Integrating Machine Learning into Language Documentation and Description

by

Sarah Moeller

B.A., Thomas Edison State College, 2002

M.A., Dallas International University, 2010

M.A., University of Colorado, 2020

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

Department of Linguistics and Institute of Cognitive Science

2021

Committee Members:

Mans Hulden, Chair

Martha Palmer

Andrew Cowell

Alexis Palmer

Katharina Kann

Moeller, Sarah (Ph.D., Linguistics and Cognitive Science)

Integrating Machine Learning into Language Documentation and Description

Thesis directed by Dr. Mans Hulden

At least 40% of the world's 7000+ languages are believed to be in danger of disappearing from human use by the end of this century. Many languages will disappear with almost no record of their existence because efforts to document and describe these languages are encountering an "annotation bottleneck" at early stages of analysis and annotation. Current annotation methods are too slow and expensive to counteract the pace of language endangerment and loss. Annotation could be sped and improved by machine learning. However, state-of-the-art supervised machine learning depends heavily on large amounts of annotated data.

This dissertation explores how to train supervised machine learning systems for morphological analysis during language documentation and description. The systems are applied to nine languages. The research investigates ways that linguists and NLP scientists may want to adjust their expectations and workflows so that both can achieve optimal results with endangered data.

New methods for tasks in morphological analysis are explored. First, various approaches to automating morpheme segmentation and glossing are compared. Second, a new task is presented for learning morphological paradigms and automatically generating new morphological resources: IGT-to-paradigms (IGT2P). Third, the impact of POS tags on segmentation, glossing, and paradigm induction is examined, showing that the presence or absence of POS tags does not have a significant bearing on the performance of machine learning systems. These results are indicators that Natural Language Processing (NLP) systems could be successfully integrated into the documentary and descriptive workflow. At the same time, the relatively high accuracy achieved from noisy field data with little or no additional human annotation hints that NLP may benefit from limited documentary linguistic data which may be the only or largest linguistically annotated resource available for some languages.

Dedication

To my parents, who have a habit of listening and then replying, “Go for it!"

Acknowledgements

This dissertation was supported by generous grants from the Institute of Cognitive Science (ICS), the Center for the Advancement of Teaching in Social Sciences (CARTSS), the Department of Linguistics at the University of Colorado Boulder, and by International Language and Development (ILAD). I am grateful to Daniel Wilson who connected me with ILAD–an innovative organization with a meaningful goal.

I could not have done this without the valuable guidance, consistent support, and encouragement of my advisor Mans Hulden. Despite his workload, he never hurries through our meetings and he contributed to my success even beyond the helpful discussions about research. As an advisor he gave an ideal balance of expectations and independence.

My committee members all contributed more to this completed manuscript than just comments and suggestions. Martha Palmer gave simple, practical advice that, especially during my first year, made this journey easier. Andy's keen interest in computational methods for minority languages has been a splendid example of cross-disciplinary research and lifelong learning. Alexis has shown the spirit of a true teacher and mentor since we first crossed paths in Portugal and has been generous with her time and with my odd question about careers in academia. Without Katharina's insightful questions and pursuit of excellence, I'm not sure chapter 5 and chapter 6 would have come into existence.

The quality of this work owes much to the other co-authors of the published versions of Chapters 5 and 6. Ling Liu was ready to listen to the vague idea and take on another project in the middle of her own PhD progress. I'm so glad Changbing Yang asked for more opportunities and I

wish her the best as she starts her PhD.

Without the linguists, fieldworkers, and annotators who prepared the IGT, this work would not have been possible. Drs. Brenda Boerger, Shobhana Chelliah, Bernard Comrie, and Andy Cowell, as well as Chuck Donet, and Andrew Brumleve generously shared their field data. Yaghut added the missing Lezgi annotations. Mary Burke, Brenda, Andy, and Andrew also did the expert cleaning for the IGT2P experiments. Changbing and my undergraduate research assistants Zachary J. Ryan and Huilin Lin helped preprocess and reformat much of the data.

I am so grateful for colleagues at CU Boulder who made time to talk about life and linguistics: Kristin, Katie, Irina, Annebeth, Adam, and of course my ever encouraging and upbeat cohort-mate Norielle.

Perhaps the biggest reasons I considered a career in academia were the examples set by Paul Kroeger, my MA advisor, and Michael Boutin, the head of the Applied Linguistics Department at Dallas International University. I didn't write a MA thesis, so this is my chance to acknowledge them. My goal is to give others what they gave to me: the sense that every time I entered their offices that they had nothing more important in their overworked life than to listen to my rambling ideas and answer my half-formed questions.

Several other people from my Texas years encouraged me to head down this road. Among them, Brenda Boerger and Will Reiman are wonderful mentors and faithful friends whose Skype pings have meant so much the past several years.

More than anyone, I am grateful for the support of my family. They seem convinced everything I do is interesting and important. And I am obliged to my 16 nephews and nieces who kept me grounded by demanding, "If you're almost-a-doctor, how come you don't know the answer to all my questions?!"

Contents

Chapter

Tables

Table

Figures

Figure

Chapter 1

Introduction

In the early 1990s it was suggested that linguistics might be the first academic discipline to preside over its own demise. As much as 90% of the world's 7,000 languages were predicted to become extinct by the end of the 21st century (Krauss, 1992, 2007; Campbell et al., 2013). Linguists responded by putting greater emphasis on the documentation and description of under-documented languages, and over the past thirty years this effort has steadily broadened our knowledge of the world's languages. Today the estimate of endangered languages is more conservative (Eberhard et al., 2020) but still little data or knowledge is available for thousands of languages. It is critical that these languages be documented and described quickly before they disappear.

The general flow of documentary and descriptive data is illustrated in Figure 1.1. A team of linguists and native speakers (a) collaborate to document, conduct basic linguistic analysis, and annotate the documented language data. The annotated data can be used for linguistic research and theory development, the expansion and testing of Natural Language Processing (NLP) algorithms, and for the benefit of the community of speakers who may wish to maintain or revitalize the language. However, because annotation is time- and labor-intensive work, a significant portion of documented language data is bottlenecked and, therefore, inaccessible or difficult to use (b) for linguistic or NLP research and, to a lesser extent, community language development. The methods presented in this dissertation will make more data accessible by integrating NLP machine learning systems into the documentary and descriptive workflow in order to speed and improve the process (c).

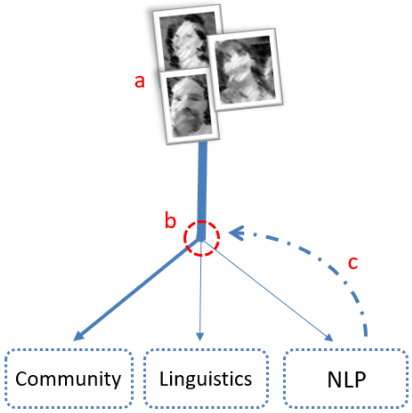

Figure 1.1: Language Data Production. A documentary and descriptive field project creates annotated data (a) which can be used for linguistic research, Natural Language Processing (NLP) development, and community efforts to maintain or revitalize the language. Manual annotation has created a bottleneck (b). This dissertation examines methods for integrating NLP into the documentary and descriptive workflow in order to increase annotated language data production (c).

Unfortunately, current methods in language documentation and description, especially the process of annotating transcribed texts and analyzing morphological paradigms, are too slow to counteract the crisis of language endangerment. For example, it can take up to 100 hours to manually transcribe a single hour of recorded speech (Seifart et al., 2018). If the process is to match the pace of language extinction, computational methods must be integrated effectively into the workflow of language documentation and description.

Natural language processing (NLP) did not respond as quickly as linguists to the language endangerment crisis. Although machine learning systems, capable of learning complex patterns in data, have gained tremendous success since the 1990s, NLP research has been focused on a handful of economically or politically powerful languages such as Chinese, Arabic, English, and other European languages. These languages are well documented and described. None are endangered.

Fortunately, this is changing. In recent years, NLP research with limited data has burgeoned. This is evidenced, for instance, by the 2015-2019 DARPA-funded Low Resource Languages for Emerging Incidents (LORELEI) project that was motivated in part by the 2014 Haiti earthquake where disaster aid was hampered by the lack of data in Haitian Creole, which is spoken widely in

the country, but linguistic resources are so limited, it is rarely encountered in language technology. When Haitian Creole speakers went to social media to cry for help, foreign aid workers struggled to process the information and to inform victims when and where medicine and supplies were available.

Despite this growth, very few NLP systems have been integrated into the process of documenting and describing endangered languages. This lack of integration is largely because state-of-the-art supervised machine learning models depend on large amounts of annotated data (on the order of hundreds of thousands or millions and more tokens), but another challenge to integration is presented by the noise in data created during documentary and descriptive field data. The dynamic, evolving nature of ongoing linguistic analysis during fieldwork and the reliance on manual annotation means that field data is peppered with inconsistencies and typos. To overcome these challenges, methods must be developed that allow documentary and descriptive linguists to benefit from NLP machine learning and *vice versa*.

This dissertation investigates methods for more effectively leveraging machine learning systems for language documentation and description, particularly for morphological analysis and annotation. The research looks at ways that linguists and NLP scientists may want to adjust their expectations and workflows so that both fields can achieve optimal results when working with limited data.

The overarching question this dissertation asks is **How might the integration of machine learning into language documentation and description affect currently accepted expectations, methods, and workflows?** It addresses that question by examining how current expectations and methods affect machine learning performance on morphological analysis and annotation and exploring new methods for effectively exploiting existing resources to boost machine learning performance on these tasks. This will be done with three studies using various NLP machine learning models and documentary and descriptive corpora from nine under-documented languages. The three studies ask their own specific questions:

(1) **Automating Segmentation and Glossing:** How do variations of research design for

morpheme segmentation and glossing which arise from differing conventional expectations in NLP and linguistics affect machine learning performance on those tasks on documentary and descriptive data?

(2) **Morphological Paradigm Induction from Interlinear Glossed Texts (IGT2P):** To what extent can machine learning models learn morphological inflection patterns from manually interlinearized texts? Can a human-in-the-loop approach improve results and overcome the inherent noisiness of field data?

(3) **Priority of Part of Speech Tagging:** What impact does NLP's traditionally high priority on part of speech tagging have on the ability of machine learning systems to perform accurate morpheme segmentation, morpheme glossing, and paradigm induction?

This work is motivated by the "yawning gap" between the amount of documented data deposited in language archives and the portion of the data that is usable for research (Seifart et al., 2018). This gap is caused by what has been described as an "annotation bottleneck" (Figure 1.2). Current tedious, time-consuming, and expensive annotation tasks are performed primarily by hand from start to finish (Simons and Lewis, 2013; Holton et al., 2017). Manual annotation is subject to

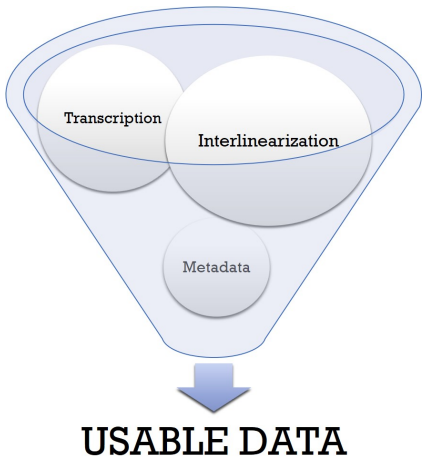

Figure 1.2: The annotation bottleneck in language documentation and description hinders linguists and NLP scientists from using new language data for research.

human error, introducing inconsistencies and typos, due often not to the difficulty of the task, but to its repetitive and monotonous nature. Other noise in the annotation arises because annotation is also the tool for linguists to begin discovering a language's structure. As linguists' understanding grows, analyses change. These changes are reflected in later annotations, but often not "corrected" in earlier annotations. Budget and time constraints often mean that substantial portions of the data produced by field projects are left uncorrected or simply unannotated. The unannotated portions of documentary corpora remain untapped resources that could inform the development of linguistic science and NLP. They could also support human language technology that would benefit the communities that speak the languages.

The contributions of this dissertation are four-fold. First, it shows that it is possible and practical to use machine learning to perform morphological annotation by developing and applying methods that improve results on typologically diverse languages with limited training data. Second, by learning inflectional paradigms and accurately generating inflected forms, it demonstrates that machine learning can be used to build and test hypotheses about a language's morphological structure. Third, by training on noisy linguistic field data rather than the curated and polished data that NLP systems are typically trained on, it proves that documentary and descriptive field data, which sometimes the only annotated data available for an under-documented language, can be used to effectively train machine learning models. Fourth, it increases annotated data in several low-resource languages which represent a range of field projects, typological structures, and language families. Increased annotated data will allow more thorough testing of linguistic theories and computational models. Finally, this work presents how new computational methods can be successfully integrated into language documentation and description.

The rest of the dissertation is organized as follows. Chapter 2 provides a background on the linguistic and computational linguistics history and concepts important to this research. Chapter 3 introduces the nine languages and their corpora used in the research and describes the machine learning systems that are implemented in the experiments. Chapter 4 describes computational methods for segmenting and glossing morphemes and is an expanded version of Moeller and Hulden

(2018) and Moeller and Hulden (2021). Chapter 5 describes methods for inducing morphological paradigms from interlinear text and is an expanded version of Moeller et al. (2020). Chapter 6 looks at the role of part-of-speech (POS) tags for both tasks and is an expanded version of a paper under review at the *North American Chapter of the Association for Computational Linguistics* (NAACL 2021). Finally, Chapter 7 summarizes the impacts of this research and outlines work that could build upon this research and further integrate machine learning into language documentation and description.

Chapter 2

Background

This chapter summarizes linguistic and NLP literature in order to establish the role that interlinearization and morphological analysis has played in language documentation and description and how NLP work is related. The research explores ways to integrate machine learning, specifically for automating the three initial tasks of interlinearization (morpheme segmentation, morpheme glossing, and free translation) as well as inflectional paradigm induction.

2.1 Language Documentation and Description

The activities that constitute language documentation and language description fieldwork are not clearly distinguished. Himmelmann (1998) defines language documentation as "a comprehensive and representative sample of communicative events [that are] as natural as possible." Woodbury (2003) defines it similarly as "comprehensive and transparent records supporting wide ranging scientific investigations of the language." Language description can be defined as work that analyzes language documentation to create "systematic presentations of the phonology, morphology, syntax, and semantics of the language" (Bird and Chiang, 2012). The emphasis in linguistics on endangered languages over the past three decades has established modern field methods for recording and analyzing data (Bowern, 2008; Czaykowska-Higgins, 2009; Lupke, 2010; Vallejos, 2014; Rice and Thunder, 2017). However, the specific activities that divide the two subfields are not rigid. Therefore, the currents work rarely distinguishes the two. Instead, it refers to the two together as "language documentation and description" or "documentary and descriptive linguistics".

The workflow of language documentation and description activities is not standardized, although most projects seem to follow a similar sequence. One common sequence of activities was described by Bird and Chiang (2012). A version is given below with numbers added which will be used in the following paragraphs to refer to each task (e.g., "task 2a" refers to the process of transcription). Bird and Chiang classify this workflow under language documentation, but each subsequent task progressively encompasses more description than documentation, excepting archiving, which comes strictly under language documentation but is logically the last step.

(1) Collect (audio/video recordings of) naturally occurring speech

(2) a) Transcribe and b) translate the recordings

(3) Perform basic morphosyntactic analysis of the transcription by segmenting the morphemes and creating morphological glosses and/or a lexicon

(4) Elicit morphological paradigms that will allow the study of specific phenomena and/or reveal underlying patterns

(5) Prepare a grammar of the language i.e., descriptive reports that outline how the language is structured

(6) Archive data in a long-term digital repository

One primary output of this workflow is interlinear glossed texts (IGT), a distinctive data format in linguistics. Interlinearization takes center stage after recorded speech has been transcribed and moves the workflow beyond simple collection of data but still serves as a "preprocessing step" to (Moon et al., 2009) language description (strictly defined). It comprises several annotation tasks that enrich the data with analytic information which can be added on lines under the original transcribed text. Several common lines of annotation are shown in Figure 2.1. Interlinearizing more and more data uncovers the rarer and unique linguistic phenomena. To assist the identification and study of these phenomena other lines of annotation, such as lexical categories (POS tags), are

Transcribed sentence	Maria	ama		las		manzanas	
Morphemes	Maria	am	-a	l	-as	manzana	-s
Gloss	Mary	love	3.SG. PRESENT	the	PL. FEMININE	apple	PL
Lexical categories (morpheme)	Proper Noun	Verb	Verb Agreement	DEF	Noun Agreement	Noun	Number
Part of speech (word)	Proper Noun	Verb		Definite article		Noun	
...any number of other lines...	…	…		…		…	
Free translation	‘Mary loves apples.’						

Figure 2.1: Interlinearization. Interlinear glossed texts add lines of annotation to the original text.

added. The lines can be added in any order, but translations (task 2a) morpheme boundaries and morpheme glosses (task 3) are usually added first. As the workflow above indicates, descriptive annotation beyond these lines is beyond the priorities of language documentation (strictly defined). Therefore, in this work “interlinearization” usually denotes three annotation tasks: 1) identifying morpheme boundaries (**morpheme segmentation**), 2) labeling each morpheme with its lexical meaning or morphosyntactic function (**glossing**), and 3) providing **free translations** of sentences in a language of wider communication.

Interlinearization opens the door for deeper linguistic analysis and lays the foundation for reference grammars, dictionaries, and language learning materials, but interlinearizing naturally occurring speech is not sufficient by itself to create complete grammars, dictionaries, etc. One additional descriptive task is often included during the fieldwork stage of the workflow: the collection of morphological inflection patterns, or paradigms, for several lemmata (task 4). Inflectional paradigms are elicited in documentary work because complete paradigms are rarely found in natural language. Complete lemma-specific paradigms are needed to infer general rules of inflection which

are an important part of any systematic linguistic description.

