.1.2
- Emily M. Bender, Michael Wayne Goodman, Joshua Crowgey, and Fei Xia. 2013. [Towards creating precision grammars from interlinear glossed text: Inferring large-scale typological properties](#). In *Proceedings of the 7th Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 74–83, Sofia, Bulgaria. Association for Computational Linguistics. 2.1.2
- Gena R Bennett. 2010. *Using corpora in the language learning classroom: Corpus linguistics for teachers*, volume 10. University of Michigan Press Ann Arbor, MI. 6.1
- Christian Bentz, Tatyana Ruzsics, Alexander Kopenig, and Tanja Samardžić. 2016. A Comparison Between Morphological Complexity Measures: Typological Data vs. Language Corpora. In *CLALC*. 3.4.1, 3.4.1
- Michele Bevilacqua and Roberto Navigli. 2020. [Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864, Online. Association for Computational Linguistics. 5.3
- Akash Bharadwaj, David Mortensen, Chris Dyer, and Jaime Carbonell. 2016. Phonologically aware neural model for named entity recognition in low resource transfer settings. In *EMNLP*. 7.1, 7.2.2, 7.4, ??
- Irshad Ahmad Bhat, Vandan Mujadia, Aniruddha Tammewar, Riyaz Ahmad Bhat, and Manish Shrivastava. 2015. [Iiit-h system submission for fire2014 shared task on transliterated search](#). In *Proceedings of the Forum for Information Retrieval Evaluation, FIRE ’14*, pages 48–53, New York, NY, USA. ACM. 6.2.2
- Riyaz Ahmad Bhat, Rajesh Bhatt, Annahita Farudi, Prescott Klassen, Bhuvana Narasimhan, Martha Palmer, Owen Rambow, Dipti Misra Sharma, Ashwini Vaidya, Sri Ramagurumurthy Vishnu, et al. 2017. The hindi/urdu treebank project. In *Handbook of linguistic annotation*, pages 659–697. Springer. 6.3
- Johannes Bjerva, Elizabeth Salesky, Sabrina J. Mielke, Aditi Chaudhary, Giuseppe G. A. Celano, Edoardo Maria Ponti, Ekaterina Vylomova, Ryan Cotterell, and Isabelle Augenstein. 2020. [SIGTYP 2020 shared task: Prediction of typological features](#). In *Proceedings of the Second Workshop on Compu-*

- tational Research in Linguistic Typology*, pages 1–11, Online. Association for Computational Linguistics. [2.1.1](#)
- Barry Blake. 1994. *Case*. Cambridge University Press, Cambridge. [4.1](#)
- Barry J Blake. 2009. History of the research on case. In *The Oxford handbook of case*. [2.2](#), [4.2.2](#)
- J. Blake. 2001. [Morphological case in linguistics](#). In Neil J. Smelser and Paul B. Baltes, editors, *International Encyclopedia of the Social Behavioral Sciences*, pages 10043–10047. Pergamon, Oxford. [4.1](#)
- Jelke Bloem and Gosse Bouma. 2013. Automatic animacy classification for dutch. *Computational Linguistics in the Netherlands Journal*. [4.2.2](#)
- Bernd Bohnet, Ryan McDonald, Gonalo Simões, Daniel Andor, Emily Pitler, and Joshua Maynez. 2018. Morphosyntactic Tagging with a meta-BiLSTM Model over Context Sensitive Token Encodings. In *ACL*. [9.1](#)
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2016. Enriching word vectors with subword information. In *arXiv*. [4.2.2](#), [7.1](#), [7.2.2](#), [7.4](#), [7.4](#), [7.4](#)
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. [Enriching word vectors with subword information](#). *Transactions of the Association for Computational Linguistics*, 5:135–146. [4.3](#)
- Eyamba G Bokamba. 1989. Are there syntactic constraints on code-mixing? *World Englishes*, 8(3):277–292. [10.2](#)
- Punyapa Boontam and Supakorn Phoocharoensil. 2016. *Effectiveness of english preposition learning through data-driven learning (DDL)*. Ph.D. thesis, THAMMASAT UNIVERSITY. [6.1](#)
- Robert D Borsley and Ian Roberts. 2005. The syntax of the celtic languages: a comparative perspective. In *Cambridge University Press*. [3.5](#)
- Jan Botha and Phil Blunsom. 2014. Compositional morphology for word representations and language modelling. In *ICML*. [7.2.2](#)
- M Bowerman and S Choi. 2001. Shaping meanings for Language: Universal and Language-specific in the Learning of spatial semantic categories. *Language acquisition and conceptual development*. [5.1](#)
- Stevo Bozinovski and Ante Fulgosi. 1976. The influence of pattern similarity and transfer learning upon training of a base perceptron b2. In *Proceedings of Symposium Informatica*, volume 3, pages 121–126. [7.1](#)
- Robert L Bradshaw. 2007. *Fuyug grammar sketch*. 53. SIL-PNG Academic Publications. [2.2](#)
- Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. 1984. Classification and regression trees. In *CRC press*. [3.2.3](#)
- Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake VanderPlas, Arnaud Joly, Brian Holt, and Gaël Varoquaux. 2013. API design for machine learning software: experiences from the scikit-learn project. In *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pages 108–122. [3.3](#)
- Lynnika Butler and Heather Van Volkinburg. 2007. Fieldworks language explorer (flex). In *Technology*

Review. 2.1.3

- Miriam Butt, Helge Dyvik, Tracy Holloway King, Hiroshi Masuichi, and Christian Rohrer. 2002. [The parallel grammar project](#). In *COLING-02: Grammar Engineering and Evaluation*. 2.1.2
- Marine Carpuat and Dekai Wu. 2007a. How phrase sense disambiguation outperforms word sense disambiguation for statistical machine translation. *Proceedings of TMI*, pages 43–52. 5.2.2
- Marine Carpuat and Dekai Wu. 2007b. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 61–72. 5.2.2
- Serkan Celik and Metin Elkatmis. 2013. The effect of corpus assisted language teaching on the learners’ proper use of punctuation marks. *Educational Sciences: Theory and Practice*, 13(2):1090–1094. 6.1
- Angela Chambers. 2010. What is data-driven learning? In *The Routledge handbook of corpus linguistics*, pages 345–358. Routledge. 6.1
- Tun-pei Chan and Hsien-Chin Liou. 2005. Effects of web-based concordancing instruction on efl students’ learning of verb–noun collocations. *Computer assisted language learning*, 18(3):231–251. 6.1
- Aditi Chaudhary, Karthik Raman, Krishna Srinivasan, and Jiecao Chen. 2020. [Dict-mlm: Improved multilingual pre-training using bilingual dictionaries](#). In *arXiv preprint arXiv:2010.12566*. 10.2
- Aditi Chaudhary, Elizabeth Salesky, Gayatri Bhat, David R. Mortensen, Jaime Carbonell, and Yulia Tsvetkov. 2019. CMU-01 at the SIGMORPHON 2019 shared task on crosslinguality and context in morphology. In *ACL-SIGMORPHON*. 8.6, 23
- Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794. 4.3
- Noam Chomsky. 1993. *Lectures on government and binding: The Pisa lectures*. Walter de Gruyter. 4.1
- Noam Chomsky. 2000. Minimalist inquiries: The framework (mitopl 15). *Step by step: Essays on minimalist syntax in honor of Howard Lasnik*, pages 89–155. 4.2.1
- H. Clahsen and D. Hansen. 1993. The missing agreement account of specific language impairment: evidence from therapy experiments. In *Cognition*. 3.1
- Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-Supervised Sequence Modeling with Cross-View Training. In *EMNLP*. 2, 9.3.2
- T Cobb. 2002. Web complet lexical tutor/vocabulary profile. Retrieved on September, 4:2011. 6.1
- Jacob Cohen. 2013. *Statistical Power Analysis for the Behavioral Sciences*. Academic press. 3
- Bernard Comrie. 1984. Reflections on Verb Agreement in Hindi and Related Languages. 3.4.1
- Alexis Conneau and Guillaume Lample. 2019. Crosslingual language model pretraining. In *NeurIPS*. 10.1
- Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. In *arXiv*. 7.1, 8.4.2, 8.4.2
- Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating Cross-lingual Sentence Representations. In *EMNLP*. 1, 10.2

- G. G. Corbett. 1979. The Agreement Hierarchy. *Journal of Linguistics*. 3.1
- Greville G Corbett. 2003. Agreement: Terms and Boundaries. In *Texas Linguistic Society Conference*. 3.2, 4.2.2
- Greville G Corbett. 2009. Agreement. In *Die slavischen Sprachen/The Slavic Languages*. 2.2, 2.2.2, 3.2.1
- Greville G Corbett. 2017. Morphology and Agreement. In *The handbook of morphology*. 3.2
- Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297. 5.2.3
- Ryan Cotterell and Georg Heigold. 2017. Cross-Lingual Character-Level Neural Morphological Tagging. In *EMNLP*. 7.1, 9.1, 9.3.2
- Ryan Cotterell and Hinrich Schütze. 2015. Morphological word-embeddings. In *NAACL-HLT*. 7.2.2
- Harald Cramér. 1946. *Mathematical Methods of Statistics*. In *Princeton U. Press, Princeton*. 3.2.4
- Dina B Crockett. 1976. *Agreement in Contemporary Standard Russian*. Slavica Publishers Inc. 3.4.1
- Peter Crosthwaite. 2020. Taking ddl online: Designing, implementing and evaluating a spoc on data-driven learning for tertiary l2 writing. *Australian Review of Applied Linguistics*, 43(2):169–195. 6.1
- Aron Culotta and Andrew McCallum. 2004. Confidence estimation for information extraction. In *NAACL-HLT*. 8.5
- Aron Culotta and Andrew McCallum. 2005. Reducing Labeling Effort for Structured Prediction Tasks. In *AAAI*. 8.3.2, 8.4.2, 8.5
- Ido Dagan and Sean P Engelson. 1995. Committee-based sampling for training probabilistic classifiers. In *Machine Learning Proceedings*. 8.3, 8.3.3, 8.3.3, 9.1
- Osten Dahl and Kari Fraurud. 1996. Animacy in grammar and discourse. *Pragmatics and Beyond New Series*. 4.2.2
- Sanjoy Dasgupta and Daniel Hsu. 2008. Hierarchical sampling for active learning. In *Proceedings of the 25th international conference on Machine learning*, pages 208–215. 8.3
- Niladri Sekhar Dash. 2008. *Corpus linguistics: An introduction*. Pearson Education India. 6.2.2
- Mark Davies. 2008. The corpus of contemporary american english (coca): 560 million words, 1990-present. 6.1
- Roshawnda A Derrick. 2015. *Code-switching, code-mixing and radical bilingualism in US Latino texts*. Wayne State University. 10.2
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*. 1, 5.3.1, 7, 7.1, 7.6, 10.1
- Gina Doggett. 1986. Eight approaches to language teaching. *ERIC*. 6.2.2
- Pinar Donmez, Jaime G Carbonell, and Paul N Bennett. 2007. Dual strategy active learning. In *European Conference on Machine Learning*, pages 116–127. Springer. 8.3
- Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *EACL*. 5.3, 7

- Matthew S. Dryer. 2007. Word order. In *Language Typology and Syntactic Description*. 4.1
- Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig. 1.1, 2.1.1, 4.2.1
- Xiangtao Du, Muhammad Afzaal, and Hind Al Fadda. 2022. Collocation use in efl learners’ writing across multiple language proficiencies: A corpus-driven study. *Frontiers in Psychology*, 13:752134–752134. 6.1
- Philipp Dufter. 2021. *Distributed representations for multilingual language processing*. Ph.D. thesis, lmu. 10.2
- Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *EMNLP*. 7.3.2
- Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A Smith. 2016. Recurrent Neural Network Grammars. In *NAACL*. 1
- Meng Fang and Trevor Cohn. 2017. Model Transfer for Tagging Low-Resource Languages using a Bilingual Dictionary. In *ACL*. 8.3, 8.3.2, 16, 9.1, 9.2, 23
- Fiona Farr. 2010. How can corpora be used in teacher education? In *The Routledge handbook of corpus linguistics*, pages 620–632. Routledge. 6.1
- Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *EACL*. 7.1
- Xiaocheng Feng, Zhangyin Feng, Wanlong Zhao, Nan Zou, Bing Qin, and Ting Liu. 2019. Improved neural machine translation with pos-tagging through joint decoding. In *AICON*. 9.1
- Lynne Flowerdew. 2011. *Corpora and language education*. Springer. 6.1
- Yoav Freund, H Sebastian Seung, Eli Shamir, and Naftali Tishby. 1997. Selective sampling using the query by committee algorithm. In *Machine learning*. 8.3.3
- Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232. 6
- Ismael Garcia-Varea, Franz Josef Och, Hermann Ney, and Francisco Casacuberta. 2001. Refined lexicon models for statistical machine translation using a maximum entropy approach. In *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, pages 204–211. 5.2.2
- Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2018. SUD or surface-syntactic Universal Dependencies: An annotation scheme near-isomorphic to UD. In *UDW*. 2.2, 3.3, 7
- Kim Gerdes, Bruno Guillaume, Sylvain Kahane, and Guy Perrier. 2019. Improving surface-syntactic universal dependencies (SUD): MWEs and deep syntactic features. In *TLT, SyntaxFest*. 2.2, 3.3, 3.3, 4.3
- David Gil. 2021. Tense–aspect–mood marking, language-family size and the evolution of predication. *Philosophical Transactions of the Royal Society B*, 376(1824):20200194. 3.4.2
- R. Godwin-Jones. 2018. *Contextualized vocabulary learning*. Language Learning and Technology. 5.1
- Yoav Goldberg. 2017. Neural network methods for natural language processing. *Synthesis lectures on human language technologies*, 10(1):1–309. 1
- Dirk Goldhahn, Thomas Eckart, and Uwe Quasthoff. 2012. Building large monolingual dictionaries at