2.2 Natural Language Processing (NLP) for Low-Resource Languages

Though the line between language documentation and description may not be clear, one thing is clear: current methods do not scale up well. Documentary and descriptive projects often archive only partially accessible corpora, meaning that only a part of the corpus has any annotation beyond transcription. Without annotations such as translations, morpheme segmentation, and glossing, the documented and transcribed data is only understandable to someone who already speaks the language. If no speakers are left, the data is inaccessible, much like Egyptian hieroglyphics before the Rosetta Stone was discovered. Corpora are only partially annotated simply because funding and time are often not sufficient to complete interlinearization (Cox et al., 2019). Current methods assume primarily manual work which is prone to human error and inconsistencies. Baldridge, Palmer, and others note that manual work is extremely inefficient and that the typical strategy of annotating texts from top to bottom is non-optimal for training a supervised machine learning model (Baldridge and Osborne, 2008; Baldridge and Palmer, 2009; Palmer, 2009). Since naturally occurring speech contains many repeated linguistic structures, manual annotation has been described as repetitive, monotonous, costly, and time-consuming (Duong, 2017; He et al., 2016). For example, it can take anywhere from 20 to 100 hours to transcribe (task 2a) a single hour of speech (Seifart et al., 2018). It is reasonable to assume that interlinearization (tasks 2b and 3) and eliciting morphological paradigms (task 4) each require significantly more time than transcription.

A few software tools specially designed for linguistic annotation do provide limited automated assistance for language documentation and description. The two most popular are ELAN (Auer et al., 2010) and FLEx (Rogers, 2010). Examples of their interlinearization interfaces are shown in Figures 2.2 and 2.3. These tools perform automatic morpheme segmentation and glossing by implementing morphological parsers. The parsers require morphological rules that are created by hand. Such parsers do not generalize to new data. In addition to a parser, FLEx has a feature that copies morpheme boundaries and glosses onto other words, but only if the words are identical to

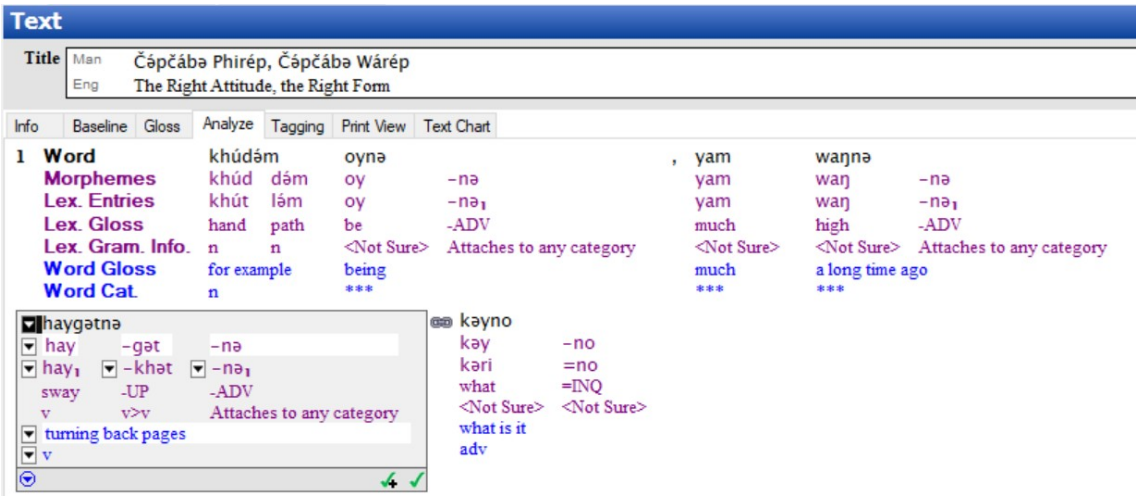

Figure 2.2: User interface for interlinearization in Fieldworks Language Explorer (FLEx) displaying a Manipuri text in the corpus described in Chapter 3

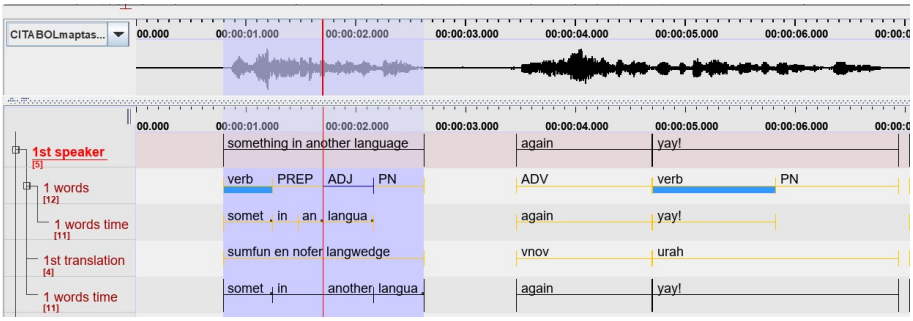

Figure 2.3: User interface for interlinearization in ELAN showing a practice session in English.

words that were previously annotated by hand. Neither tool incorporates machine learning.

A recent growth of interest in low-resource languages[1] has developed models and methods that improve machine learning results with limited data. This includes machine translation (Abbott and Martinus, 2018; Gu et al., 2018; Shearing et al., 2018; Al Mumin et al., 2019; Duh et al., 2020),

[1] According to LORELEI (https://www.darpa.mil/program/low-resource-languages-for-emergent-incidents), "low-resource" refers to languages for which no automated human language technology exists. This is due to a lack of linguistic resources. Szymanski (2012) estimates that 99% of the world's languages are "resource-poor". In linguistics, it is more common to hear other terms. The current work uses the term "under-described languages" to refer to languages with minimal published linguistic resources; these have been called "very scarce-resource language" (Duong, 2017). The term "under-documented languages" (Duong's "extremely scarce-resource languages") refer to languages that lack sufficient raw or annotated data to write a full reference grammar. The term "endangered languages" refers to languages that are predicted to have no native speakers within a generation or two. Most endangered languages are under-documented and/or under-described, as well as fitting the definition of low-resource languages. The distinctions between the terms are rarely crucial in the current work. In practice, these terms can be used almost interchangeably.

computational morphology (Ruokolainen et al., 2013; Baumann and Pierrehumbert, 2014; Micher, 2017; Moeller et al., 2019), syntactic parsing (Baldridge and Garrette, 2013; Duong et al., 2015; Duong, 2017), and automatic speech recognition (Adams, 2017; Anastasopoulos, 2019). Several experiments have demonstrated that these models and methods are both practicable and beneficial for language documentation and description. Machine learning has already been leveraged to develop tools to automate transcription of documentary and descriptive audio recordings. A notable example is ELPIS (Foley et al., 2018), an online tool that includes a user interface accessible to those with no programming background. Machine translation (MT) has been applied to documentary data, using the output of an automatic speech recognition system as input to the MT system (Anastasopoulos et al., 2016; Duong et al., 2016). The potential for integrating machine learning into interlinearization has been clearly demonstrated (Baldridge and Palmer, 2009; Palmer, 2009; Palmer et al., 2010; Xia et al., 2016). For example, Felt (2012) found that automated "pre-annotation" improves human annotators' accuracy if the machine learning model achieves only 60% accuracy and significantly speeds human annotation with an accuracy of 80%. In the area of morphological paradigm induction, the annual SIGMORPHON and CoNLL-SIGMORPHON shared tasks (Cotterell et al., 2016b, 2017b, 2018b; McCarthy et al., 2019; Nicolai et al., 2020; forthcoming) have developed many successful methods to improve learning of inflection with limited training data.

Although NLP interest in low-resource languages has grown noticeably in the past few years, it is not a new area of research. Since the late 20$^{\text{th}}$ century, NLP has taken several approaches to low-resource languages. These approaches can be classified as either rule-based (i.e., finite state transducers) (Cotterell et al., 2015; Forsberg and Hulden, 2016; Moeller et al., 2018, 2019) or machine learning that "learn" rules from data. Machine learning approaches to low-resource languages can be further divided into three types according to whether the training data was annotated completely (supervised) (Bergmanis et al., 2017; Sudhakar and Singh, 2017; Makarov et al., 2017; Liu et al., 2018; Makarov and Clematide, 2018b), partially (semi-supervised) (Ahlberg et al., 2014), or not at all (unsupervised) (Moon et al., 2009; Palmer et al., 2010; Kirschenbaum et al., 2012; Soricut and Och, 2015).

At first glance, unsupervised and semi-supervised learning seem most promising for language documentation and description because they do not require copious amounts of manually annotated data as input. However, even though supervised learning requires annotation, it needs much less data than unsupervised learning and almost always yields better results (Ruokolainen et al., 2013; Cotterell et al., 2015). Additionally, without annotated labels, unsupervised learning can only really cluster data by the latent patterns in the data. Discovering latent patterns might be quite useful for linguists when first exploring the data; for example, frequent character patterns and substrings that a model discovers could provide an initial hypothesis to the linguist about the language's morphological structure. However, no matter how accurate an unsupervised model may be, it cannot substitute the valuable process of manually analyzing and discovering patterns in the data. Detailed analysis of new data is vital for linguists because through that process the linguist becomes familiar with the data and begins to absorb an intuitive knowledge of the language. Nevertheless, the latent patterns discovered by unsupervised models can have many uses such as being leveraged in a semi-supervised approach. Semi-supervised learning combines some supervised data with a larger set of unsupervised data (Kohonen et al., 2010; Poon et al., 2009). This approach may be more suitable than supervised learning if available annotated data is not adequate to effectively train a supervised model. Semi-supervised learning may be ideal for language documentation and description because having substantial amounts of unannotated data with a small amount of annotated data is a common situation, due to the annotation bottleneck. However, like unsupervised learning, semi-supervised learning requires some strong initial hypothesis about the data. These assumptions may hold true in pre-processed datasets but "tend to be violated in real-world data" (Druck et al., 2007). Unfortunately, real applications of semi-supervised learning, specifically for computational morphology, are relatively rare, particularly with neural networks. There are exceptions, such as Ahlberg et al. (2014), where semi-supervised learning was used to induce morphological paradigms in low-resource settings.

Supervised learning is trained on "gold standard" annotated data. It learns the patterns of the annotation labels. A successful model can label new data instances with high accuracy. The new

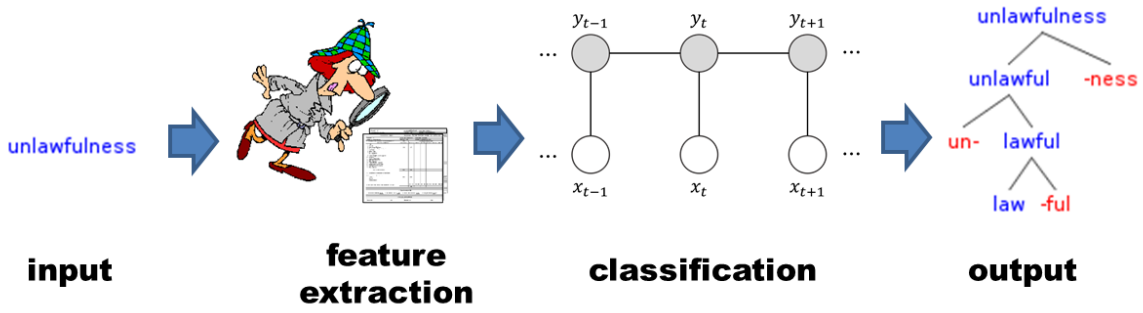

Figure 2.4: Feature-based machine learning requires a human to identify and extract features that a feature-based classification model such as a CRF uses to provide the correct output.

data instance might be a morpheme segment, morpheme gloss, or translation of a word or phrase that was not present in the training data.

Until the 2010s, most machine learning models were feature-based with hand-designed features, illustrated in Figure 2.4. A hand-designed feature function for a task such as morpheme segmentation might have 1) the whole word, 2) the position of the word in the sentence, 3) surrounding words or morphemes, 4) the POS tag of the previous morpheme/word. Features are assigned weights by the model during training to achieve optimal performance according to some objective function such as classification accuracy. These weights put the model's attention on the most helpful features for accurate performance. For example, in a morpheme segmentation task where one chosen feature is the previous word and the previous word is some form of the English "to be" verb, and the target word ends in "ing", then the model might give a high weight to the previous word so that the model pays attention to it when deciding how to segment a word ending in "ing".

The performance of feature-based models, such as the CRF and SVM (see Chapter 3), relies heavily on the manual choice of features. This could be a drawback for under-described languages, because if little linguistic description is available, how does one know which features are optimal for that language? Fortunately, some feature-based models have been shown to perform reasonably well using language-independent features (Ruokolainen et al., 2016; Moeller and Hulden, 2018).

Currently, neural networks models, or deep learning models, are dominating NLP (Goldberg, 2017). Even though they outperform older, feature-based models on almost all tasks, they did not become popular until the mid-2010s because they require greater computing power and, for some tasks, train more slowly (Cotterell and Heigold, 2017). Neural networks, illustrated in Figure 2.5, refers to a family of supervised machine learning models that are composed of layers of statistical units. The layers essentially substitute the feature engineering needed in non-neural machine learning. Multiple embedded layers allow the model to look at an exponential number of "semantically" neighboring instances of each training instance it encounters (Bengio et al., 2003). The layers create intermediate representations of the data that allow the model to "learn" a distributed representation of elements within each instance (e.g., a distributed representation of words within a sentence). This ability of the model to learn requires no (or at most, quite simple) manual feature design. Each unit in each layer is connected to each unit in the adjacent layers. Vector representations of the data are received by an input layer and transformed in "hidden" layers. The hidden layers feed into a final logistic function layer (i.e., softmax) that outputs a prediction of each possible class as a probability between 0 and 1. The connections between layers are represented by learnable weights; the higher the weight the more influence a unit has on the result. Since deep learning is supervised[2] the weights are adjusted with feedback from the gold standard. This is done via stochastic gradient descent or some similar optimization algorithm (Goldberg, 2017) with backpropagation that tells the model how to change the parameters which build the representation of each layer from the previous layer (LeCun et al., 2015).

Until recently, neural networks had the same great disadvantage for language documentation and description that unsupervised learning has. Superior performance required a great deal of data. Until recently data from language documentation and description would have been considered inadequate to train neural networks (Duong, 2017). Even now, a non-neural model can outperform any given neural model that is not tuned to low-resource settings, yet neural models can be difficult

[2] Deep learning morphological segmentation has been performed on unsupervised texts with some success (Wang et al., 2016)

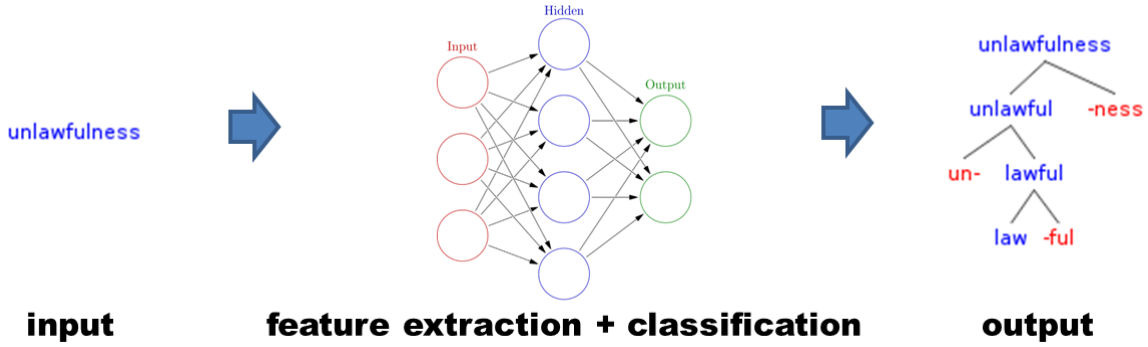

Figure 2.5: Neural Networks, or deep learning, models learn what features in the data are important for giving the correct output.

to optimize and tune for low-resource settings (Popel and Bojar, 2018). Low-resource settings do not always respond to methods that are often effective for improving neural models. For example, fine-tuning hyperparameters or adding hidden layers may sometimes reduce accuracy with smaller amounts of data (Cotterell et al., 2017b; Popel and Bojar, 2018).

New methods are being developed that overcome neural models' dependence on large corpora. Methods include fine-tuning a model to the specific task and input data, training intermediate steps, or augmenting the training data. van Biljon et al. (2020) looked at fine-tuning a model for limited data and determined that shallow- or medium-depth size Transformer models, for example only 3 encoder and 3 decoder layers, give better results at tasks with limited training data. An example of an intermediate training step would be first training a segmentation model to produce surface morpheme breaks and the learn underlying forms of morphemes (e.g., "impossible" → "in-possible" → "NEG-possible") (Cotterell et al., 2016c; Liu et al., 2018; Moeller et al., 2019). A similar strategy could be used in machine translation by first training the model to produce glosses from the source language and then "translating" the glosses into a more correct version of the target language (e.g., from Russian to English: Vecherom ya pobejala v magazin. → evening-INS 1.SG.NOM run-PFV.PST.SG.FEM in store.ACC → In the evening I ran to the store).

Another successful method is augmenting the training data. Data augmentation can be done

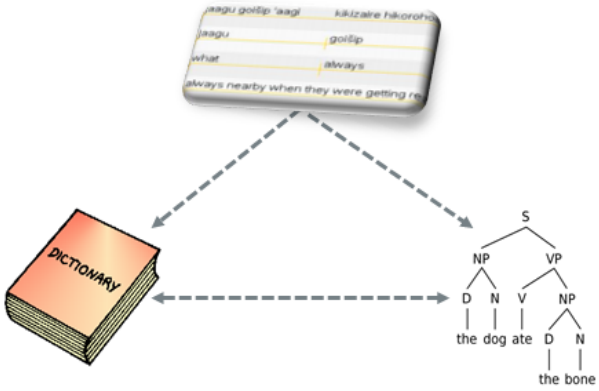

Figure 2.6: The Boasian Triad is a grammatical description, a bilingual lexicon, and a corpus of IGT.

with artificial word forms (Liu et al., 2018) or with information extracted from other resources such as grammars and dictionaries. Leveraging multiple resources to train machine learning models is not impractical if language documentation and description projects have been undertaken in the language. Traditionally, documentary and descriptive projects produce some form of the Boasian triad, illustrated in Figure 2.6.