- the leipzig corpora collection: From 100 to 200 languages. In *LREC*. [3.3](#), [3.5](#)
- Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. *BilBOWA: Fast Bilingual Distributed Representations without Word Alignments*. *ICML*. [7.3.2](#)
- Stephan Gouws and Anders Søgaard. 2015. Simple task-specific bilingual word embeddings. In *NAACL-HLT*. [7.1](#)
- Peter JM Groot. 2000. Computer assisted second language vocabulary acquisition. *Language learning and technology*. [5.1](#), [5.3.2](#)
- Masato Hagiwara, Joshua Tanner, and Keisuke Sakaguchi. 2021. Grammartagger: A multilingual, minimally-supervised grammar profiler for language education. *arXiv*. [2.1.3](#)
- Ken Hale, Michael Krauss, Lucille J Watahomigie, Akira Y Yamamoto, Colette Craig, LaVerne Masayesva Jeanne, and Nora C England. 1992. Endangered Languages. *Language*. [1](#)
- Morris Halle, Alec Marantz, Kenneth Hale, and Samuel Jay Keyser. 1993. Distributed morphology and the pieces of inflection. *The view from Building 20*. [4.1](#)
- Harald Hammarström. 2015. "ethnologue" 16/17/18th editions: A comprehensive review. [1](#), [2.1.1](#), [7.1](#)
- Harald Hammarström, Robert Forkel, and Martin Haspelmath. 2018. Glottolog 3.3. Max Planck Institute for the Science of Human History. Jena. [2.1.1](#)
- Zellig S Harris. 1954. Distributional structure. In *Word*. [1](#), [3](#), [7.2.1](#)
- Jean Hausser and Korbinian Strimmer. 2009. Entropy inference and the james-stein estimator, with application to nonlinear gene association networks. In *JMLR*. [3.4.1](#)
- Lars Hellan. 2010. [From descriptive annotation to grammar specification](#). In *Proceedings of the Fourth Linguistic Annotation Workshop*, pages 172–176, Uppsala, Sweden. Association for Computational Linguistics. [2.1.2](#)
- Daniel Hershcovich, Stella Frank, Heather Lent, Miryam de Lhoneux, Mostafa Abdou, Stephanie Brandl, Emanuele Bugliarello, Laura Cabello Piqueras, Ilias Chalkidis, Ruixiang Cui, et al. 2022. Challenges and strategies in cross-cultural nlp. *arXiv preprint arXiv:2203.10020*. [10.2](#)
- Nikolaus P Himmelmann. 1998. Documentary and descriptive linguistics. *Linguistics*. [\(document\)](#), [1](#)
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. In *Neural Computations*. [10.1](#)
- Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python. *Zenodo*. [5.3](#)
- Kristen Howell, Emily M. Bender, Michel Lockwood, Fei Xia, and Olga Zamaraeva. 2017. [Inferring case systems from IGT: Enriching the enrichment](#). In *Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 67–75, Honolulu. Association for Computational Linguistics. [2.1.2](#)
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalization. In *ICML*. [1](#), [10.1](#)
- Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional LSTM-CRF models for Sequence Tagging. In

*arXiv*. [9.1](#)

- Jan H Hulstijn, Merel Hollander, and Tine Greidanus. 1996. Incidental Vocabulary Learning by Advanced Foreign Language Students: The Influence of Marginal Glosses, Dictionary use, and Reoccurrence of Unknown Words. *The Modern Language Journal*. [5](#)
- Itisree Jena, Riyaz Ahmad Bhat, Sambhav Jain, and Dipti Misra Sharma. 2013. Animacy annotation in the Hindi treebank. In *ACL-LAW-ID*. [4.2.2](#)
- VR Jeyasala. 2014. A prelude to practice: Interactive activities for effective communication in english. *Alternative pedagogies in the English language & communication classroom*, pages 164–170. [6.2.2](#)
- N. Jiang. 2002. Form–meaning mapping in Vocabulary Acquisition in a Second Language. *Studies in Second Language Acquisition*. [5.1](#)
- TF John. 1991. Should you be persuaded: Two examples of data-driven learning. *Johns TF, King P. Classroom Conlcor-Idanlcing. Birmingham: ELR*. [6.1](#)
- Michael A Johns, Jorge R Valdés Kroff, and Paola E Dussias. 2019. Mixing things up: How blocking and mixing affect the processing of codemixed sentences. *International Journal of Bilingualism*, 23(2):584–611. [10.2](#)
- Keith Johnson and Christopher Brumfit. 1979. *The communicative approach to language teaching*. Oxford University Press. [6.2.2](#)
- Christian Jones and Daniel Waller. 2015. *Corpus linguistics for grammar: A guide for research*. Routledge. [1.1](#)
- Braj B Kachru. 1978. Code-mixing as a communicative strategy in india in international dimensions of bilingual education. *Georgetown University Round Table on Languages and Linguistics Washington, DC*, pages 107–124. [10.2](#)
- Liu Kanglong and Muhammad Afzaal. 2020. Lexical bundles: A corpus-driven investigation of academic writing teaching to esl undergraduates. *International Journal of Emerging Technologies*, 11:476–482. [6.1](#)
- Ronald M Kaplan, Joan Bresnan, et al. 1981. *Lexical-functional grammar: A formal system for grammatical representation*. Citeseer. [2.1.2](#), [4.2.2](#)
- Anisia Katinskaia, Javad Nouri, and Roman Yangarber. 2017. Revita: a system for language learning and supporting endangered languages. In *NLP4CALL LA*. [5.1](#)
- E. Keenan. 1974. The Functional Principle: Generalizing the Notion ‘subject of’. *Chicago Linguistic Society*. [3.1](#)
- Tracy Holloway King, Martin Forst, Jonas Kuhn, and Miriam Butt. 2005. The feature space in parallel grammar writing. *Research on Language and Computation*, 3(2-3):139–163. [2.1.2](#)
- Eliyahu Kiperwasser and Yoav Goldberg. 2016. [Simple and accurate dependency parsing using bidirectional LSTM feature representations](#). *Transactions of the Association for Computational Linguistics*, 4:313–327. [1](#)
- James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. Overcoming