NLP research in low-resource languages has been growing quite a bit in the past 5-6 years, but most research has been applied to tasks relevant to common NLP goals such as learning to predict unseen inflected forms, and very little has been applied to linguistic research such as discovering and describing the inflectional classes of a language's verbs. One notable exception is the AGGREGATION project (Bender, 2014) which has used IGT to automatically infer grammatical structure for multiple languages, including the construction and visualization of morphosyntax (Lepp et al., 2019; Wax, 2014). Much of their IGT data comes from the Online Database of INterlinear Text (Lewis and Xia, 2010, ODIN) which is a collection of IGTs extracted from published linguistic articles or books. These IGT excerpts that are used in published work differ from IGTs produced by field linguists, and that are used in this dissertation, in at least one important way: noise (i.e., typos, inconsistencies, etc.). Noise is generally removed from the IGT before publication, so ODIN does not have the level of noise that field IGT does.

2.3 Morphological Analysis

Morphological analysis is a key activity in language documentation and description. This is the study of word-building properties and their accompanying (morpho-)syntactic phenomena. Historically, computational linguists and "paper-and-pencil linguists" have taken sometimes seemingly incompatible approaches to morphology (Sproat, 1992; Karttunen and Beesley, 2005). Yet, despite their out-of-sync approaches, both computational linguistics and "traditional" linguistics benefit from the analysis of a language's morphology (Cotterell et al., 2015). Morphological analysis is particularly important when working with morphologically complex languages. Languages that build words from multiple morphemes or via significant morphophonological changes produce a high number of inflected and compound words which appear to the machine as brand new, unrelated words (Dreyer and Eisner, 2011; Goldsmith et al., 2017; Hammarström and Borin, 2011; Kann et al., 2016; Ruokolainen et al., 2013). These include agglutinating morphologies (common in central and north Asia, South America, central and southern Africa, Australia), polysynthetic (North America, the Far East of Russia), or non-concatenative (north Africa and the Middle East, southeast Asia). NLP systems that account for morphology can reduce data sparsity that is caused by an abundance of individual word forms (McCarthy et al., 2019; Vylomova et al., 2020) and help mitigate bias in training data for natural language processing (NLP) systems (Zmigrod et al., 2019). Such systems have often been limited to languages with publicly available structured data, i.e., languages for which tables of inflectional patterns can be easily found, for example, in online dictionaries like Wiktionary.[3] Unfortunately, easily available or complete inflectional tables are not available for many of the world's languages. The current work leverages unstructured data produced by documentary and descriptive field projects.

Morphological analysis can be separated into two core tasks (Cotterell et al., 2015; Hammarström and Borin, 2011; Nicolai and Kondrak, 2017; Palmer, 2009). The first task is identifying morphemes by determining their shapes and marking boundaries between them, as was done for

[3] `https://www.wiktionary.org`

the Lezgi noun in (1b) below. This is known as (unlabeled) morpheme segmentation (Creutz and Lagus, 2007; Snyder and Barzilay, 2008). The second task is deducing each morpheme's meaning, which is known as parsing, or sometimes called morphological analysis by itself.[4] This single step is known in linguistics as glossing, and in computational linguists as labeled morpheme segmentation or, merely, labeling, or tagging.

Together segmentation and glossing make up a significant part of interlinearization in documentary and descriptive linguistics. These two tasks (step 3 of Bird and Chiang's workflow on page 7) are often the most detailed analytical tasks undertaken while still in the field. They are also perhaps the most time-consuming tasks, requiring at least as much, and probably more, time than transcription which can take up to 100 hours for each hour of recorded speech. The linguistic information provided by morpheme segments and glosses lays a vital foundation for subsequent descriptive work.

Many NLP models have been applied to segmentation and glossing of low-resource languages. Automatic morpheme segmentation is commonly traced to the early work of Harris (1955) and much segmentation research since then has implemented unsupervised learning which he inspired (Goldsmith, 2001; Creutz and Lagus, 2002; Poon et al., 2009). The earlier preponderance of unsupervised models was probably motivated by the difficulty of finding the high quantity and quality manually segmented data needed to train supervised models. The lack of sufficient training data is illustrated by a recent supervised segmentation experiment (Ansari et al., 2019) which had to manually segment the Persian corpus before being able to conduct the experiment.

Glossing-only NLP experiments assume that the data is already segmented into morphemes or that it does not need to be segmented. McMillan-Major (2020) trained conditional random field (CRF) systems to produce a gloss line for several high-resource languages and three low-resource languages. The systems incorporated predictions made directly from the segmented line

[4] Nicolai and Kondrak (2017) subdivide morphological analysis slightly differently, making a distinction between morphological "analysis" and morphological tagging. They describe morphological analysis as a combination of segmentation and labeling, though they later state that "morphological tagging can be performed as a downstream application of morphological analysis" (p. 211), thereby adhering to the same two distinctions described above.

and predictions made with information from the free translation line that was enriched with INTENT Georgi (2016). The low-resource language data came from field projects, as does the data in the current work. Both McMillan-Major and Samardzic et al. (2015) used information from other lines of interlinearized texts such as translation and part-of-speech tags, whereas our work assumes the texts have not yet been annotated with any other information.

Segmentation-only NLP experiments may take different strategies. The choice of strategy may depend in part on available data or the type of learning model employed. Unsupervised learning of morphology naturally leans towards surface segmentation (simply indicating segment boundaries in the orthographic representation, rather than determining underlying morpheme shapes). Supervised models depend on annotated data provided by linguists which is preprocessed to reduce inconsistencies. Moeller and Hulden (2018) trained a joint system with highly accurate results on canonical affixes in languages with little allomorphy or morphophonological processes. In languages with more complicated morphophonology and allomorphy—including null morphemes that must be "segmented" and glossed, or circumfixation—the effect of canonical segmentation may be unclear.

NLP experiments with low-resource languages often treat segmentation and glossing as separate tasks. This approach seems to assume that the two tasks are performed sequentially and that it is reasonable to expect morpheme segments to be available before glosses. Some computational models, however, have taken a tip from language documentation and description and joined the two tasks. Joint learning of segmentation and glossing, or labeled segmentation, is less common but has been successful in NLP for low-resource languages (Cotterell et al., 2015; Moeller and Hulden, 2018), usually with non-neural models. In general, joint learning is characterized by training on different types of information and is based on the intuition that one type of linguistic knowledge (e.g., syntax) can improve results in another domain (e.g., morphology) (Goldsmith et al., 2017).

In documentary and descriptive linguistics, the segmentation and glossing are typically tackled simultaneously. It is more likely that general linguistics would define morphological analysis as all and any tasks related to identification of morphemes, their meanings, as well as the description of a language's systematic rules of morphology. (Cotterell et al., 2015). When the two tasks are done

separately, parsing usually refers to labeling each segmented morpheme with a gloss, for example, the OBL (oblique stem) and GEN (genitive case) of the Lezgi noun in (1c). But parsing does not require segmentation. Words can be parsed without identifying morpheme boundaries, i.e., parsing by itself would only provide the information in (1c) but without any indication of morpheme boundaries.

(1) a. paçahdin
    b. paçah-di-n
    c. king-OBL-GEN
    d. 'king's'

2.4 Inflectional Paradigm Induction

Morphology includes the inference of rules that govern a language's word building strategies and the discovery of how word forms are systematically related through derivation or inflection (Roark and Sproat, 2007). Therefore, Virpioja et al. (2011) add a third task to morphological analysis: identification of morphologically related words through patterns of inflection.

Durrett and DeNero (2013) claim that the inference of inflectional patterns must be based on three assumptions. First, each lexical category is dictated by a subsystem of rules. Russian nouns, for example, can be generalized into three simplified patterns of inflection that are usually labeled "masculine", "feminine", and "neuter". Lexemes that adhere to the same pattern are grouped into inflectional classes (sometimes called "declensions" for nouns and adjectives and "conjugations" for verbs). The patterns themselves are known as inflectional paradigms. Second, inflectional changes are triggered by context and, therefore, the patterns can be inferred from context. Descriptive studies look to phonology or else to both phonological structure and the semantic content of the lexeme for the triggering context. Computational models, due to the nature of their input, look to orthographic context. The third assumption is that each stem morpheme is inflected consistently according to the inflectional class it belongs to, plus any idiosyncrasies of the stem.

Monson et al. (2007) give two guiding principles for computational paradigm induction. One is that inflected forms of a lemma will look similar to each other. This principle does not always

	present		past	
	sing.	**pl.**	**sing.**	**pl.**
1 person	am	are	was	were
2 person	are	are	were	were
3 person	is	are	was	were

Table 2.1: Inflectional paradigm of the English verb "to be".

hold because languages abound with exceptions. Also, inflection can be suppletive (e.g., *is* vs. *are*, etc.). However, the principle holds often enough to serve as a solid working assumption.

The second principle is that "in any given corpus, a particular lexeme will likely not occur in all possible inflected forms". Paradigms can be quite large. For example, a typical Polish verb can have 30 or more inflected forms and many languages may have hundreds or even thousands of forms per lemma (Corbett, 2013). Even with a large corpus, attempts to learn paradigms like the one illustrated in Table 2.1 by only using the data in the corpus will often leave empty cells in the paradigm's table. In fact, it is possible that certain forms may never occur in natural language even though they are grammatically possible (Silfverberg and Hulden, 2018). Even if it is possible to find all the inflected forms of very frequent lexemes, frequent words often follow irregular patterns, as, for example, does the English "be". This is why, despite documentary linguistics' emphasis on language in natural use, the language documentation and description workflow (cf. section 2.1) includes elicitation of morphological paradigms (Lupke, 2010; Boerger et al., 2016).

Computational models have successfully learned frequent and regular paradigmatic patterns with high accuracy even in low-resource settings (Hammarström and Borin, 2011; Durrett and DeNero, 2013; Ahlberg et al., 2014). Most early work on paradigm induction applied unsupervised learning to concatenative morphology (Goldsmith, 2001; Chan, 2006; Monson et al., 2007). Semi-supervised models have been more recently applied on concatenative and non-concatenative languages (Dreyer and Eisner, 2011; Durrett and DeNero, 2013).

Supervised learning has also been applied to inflectional morphology. Some work focuses on generating inflected forms, including work motivated by the Paradigm Cell Filling Problem (PCFP),

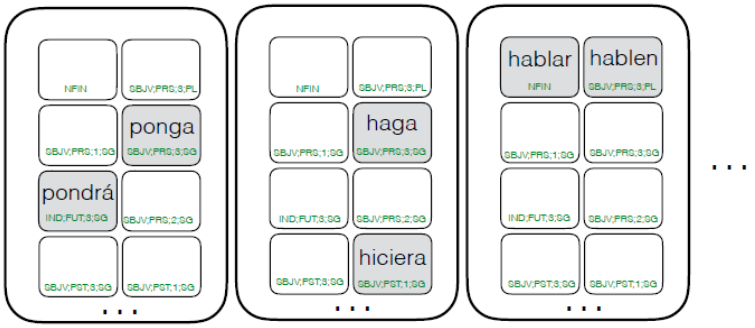

Figure 2.7: Illustration of the Paradigm Cell Filling Problem (Silfverberg and Hulden, 2018) with Spanish verb paradigms.

illustrated in Figure 2.7 (Ackerman et al., 2009). The PCFP is framed as an attempt to model how new speakers (e.g., young children) infer the inflected forms they have not yet encountered (Dreyer and Eisner, 2011; Ahlberg et al., 2015; Malouf, 2016; Silfverberg and Hulden, 2018).

Other work with supervised learning has attempted to induce inflectional paradigms from text. With this method, paradigms are completed by finding overlapping patterns from several incomplete paradigms that are found in text. One method abstracts the longest common subsequence of characters in inflected forms of the same lexeme and then clusters words with same or similar patterns (Ahlberg et al., 2014, 2015). This is illustrated in Figure 2.8. Exceptions or irregularities in the paradigms can be accounted for by collapsing the similar patterns. The experiment has been quite successful for a few Indo-European languages (German, Spanish, Catalan, French, Galician, Italian, Portuguese, Russian), as well as Maltese and Finnish. Paradigm completion work includes Malouf (2016), who trained recurrent neural networks and applied them successfully to Irish, Maltese, and Khaling, among other languages. Silfverberg and Hulden (2018) also trained neural networks for the task. Kann et al. (2017a) differed from other approaches in that they encoded multiple inflected forms of a lemma to provide complementary information for the generation of unknown forms of the same lemma. Finally, Cotterell et al. (2017c) introduced neural graphical models which completed paradigms based on principal parts.

The unsupervised version of the paradigm completion task (Jin et al., 2020) has been the

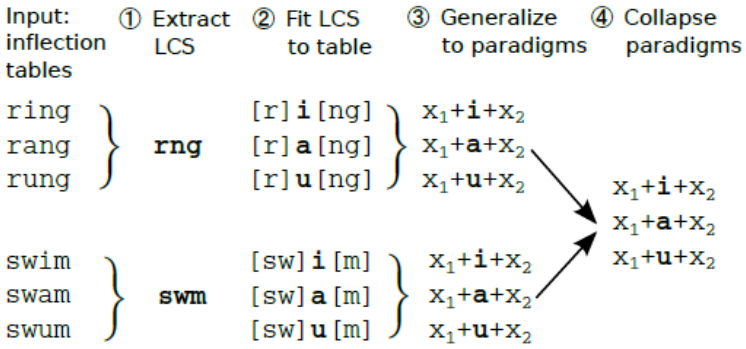

Figure 2.8: Ahlberg et al. (2015): Inducing paradigms. The longest common subsequences (LCS) *rng* or *swm* are extracted (step 1) and represented as x_1 and x_2 which replace the LCS (step 2). Words with the same inflectional patterns will be identical (step 3) and can be generalized into paradigms (step 4). The remaining characters *i*, *a*, *u* are assumed to be inflectional affixes.

subject of a recent shared task (Kann et al., 2020b), with the conclusion that it is extremely challenging for current state-of-the-art systems.

Most recent work in paradigm induction has been concerned with generation (as opposed to analysis) of inflected words and has focused on morphological inflection or reinflection. Approaches include Durrett and DeNero (2013); Nicolai et al. (2015); Faruqui et al. (2016); Kann and Schütze (2016); Aharoni and Goldberg (2017). Partially building on these, other research has developed machine learning models which are more suitable for low-resource languages and perform well with limited data (Kann et al., 2017b; Sharma et al., 2018; Makarov and Clematide, 2018a; Wu and Cotterell, 2019; Kann et al., 2020a; Wu et al., 2021). These are relevant approaches and models for this dissertation, since it aims to aid the documentation and description of low-resource languages.

With formal notation, we can describe the most important generation tasks from the computational morphology literature.

Formally, paradigm of a lemma ℓ can be denoted as:

$$\pi(\ell) = \left\langle f(\ell, \vec{t_\gamma}) \right\rangle_{\gamma \in \Gamma(\ell)} \tag{2.1}$$

where $f : \Sigma^* \times \mathcal{T} \rightarrow \Sigma^*$ defines a mapping from a tuple consisting of the lemma and a vector $\vec{t_\gamma} \in \mathcal{T}$ of morphological features to the corresponding inflected form. Σ is an alphabet of discrete symbols,

i.e., the characters used in the natural language. $\Gamma(\ell)$ is the set of slots in lemma ℓ's paradigm. We will abbreviate $f(\ell, \vec{t}_\gamma)$ as $f_\gamma(\ell)$ for simplicity.

Morphological inflection. The task of morphological inflection consists of generating unknown inflected forms, given a lemma ℓ and a feature vector $\vec{t}_\gamma$. Thus, it corresponds to learning the mapping $f : \Sigma^* \times \mathcal{T} \to \Sigma^*$. Training data consists of lemmas and paradigms.

Morphological reinflection. Morphological *re*inflection is a generalized version of the previous task. Here, instead of having a lemma as input, systems are given some *inflected form* $f(\ell, \vec{t}_{\gamma_1})$ – optionally together with $\vec{t}_{\gamma_1}$ – and a target feature vector $\vec{t}_{\gamma_2}$. The goal is then to produce the inflected form $f(\ell, \vec{t}_{\gamma_2})$. This task is conducted in Chapters 5 and 6.

Paradigm completion. The task of paradigm completion consists of, given a *partial* paradigm $\pi_P(\ell) = \left\langle f(\ell, \vec{t}_\gamma) \right\rangle_{\gamma \in \Gamma_P(\ell)}$ of a lemma ℓ, generating all inflected forms for all slots $\gamma \in \Gamma(\ell) - \Gamma_P(\ell)$. Training data for this task consists of entire paradigms.

Unsupervised morphological paradigm completion. For the *unsupervised* version of the paradigm completion task, systems are given a corpus $\mathcal{D} = w_1, \ldots, w_{|\mathcal{D}|}$ with a vocabulary V of word types $\{w_i\}$ and a lexicon $\mathcal{L} = \{\ell_j\}$ with $|\mathcal{L}|$ lemmas belonging to the same part of speech. However, no explicit paradigms are observed during training. The task of unsupervised morphological paradigm completion then consists of generating the paradigms $\{\pi(\ell)\}_{\ell \in \mathcal{L}}$ of all lemmas $\ell \in \mathcal{L}$.

2.5 POS Tagging and Computational Morphology

Parts of speech (POS), also known as word classes or lexical categories, communicate information about a word, its morphological structure and inflectional paradigm, and its potential grammatical role in a clause. POS tagging is a well-studied problem in NLP. Work on POS tagging has led to the development of several related resources in NLP and linguistics including numerous methods for automatic tagging (e.g., Kupiec (1992); Toutanova and Johnson (2008)) as well as tag sets. The most popular tag set for English was developed by the Penn Treebank Project (Marcus et al., 1993). A universal POS tag set was proposed by Petrov et al. (2012) and has been widely

adopted. It closely follows traditional linguistic conventions for common lexical categories as can be seen by comparing to the Leipzig Glossing Rules (Institute, 2008) which also has recommended tags for less common categories.

Work in computational morphology for low-resource languages generally assumes that other interlinear information is available. POS tags are an example of information frequently assumed to be available. For example, McMillan-Major (2020) and other experiments such as Samardzic et al. (2015) assumed information from lines of interlinearized texts such as translation and POS tags. This assumption is also visible in work on morphological inflection paradigm learning or reinflection, such as in the universal presence of POS tags in work developed as part of the SIGMORPHON Shared Tasks.

Chapter 3

Data and Models

This chapter presents the languages that appear in the following chapters and the details about the corpora in these languages that provide training and test data for the supervised machine learning models. The models themselves are presented in the second part of this chapter.

3.1 Data

This work addresses the question of machine learning integration by experimenting with documentary and descriptive data. Using field data will 1) reveal how current annotation methods affect the performance of NLP systems, 2) test the practicality of incorporating machine learning into the documentary and descriptive workflow, 3) demonstrate that IGT output of documentary and descriptive projects is sufficient to train machine learning models, and 4) enrich the selected data with the results of the experiments.