- catastrophic forgetting in neural networks. In *NAS*. 7.5.1
- Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian J. Mielke, Arya McCarthy, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2018. UniMorph 2.0: Universal Morphology. In *LREC*. 9.4
- Seppo Kittilä, Katja Västi, and Jussi Ylikoski. 2011. *Introduction to case, animacy and semantic roles*. John Benjamins Publishing. 4.2.2
- Stav Klein and Reut Tsarfaty. 2020. [Getting the ##life out of living: How adequate are word-pieces for modelling complex morphology?](#) In *Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology*, pages 204–209, Online. Association for Computational Linguistics. 10.1
- Jan-Christoph Klie, Michael Bugert, Beto Boulosa, Richard Eckart de Castilho, and Iryna Gurevych. 2018. The INCEpTION Platform: Machine-Assisted and Knowledge-Oriented Interactive Annotation. In *COLING: System Demonstrations*. 2.1.3
- Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Machine Translation Summit*, Phuket, Thailand. 5.3
- Philipp Koehn. 2020. *Neural Machine Translation*. Cambridge University Press. 1
- Dan Kondratyuk and Milan Straka. 2019. 75 languages, 1 model: Parsing universal dependencies universally. In *EMNLP*. 1, 2.1.1, 3.5, 3.5, 4.2.3, 4.5, 4.5, 6.3, 7
- Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Alahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, et al. 2021. Quality at a glance: An audit of web-crawled multilingual datasets. *arXiv preprint arXiv:2103.12028*. 6.2.1
- Artur Kulmizev, Miryam de Lhoneux, Johannes Gontrum, Elena Fano, and Joakim Nivre. 2019. [Deep contextualized word embeddings in transition-based and graph-based dependency parsing - a tale of two parsers revisited](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2755–2768, Hong Kong, China. Association for Computational Linguistics. 1, 2.1.1
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL-HLT*. 9.3.2
- Chuen-Yi Lee and Hsien-Chin Liou. 2003. A study of using web concordancing for english vocabulary learning in a taiwanese high school context. *English teaching and learning*, 27(3):35–56. 6.1
- Els Lefever and Veronique Hoste. 2010. [SemEval-2010 task 3: Cross-lingual word sense disambiguation](#). In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 15–20, Uppsala, Sweden. Association for Computational Linguistics. 5.2.1
- Els Lefever and Véronique Hoste. 2013. SemEval-2013 task 10: Cross-lingual word sense disambiguation. In *SemEval*. 5.3.1
- Julie Anne Legate. 2008. Morphological and abstract case. *Linguistic inquiry*. 4.1
- Christian Lehmann, C. Lehmann. 1968. Universal and Typological aspects of Agreement. *Introduction to Theoretical Linguistics*. 3.1

- Lekakou, Marika and Baldissera, Valeria and Anastasopoulos, Antonios. 2013. Documentation and Analysis of an Endangered Language: aspects of the grammar of Griko. [9.6](#)
- Agnieszka Leńko-Szymańska. 2017. Training teachers in data driven learning: Tackling the challenge. *Language Learning & Technology*, 21(3):217–241. [6.1](#)
- Colin Leong and Daniel Whitenack. 2022. [Phone-ing it in: Towards flexible multi-modal language model training by phonetic representations of data](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5306–5315, Dublin, Ireland. Association for Computational Linguistics. [7.6](#), [10.1](#)
- David D Lewis. 1995. Evaluating and optimizing autonomous text classification systems. In *ACM-SIGIR*. [8.1](#)
- David D Lewis and William A Gale. 1994. A sequential algorithm for training text classifiers. In *SIGIR (document)*, 8, [8.1](#), [8.2](#), [8.3](#), [8.3.2](#)
- William D. Lewis and Fei Xia. 2008. [Automatically identifying computationally relevant typological features](#). In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*. [1.1](#), [2.1.2](#)
- William D Lewis and Fei Xia. 2010. Developing odin: A multilingual repository of annotated language data for hundreds of the world’s languages. *Literary and Linguistic Computing*, 25(3):303–319. [2.1.2](#)
- Cathy Li and Farah Lalani. 2020. The covid-19 pandemic has changed education forever. In *World economic forum*, volume 29. The rise of online learning during the COVID-19 pandemic| World Economic .... [5.1](#), [6.1](#)
- Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, et al. 2020. Xglue: A New Benchmark Dataset for Cross-lingual Pre-training, Understanding and Generation. *arXiv*. [1](#)
- R Lieber. 2009. *Inflection*. In *Introducing Morphology*. Cambridge Introductions to Language and Linguistics. [2.2](#)
- Rochelle Lieber. 2021. *Introducing morphology*. Cambridge University Press. [2.2.2](#)
- Ming Huei Lin and Jia-Ying Lee. 2015. Data-driven learning: Changing the teaching of grammar in efl classes. *Elt Journal*, 69(3):264–274. [6.1](#)
- Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing Transfer Languages for Cross-Lingual Learning. In *ACL*. [9.3.2](#)
- Hans Lindquist. 2018. *Corpus linguistics and the description of English*. Edinburgh University Press. [6.1](#)
- Wang Ling, Chris Dyer, Alan W Black, Isabel Trancoso, Ramon Fernandez, Silvio Amir, Luis Marujo, and Tiago Luis. 2015. Finding function in form: Compositional character models for open vocabulary word representation. In *EMNLP*. [7.1](#)
- Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *LREC*. [5.3](#)
- Fangyu Liu, Emanuele Bugliarello, Edoardo Maria Ponti, Siva Reddy, Nigel Collier, and Desmond Elliott.