The selected data corpora are representative of a range of typical documentary and descriptive projects. They consist of manually interlinearized glossed texts (IGT) from the nine under-documented and endangered languages summarized in Table 3.1.

3.1.1 Languages

The selected languages are spoken across five continents. Published linguistic resources are limited, meaning all these languages are under-documented. Information about speaker population and endangerment status was retrieved from the **Ethnologue** (Eberhard et al., 2020) and

Language	ISO	Family	Morphology	Status
Alas	btz	Austronesian	Agglutinative	Stable
Arapaho	arp	Algonquian	Polysynthetic	Severely Endangered
Lamkang	lmk	Sino-Tibetan	Agglutinative	Endangered/Stable
Lezgi	lez	Nakh-Daghestanian	Agglutinative	Vulnerable
Manipuri	mni	Sino-Tibetan	Agglutinative	Vulnerable
Natügu	ntu	Austronesian	Agglutinative	Stable
Southern Sierra Miwok	skd	Utian	Polysynthetic	Nearly Extinct/Dormant
Tsez	ddo	Nakh-Daghestanian	Agglutinative	Endangered
Upper Tanana	tau	Athabaskan	Polysynthetic	Critically Endangered

Table 3.1: Language name, ISO 639-3 code, language family, predominant morphological type, endangerment status.

Catalogue of Endangered Languages (elc, 2020).

Alas (Alas-Kluet, Batak Alas, Batak Alas-Kluet) belongs to the Malayo-Polynesian branch of the Austronesian family. It is spoken by 200,000 people on the Indonesian island of Sumatra. It is unclear if the three dialects (Alas, Kluet, and Singkil) constitute one language. The selected corpus is from the Alas dialect. Its morphology features reduplication, infixation, and circumfixation. The corpus consists of 12 transcribed texts and one set of elicited sentences written with the Indonesian orthography.

Arapaho is an Algonquian language spoken by about 200 people in Wyoming, USA, but is spoken fluently by less than 50 people. It is highly agglutinating and polysynthetic, with the verb carrying the heaviest morphological load (Cowell and Moss, 2008). Polysynthesis in Arapaho includes noun incorporation, where special forms of certain nouns become part of the verb. The corpus contains narratives and conversation from the 1880s until the present day, including a few religious texts that are translations from English. The corpus is in the popular Arapaho orthography. Much of the data is available through the Endangered Languages Archive[1] or the Center for the Study of Indigenous Languages of the West[2] at the University of Colorado.

[1] `https://elar.soas.ac.uk/Collection/MPI189644`
[2] `https://www.colorado.edu/center/csilw/arapaho-language-archives`

Lamkang is a Northern Kuki-Chin language in the Tibeto-Burman family. Depending on the primary source of population numbers, speakers are estimated between 4 to 10 thousand people. Speakers live primarily in Manipur, India but also in Burma (Thounaojam and Chelliah, 2007). Its endangerment status is also not clear. It tends toward agglutination with many stem-stem patterns to signal syntactic categories. Many morphemes are written as separate words in the corpus. There is limited literacy material available. The corpus is at the Computational Resources for South Asian Languages (CoRSAL) digital archive at the University of North Texas.[3]

Lezgi (Lezgian) belongs to the Lezgic branch of the Nakh-Daghestanian (Northeast Caucasian) family. It is spoken by over 400,000 speakers in Russia and Azerbaijan. Lezgi is a highly agglutinative language with overwhelmingly suffixing morphology. The corpus contains oral texts in the endangered Qusar dialect of Azerbaijan. This dialect differs from the standard written dialect in a few ways, such as a locative case suffix borrowed from Azerbaijani which is used alongside the native inessive case suffix with the same meaning. The texts are transcribed into the language's Cyrillic orthography and is the only corpus used in this research that is transcribed in a non-Latin alphabet. The corpus is being deposited at SIL Language and Culture Archives.

Manipuri (Meitei, Meetei) belongs to the Tibeto-Burman branch of the Sino-Tibetan family. It is spoken by nearly two million people, primarily in the state of Manipur, and is one of India's official languages. It has nevertheless been classified as vulnerable to extinction by UNESCO. It is a tonal language and has weakly suffixing, agglutinative morphology (Chelliah, 1997). It is the only Tibeto-Burman language in India with its own script, but the corpus was transcribed with the international Phonetic Alphabet (IPA). The corpus is available through the Computational Resources for South Asian Languages (CoRSAL) digital archive.[4]

Natügu belongs to the Reefs-Santa Cruz group in the Austronesian family. It is spoken by about 4,000 people in the Temotu Province of the Solomon Islands. It has a mainly agglutinative morphology with complex verb structures (Åshild Næss and Boerger, 2008). The corpus contains

[3] `https://digital.library.unt.edu/explore/collections/SAALT/`
[4] `https://digital.library.unt.edu/explore/collections/MDR/`

transcribed narratives and a large written text. Part of the data is available through SIL Language and Culture Archives[5] ; part is being deposited at the Pacific and Regional Archive for Digital Sources in Endangered Cultures (PARADISEC).

Southern Sierra Miwok is a member of the Utian (Miwok-Costanoan) family of central California (USA) (Broadbent, 1964). There are likely no fully fluent native speakers remaining. Verbs inflect for both subjects and objects and have complex derivational morphology with extensive allomorphy. The corpus consists of narratives and conversation from the early 1900s to the 1980s and is archived and available upon request from the Center for the Study of Indigenous Languages at the University of Colorado.[6]

Tsez (Dido) belongs to the Tsez-Hinukh branch of the Nakh-Daghestanian family. It is spoken by about 12,500 speakers in Daghestan, Russia. It has a rich agglutinative, suffixing morphology. The corpus is made up of folklore and is part of the Tsez Annotated Corpus Project (Abdulaev and Abdullaev, 2010). It is available online[7] stored and preserved by Zenodo.org.

Upper Tanana (Nabesna) belongs to the Alaskan sub-group of the Northern Dene (Athabaskan) family. It is one of the official languages of Alaska and the language has been taught at a school in the one village where it is spoken in Canada. Despite its status in both countries, the language is critically endangered, with barely 50 adult speakers living in the eastern interior of Alaska (USA) and the Yukon Territory (Canada) (Lovick, 2020). Its complex morphology features non-continuous lexical, derivational, and inflectional prefixes on verbs. The corpus was collected in 2006-2019 in Alaska; most speakers represented the Tetlin and Northway dialects. The primary data (recordings, transcripts) is preserved at the Alaska Native Language Archive.[8]

3.1.2 Corpora

The data was selected primarily to represent a range of "typical" outputs by documentary and descriptive projects. Less effort was made to represent typological structures, geographic areas,

[5] https://www.sil.org/resources/search/language/ntu
[6] https://www.colorado.edu/center/csilw/arapaho-language-archives
[7] https://tsezacp.clld.org/
[8] https://www.uaf.edu/anla

Language	Tokens	Inflected	Segmented		Glossed		POS-tagged	
Alas	4.5k	412	3,840	86%	3,775	85%	3,845	86%
Arapaho	300k	56,922	202,760	69%	201,915	68%	0	0%
Lamkang	101k	n/a	49,699	49%	50,252	50%	46,557	46%
Lezgi	14k	588	13,625	98%	13,353	94%	13,636	96%
Manipuri	12k	2,192	11,907	98%	11,907	98%	2,067	17%
Natügu	16.5k	1,646	14,136	86%	13,925	84%	10,994	66%
So. Sierra Miwok	10k	n/a	7,422	72%	7,413	72%	0	0%
Tsez	53k	7,315	53,025	100%	53,025	100%	0	0%
Upper Tanana	17.5k	n/a	11,930	68%	11,867	67%	11,198	64%

Table 3.2: Tokens include multiple word expressions (when parsed as such) as single tokens and but do not include English words, proper nouns, digits, and punctuation when they are tagged as such. "Inflected" includes the number of unique inflected forms which are used to train the inflection task in Chapter 5. The percentage and total number of tokens that are segmented or glossed are shown in the next two columns. The segmentation and glossing task in Chapter 4 trains on both. The experiments with POS tags in Chapter 6 used all the tokens that are POS-tagged.

or a range of language families, although some attempt was made to keep an even distribution of morphological types. No attempt was made to include fusional morphology since it is already well-represented in high-resource languages. The texts are mostly narratives that were transcribed from recorded speech. Since the sample of languages is small, the analyses in this work avoid sweeping claims that depend on linguistic typology or discourse genre. The corpora were generously shared in the form of backup XML or CSV files. The rights holders gave informed consent to use the data for research purposes.

Even though most of the corpora resulted from many years of work, they still stand as prime examples of the annotation bottleneck in language documentation and description. More data was recorded and transcribed than could be interlinearized within the project's budget. Each project's unique priorities and workflow resulted in different proportions of fully segmented and glossed data, as shown in Table 3.2. Only the Arapaho and Tsez corpora can be considered completed (though missing annotations were found in both during preprocessing).

Each corpus required preprocessing because of errors, typos, changing analyses, or variable

formatting. Even projects that employed the same software tool had variable formatting. For example, both Natügu and Bahasa Alas have circumfixing, but in one corpus the prefixed part was labeled as a circumfix and the suffixed part as a suffix, while the other corpus took the opposite approach.

Gold standard data for training and testing was produced by filtering incomplete annotations. Only tokens that were completely segmented and glossed, and only sentences that were translated are included in the gold standard for the relevant experiments. The resulting sizes are in Table 3.2.

The projects that produced the corpora spent different amounts of time on interlinearization, had different short-term goals, and varied by team size, team members' education or linguistic training. For example, the Alas corpus was produced in a matter of months while the Arapaho and Natügu corpora were produced by projects that have extended over many years. These two corpora, as well as the Tsez corpus, were annotated by teams with multiple linguists and native speakers. In contrast, the projects that produced the Alas corpus and most of the Lezgi corpus had one linguist and one or two native speaker annotators. The smaller, shorter projects were usually not able to provide as much training to annotators. Most projects were undertaken with the primary goal of documenting and describing the language, but the shorter-term goal of the Alas project is to support literacy efforts while most of the work on the Lezgi corpus was done to support an MA thesis on verbs. These factors are reflected in the quality of the interlinearization. For example, in the Lezgi corpus only verbs were consistently annotated by a trained linguist. The rest was annotated by a native speaker with minimal linguistic training and, therefore, contains more errors and inconsistencies.

The analyses of results in this research often refers to issues that arise from the annotation process. Variation in the IGT corpora is problematic for comparative analysis. The IGT vary in size and annotation quality. The issues due to size become clear in most cases when comparing results, and the answer from consensus would be that more data is always better. Some of the quality factors are brought to focus in the error analyses and discussions. The less significant factors, such as typos in glosses and inconsistent morpheme segmentation, are ignored or handled during preprocessing.

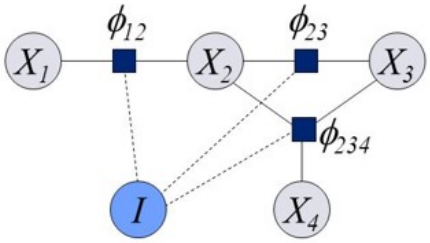

Figure 3.1: Conditional Random Fields sequence classifier.[9]

Some of the more significant issues, such as different segmentation strategies (surface vs. canonical) and varying proportions of POS tagged data are addressed directly in this work.

3.2 Models

The experiments use supervised machine learning systems. All implemented models have achieved state-of-the-art NLP results in low-resource settings. They include the feature-based Conditional Random Fields and Support Vector Machine, and the neural LSTM-based encoder-decoder and Transformer. All neural models have been trained on an NVIDIA GP102 [TITAN Xp] GPU unless otherwise mentioned.

Conditional Random Fields (CRF). The best performing non-neural model for sequence prediction such as morpheme segmentation and glossing is Conditional Random Fields (CRF) (Lafferty et al., 2001; Müller et al., 2013; Ruokolainen et al., 2016). The CRF is a sequence classifier that considers the whole input sequence of symbols (words, letters, glosses, etc.) when making an individual prediction of one output symbol, rather than only consider one input symbol at time. It tries to optimize the probability of a complete sequence of labels, where each individual label is given a conditional probability based on the previous label and an arbitrary number of surrounding inputs. The CRF has performed well on boundary detection (segmentation) and labeling (glossing).

This work implements a linear-chain CRF (Lafferty et al., 2001) with *CRFsuite* (Okazaki, 2007) and its Python API.[10] The training parameters use L-BFGS optimization (Liu and Nocedal,

[10] `https://python-crfsuite.readthedocs.io/en/latest/`

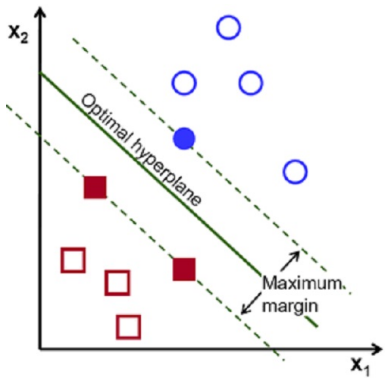

Figure 3.2: A Support Vector Machine (SVM).[11]

1989) and Elastic Net regularization, i.e., a linear combination of L_1 and L_2 penalties. Maximum iterations for early stopping are set at 50.

For the CRF (Fig. 3.1), predictions are scored over the whole sequence and then transformed into a probability distribution. The conditional distribution of the output sequence $\mathbf{y}$, given the input $\mathbf{x}$ can be modeled as

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{Z}\exp\Big(\sum_{i=1}^{n}\phi(y_{i-1}, y_i, \mathbf{x}, i)\Big) \tag{3.1}$$

where ϕ is the feature extraction function which can be expressed through a sum of k individual component functions

$$\phi(y_{i-1}, y_i, \mathbf{x}, i) = \sum_k w_k f_k(y_{i-1}, y_i, \mathbf{x}, i) \tag{3.2}$$

Here, Z is the "partition function" which normalizes the expression to a proper distribution over all possible tagging sequences given an input.

Support Vector Machine (SVM). The experiments in Chapter 4 with segmentation and glossing compare sequential and joint approaches. When feature-based models are used for the sequential approach, words are first segmented into morphemes with the CRF and then glossed with a Support Vector Machine (SVM) (Cortes and Vapnik, 1995). The SVM labels only after segmenting is done by the CRF, using its own feature scheme. The SVM is implemented as a

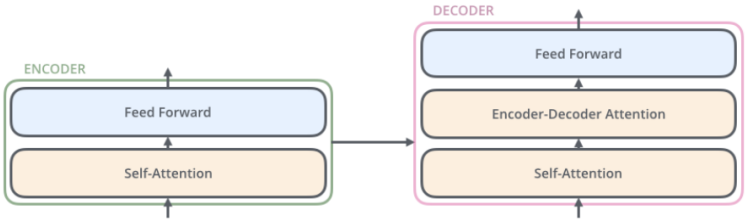

Figure 3.3: The Transformer encoder.[12]

multi-class linear SVM using the LIBLINEAR package (Fan et al., 2008). It creates hyperplanes to separate the data into classes. The optimal hyperplane is found by maximizing the margin between the closest data points in each class, as illustrated in Figure 3.2. These data points are the support vectors. Like CRF, the SVM learns from extracted features.

Transformer. The primary model used in this work is the neural Transformer model (Vaswani et al., 2017) in all three studies. The Transformer is a state-of-the-art neural model architecture for morphological tasks (Vylomova et al., 2020) that has achieved promising results for NLP in low-resource languages (Abbott and Martinus, 2018; Martinus and Abbott, 2019). It is a encoder-decoder model that uses self-attention to boost speed and performance, as shown in Figure 3.3. Attention allows the model to look at different elements in the sequence to help it encode/decode a given element before feeding it to the next encoder. The decoder has an additional attention layer that allows it to focus not only on the input sentence but also on the output sequence up to the element it is decoding. The order of elements in the sequence is represented as an embedding which is input to the encoder/decoder.

In all cases, the model is implemented with the Fairseq[13] toolkit (Ott et al., 2019) with modifications and code (Wu et al., 2021) that have been successful with character-level transduction for morphology learning in low-resource settings.[14] The parameters employed were $N = 4$ layers for the encoder and the decoder, each with 4 self-attention heads. The embedding size for the encoder and decoder is 256, and the hidden layer size is 1024. A dropout rate of 0.3 for encoding

[13] `https://fairseq.readthedocs.io/en/latest/`

[14] 4 encoder-decoder layers, 4 self-attention heads, 256 embedding size, 1024 hidden size of feed-forward layer, layer normalization before self-attention, decoding left-to-right in a greedy fashion, early stopping after no improvement on the development for 4 epochs

and beam search with a width of 5 at decoding time was used. The Adam algorithm (Kingma and Ba, 2015) ($\beta_1 = 0.9$, $\beta_2 = 0.98$) was used to optimize the cross-entropy loss with label smoothing (Szegedy et al., 2016) of 0.1.

LSTM encoder-decoder. In Chapter 5 a LSTM with exact hard monotonic attention for character-level transduction (Wu and Cotterell, 2019) serves as a baseline for the Transformer.

Chapter 4

Automated Segmentation and Glossing for Documentary and Descriptive Linguistics

This chapter examines how variations in research design could affect the integration of machine learning for morpheme segmentation and glossing of under-documented languages. Morpheme segmentation and glossing are traditionally the first tasks undertaken in the documentary and descriptive workflow after transcription. Both segmentation and glossing provide essential linguistic information from which deeper analysis can be done. Segmenting words into morphemes can reduce confusion in NLP models that is caused by data sparsity because it clarifies relationships between word forms. Glosses make implicit morphosyntactic structures explicit and accessible for analysis. Segmented and glossed text can be leveraged to improve NLP systems in low-resource settings, such as for machine translation (Shearing et al., 2018; Zhou et al., 2020). Therefore, automating these tasks and integrating that automation as technological assistance for documentary and descriptive linguists would benefit both linguistics and NLP.