2021. [Visually grounded reasoning across languages and cultures](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10467–10485, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. [10.2](#)
- Michael H Long. 2000. Focus on form in task-based language teaching. *Language policy and pedagogy: Essays in honor of A. Ronald Walton*, 179:192. [1.1](#)
- Ken Longenecker, David Lacho, John Wagner, and Christine Schreyer. 2019. Evaluating Success and challenges of using Technology in remote villages to document the Kala language in Papua New Guinea. *arXiv*. [5.1](#)
- Qing Ma, Rui Yuan, Lok Ming Eric Cheung, and Jing Yang. 2022. Teacher paths for developing corpus-based language pedagogy: a case study. *Computer Assisted Language Learning*, pages 1–32. [6.1](#)
- Xuezhe Ma and Eduard Hovy. 2016. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In *ACL*. [1](#), [7](#), [7.5](#), [16](#), [9.3.2](#)
- Paul Macaruso and Alyson Rodman. 2009. [Benefits of computer-assisted instruction for struggling readers in middle school](#). *European Journal of Special Needs Education*, 24(1):103–113. [6.1](#)
- Chaitanya Malaviya, Matthew R. Gormley, and Graham Neubig. 2018. Neural Factor Graph Models for Cross-Lingual Morphological Tagging. In *ACL*. [1](#)
- Chaitanya Malaviya, Graham Neubig, and Patrick Littell. 2017. [Learning language representations for typology prediction](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2529–2535, Copenhagen, Denmark. Association for Computational Linguistics. [2.1.1](#)
- Andrej Malchukov. 2018. Grammatical case: Morphology, syntax, and word order. [4.1](#)
- Christopher D Manning. 1994. *Ergativity: Argument structure and Grammatical relations*. Ph.D. thesis, Stanford University. [3.2](#)
- Alec Marantz. 2000. Case and licensing. *Arguments and case: Explaining Burzio’s generalization*, pages 11–30. [4.2.1](#)
- Diego Marcheggiani and Thierry Artières. 2014. An Experimental comparison of Active Learning Strategies for Partially Labeled Sequences. In *EMNLP*. [9.1](#), [9.2](#)
- Stephen Mayhew, Chen-Tse Tsai, and Dan Roth. 2017. Cheap translation for cross-lingual named entity recognition. In *EMNLP*. [7.4](#), [??](#), [9.3.2](#)
- Andrew Kachites McCallumzy and Kamal Nigamy. 1998. Employing em and pool-based active learning for text classification. In *ICML*. [8.2](#)
- Arya D. McCarthy, Miikka Silfverberg, Ryan Cotterell, Mans Hulden, and David Yarowsky. 2018. Marrying Universal Dependencies and Universal Morphology. In *UDW*. [9.4](#)
- Ryan McDonald, Joakim Nivre, Yvonne Quirnbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. [Universal Dependency annotation for multilingual parsing](#). In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria. Association for Computational Linguistics. [2.1.2](#), [3.2.1](#), [6.3](#)

- Robert A McLean, William L Sanders, and Walter W Stroup. 1991. A unified approach to mixed linear models. *The American Statistician*. [5.3.2](#)
- Rada Mihalcea, Ravi Sinha, and Diana McCarthy. 2010. SemEval-2010 task 2: Cross-Lingual Lexical Substitution. In *ACL-SemEval*. [5.2.1](#)
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *CoRR*. [7.1](#)
- Tomas Mikolov, Quoc V Le, and Ilya Sutskever. 2013b. Exploiting similarities among languages for machine translation. In *arXiv*. [7](#), [7.1](#)
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013c. Distributed representations of words and phrases and their compositionality. In *NeurIPS*. [4.2.2](#), [7.2.1](#), [7.2.1](#), [7.2.2](#)
- George A Miller. 1995. Wordnet: a lexical database for english. *ACM*. [4](#), [6.2.2](#)
- Emily Ariel Moline. 2020. Indigenous Language Teaching Policy in California/the US: What’s Left Unsaid in Discourse/Funding. In *Issues in Applied Linguistics*. ([document](#)), [5.1](#), [6.1](#)
- Joshua Moore, Christopher J.C. Burges, Erin Renshaw, and Wen-tau Yih. 2013. Animacy detection with voting models. In *EMNLP*. [4.2.2](#)
- Steven Moran, Daniel McCloy, and Richard Wright. 2014. Phoible online. *Max Planck Institute for Evolutionary Anthropology*. [2.1.1](#)
- David R. Mortensen, Siddharth Dalmia, and Patrick Littell. 2018. Epitran: Precision G2P for many languages. In *LREC*. [7.4](#)
- David R. Mortensen, Xinyu Zhang, Chenxuan Cui, and Katherine J. Zhang. forthcoming. A hmong corpus with elaborate expression annotations. In *LREC 2022*. [4.5](#)
- Christopher Moseley. 2010. *Atlas of the World’s Languages in Danger*. UNESCO. [1](#)
- Joybrato Mukherjee. 2004. Bridging the gap between applied corpus linguistics and the reality of english language teaching in germany. In *Applied Corpus Linguistics*, pages 239–250. Brill. [6.1](#)
- John Munby. 1981. *Communicative syllabus design: A sociolinguistic model for designing the content of purpose-specific language programmes*. Cambridge university press. [6.1](#)
- Pieter Muysken, Pieter Cornelis Muysken, et al. 2000. *Bilingual speech: A typology of code-mixing*. Cambridge University Press. [10.2](#)
- David Nadeau and Satoshi Sekine. 2007. A survey of Named Entity Recognition and Classification. In *Linguisticae Investigations*. [7.5](#)
- ISP Nation. 2005. Teaching and learning vocabulary. In *Handbook of research in second language teaching and learning*, pages 605–620. Routledge. [5.3.2](#)
- Paul Nation. 2021. Is it worth teaching vocabulary? *TESOL Journal*, 12(4):e564. [6.2.2](#)
- Hloniphani Ndebele. 2012. *A socio-cultural approach to code-switching and code-mixing among speakers of IsiZulu in KwaZulu-Natal: a contribution to spoken language corpora*. Ph.D. thesis. [10.2](#)
- John Nerbonne, Duco Dokter, and Petra Smit. 1998. Morphological Processing and Computer-Assisted Language Learning. *CALL*. [5.1](#)

- Alaska Native Knowledge Network. 2001. Guidelines for strengthening indigenous languages. *Fairbanks, AK: Author*. 6.1
- Graham Neubig, Yosuke Nakata, and Shinsuke Mori. 2011. Pointwise prediction for robust, adaptable Japanese morphological analysis. In *ACL*. 8.1, 8.4.2
- Graham Neubig, Matthias Sperber, Xinyi Wang, Matthieu Felix, Austin Matthews, Sarguna Padmanabhan, Ye Qi, Devendra Singh Sachan, Philip Arthur, Pierre Godard, John Hewitt, Rachid Riad, and Liming Wang. 2018. XNMT: The extensible neural machine translation toolkit. In *AMTA*. 7.5.2
- Hieu T Nguyen and Arnold Smeulders. 2004. Active learning using pre-clustering. In *Proceedings of the twenty-first international conference on Machine learning*, page 79. 8.3
- Minh Van Nguyen, Viet Dac Lai, Amir Pouran Ben Veyseh, and Thien Huu Nguyen. 2021. [Trankit: A light-weight transformer-based toolkit for multilingual natural language processing](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 80–90, Online. Association for Computational Linguistics. 2.1.1, 4.2.3
- J. Nichol. 1995. Effects of clausal structure on subject-verb agreement errors. In *booktitle of Psycholinguistic Research*. 3.1
- Johanna Nichols. 1985. The directionality of agreement. In *Annual Meeting of the Berkeley Linguistics Society*. 3.2
- Joakim Nivre, Rogier Blokland, Niko Partanen, Michael Rießler, and Jack Rueter. 2018. Universal Dependencies 2.3. In *Universal Dependencies Consortium*. 2.1.1, 3.3, 7, 9.4
- Joakim Nivre, Marie-Catherine De Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Reut Silveira, Natalia Tsfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *LREC*. 1, 2.1.3, 2.2, 3.3, 4.2.2, 9.4
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Jan Hajic, Christopher D. Manning, Sampo Pyysalo, Sebastian Schuster, Francis Tyers, and Daniel Zeman. 2020. Universal Dependencies v2: An Evergrowing Multilingual Treebank Collection. In *LREC*. 3.3
- Joakim Nivre, Jens Nilsson, and Johan Hall. 2006. Talbanken05: A Swedish treebank with phrase structure and dependency annotation. In *LREC*. 2.1.1, 4.2.2, 7
- Jeremy Nixon, Mike Dusenberry, Linchuan Zhang, Ghassen Jerfel, and Dustin Tran. 2019. Measuring calibration in Deep Learning. In *arXiv*. 2, 9.5.2
- Sebastian Nordhoff and Harald Hammarström. 2011. Glottolog/langdoc: Defining dialects, languages, and language families as collections of resources. In *First International Workshop on Linked Science 2011-In conjunction with the International Semantic Web Conference (ISWC 2011)*. 2.1.1
- Patrick D. Nunn and Nicholas J. Reid. 2016. Aboriginal Memories of Inundation of the Australian Coast Dating from More than 7000 Years Ago. *Australian Geographer*. 1
- Antoine Nzeyimana and Andre Niyongabo Rubungu. 2022. [KinyaBERT: a morphology-aware Kinyarwanda language model](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5347–5363, Dublin, Ireland. Association for Computational Linguistics. 7.6, 10.1

- Michael P. Oakes. 1998. Statistics for Corpus Linguistics. In *Edinburgh Textbooks in Empirical Linguistics*. [3.2.4](#)
- Anne O’keeffe, Michael McCarthy, and Ronald Carter. 2007. *From corpus to classroom: Language use and language teaching*. Cambridge University Press. [6.1](#)
- Lourdes Ortega. 2015. Usage-based SLA: A research habitus whose time has come. *Usage-based Perspectives on Second Language Learning*. [5.1](#)
- Robert Östling. 2015. Word Order Typology through Multilingual Word Alignment. In *ACL*. [2.1.2](#), [4.1](#)
- Lilja Øvrelid. 2006. Towards robust Animacy classification using morphosyntactic distributional features. In *Student Research Workshop*. [4.2.2](#)
- Lilja Øvrelid. 2009. Empirical Evaluations of Animacy Annotation. In *EACL*. [4.2.2](#)
- Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *ACL*. [1](#)
- Tommaso Pasini, Alessandro Raganato, and Roberto Navigli. 2021. XL-WSD: An extra-large and cross-lingual evaluation framework for word sense disambiguation. In *Proc. of AAAI*. [6.2.2](#)
- 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. 2011. Scikit-learn: Machine Learning in Python. In *booktitle of Machine Learning Research*. [3.3](#), [5.3](#)
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014a. GloVe: Global vectors for word representation. In *EMNLP*. [7.1](#)
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014b. Glove: Global vectors for word representation. In *EMNLP*. [8.5](#)
- Haroldo Vargas Pereira, José Vargas Pereira, Lev Michael, Christine Beier, and Zachary O’Hagan. 2011. Matsigenka texts. In *Archive of the Indigenous Languages of Latin America*. [2.1.3](#)
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*. [1](#), [7.1](#), [7.6](#)
- Deborah Chen Pichler, Julie A Hochgesang, Diane Lillo-Martin, and Ronice Müller de Quadros. 2010. Conventions for sign and speech transcription of child bimodal bilingual corpora in elan. *Language, Interaction and Acquisition*, 1(1):11–40. [2.1.3](#)
- Carl Pollard and Ivan A Sag. 1994. *Head-driven phrase structure grammar*. University of Chicago Press. [2.1.2](#), [3.1](#), [4.2.2](#)
- Shana Poplack. 2001. Code-switching (linguistic). *International encyclopedia of the social and behavioral sciences*, 12:2062–2065. [10.2](#)
- Adithya Pratapa, Antonios Anastasopoulos, Shruti Rijhwani, Aditi Chaudhary, David R Mortensen, Graham Neubig, and Yulia Tsvetkov. 2021a. Evaluating the morphosyntactic well-formedness of generated texts. *arXiv*. [3.1](#), [4.6](#)
- Adithya Pratapa, Antonios Anastasopoulos, Shruti Rijhwani, Aditi Chaudhary, David R. Mortensen, Graham Neubig, and Yulia Tsvetkov. 2021b. [Evaluating the morphosyntactic well-formedness of generated](#)



- [texts](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7131–7150, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. [10.1](#)
- Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *ACL*. [10.1](#)
- Geoffrey K Pullum. 1984. Syntactic and semantic parsability. In *COLING*. [3.2](#)
- Zorica Puškar and Gereon Müller. 2018. Unifying structural and lexical case assignment in dependent case theory. *Advances in formal Slavic linguistics 2016*, 1:357. [4.2.1](#)
- Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. [Stanza: A python natural language processing toolkit for many human languages](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics. [5.3](#)
- Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation? In *NAACL-HLT*. [7.5.2](#)
- J. Ross Quinlan. 1986. Induction of decision trees. In *Machine learning*. [3.2.3](#), [4.2.3](#)
- C. Quinn. 2001. A preliminary survey of animacy categories in penobscot. *Algonquian Conference*. [4.2.2](#)
- Lawrence R Rabiner. 1989. A tutorial on hidden markov models and selected applications in speech recognition. In *IEEE*. [8.4.2](#)
- Gowtham Ramesh, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Divyanshu Kakwani, Navneet Kumar, et al. 2022. Samanantar: The largest publicly available parallel corpora collection for 11 indic languages. *Transactions of the Association for Computational Linguistics*, 10:145–162. [6.3](#)
- Randi Reppen. 2010. *Using corpora in the language classroom*. Cambridge University Press. [6.1](#)
- Jack C Richards, David Singleton, and Michael H Long. 1999. *Exploring the second language mental lexicon*. Cambridge University Press. [5.3.2](#)
- Eric Ringger, Peter McClanahan, Robbie Haertel, George Busby, Marc Carmen, James Carroll, Kevin Seppi, and Deryle Lonsdale. 2007. Active Learning for part-of-speech Tagging: Accelerating corpus annotation. In *Linguistic Annotation Workshop*. [9.2](#)
- Annette Rios, Mathias Müller, and Rico Sennrich. 2018. The Word Sense Disambiguation Test Suite at WMT18. In *ACL-WMT*. [5.3.1](#)
- Frankie Robertson. 2020. [Show, don't tell: Visualising Finnish word formation in a browser-based reading assistant](#). In *Proceedings of the 9th Workshop on NLP for Computer Assisted Language Learning*, pages 37–45, Gothenburg, Sweden. LiU Electronic Press. [5.1](#)
- Bruce M Rowe and Diane P Levine. 2018. *A concise introduction to linguistics*. Routledge. [7.1](#)
- Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. In *booktitle of Artificial Intelligence Research*. [7.1](#)
- Tatyana Ruzsics, Olga Sozinova, Ximena Gutierrez-Vasques, and Tanja Samardzic. 2021. [Interpretability for morphological inflection: from character-level predictions to subword-level rules](#). In *Proceedings*