Whenever two disciplinary fields are brought together for mutual benefit, different expectations or accepted conventions will also meet. These expectations may seem to clash. This chapter addresses potential clashes that stem from differing approaches to the same tasks in natural language processing (NLP) and linguistic analysis. The approaches are based on, or have led to, differing conventional expectations about methods or data. For example, it is generally expected in NLP that the textual data to be processed will be an orthographic representation, whereas linguists may prefer to work with a morphophonological representation with the goal of processing underlying lin-

guistic forms. Such differences can make interdisciplinary collaboration unnecessarily complicated or perplexing.

When these differences affect overall research design, it is easy to simply choose one or the other conventional approach without testing which choice might make the task at hand more accurate or more efficient for long-term goals. This chapter compares the short-term effect of three pairs of differing expectations which have arisen during the authors' research. The first study examines *how a choice of morpheme segmentation strategies affects NLP performance.* Linguistic theory assumes the existence of underlying, or canonical, morphemes and the segmentation strategy choice is guided by the goal to discover those forms. Canonical segmentation represents morphemes in their theoretical, underlying forms, which allows orthographic changes triggered by surrounding phones to be ignored. This contrasts with surface segmentation which simply inserts segment breaks in the orthographic representation, thereby indicating surface segments or "morphs" (Virpioja et al., 2011). Since NLP almost always deals with orthographic representations, its systems are trained to perform surface segmentation almost exclusively. It might seem reasonable that linguists who want to integrate automated assistance should adjust their strategy to match NLP expectations. But without testing, are we sure that NLP systems perform better at one strategy over the other?

The second study asks *whether morpheme segmentation and glossing should be approached jointly or sequentially.* In other words, is an NLP system trained to do segmentation and glossing simultaneously as a joint task better than a system trained to treat them as two separate tasks? Instead of arbitrarily choosing one or the other method, we should test whether one approach achieves more accurate results on its task. If the sequential approach is more accurate, then linguists might want to consider adjusting their workflow in order to gain optimal benefit from NLP integration, but if the joint task approach performs better, then perhaps NLP scientists would benefit by adjusting their approach to match how linguists produce new language data.

The third pair of differing expectations is not between linguistics and NLP but within NLP. This study looks at *whether a state-of-the-art deep learning model can outperform feature-based models.* Until recently, feature-based models regularly outperformed deep learning models in low-

Language	Tokens	Seg/Gloss	
Alas	4.5k	3,775	85%
Lezgi	14k	13,353	94%
Manipuri	12k	11,907	98%
Natügu	16.5k	12,435	75%
Upper Tanana	17.5k	11,867	67%

Table 4.1: The percentage and total number of tokens in the corpora that are both segmented (canonical and surface) and glossed.

resource settings. For example, previous work on Lezgi (Moeller and Hulden, 2018) used the same corpus as the current work does and found the CRF outperformed the then state-of-the-art LSTM.[1] Yet, as deep learning models improve, it is becoming less certain which type of model is best in low-resource settings, so it is still important to compare the two types. Linguists may want to know what models to recommend when they begin collaboration with computer scientists in NLP. Additionally, if feature-based models are consistently more accurate, then both linguists and NLP scientists will want to know what features can be extracted for best results across multiple languages.

Only four corpora that provided both surface and canonical morpheme segments could be used for all experiments. The relevant statistics of these corpora from Table 3.2 are repeated in Table 4.1. Results from other languages are included in the tables of results in this chapter when they are available; however, results from the four languages in Table 4.1 are the primary focus of the error analyses and discussions.

The rest of this chapter describes the experiments that test segmentation and glossing systems and compare the results between a surface and canonical strategy, between a joint and sequential approach, and between feature-based and deep learning models. The experiments are described in section 4.1. Their results are presented in section 4.2. The deep learning results are analyzed and discussed in subsection 4.2.4 and the feature-based models are discussed in subsection 4.2.3.

[1] A direct comparison cannot be made to this work because in that work only affixes were glossed while the current work also glosses roots.

4.1 Experiments

All tasks are treated as a problem of converting an input sequence of characters $\mathbf{x} = (x_1, \ldots, x_n)$ to an output sequence of labels $\mathbf{y} = (y_1, \ldots, y_n)$. The output sequence of labels indicates the (canonical or surface) morpheme and/or the morpheme's gloss. Pilot work showed that when the context of the whole sentence was provided during training performance decreased, so each input data instance is a word (or in the case of the SVM, a single morpheme because training on one morpheme at a time increases accuracy with the SVM).

It was assumed that the documentary and descriptive field data had only been segmented and glossed. No other information was leveraged from the IGT or other resources. Gold standard data was created by filtering out tokens that were not completely segmented (both canonical and surface) or glossed. This was determined by assuring that the surface, canonical, and gloss lines aligned with each other. Glosses were standardized by capitalizing affix glosses. When morphemes had multiple English words or symbols in their glosses, the words were joined by periods. Morpheme boundary markers such as hyphens (-) and equal signs (=) were preserved to distinguish clitics from bound morphemes and to indicate relative ordering of morphemes (i.e., pre-/suf-/infixing); angle brackets ($\langle\rangle$) were used to denote circumfixes.

Ten percent of each corpus was withheld as a test set on which the experiments could be compared. The rest of the data was divided into two equal parts. This accommodated accommodate a simple experimental setup for the sequential approach. One part is used to train the segmentation model and joint model while the second part is used to train the glossing model. Ten percent of each part was used as a development set for the Transformer. For easier comparison, the joint models were trained on only one part of the data; this part is the same part used for the segmentation-only step in the sequential approach. For each experiment with the Transformer model, a ten-fold cross validation was run, but since the CRF took significantly longer to train, the feature-based models were run only once.

4.1.1 Surface vs. Canonical Segmentation Strategies

This experiment compares the Transformer's performance when trained on different segmentation strategies. Canonical segmentation gives more information about a language's underlying morphological structure, but at the same time, it reduces the number of unique labels and reflects allomorphy and morphophonological processes that might not be represented in the orthography. On the other hand, surface segmentation does not require the computational models to learn allomorphy or morphophonology (Goldsmith et al., 2017) but also does not provide a thorough analysis of the language's morphology. Surface segmentation simply divides the strings of surface text into surface segments known as "morphs" without regard to potential relationships between them.

The two segmentation strategies are compared in (2) where the first two surface letters of each word in (2a) are represented by identical canonical segments in (2b). In practice, both strategies are encountered during language documentation and description, the initial strategy depending in part on software tools. For example, the older, but still popular, Toolbox[2] allows surface segmentation whereas ELAN (Auer et al., 2010) supports both but as separate tasks, while FLEx (Baines, 2018) requires surface segmentation but facilitates simultaneous canonical segmentation.[3]

(2) a. il-legal in-capable im-mature

b. in-legal in-capable in-mature

c. NEG-legal NEG-capable NEG-mature

The intention of this study is not to provide a direct comparison between models trained on the two strategies because technically the surface and canonically segmented data are different datasets. The study assumes that if one strategy was conducted first, then the other type of segmentation might be more easily learned from it. Therefore, if one strategy consistently results in a more accurate model, it might serve linguists well to adopt this strategy at an earlier step in the documentary and descriptive workflow, because the output of an accurate model at one strategy

[2] https://software.sil.org/toolbox/

[3] The Arapaho and Southern Sierra Miwok corpora could not be used for this experiment because they were annotated in Toolbox.

might be leveraged to automate the other segmentation strategy and this approach might be more efficient in the long-term. In other words, if a corpus could be surface segmented with extremely high accuracy automatically, then first predicting surface segments for the whole corpus might make it easier and faster to later discover the canonical, underlying morphemes. This approach would also match the conventional expectation in NLP.

The model inputs for both strategies do not change. The input is a single word as a sequence of letters, as show in (3a). The difference between the two machine learning models is their outputs. An English example of the output for surface segmentation is shown in (3b) and the corresponding output for canonical segmentation is in (3c).

(3) a. **SEGMENTATION INPUT:** t a x e s

b. **SURFACE OUTPUT:** tax#levy -es#PL

c. **CANONICAL OUTPUT:** tax#levy -s#PL

For this study, only the corpora that had been interlinearized with FLEx were selected because FLEx allows the annotators to provide both surface and canonical segments, but the other methods used only one or the other strategy. Even though those five projects employed the same software tool, the corpora still had differences in formatting. For example, in FLEx only one part of the circumfix is labeled as circumfix, the other part as prefix or affix. Both Natügu and Alas were annotated for circumfixing, but in one corpus the prefixed part was labeled as a circumfix and the suffixed part as a suffix, while in the other corpus the opposite approach was taken. Such variations had to be identified and re-formatted.

The surface morphs and underlying morphemes datasets had to be handled slightly differently for the two strategies. The most obvious difference is the handling of circumfixes. Surface representation preserves the ordering of morphs and does not require knowledge of morpheme types, so the two parts of circumfixes were treated as two different prefix and suffix morphs. On the other hand, canonical segmentation represents the circumfixes as a single morpheme that repeats before and after the stem. The two ways of handling the two strategies are shown in (4).

(4) a. **SURFACE:** ke- STEM -en

b. **CANON.:** ke⟨⟩en- STEM -ke⟨⟩en

4.1.2 Joint vs. Sequential Segmentation and Glossing

Joint versus sequential approaches to segmentation and glossing were tested and compared to see whether joint or sequential segmentation and glossing is a more accurate approach to interlinearization when integrating automated assistance. Joint segmentation assumes that segmented data without glosses is unlikely to be commonly available because when linguists identify and annotate a morpheme it is because they have already determined the morpheme's meaning.[4] Joint segmentation requires the model to learn the morpheme boundary and gloss simultaneously. The sequential approach presupposes that glossing happens after the whole text is segmented. It assumes that segmentation is easier or faster to do than joint segmentation and glossing or that unsupervised segmentation tools such as Morfessor (Smit et al., 2014) are reasonably accurate.

The models of joint learning take an input that is a character-level representation of a word, as shown in (5a). Each character is treated as separate symbol by the model. The output is a sequence of labels, one label per morpheme, as shown in (5b). The label combines the morpheme's shape and gloss, separated by a hashtag or pound symbol; this symbol is chosen because it did not appear in the first eight corpora.[5] This combined label allows the system to learn segmentation and glossing simultaneously.

(5) a. **JOINT IN:** t a x e s

b. **JOINT OUT:** tax#levy -es#PL

The sequential system involves two models. One model learns morpheme segments and the other learns glosses of the predicted morphemes. The first half of the data is used to train the

[4] Field linguist Lindy Pate (p.c.) believed that poorly educated native speakers in Papua New Guinea could segment morphemes without knowing the glosses. In my experience with Lezgi, this is true. It may result in more frequent segmentation errors, but it certainly seems easier for non-linguist native speakers to segment than to gloss.

[5] It was used infrequently in Upper Tanana glosses, but was substituted by NUM.

segmentation step and the segments are predicted for the second half of the data as well as the test set. In the glossing step, it is assumed only these predicted segments have been glossed and can be used for training the sequential system. The output to the first model is a sequence of segments only, shown in (6b). These predicted segments are used as input for the glossing model. The glossing model outputs predicted glosses for the predicted segments, as shown in (6c).

(6) a. **SEGMENTATION IN:** t a x e s

b. **SEG. OUT / GLOSS. IN:** tax -es

c. **GLOSSING OUT:** levy PL

4.1.3 Feature-based vs. Deep Learning Models

Since the mid-2010s, it has been reasonable to expect that deep learning models will outperform feature-based models on any NLP task, except with limited data. With the success of the Transformer (Vaswani et al., 2017) in low-resource settings, this expectation may hold true even in low-resource settings. This third study in segmentation and glossing compares the performance of feature-based and deep learning models. It repeats the experiments described in subsection 4.1.1 and subsection 4.1.2. The only difference is that feature-based models are used instead of the Transformer.

The joint approach to segmentation and glossing is carried out with Conditional Random Fields (CRF) (Lafferty et al., 2001). The sequential approach uses two feature-based models: the CRF for segmentation and a multi-class linear Support Vector Machine (SVM) for glossing.

The input of feature-based models is not a list of letters, but a list of features extracted for each letter in a word. The features were chosen to be cross-linguistically applicable. For the CRF in both the joint and sequential approach the features extracted are 1) the letter as represented in text, 2) the letter in lower case, 3) the whole word as represented in the text, 4) the whole word in lower case, 5) the length of the word as a number of characters, 6-7) position of letter in word

counted from both last and first letter, 8-15) the 1-4 preceding and subsequent letters that surround the current letter.

The sequential approach allows a richer set of contextual features for the glossing-only step. The input to the SVM is a list of features for each predicted morpheme segment. The first features are a concatenation of the above features for each letter in the predicted segment. Then morpheme-specific features are added. They are 1-2) surrounding morpheme, 3) the shape of the current morpheme.

The CRF model gives a sequence of BIO-labels (Ramshaw and Marcus, 1999) as output. Each character in the input is aligned with a Begin-Inside-Outside (BIO) label. This is a type of tagging where each input token is declared either the beginning (B) of a morpheme or gloss, or the inside (I).[6] For the joint task, the BIO label includes the morpheme shape and its gloss, as shown in (7b). For the sequential task only the morpheme shape is included for the segmentation-only step, as shown in (7c).

(7) a. **INPUT:** a v a y d i

b. **JOINT OUTPUT:** B-ava#BE I-ava#BE I-ava#BE B-d#PTP B-i#SBST I-i#SBST

c. **SEGMENTATION OUTPUT:** B-BE I-BE I-BE B-PTP B-SBST I-SBST

BIO-labeling is not used for the glossing-only because each input instance is a single morpheme, and each output is a single gloss. The input and output are similar to that shown in (6a) and (6c), except that there is only one morpheme per input instance. This allows the model to train much faster than when having one word per line and seems to improve results.

These feature-based models require the input and output sequences be of equal length so the number of predicted morpheme segments and the number of gold glosses must match during training of the glossing-only step. When they did not, the number was normalized by adding an 'UNPREDICTED' gloss for every extra predicted segment or by adding a "NULL" feature for every morpheme segment that was not predicted as it should have been.

[6] Since every letter is part of a morpheme the Outside (O) label is not needed as it would be in the BI0-labeling common application in Named Entity Recognition.

	Transformer				CRF (and SVM)			
	Surface		Canonical		Surface		Canonical	
	Joint	**Seq**	**Joint**	**Seq**	**Joint**	**Seq**	**Joint**	**Seq**
Alas	.4280	.4565	.5166	.5291	.5573	.6319	.5792	**.6360**
Arapaho*	.7630	**.7780**	n/a	n/a	n/a	n/a	n/a	n/a
Lamkang	.7091	.7391	.5414	.5785	n/a	**.8376**	n/a	.8197
Lezgi	.5489	.6062	.4993	.5371	.6696	**.7090**	.6518	.6888
Manipuri	.4719	.5067	.6401	.6675	.7766	.8063	.7904	**.8191**
Natügu	.5423	.5263	.6083	.6335	.8388	.8395	.8349	**.8398**
Tsez**	n/a	n/a	.8592	**.8997**	n/a	n/a	n/a	n/a
So. Sierra Miwok*	.6848	**.6982**	n/a	n/a	n/a	n/a	n/a	n/a
Upper Tanana	.7240	.7849	.7459	.7886	.7117	.7942	.7183	**.7970**

Table 4.2: F_1-scores of Transformer and CRF joint and Transformer+Transformer and CRF+SVM sequential models with both segmentation strategies. Transformer scores are an average across a 10-fold cross-validation. CRF and SVM are results of one run. The sequential approach (Seq) results are the average of the segmentation and glossing models' results. Best overall score for each language is bolded. * = a language with only surface segments available. ** = a language with only canonical segments available.

4.2 Results

The system predictions were automatically evaluated against the gold standard test set that was withheld from the corpus. Only the five corpora that were annotated in FLEx could be used for all experiments but scores from the three other languages are included for comparison when available.[7] F_1-scores were calculated as a micro-average on all labels, rather than of each word. F_1-scores are used because they give a better measure of success than mere accuracy when data is as imbalanced as segments and glosses are. The F_1-score provides a harmonic mean of precision (proportion of segments/glosses identified as x that were correctly identified) and recall (proportion of segments/glosses that were correctly identified as x out of those that should have been identified as x).

Table 4.2 displays all F_1-scores. On average all models achieved over 0.60 F_1-score. Only the

[7] *Because of a server crash, the Lamkang CRF model is still training. It will take about a week but should be completed before the April 13 deadline to submit the dissertation to the CU Graduate School. The joint and sequential Arapaho the sequential Southern Sierra Miwok results may or may not be completed by then.*

smallest corpus, Alas [btz], barely scored above that; it only scored higher on the sequential approach with the feature-based models. The larger corpora (Lamkang and Manipuri) scored over 0.70 average F_1 score with the Transformer and over 0.84 with the feature-based models. The feature-based models gave scores just as high with Natügu but the Transformer results were noticeably lower.

The evaluation of the various tasks differed slightly. For the joint task, the scores indicate morphemes that were correctly segmented and glossed. For the sequential systems, the scores are an average of the scores from the segmentation and glossing models. Averaging the results of the segmentation and the glossing models gives a better sense of how well the sequential approach performs when measured on both tasks and provides a more direct comparison to the joint model because both scores—the score of the joint approach and an average of the scores from the two tasks in the sequential approach—measure how well the given approach performs overall on both tasks of segmentation and glossing. The performance of the Transformer was evaluated by a cross-validation on ten random training/development sets with a 9/1 split from each half of the data used for the given experiment. The feature-based models were evaluated on a single run without a development set.

4.2.1 Surface vs. Canonical Results

The differences between surface and canonical segmentation in both the joint and sequential approaches are shown in Table 4.3. The difference in performance between the segmentation strategies on the joint approach is roughly the same as the difference between the strategies on the sequential approaches. The results of each segmentation strategy are not increased or decreased simply because a different approach to segmentation and glossing is taken. In other words, surface segmentation does not do noticeably better or worse on the joint approach than it does on the sequential approach; the same is true for the canonical segmentation. The differences between the two strategies almost disappear with the feature-based models. Doubling the training data also makes the relative performance of the two strategies become nearly the same.