- of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 3189–3201, Online. Association for Computational Linguistics. [6.3](#)
- Ivan A Sag, Ronald Kaplan, Lauri Karttunen, Martin Kay, Carl Pollard, Stuart M Shieber, and Annie Zaenen. 1986. Unification and grammatical theory. In *Proceedings of the West Coast Conference on Formal Linguistics*. Cascadilla Press. [3.1](#)
- Mike Schuster and Kaisuke Nakajima. 2012. Japanese and korean voice search. In *ICASSP*. [10.1](#)
- Ali Fuad Selvi and Ali Shehadeh. 2018. *Approaches and Methods in English for Speakers of Other Languages & Non-native English-speaking Teachers (NNESTs)*. Wiley-Blackwell. [6.2.1](#)
- Ozan Sener and Silvio Savarese. 2018. Active Learning for Convolutional Neural Networks: A Core-Set Approach. In *ICLR*. [8.2, 23](#)
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *ACL*. [7.1, 10.1](#)
- Burr Settles. 2009. Active Learning literature survey. *University of Wisconsin-Madison Department of Computer Sciences*. [9.2](#)
- Burr Settles, Chris Brust, Erin Gustafson, Masato Hagiwara, and Nitin Madnani. 2018. Second Language Acquisition Modeling. In *ACL-BEA*. [5](#)
- Burr Settles and Mark Craven. 2008. An Analysis of Active Learning Strategies for Sequence Labeling Tasks. In *EMNLP*. ([document](#)), [8.1, 8.2, 8.3.2, 8.3.3, 9.1](#)
- H Sebastian Seung, Manfred Opper, and Haim Sompolinsky. 1992. Query by committee. In *Fifth annual workshop on Computational learning theory*. [8.3, 8.3.3](#)
- Claude Elwood Shannon. 2001. A mathematical theory of communication. In *ACM SIGMOBILE*. [8.3.2](#)
- Qinlan Shen, Daniel Clothiaux, Emily Tagtow, Patrick Littell, and Chris Dyer. 2016. The role of context in neural morphological disambiguation. In *COLING*. [7.4](#)
- Stuart M Shieber. 2003. *An introduction to unification-based approaches to grammar*. Microtome Publishing. [1.1](#)
- Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training multi-billion parameter language models using model parallelism. *arXiv preprint arXiv:1909.08053*. [10.1](#)
- Aditya Siddhant, Ankur Bapna, Henry Tsai, Jason Riesa, Karthik Raman, Melvin Johnson, Naveen Ari, and Orhan Firat. 2020. Evaluating the cross-lingual effectiveness of massively multilingual neural Machine Translation. In *AAAI*. [10.1](#)
- Han Sloetjes and Peter Wittenburg. 2008. Annotation by category-elan and iso dcr. In *LREC*. [2.1.3](#)
- Philip D Smith Jr. 1981. *Second Language Teaching: A Communicative Strategy*. *The Foreign & Second Language Educator Series*. ERIC. [6.2.2](#)
- Matthias Sperber, Mirjam Simantzik, Graham Neubig, Satoshi Nakamura, and Alex Waibel. 2014. Segmentation for efficient supervised language annotation with an explicit cost-utility tradeoff. In *TACL*. [8.1, 8.4.2](#)

- S. Steele. 1978. Word order variation: a typological study. In *Universals of Human Language*. 3.1
- Pontus Stenetorp, Sampo Pyysalo, Goran Topić, Tomoko Ohta, Sophia Ananiadou, and Jun'ichi Tsujii. 2012. brat: a web-based tool for NLP-assisted text annotation. In *EACL*. 2.1.3
- Rosetta Stone. 2010. Rosetta stone. 2.1.3, 5.1, 6.2.1
- Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. Spine: Sparse interpretable neural embeddings. *Proceedings of the Thirty Second AAAI Conference on Artificial Intelligence (AAAI)*. 4.2.2, 4.3
- Gail M Sullivan and Richard Feinn. 2012. Using Effect Size—or Why the P Value is not Enough. In *booktitle of Graduate Medical Education*. 3.2.4
- Kristin Sverredal. 2018. A grammar sketch of north tanna. 2.2
- Oscar Täckström, Dipanjan Das, Slav Petrov, Ryan McDonald, and Joakim Nivre. 2013. Token and type constraints for cross-lingual part-of-speech tagging. In *TACL*. 2.2
- Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, and Chunfang Liu. 2018. A survey on deep transfer learning. In *International conference on artificial neural networks*, pages 270–279. Springer. 7.1
- Chuanqi Tan, Furu Wei, Nan Yang, Weifeng Lv, and Ming Zhou. 2017. S-net: From answer extraction to answer generation for machine reading comprehension. In *arXiv*. 7
- Juhi Tandon, Himani Chaudhry, Riyaz Ahmad Bhat, and Dipti Sharma. 2016. [Conversion from paninian karakas to Universal Dependencies for Hindi dependency treebank](#). In *Proceedings of the 10th Linguistic Annotation Workshop held in conjunction with ACL 2016 (LAW-X 2016)*, pages 141–150, Berlin, Germany. Association for Computational Linguistics. 6.3
- Duyu Tang, Furu Wei, Bing Qin, Nan Yang, Ting Liu, and Ming Zhou. 2016. Sentiment embeddings with applications to sentiment analysis. In *IEEE Transactions on Knowledge and Data Engineering*. 7
- Wilson L Taylor. 1953. “cloze procedure”: A new tool for measuring readability. *Journalism quarterly*. 5.1
- Cynthia A Thompson, Mary Elaine Califf, and Raymond J Mooney. 1999. Active learning for natural language parsing and information extraction. In *ICML*. 8.2
- Juliette Thuilier, Margaret Grant, Benoît Crabbé, and Anne Abeillé. 2021. Word order in french: the role of animacy. *Glossa: a journal of general linguistics*, 6(1). 4.2.2
- Jörg Tiedemann. 2012. Parallel data, tools and interfaces in opus. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA). 5.3
- Erik F. Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. In *COLING*. 8.5
- Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *NAACL-HLT*. 8.5
- Erik F. Tjong Kim Sang and Jorn Veenstra. 1999. Representing text chunks. In *EACL*. 8.4.1