Although the Transformer shows a bigger difference than the feature-based models, the general

	Transformer.		CRF (SVM)	
	Joint	**Seq**	**Joint**	**Seq**
Alas	-.09	-.07	-.02	.00
Alas all	-.01	n/a	n/a	n/a
Lamkang	+.17	+.16	n/a	+.02
Lamkang doubled	+.13	n/a	n/a	n/a
Lezgi	+.05	+.07	+.02	+.02
Lezgi doubled	+.01	n/a	n/a	n/a
Manipuri	-.17	-.16	-.01	-.01
Manipuri doubled	-.02	n/a	n/a	n/a
Natügu	-.07	-.11	.00	.00
Natügu doubled	.00	n/a	n/a	n/a
Upper Tanana	-.02	.00	-.01	.00
Upper Tanana doubled	-.02	n/a	n/a	n/a

Table 4.3: The F_1 differences between the average results on surface and canonical segmentation strategies with the Transformer. Positive numbers mean surface segmentation outperformed canonical segmentation.

trend of both is the same. When comparing segmentation strategies languages with a higher ratio of unique labels to total tokens tend to do better with canonical segmentation. The differences are quite small for Alas [btz], Lezgi, and Natügu [ntu]. The biggest differences are found in Lamkang and Manipuri [mni]; their numbers are the same with the Transformer but the segmentation strategy that performed better at its task is different for each language. If one were to chose the segmentation strategy with the most accurate results, then surface segmentation would be the best choice for Lamkang and canonical segmentation would be the best choice for Manipuri. Interestingly, these two languages have the largest difference of the number of unique labels between surface and canonically segmented data. In Lamkang and Manipuri training data, the average number of unique joint labels increased by over 500 and 400, respectively, and in the segmentation step of the sequential system the number of segments increased by over 350. In the other languages the largest average increase of labels is 88 but usually the differences are less than 15. Since Lamkang and Manipuri belong to the same family, it is possible that significant differences in segmentation strategies are due to

	Transformer				CRF (and SVM)			
	Surface		Canonical		Surface		Canonical	
	Joint	**Seq**	**Joint**	**Seq**	**Joint**	**Seq**	**Joint**	**Seq**
Average	.6090	**.6370**	.6301	**.6620**	.7007	**.7697**	.7149	**.7667**

Table 4.4: Average F_1-scores across all available languages on the Transformer and CRF joint and Transformer+Transformer and CRF+SVM sequential models with both segmentation strategies.

characteristics of their familial morphological structure, but it could be due to other factors such as idiosyncratic choices in the orthographic representation. It is also possible that the differences are due to similar annotation methods as the two corpora were sourced from the same origin.

The relative performance on the two segmentation strategies were compared again, but this time with the joint approach only after training the Transformer on all available data. When all the available data is used, the comparison of the models' performances on the surface and canonical segmentation tasks paints a clearer picture. The differences becomes less noticeable as shown in Table 4.3. Doubling the training data improves overall F_1-scores by about .2 to .4 points. The difference becomes quite small with the doubled data – roughly .1 points or less. This size of difference is closer to the differences between the strategies when using feature-based models trained on only one half the data. At the same time, however, canonical segmentation tends more consistently to achieve better results at its task.

4.2.2 Joint vs Sequential Results

Overall, the sequential system does better than the joint approach to segmentation and glossing, but the difference is not great. The average scores across Alas, Lamkang, Lezgi, Manipuri, Natügu, and Upper Tanana are shown in Table 4.4. The best improvement with the Transformer is slightly over .06 points on Upper Tanana. The best improvement with feature-based models is .08, also on Upper Tanana.

The performance on the Natügu data is the only case where the sequential system does not consistently improve over the joint system. The joint approach outperformed the sequential system

	Surface		Canonical	
	Joint	**Seq**	**Joint**	**Seq**
Alas	+.13	+.18	+.06	+.11
Lamkang	n/a	+.10	n/a	+.24
Lezgi	+.12	+.10	+.15	+.15
Manipuri	+.30	+.30	+.15	+.15
Natügu	+.30	+.31	+.23	+.21
Upper Tanana	-.01	+.01	-.03	+.01
Average	+.21	+.20	+.28	+.17

Table 4.5: The differences between the CRF or CRF+SVM models and the Transformer. Positive numbers mean the feature-based model outperformed the Transformer.

on surface segmentation with the Transformer, but by only .016 points. However, the differences between the various experiments are so slight in Natügu compared to the other corpora that any system or strategy may perform nearly equally well on that language. Interestingly, the Natügu corpus has the smallest difference in the number of unique labels between surface and canonical segmentation (an increase of 14 labels, compared to next lowest of 46). With so few languages, it is difficult to say whether the relative number of unique labels significantly affects performance. More corpora should be included for this question to be explored further.

4.2.3 Feature-Based Results and Discussion

The results of the deep learning and the feature-based models were compared. The differences between the CRF and the Transformer on joint approach and the differences between the CRF+SVM and Transformer+Transformer on the sequential approach are shown in Table 4.5. Transformer results were subtracted from the feature-based results. The feature-based models were evaluated on a single run because less variation is expected and because the models take significantly longer to train. All Transformer results are reported on the average score from a 10-fold cross-validation.

In all cases, except for Upper Tanana, the feature-based models outperformed the Transformer by as much as .31 F_1-score and as little as .01. On average, the feature-based models boost

performance by about .2 points. If only these numbers are considered, the feature-based model would be clearly a better choice for integrating into language documentation and description.

However, other factors need to be considered. Although feature-based models can still outperform a state-of-the-art deep learning model in low-resource settings, they are not necessarily always a better choice because they are more complicated to use. The set-up, feature extraction, and data formatting for feature-based models is significantly more difficult and time-consuming than for deep learning models. The deep learning models are more flexible since they do not require features. For the older models, the features must be determined and extracted manually. This requires some basic knowledge of morphology and extra coding. It also requires time to test various sets of features to find the one that yields best results.

The complications are increased when dealing with the different segmentation strategies. For corpora with only canonical segmentation, there are no surface segments by which morph breaks between letters can be located. This makes it extremely difficult to align the BIO-labels of each letter to the input sequence of letters from the transcription. This is why Table 4.2 displays no feature-based model results for Tsez. This alignment issue is a problem for future work.

Another complication arises from inconsistent segmentations. The CRF input and output sequences must be the same length, one label per letter, while also preserving the surface morph breaks. It turns out that the projects that only surface-segmented (Arapaho and Southern Sierra Miwok) did not always strictly follow that strategy. The surface morphs provide reference for morph breaks which are matched to the appropriate B or I label. When the number of characters in the transcribed word did not match the number of letters in the morphs, those words had to be eliminated during training. The number of words were roughly the same as the size of the development set for the Transformer.

4.2.4 Discussion of Deep Learning Results

A closer look at the results of the Transformer models reveals interesting patterns. One significant factor in system performance is sparsity of data. Unsurprisingly, most errors occur on

rarer forms. The larger class of stems means these are more often segmented/glossed incorrectly than affixes. Another factor is the number of inconsistencies or errors in the manually annotated data. Poor annotation quality can amplify data sparsity.

Allomorphy and isomorphy (same character sequence, different meaning) caused repeated errors during the glossing step and joint learning, where it becomes obvious that the model must deal with multiple options. For example, the Lezgi suffix *-di*[8] has five possible glosses as shown by the joint labels in (8). These morphological phenomena are a moot issue during the segmentation step.

(8) -di#ENT

-di#DIR

-di#ERG

-di#OBL

-di#SBST

Sometimes multiple glosses are not due to morphological structure, but because the same morph(eme) was given different glosses. For example, interchanging 'be' and 'is' and 'COP' for copular verbs or alternating between lexical glosses (e.g., 'you') and grammatical glosses (e.g., '2SG.ERG'). Sometimes different glosses appear because the item can be translated by different English words depending on the context. For example, one Lezgi word can be, and is, translated as 'be' in some context or 'happen' in others. If alternative labels such as *bahaye#danger* and *bahaye#dangerous* are equally frequent, the model must choose randomly. Such inconsistency is to be expected from manual work and could be reduced with more automated assistance from machine learning.

Another pattern of errors is caused by tokens that were only partially segmented (and therefore, not correctly glossed). We knew that many such tokens were included in the gold standard data but there was no reliable way to eliminate them automatically. It is unclear how many exist in

[8] In running text, Lezgi text is transliterated from the Cyrillic orthography for the reader's convenience.

each corpus, although Alas and Natügu seem to have the least. Manipuri [mni] and Lezgi seem to have the most incomplete segmentation. Th4e effect of incomplete segmentation became clear for Manipuri during the experiment described in Chapter 5 when a language expert was asked to the correct the glosses for several inflected words. It appears that, in the original corpus, the annotators had been focused only on segmenting and glossing certain morphemes on each word, leaving other affixes on the word unsegmented. The Lezgi data was annotated by a non-linguist who was trained to use FLEx but did not fully grasp Lezgi's unique morphology or simply did not finish segmenting all words.

Many quality issues unpredictably increase the number of possible labels and amplify data sparsity. An example is repeated misspelling of glosses (e.g., apperance—appereance—appearance, fourty—forty). Other misspellings originate in transcription. In the Lezgi test data, over 50 misspelled or incorrectly segmented strings were found in the first 200 hundred unique segments, although a few spelling changes are representative of dialectical variations.

The results from the Alas corpus were quite good when compared to the much larger corpora. However, the errors are less predictable and more random. It seems likely that the small data set increased the noise in comparison to the consistent and complete annotations and this increase noise ratio obscured general patterns. The Alas model is less likely to get rare morphemes correct. One noticeable confusion was caused by the canonical representation of circumfixes. This is shown in (9) where the model predicted a prefix *n-*. This prefix is a correct surface allomorph of the circumfix at that position. The promiscuous attachment of Alas clitics also seems too difficult to learn with such limited data.

(9) a. **GOLD:** n⟨⟩ken- nindekh -n⟨⟩ken

OUTPUT: n- nindekh -n⟨⟩ken

Nevertheless, error analysis shows that the models deal with data sparsity quite well. Even incorrect segments often have similar character sequences to the correct choice, particularly when the difference is due to a change in the root vowel (e.g., dakhi ∼ dikhi). One of the most interesting

errors, indicating the model's strong ability to learn patterns even in the face of data sparsity, occurred in Lezgi. The transcribed oral speech has a few dozen codeswitched Russian words. The test data include one or two examples, and in one case the model substituted one codeswitched word with another codeswitched word.

Many errors noted during error analysis were not actually errors. Since the annotation was originally done by hand, sometimes by multiple annotators, the glosses varied due to misspellings or synonymous glossing choices (e.g., 'BE.PST' vs. 'was'). There was a clear pattern in all datasets for one of the variants to be predicted rather than a random, unrelated label. These cases would not be considered errors by human annotators but were evaluated automatically as errors in the test data. For instance, one Lezgi demonstrative pronoun was sometimes glossed as 'these' and sometimes as 'this -ABS.PL'. In at least once case, the second (and more linguistically precise) analysis was predicted. Unfortunately, because we did not have access to language experts for every corpus, we were not able to normalize our scores based on this knowledge; however, in the future it may be useful to consider that the performance of models trained on field data may, for all practical purposes, be better than the initial scores indicate.

In other cases, the labels in the test data were evaluated as errors, but closer examination revealed that the original human annotations were incorrect in that particular instance and the predicted label was the best fit to the data. So, a human error had been "corrected". Word instances that had been incorrectly segmented by the human annotators were sometimes correctly segmented by the model, although again these examples were evaluated as incorrect because they did not match the gold standard data. For Lezgi, these examples of "correction" by the model were more frequent in the sequential system and may explain why the biggest improvement by the sequential system over the joint system is found in the Lezgi data, which we know had many incorrect or incomplete segmentations. Again, due to the lack of language experts, we are unable to say whether this holds true for all corpora, but this should be explored deeper in future research.

4.3 Conclusion

This chapter is aimed at smoothing the road to more interdisciplinary work with NLP and linguistics by articulating and examining the results of different research designs. Different research designs arise from different expectations or conventions in the two fields. Although they do not present barriers to mutually beneficial research, different expectations, such as in segmentation strategies, and different workflows, such as joint or separate segmentation/glossing, should not be dismissed when they arise. This chapter tests the possible effects of these two differences.

The small difference between surface and canonical segmentation for three of the five languages suggests either strategy is a useful approach with minimal data, although this changes when data is increased in the joint model. Even though surface segmentation increases the number of labels in a dataset, this appears to be balanced by the abstract character of canonical morphemes, most noticeably by circumfixes. The fact that the difference almost disappears when the data size is doubled indicates that the question of segmentation strategy can be eliminated by simply annotating more data with whatever strategy suits the project at hand. However, larger differences on Lamkang and Manipuri corpora indicate that the reasons why segmentation strategies do sometimes differ in performance on the same corpus should be explored more across other Tibeto-Burman languages. Testing the differences in related languages might indicate whether certain linguistic features influence the results of different segmentation strategies when integrating NLP systems.

The consistent improvement of the sequential system over joint learning may be a reason to consider separating segmentation and glossing tasks in order to leverage the higher accuracy of segments, and a more completely segmented corpus, when glossing the corpus. The strength of the sequential system might be applied when a corpus cannot be completely segmentation and glossed due to budget or time constraints. Instead, a strategy would be to prioritize segmenting and benefit from computational assistance when glossing.

The feature-based models consistently outperformed the deep learning model by up to .3 F_1-score. This might be a reason for linguists and NLP practitioners to prefer feature-based models.

However, better results need to be balanced against the ease of setting up and training a deep learning model like the Transformer.

These studies could serve as a foundation towards more efficient use of computational methods in linguistic analysis and annotation. This chapter shows, for example, that the glossing-only model performs well even on inaccurate segmentation predictions and can even "correct" manual segmentation errors. The study presented here assumes that the model's segmentation is not corrected by the language experts before training the glossing model. If a human-in-the-loop workflow was introduced to first correct segments, then the glossing-only model could improve even more. Such methodological considerations should be tested to see to what extent linguistic analysis and annotation of endangered language might benefit.

Finally, as McMillan-Major (2020) noted in glossing research, consistency of the annotations has a strong effect on system performance. This is most clearly seen in Lezgi which is known to be particularly noisy. Random strange characters were found at morpheme boundaries (e.g., ∗ instead of -). The human annotators frequently segmented one pair of characters whenever it occurred because it matched a frequent suffix. Allomorphs were frequently glossed as if they were different morphemes, undoing the benefit of canonical segmentation. Finally, its unique case-stacking caused confusion both to the human annotator and to the system results. Morphemes with several allomorphs are (incorrectly) glossed one way when they serve as a particular case marker and another way when they are one of several case markers. For example, the second morpheme in (10) is identical to the second morpheme in (11), but in (10) it is a single case marker and should be glossed as 'ERG' but in (11) it is part of a case-stacking sequence and should be glossed as 'OBL. The model often confused such cases of two possible glosses for an identical second morpheme when the gloss depended on context.

(10) a. imi -di
b. 3.SG -ERG
c. 'he/she/it'

(11) a. imi -di -n
b. 3 -OBL -GEN
c. 'his/her/its'

So, what would happen if linguists emphasized quality over quantity? We can answer this question by comparing Lezgi to Alas. According to the accounts of the linguists involved, and evidenced by our experimental results, the Alas data was annotated much more consistently and meticulously. With a corpus one third the size of the Lezgi corpus, the Alas model performs almost equally well. It is possible but seems unlikely that this is due to differing morphological structure. Unlike Lezgi—which is overwhelmingly suffixing and has fairly limited morphophonological changes—Alas features prefixing, suffixing, circumfixing, and infixing with various morphophonological processes.

Interestingly, Alas showed the least marked preference between sequential and joint learning. This may indicate that higher consistency may eliminate the need to consider any change to segmentation/glossing workflow, but it should be investigated with further experiments focused on differences in annotation quality. Preferably these experiments would be conducted on closely related languages to reduce effects due to different typology.

When considering low-resource settings, consistency for machine learning seems more important than data size, strategy, or workflow. Ruthless consistency is not something linguists have had reason to put high value on and it is not something to be expected from manual annotation, Consistency can be provided by machine learning integration, but ironically, supervised machine learning needs high consistency in annotated data before it can perform accurately enough to assist human annotators by increasing their speed or accuracy. Our best estimate of the accuracy threshold for practical integration of machine learning into annotation is 60% (Felt, 2012). This threshold on F_1-scores was passed soundly by Lamkang because the corpus has over 18k manually annotated tokens for training, but it was barely reached by the corpora with 4.5k-5.5k tokens. However, the meticulously annotated Alas corpus came close to this threshold with only 1.5k training tokens. If linguists wish to successfully integrate machine learning into the documentation and description

of under-documented and endangered languages, then they must adopt from NLP an emphasis on highly consistent annotation.

Chapter 5

IGT2P: From Interlinear Glossed Texts to Paradigms

The typical next step in the language documentation and description workflow following morpheme segmentation and glossing is finding and organizing inflected forms that are attested in the transcribed natural speech. Then, because full paradigms rarely occur in natural speech, complete morphological paradigms are elicited. This can be done by using the inflected forms found in IGT to hypothesize and generate new inflected forms that might complete a lexeme's inflectional paradigm. This chapter describes a task that can speed this process and automatically generate the "missing" morphological forms.

This task, which we call **IGT-to-paradigms**[1] (IGT2P), differs from the existing *morphological inflection* (Yarowsky and Wicentowski, 2000; Faruqui et al., 2016) task in three aspects: (1) inflected forms extracted from IGT are noisier than curated training data for morphological generation, (2) since lemmas are not explicitly identified in IGT, systems cannot be trained on typical lemma-to-form mappings and, instead, must be trained on form-to-form mappings, and (3) part-of-speech (POS) tags are often unavailable in IGT. IGT2P can thus be seen as a noisy version of morphological *re*inflection (Cotterell et al., 2016a), but without explicit POS information. Our experiments show that after preprocessing the data existing morphological reinflection systems are strong baselines for this task.

IGT2P generates entire morphological paradigms from IGT input and can be used to generate new morphological resources for natural language processing systems in low-resource settings. NLP

[1] IGT2P was introduced as a new task by Moeller et al. (2020)

systems that account for morphology can reduce data sparsity caused by an abundance of individual word forms in morphologically rich languages (Cotterell et al., 2016a, 2017a, 2018a; McCarthy et al., 2019; Vylomova et al., 2020) and help mitigate bias in training data for natural language processing (NLP) systems (Zmigrod et al., 2019). Over the last few years, multiple shared tasks have encouraged the development of systems for learning morphology, including those that generate inflected forms of the canonical form, or lemma, of a lexeme. However, such systems have often been limited to languages with publicly available structured data, i.e., languages for which complete tables containing inflectional paradigmatic information can be found, for example, in Wiktionary.[2] This limits the development of NLP systems for morphology to those languages for which morphological information can be easily extracted.

IGT2P instead makes use of a resource which is much more common, especially for low-resource languages: we explore how to leverage interlinear glossed text (IGT) to generate unseen forms of inflectional paradigms, as illustrated in Figure 5.1. Field IGT has not been used in NLP for various reasons. One reason is the nature of the data itself which is discussed in 5.3. Another reason is the difficulty of bringing the data to the state found in resources such as Wiktionary or Unimorph.[3] This chapter explores whether having a language expert spend only a few hours cleaning the noisy IGT data improves the task's performance.

Thus, this chapter asks two related questions:

- To what extent can manually interlinearized texts be utilized for computational induction of morphological inflection paradigms?
- How much does manual cleaning of IGT data by a domain expert improve performance?

The first question is answered using existing morphological reinflection models and documentary and descriptive data and the IGT2P task. IGT2P can successfully induce morphological paradigms with 21% to 64% accuracy. The second question is answered by examining which inflec-

[2] `https://www.wiktionary.org`
[3] `https://www.unimorph.org`

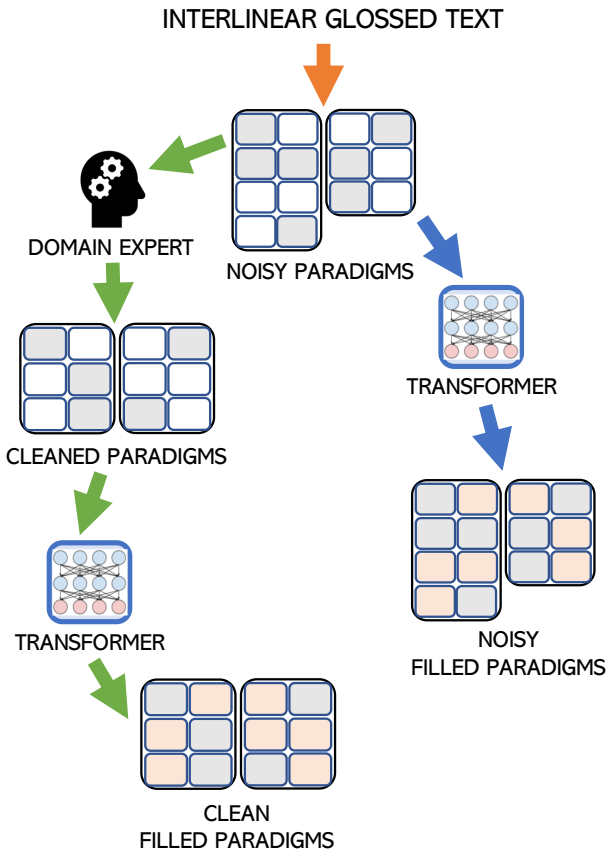

Figure 5.1: Inflected word forms attested in interlinear glossed texts (IGT) train Transformer encoder-decoder to generalize morphological paradigmatic patterns and generate word forms when given known morphosyntatic features of missing paradigm cells. Noisy paradigms are automatically constructed from IGT and a language expert creates "cleaned" paradigms. Both sets are tested on the same missing word forms and the results are compared.

tion model performs better on noisy and cleaned IGT data. Cleaning the data improves performance across the board with a Transformer by 1.27% to 16.32%.

5.1 IGT-to-Paradigms (IGT2P)

The task presented here, IGT-to-paradigms (IGT2P), can be described formally as the paradigm completion problem described in Chapter 2, with an additional step of inference regarding which of the attested forms is associated with which lemma. Formally, during training the systems are given a list of input words with – potentially empty – morphological feature vectors $\mathcal{D} = (w_1, \vec{t}_1) \ldots, (w_{|\mathcal{D}|}, \vec{t}_{|\mathcal{D}|})$ and a list $\mathcal{U} = \{u_j\}$ of $|\mathcal{U}|$ inflected words, $u_j = f(\ell_j, \vec{t}_{\gamma_j})$. The goal of

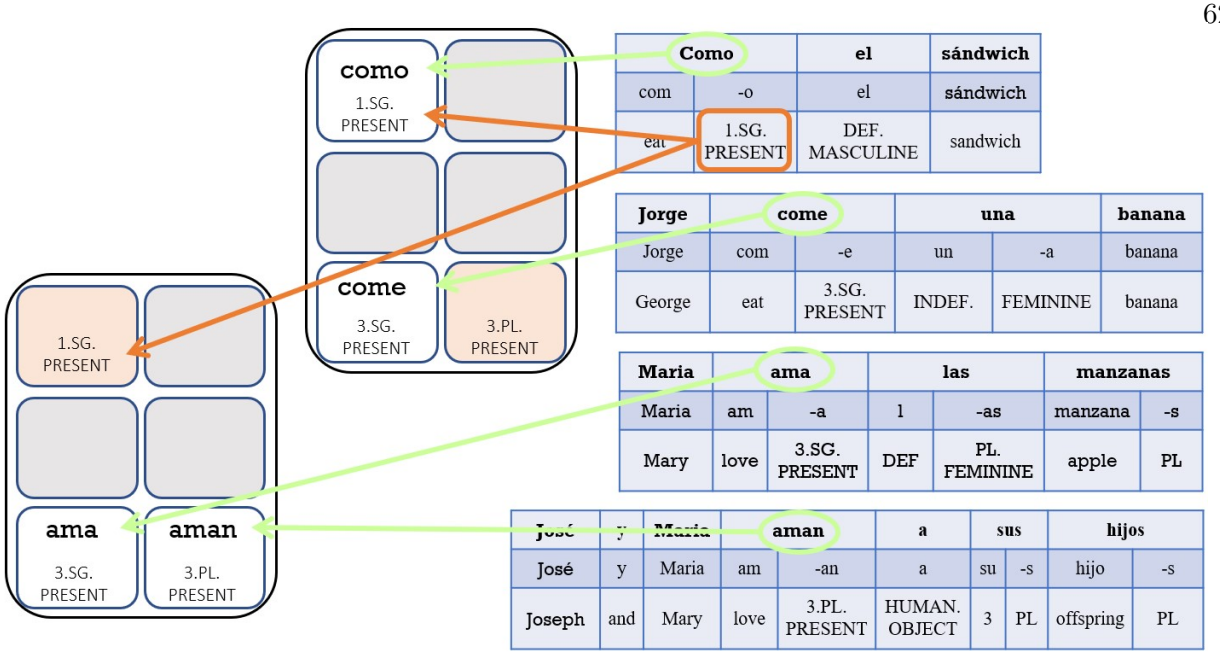

Figure 5.2: IGT2P learns inflectional paradigms by filling paradigm table cells with inflected forms attested in IGT data such as this mini-Spanish corpus gathered from fluent speakers. The grayed cells represent forms that are not attested in the mini IGT corpus and can only be hypothesized to exist in the language. The orange cells also represent forms not attested in the data, but they are known to exist because of the morphosyntactic features of the cell are attested by other lemmas (i.e., *como* 'I eat'). A machine learning model predicts the missing forms (i.e., *amo* 'I love' and *comen* 'they eat') of the orange cells. In documentary and descriptive fieldwork, the predicted forms can serve as a hypothesis that linguists can use to help elicit further information about the language's inflectional patterns.

IGT2P is to generate the paradigms $\{\pi(\ell_j)\}_{f(\ell_j,\vec{t}_{\gamma_j})\in\mathcal{U}}$ from the information available in IGT. This method is illustrated in Figure 5.2.

Like unsupervised paradigm completion, IGT2P does not assume information about the lemma to be explicit. Like morphological reinflection, the input includes word forms with features, and a system must learn to generate inflections from other word forms and morphological feature vectors. IGT2P is further like paradigm completion in that it aims at generating *all* inflected forms for each lemma.[4] Instead of generating paradigms from raw text, IGT2P generates them from IGT, a resource available for many under-studied languages.

[4] Currently this is approximated during evaluation, since gold standard paradigms do not exist for all the languages/dialects. Also, the list $\mathcal{U}$ consists of words in $\mathcal{D}$ but we do not include those in the input so that the words in $\mathcal{D}$ have not been seen by the model.

Text	Vecherom	ya	pobejala	v	magazin.
Segmented	vecher-om	ya	pobeja-la	v	magazin
Glossed	evening-INS	1.SG.NOM	run-PFV.PST.SG.FEM	in	store.ACC
Translation	'In the evening I ran to the store.'				

Table 5.1: An example of typical interlinear glossed text (IGT) with a transliterated Russian sentence, including translation. IGT2P leverages the original text and gloss lines.

5.2 Why IGT2P?

Descriptive linguistics aims to objectively analyze primary language data in new languages and publish descriptions of their structure. This work informs our understanding of human language and provides resources for NLP development through academic literature, which informs projects such as UniMorph (Kirov et al., 2016), or through crowdsourced efforts such as Wiktionary. Yet since most descriptive work is performed manually with minimal, if any, NLP assistance, language resources for thousands of under-described languages remain limited. This includes languages with millions of speakers, such as Manipuri in India.

However, there exists a type of labeled data that is available in nearly all languages that a linguist has documented or described: *interlinear glossed texts* (IGT), illustrated in Table 5.1. They are the output of early steps in a field linguist's pipeline which consist of recording natural speech, transcribing it, and then identifying minimal meaningful units—the morphemes—and using internally consistent tags to label the morphemes' morphosyntactic features. IGTs serve as vital sources of morphological, syntactic, and higher levels of linguistic information. They are often archived in long-term repositories, and openly accessible for non-commercial purposes, yet they are rarely utilized in NLP.

IGT2P has potential benefits for NLP (by increasing available resources in low-resource languages) but also for linguistic inquiry. First, since machine-assistance has been shown to increase speed and accuracy of manual linguistic annotation with just 60% model accuracy (Felt, 2012), such a model could assist the initial analysis of morphological patterns in IGT. Second, by quickly learning morphological patterns from word forms attested in IGT, IGT2P generates forms that fill

empty cells in a lemma's paradigm. Since IGTs are unlikely to contain complete paradigms of lemmas, an accompanying step in fieldwork is that of elicitation of inflectional paradigms for selected lemmas. Presenting candidate words to a native speaker for acceptance or rejection is often easier than asking the speaker to grasp the abstract concept of a paradigm and to generate the missing cells in a table. With the help of IGT2P, linguists could use the machine-generated word forms to support this elicitation process. IGT2P then becomes a tool for the discovery of morphological patterns in under-described and endangered languages.

5.3 Issues specific to IGT

The most notable issue with IGT is the "noise". An inevitable cause is the dynamic nature of ongoing linguistic analysis. As the linguist gains a better understanding of the language's structure by doing interlinearization, early decisions about morpheme shapes and glosses differ from later ones. Another cause is that limited budget and time means IGT are often only partially completed. Another source of noise comes when the project is focused on annotating one particular phenomenon. For example, frequently only one morphosyntactic feature in Manipuri was glossed in each word, meaning different inflected forms looked like they had the same morphosyntactic features. Another source of noise is imprecision introduced by human errors or choices made for convenience to speed tedious annotation. One example of imprecision is glossing different stem morphemes with the same English word. For example, Lezgi has several copula verbs which can be narrowly translated as 'be in', 'be at', etc., but most were merely glossed as 'be'. So, all copula verbs were initially grouped into one paradigm. An analogous situation happened with Arapaho: nuances of meaning were not often distinguished in the glosses; thus, different verb stems are glossed simply as 'give', when they should be divided into 'hand to someone' in one case, 'give as a present' in another case, and 'give ceremonially, as an honor' in third case.

Another issue is that IGT annotators do not usually differentiate between distinct types of morphemes. Thus, we do not always distinguish them. Derivational and inflectional morphemes were only differentiated where we were able to easily identify and eliminate derivational glosses.

Language	paradigms	single-entry	tokens	train	dev	test	unannot
arp clean	16,857	10,857	56,644	283,714	14,151	14,150	6,877
arp noisy	14,389	8,855	56,922	435,430			
btz clean	247	172	386	354	52	52	1,106
btz noisy	235	150	412	575			
ddo clean	982	330	7,221	35,773	2,173	2,172	9,408
ddo noisy	945	295	7,315	36,875			
lez clean	301	202	543	539	88	88	3,054
lez noisy	298	188	588	1,254			
mni clean	479	126	2,860	9,917	853	852	2,593
mni noisy	428	165	2,192	15,958			
ntu clean	316	123	1,654	5,774	473	472	1,661
ntu noisy	365	167	1,646	7,886			

Table 5.2: Data sizes for noisy extracted paradigms and paradigms cleaned by experts. The columns show the total number of inflectional paradigms extracted from the IGT, the number of paradigms with only a single word entry, the number of three-tuples (source, target, features) in the train/validation/test sets before adding unannotated forms and finally the number of additional unannotated and uninflected (unannot) word forms.

For example, in Arapaho we were able to group derived stems into separate paradigms because they were glossed distinctly. Also, clitics are often not distinguished from affixes. This means that the morphological patterns that the models learn are not always, strictly speaking, inflectional paradigms, but it does mean that the models learn all attested forms related to one lemma.

5.4 Experimental Approach

From the IGT corpora described in Chapter 3 this study used Arapaho, Alas, Lezgi, Manipuri, Natügu, and Tsez. As a first step, partial inflectional paradigms were automatically extracted from the IGT. Words were organized into paradigms based on the gloss of the stem morpheme. Then, these stem glosses were removed, leaving only the affix glosses which serve as morphosyntactic feature tags. The data is summarized in Table 5.2.

Step 1: Preprocessing paradigms. The automatically extracted paradigms were preprocessed in two ways. The resulting data is publicly available.[5] In the first preprocessing method, a language domain expert was asked to "clean" the automatically extracted paradigms. Example results are in shown Figure 5.3. Experts reorganized words into correct inflectional paradigms, for example, by regrouping Lezgi copula verbs. They also completed missing morphosyntatic information; for example, adding PL (plural) or SG (singular) where the nouns were otherwise glossed identically. Finally, they removed any words that are not inflected in the language. This usually included words that are morphologically derived from another part of speech but not inflected. For example, an affix might derive an adverb from a noun root, and if the adverbializing affix was glossed, then the word form would have been extracted automatically, resulting in more noise since it displays derivational morphology and no inflectional morphology. Experts were asked to spend no more than six hours on the cleaning task.

For the second preprocessing method, the automatically extracted paradigms were surveyed by a non-expert. Since non-experts could not be expected to identify and correct most issues, they simply removed obvious mistakes such as glosses of stem morphemes that were misidentified as affix glosses and word forms with obviously incomplete glosses or ambiguous glosses (due to identical glosses on one or more word forms). For some languages, this cleaning-by-removal made these paradigms smaller than the "cleaned" dataset. The output of non-expert cleaning is shown in the left-hand column of Figure 5.3.

Step 2: Preparing reinflection data. The typical morphological reinflection data is in tuple format of `(source form, target form, target features)`. The paradigm data is converted into this format in preparation for reinflection. Table 5.2 presents the data sizes.[6]

For each language, the validation and test sets are prepared by using the expert-cleaned data language in the following way: If the paradigm has more than one form, pick a random form as the source form and select the remaining forms in the paradigm with a probability of 0.3 to be

[5] `https://github.com/LINGuistLIU/IGT`
[6] Inflection data available at: `https://github.com/LINGuistLIU/IGT`

вав	SG;AD	вав	SG;AD	вав	SG;AD
ваз	SG;DAT	ваз	SG;DAT	ваз	SG;DAT
вакай	SG;SBSS;EL	вакай	SG;SBSS;EL	вакай	SG;SBSS;EL
		вавай	SG;AD;EL	вавай	SG;AD;EL
bун	SG;ABS	вун	SG;ABS	вун	SG;ABS
вуна	SG;ERG	вуна	SG;ERG	вуна	SG;ERG
		ви	SG;GEN	вак	SG;SBSS
ви	SG;ABS			ви	SG;GEN
				ва	SG;INESS
				вавди	SG;AD;DIR
				вахъ	SG;POES
				вахъай	SG;POES;EL
				вахъди	SG;POES;DIR
				вакди	SG;SBSS;DIR
				вал	SG;SPSS
				валай	SG;SPSS;EL
				валди	SG;SPSS;DIR
				вай	SG;INEL

Figure 5.3: Lezgi paradigms were automatically constructed from IGT (left column) which have typos, incorrect glosses, and wrongly clustered paradigms. Domain experts "cleaned" the automatically extracted paradigms (middle column). These can be compared with the published description (right column) which includes several forms that are rarely used in modern spoken Lezgi.

"unknown", i.e., to be predicted from the first form. Half of the "unknown" data transformed in this way is used for validation and the other half for testing. The validation and test sets for each language are shared across all the experiments conducted for that language.

To create training data from both noisy and clean paradigms, each inflected form is matched with its own morphosyntactic features and mapped to itself. Paradigms with a single entry have only this self-to-self mapping. If a paradigm has more than one form, all possible pairs of inflected forms plus morphosyntactic features within each paradigm are generated, excluding those that are part of testing or validation set, i.e., "unknown".

Step 3: Reinflection models and experimental setup. This experiment compares two state-of-the-art models for morphological reinflection, the Transformer model for character-level transduction (Wu et al., 2021) and the LSTM sequence-to-sequence model with exact hard monotonic attention for character-level transduction (Wu and Cotterell, 2019). For all the models, the implementation of the SIGMORPHON 2020 shared task 0 baseline was used (Vylomova et al., 2020),[7] and the hyperparameters for this study are the same as the shared task baseline.

After paradigms are extracted and preprocessed, two experiments were conducted to generate "unknown" inflected forms. Those experiments were then expanded by two data augmentation techniques. First, all unannotated/uninflected words from the IGT data are added to the training data. When tokens that were either unannotated or uninflected are added, they are self-mapped as source and target forms (exactly as with single-entry paradigms), and their morphosyntactic features are annotated with a special tag: `XXXX`. Second, the training data was augmented by generating 10,000 artificial instances with the implementation in the SIGMORPHON 2020 shared task 0 baseline of the data hallucination method proposed by Anastasopoulos and Neubig (2019). Finally, both additions are combined. These augmentations are intended to overcome data scarcity.