- Simon Tong and Edward Chang. 2001. Support vector machine active learning for image retrieval. In *ninth ACM Multimedia conference*. [8.2](#), [8.3](#)
- Kristina Toutanova and Christopher D. Manning. 2000. [Enriching the knowledge sources used in a maximum entropy part-of-speech tagger](#). In *2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 63–70, Hong Kong, China. Association for Computational Linguistics. [1](#)
- Yuta Tsuboi, Hisashi Kashima, Shinsuke Mori, Hiroki Oda, and Yuji Matsumoto. 2008. Training conditional random fields using incomplete annotations. In *COLING*. [8.4.2](#), [16](#)
- Yulia Tsvetkov and Chris Dyer. 2016. Cross-lingual bridges with models of lexical borrowing. In *booktitle of Artificial Intelligence Research*. [7.2.2](#)
- Jan Ullrich, Elliot Thornton, Peter Vieira, Logan Swango, and Marek Kupiec. 2020. Owóksape-an Online Language Learning Platform for Lakota. In *SLTU-CCURL*. [5.1](#)
- Shyam Upadhyay, Manaal Faruqui, Chris Dyer, and Dan Roth. 2016. Cross-lingual models of word embeddings: An empirical comparison. *arXiv preprint arXiv:1604.00425*. [10.2](#)
- Anders Vaa. 2013. A grammar of engdewu. In *Thesis*. [2.1.3](#)
- Bill VanPatten and Megan Smith. 2019. Word-order typology and the acquisition of case marking: A self-paced reading study in latin as a second language. *Second Language Research*, 35(3):397–420. [4.1](#)
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017a. Attention is all you need. In *NeurIPS*. [9.3.2](#)
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017b. Attention is all you need. In *NIPS*. [10.1](#)
- Olavi Vesalainen. 2014. *A grammar sketch of Lhomi*. SIL International. [2.2](#)
- G. Vigliocco, B. Butterworth, and M. F. Garrett. 1996. Subject-verb agreement in spanish and english: Differences in the role of conceptual constraints. In *Cognition*. [3.1](#)
- G. Vigliocco and J. Nicol. 1998. Separating hierarchical relations and word order in language production: is proximity concord syntactic or linear? In *Cognition*. [3.1](#)
- Dingquan Wang and Jason Eisner. 2017. [Fine-grained prediction of syntactic typology: Discovering latent structure with supervised learning](#). *Transactions of the Association for Computational Linguistics*, 5:147–161. [2.1.2](#), [4.1](#)
- Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2021. [Multi-view subword regularization](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 473–482, Online. Association for Computational Linguistics. [10.1](#)
- Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2022. [Expanding pretrained models to thousands more languages via lexicon-based adaptation](#). In *Annual Conference of the Association for Computational Linguistics (ACL)*, Dublin, Ireland. [10.2](#)
- Zhe Wang, Xiaoyi Liu, Limin Wang, Yu Qiao, Xiaohui Xie, and Charless Fowlkes. 2018. Structured Triplet Learning with POS-tag Guided Attention for Visual Question Answering. In *IEEE-WACV*. [9.1](#)

- Dittaya Wanvarie, Hiroya Takamura, and Manabu Okumura. 2011. Active learning with subsequence sampling strategy for sequence labeling tasks. In *Information and Media Technologies*. 8.4.2
- Yuichi Watanabe. 1997. Input, intake, and retention: Effects of Increased Processing on Incidental Learning of Foreign Language Vocabulary. *Studies in Second Language Acquisition*. 5, 5.1
- Lorna Williams. 2019. Wa7 szum'in'stum' ti nqwelutenlhkalkha: Technology and Indigenous language revitalization, recovery and normalization. *Keynote LT4All*. 1, 1
- Fei Xia, William Lewis, Michael Wayne Goodman, Joshua Crowgey, and Emily M Bender. 2014. Enriching odin. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 3151–3157. 2.1.2
- Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. In *EMNLP*. 8.4.2, 8.5, 8.5, 9.3.2
- Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *NAACL-HLT*. 7, 7.1
- Zhao Xu, Kai Yu, Volker Tresp, Xiaowei Xu, and Jizhi Wang. 2003. Representative sampling for text classification using support vector machines. In *European conference on information retrieval*, pages 393–407. Springer. 8.3
- Mutsumi Yamamoto. 1999. *Animacy and reference: A cognitive approach to corpus linguistics*. John Benjamins Publishing. 4.2.2
- Zhilin Yang, Ruslan Salakhutdinov, and William W Cohen. 2017. Transfer Learning for Sequence Tagging with hierarchical recurrent networks. In *arXiv*. 9.3.2
- David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *ACL*. 9.3.2
- David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *HLT*. 8.1
- Kayo Yin, Patrick Fernandes, Danish Pruthi, Aditi Chaudhary, André F. T. Martins, and Graham Neubig. 2021. [Do context-aware translation models pay the right attention?](#) In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 788–801, Online. Association for Computational Linguistics. 5.4, 10.1
- Hyunsook Yoon. 2005. *An investigation of students' experiences with corpus technology in second language academic writing*. Ph.D. thesis, The Ohio State University. 6.1
- Hyunsook Yoon and JungWon Jo. 2014. Direct and indirect access to corpora: An exploratory case study comparing students' error correction and learning strategy use in l2 writing. *Language Learning & Technology*, 18(1):96–117. 6.1
- Yukio Yotsumoto. 2020. Revitalization of the Ainu Language: Japanese Government Efforts. *Handbook of the Changing World Language Map*. 5.1
- Hwanjo Yu. 2005. Svm selective sampling for ranking with application to data retrieval. In *ACM SIGKDD*. 8.2

- Liang-Chih Yu, Jin Wang, K. Robert Lai, and Xuejie Zhang. 2018. Refining word embeddings using intensity scores for sentiment analysis. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing*. 7
- Annie Zaenen, Jean Carletta, Gregory Garretson, Joan Bresnan, Andrew Koontz-Garboden, Tatiana Nikitina, M. Catherine O'Connor, and Tom Wasow. 2004. Animacy encoding in English: Why and How. In *ACL-DiscAnnotation*. 4.2.2
- Morteza Zahedi, Philippe Dreuw, David Rybach, Thomas Deselaers, and Hermann Ney. 2006. Continuous sign language recognition-approaches from speech recognition and available data resources. In *Second workshop on the representation and processing of sign languages: lexicographic matters and didactic scenarios*, pages 21–24. 2.1.3
- Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *ACL*. 7.1
- Leonardo Zilio, Rodrigo Wilkens, and Cédric Fairon. 2017. Using NLP for Enhancing Second Language Acquisition. In *RANLP*. 5.1
- Indrè Žliobaitė, Albert Bifet, Bernhard Pfahringer, and Geoff Holmes. 2011. Active learning with evolving streaming data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. 8.2
- Ran Zmigrod, Sabrina J. Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. In *ACL*. 3.1
- Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low resource neural machine translation. In *EMNLP*. (document), 7.3.2