5.5 Results

All models and techniques were tested on the same held-out set chosen randomly from multi-entry paradigms in each language. Results were compared when trained on the noisy paradigms and on the expertly cleaned paradigms. The results are displayed in Table 5.3. With few exceptions, it shows that the limited involvement of experts cleaning the data improves results over training on the noisy data. Also, the best Transformer results surpassed the best LSTM with hard monotonic attention. The primary exception to both these statements is the Alas [btz] results. This perhaps because it is the smallest corpus, roughly half or two-thirds the size of the next largest corpus (Lezgi). Alas also has the poorest results by far of all the languages. Comparing results from augmenting the data by artificial and uninflected/unannotated tokens gave varied results. The factors affecting

[7] `https://github.com/shijie-wu/neural-transducer/tree/f1c89f490293f6a89380090bf4d6573f4bfca76f`

	T	**+aug**	**+uninfl**	**+both**	**mono**	**+aug**	**+uninfl**	**+both**
arp clean	**62.08**	61.39	61.58	60.78	15.93	15.75	15.58	15.94
arp noisy	57.77	57.64	*58.04*	57.51	14.51	14.64	14.52	14.69
btz clean	7.69	3.85	1.92	1.92	1.92	5.77	1.92	1.92
btz noisy	9.62	9.62	13.46	3.85	5.77	13.46	1.92	**17.31**
ddo clean	65.38	**66.53**	65.19	65.42	59.9	60.87	59.53	60.64
ddo noisy	63.54	63.95	62.89	*64.04*	59.12	58.66	57.87	57.97
lez clean	46.59	32.95	46.59	**48.86**	32.95	35.23	31.82	31.82
lez noisy	*35.23*	29.55	32.95	27.27	30.68	28.41	20.45	31.82
mni clean	30.63	30.87	31.81	**32.04**	23.24	25.7	21.95	24.77
mni noisy	21.48	*22.3*	21.60	21.83	18.78	18.31	19.37	20.31
ntu clean	**53.18**	46.82	49.15	48.52	29.66	33.9	28.18	33.05
ntu noisy	36.86	45.55	45.34	*45.76*	31.99	33.69	31.78	30.93

Table 5.3: Accuracy percentages of reinflection task from the Transformer (T) and the LSTM seq2seq model with exact hard monotonic attention (mono) with/out artificial data augmentation (+aug), unannotated/uninflected word forms (+uninfl) and both together. Boldface indicates best result; italics indicate best result on noisy paradigms.

these varied results are unclear although the consistency of the noise and the extent of cleaning may be factors.

There is no clear correlation between accuracy and the total number of annotated tokens or training paradigms (see Tables 3.2 and 5.2). Tsez and Arapaho [arp] achieved over 60% accuracy and these languages do have more training data (35K and 283K triples, respectively) than the others (less than 10K). However, even though Arapaho has considerably more training data, its accuracy is lower than Tsez. A slight correlation between accuracy and amount of multi-entry paradigms does exist. Languages with a higher proportion of multi-entry paradigms tend to have better results. Fewer single-entry paradigms may indicate more complete paradigm information.

Any correlation between results and linguistic factors such as language family or morphological type is uncertain because of the limited number of languages. Tsez [ddo] gave best results overall. This could be due to its limited allomorphy and very regular inflection which may explain why Lezgi [lez], which is closely related to Tsez, performs better than languages with more data. Arapaho's poorer performance could be due to its polysynthetic morphology (Cowell and Moss,

2008) which is more complex than the fairly straightforward agglutination in Tsez (Job, 1994) and Lezgi (Haspelmath, 1993). The models do seem less sensitive to recognizing the word structure in Arapaho. When the first part of a stem is incidentally the same as a common inflectional affix, the stem is often generated incorrectly. This result in a misspelled stem.

The factor that seems most clearly correlated with accuracy is the consistency and thoroughness of IGT annotations. The Arapaho, Tsez, and Natügu [ntu] corpora were noticeably more complete (i.e., most morphemes were glossed) and consistent. This probably explains why Tsez not only had the best results but also showed the smallest improvements after cleaning. Interestingly, augmentation techniques helped these languages the least (only artificial data augmentation helped Tsez slightly). It seems, therefore, that data augmentation is most helpful when original manual annotations are most consistent or complete.

As might be expected with limited data, errors were most common with irregular or rare forms. For example, the best performing model incorrectly inflected many Lezgi pronouns which have an inflection pattern identical to nouns except for an unpredictable change in the stem vowel. Perhaps related to this, the model also misidentified some epenthetic vowels in several Lezgi nouns. Another interesting pattern involved unique Nakh-Daghestanian (Tsez and Lezgi) case-stacking, where nominal affixes concatenate, rather than substitute each other, to form several peripheral cases such as GENITIVE (cf. (11)). The more common affixes in the concatenated strings were often generated correctly but the less common concatenated affixes were not. Allomorphy also causes difficulty. Models tend to generate the right form less often when multiple forms are possible. This seems particularly true for Arapaho which also has many similar forms that mark similar, but not identical, information. For example, in Arapaho similar agreement affixes on verbs were often "confused", such as -oo 'OBJ (of transitive verb with animate object)', -o' '1.SG.ACTOR/3.SG.UNDERGOER (of transitive verb with animate object)', and -' '3.SG.ACTOR (of intransitive verb with inanimate subject)'. All these affixes had a low accuracy of prediction by the model. On the other hand, models learned regular inflectional patterns well enough to correctly inflect forms to the extent that where the expert had left misspellings of that form in the cleaned data, the model "corrected" it.

Finally, we clearly see expert cleaning improved performance across the board (with two negligible exceptions for Tsez and Lezgi on the hard monotonic attention model). Experts were asked to spend no more than six hours and actually spent up to seven but as little as two hours on each language. This indicates that expert labor is well worth its cost.

5.6 Conclusion

We proposed a new morphological generation task called IGT2P, which aims to learn inflectional paradigmatic patterns from interlinear gloss texts (IGT) produced in linguistics fieldwork. We experimented with neural models that have been used for morphological reinflection and new preprocessing steps as baselines for the task. Our experiments show that IGT2P is a promising method for creating new morphological resources in a wide range of low-resource languages.

With sufficient IGT annotations, IGT2P obtains reasonable performance from noisy data. We investigated the effect of manual cleaning on model performance and showed that even limited cleaning effort (2-7 hours) drastically improves results. The inherent noisiness in IGT and other linguistic field data can be overcome with limited input from domain experts. This is a significant contribution considering the extensive effort—on the order of months and years—to produce the curated structured data normally used to train NLP models. In languages with the noisiest data the model's performance is improved even further by data augmentation techniques.

There is room for future improvement. Better techniques for further cleaning might be useful since accuracy seems to be closely related to data quality. However, at some point more cleaning will return less improvement. Upper bounds could be established by comparing results on languages with gold standard inflection tables, although polysynthetic languages like Arapaho would make this difficult since their tables do not always include noun incorporation. Better use of experts' time might involve identification of lemmata that could be used to train a lemma-to-form model, rather than the form-to-form mapping used here. Another approach would be to compare improvements between manual-only cleaning and cleaning done by a linguist working with someone who can write scripts to automatically correct repeated patterns of noise.

IGT2P also has implications for the documentation of endangered languages and addressing digital inequity of speakers of marginalized languages. It could be integrated into linguists' workflow in order to improve the study of inflection and increase IGT data. For example, the generated inflected forms could be used for automated glossing of raw text. IGT2P could speed the discovery and description of a language's entire morphological structure. An elicitation step with native speakers could be added to strategically augment data. This would integrate well with linguists' workflow. IGT2P results could serve to prompt speakers for forms that are rare in natural speech. It might also be integrated into linguistic software such as FLEx.

Chapter 6

To POS Tag or Not to POS Tag: The Impact of POS Tags on Morphological Analysis in Low-Resource Settings

Parts of speech (POS), also known as word classes or lexical categories, communicate information about a word, its morphological structure and inflectional paradigm, and its potential grammatical role in a clause. POS tagging is a well-studied problem in NLP. It is one of the first tasks undertaken for a new data set and a POS tagger is often one of the first NLP resources built for low-resource languages (Yarowsky and Ngai, 2001; Cox, 2010; De Pauw, 2012; Baldridge and Garrette, 2013; Duong, 2017; Anastasopoulos, 2019; Millour and Fort, 2019; Eskander et al., 2020).

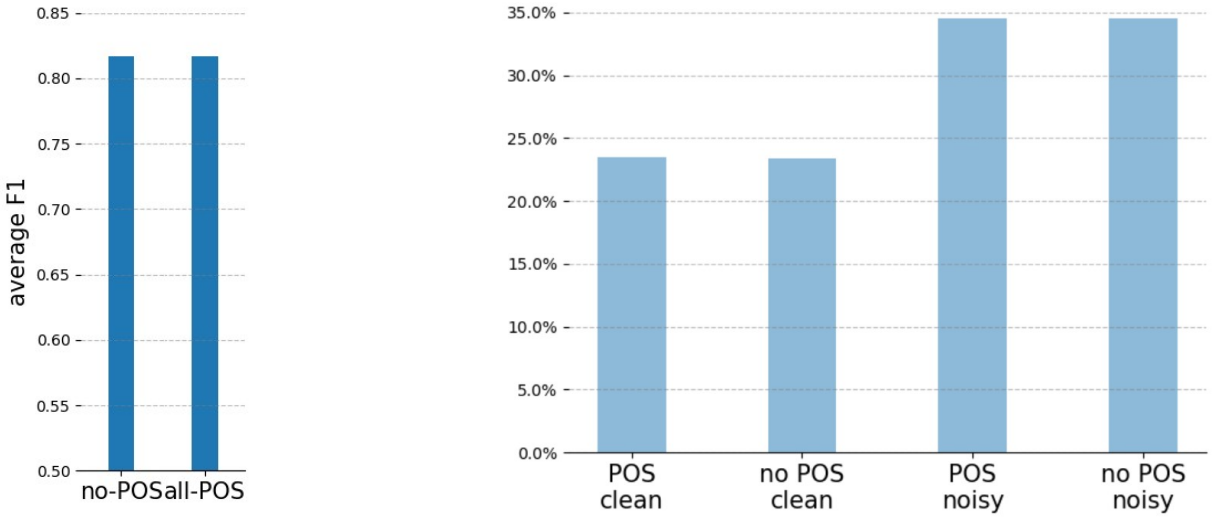

Figure 6.1: Average F_1-scores (right) on joint segmentation and glossing on interlinear glossed texts from fieldwork in five languages found that POS-tags have little and irregular impact. Also, reinflection (left) on four interlinear field corpora and four “cleaned" versions of those corpora who POS tags make minimal significant or consistent difference in accuracy.

Although this priority on early POS tagging may be simply due to relative ease of building a POS tagger, it seems to reflect an assumption that POS tags simplify many other NLP tasks (Krauwer, 2003). As far as we are aware, this assumption has not been methodically tested.

The impact of POS tags on computational morphology may hold implication for linguistic theory. The nature of lexical categories (Rauh, 2010), the criteria for identifying them (Croft, 2000), and even their very reality as a universal property of language (Gil, 2005) are not entirely settled among linguists. If the morphological structure of unseen words can be analyzed and generated without reference to lexical categories, then perhaps such categories should not be considered an inherent property of the lexicon (Rauh et al., 2016).

This chapter examines the impact of POS tags on morphological analysis, a critical area for low-resource languages, many of which are more morphologically complex than English, Mandarin, or other large-resource languages. Morphological analysis also holds high priority in documentary and descriptive linguistics as a necessary foundation for further descriptive work. We focus on two related tasks that both involve morphological analysis: joint segmentation and glossing (segmenting a word into its component morphemes with their glosses) and re-inflection (learning a language's inflectional patterns well enough to generate inflected word forms from morphological features). Since lexical categories are identified in great part through morphological structure, it seems reasonable to assume that knowing a word's part of speech would make it easier to analyze its morphological structure. For example, knowing that a word is a noun makes it extremely unlikely that a final substring *(e)n* could be a participial affix (e.g., *oven* - noun vs. *driven* - verb). On the other hand, POS tags may be providing redundant information when, for example, an affix marking a given inflected feature is identical across all categories whenever that feature appears (e.g., the Russian morpheme *-i* 'PL' is identical for plural nouns and plural verb agreement). However, we must test either hypothesis before claiming it.

Perhaps the impact of (not) having POS tags has not been looked at closely because it seems safe to assume that POS tags or a POS tagger will be available. However, as NLP expands its reach to new languages, POS tags may not be readily available. In fact, the lexical categories present in the

language may not be described yet when data becomes available. In documentary and descriptive linguistics, the description and tagging of lexical categories takes a relatively low priority compared to its place in NLP (cf. Bird and Chiang (2012)'s workflow). Yet interlinear glossed texts (IGT) are sometimes the largest available, or only, annotated resource for a low-resource language.

This chapter describes experiments that were run on corpora differing only in the presence or absence of POS tags. The results, which are generalized in Figure 6.1, indicate that POS tags do not have significant impact on computational morphological analysis. Section 6.1 describes the data that is used. The segmentation and glossing task and results are presented in Section 6.2. The reinflection task and results are presented in Section 6.3. Implications of both experiments are discussed in Section 6.4.

6.1 Data

The study described here uses published data in ten languages and unpublished data from five IGT corpora. The published and unpublished data is used for morphological reinflection but only the unpublished data for segmentation and glossing.

SIGMORPHON/Unimorph Data. As a baseline for the impact of POS tags on the morphological reinflection task the experiment was run on data released for the CoNLL-SIGMORPHON 2018 shared task 1 (Cotterell et al., 2018b). Ten languages were selected that belong to different families and are typologically diverse with regards to morphology. The languages, their families, morphology type, and the inflected lexical categories available for the shared task are listed in Table 6.1. The language family and morphological typology for each family are also available on the UniMorph official website.[1]

IGT Data. Table 6.2 describes the IGT corpora that were selected for this experiment. Only the tokens that were segmented, glossed, and POS-tagged could be used. For the reinflection task, the data was further limited to inflected forms and the corpora that were used in Chapter 5. Both noisy and clean versions of the inflection data were used.

[1] `https://unimorph.github.io`

Language	Family	Morphology	POS
Adyghe	Northeast Caucasian	agglutinative	N, ADJ
Arabic	Semitic	templatic	N, V, ADJ
Basque	isolate	agglutinative	V
Finnish	Uralic	agglutinative	N, V, ADJ
German	Indo-European	fusional	N, ADJ
Persian	Indo-European	fusional	V
Russian	Indo-European	fusional	N, V, ADJ
Spanish	Indo-European	fusional	N, V
Swahili	Niger-Congo	fusional	N, V, ADJ
Turkish	Turkic	agglutinative	N, V, ADJ

Table 6.1: SIGMORPHON/Unimorph languages, families, morphological types, and the lexical categories available in the data.

It is worth emphasizing that the Lamkang (used only for the segmentation and glossing study), Manipuri, and Natügu corpora are the result of many years of work and these extended projects eventually led to a greater proportion of the corpora being POS-tagged. The Arapaho and Tsez corpora, which are large and almost completely segmented/glossed, could not be used for this study because no word-level lexical categories had been tagged. The Lezgi project had annotated POS tags at such an early stage only because the original research was focused on verb tenses (Donet, 2014), so POS tagging was done to identify the verbs. The smaller Alas corpus has POS tags because they were added specifically for this research, as were many POS tags in the Lezgi corpus.

6.2 POS for Segmentation and Glossing

The first study asks whether POS tags make a significant impact on automated morpheme segmenting and glossing. The experiment tests and compares two models on data that is identical expect for the presence/lack of POS tags. Segmenting words into morphemes and glossing (strictly translating) those morphemes are usually the first tasks undertaken after new data is transcribed. Automatic systems could greatly benefit the analysis of endangered languages because current manual methods caused an "annotation bottleneck" (Simons and Lewis, 2013; Holton et al., 2017;

Language	Tokens	Inflected	POS Tags
Alas	3,845	412	**ADJ**, **ADV**, AUX, CARDNUM, CLF, **CONJ**, COP, DEM, DISTRNUM, EXISTMRKR, INTERJ, **N**, NPROP, ORDNUM, **PREP**, **PRO**, PRT, QUANT, REFL, RELPRO, **V**, VD, **VI**, **VT**
Lamkang*	46,557	n/a	ADN, ADVL, DEM, CONN, COORDCONN, COP, INTERJ, N, NPR, NUM, ORDNUM, POSTP, PRON, PTC, QUANT, SUBO, UNK, V, VC, VI, VT
Lezgi	13,636	588	**ADJ**, **ADV**, CARDNUM, CONN, COORDCONN, **DEM**, DET, INDFPRO, INTERJ, INTERROG, MSD, MULTIPNUM, **N**, **NPROP**, **NUM**, ORDNUM, **PERS**, **POSS**, POST, PREP, **PRO**, PROFORM, PRT, PTCP, **RECP**, SUBORDCONN, **V**, **VERBPRT**, **VF**, **VNF**, VOC
Manipuri	2,067	2,192	**ADV**, **INTERJ**, **N**, **PROFORM**, **UNK**, **V**
Natügu	10,994	1,954	**A-D-P2**, **ADJ**, **ADV**, CLAUSE, **CONJ**, **DEM**, **DET**, **GEN**, **GERUND**, INTERROG, INTJ, **N**, **N.(KX.CL)**, NCOMP, **NEG**, **NOM1**, NP, NP(COMP), NPROP, **NUM**, **ORD**, **PARTICLE**, **PCLF**, **PERSPRO**, **PHRASE**, **PN**, **POSSPRO**, **PREP**, **PRO**, **RPRN**, **SUBR**, **UNK**, **V**, **VI**, VP, **VT**, Z-GERUND
Upper Tanana*	11,198	n/a	ADJ, ADV, ADVLIZER, CARDNUM, COORDCONN, DEM, DIR, DM, INTER, INTERJ, IMP, MOD, N, NOMPRT, NPROP, NVP, ONOM, POST, PRO, PROFORM, QUANT, VERBPRT, V

Table 6.2: The number of segmented, glossed, and POS-tagged tokens. The number of unique inflected words. All POS tags were used for the segmentation and glossing task but tags in boldface were found on inflected words. Since only inflected words are used for the reinflection task, only the boldfaced tags are relevant for that task. * = language were not used for reinflection task.

Seifart et al., 2018). Tagging parts of speech would add to that bottleneck; however, if the tags have a significant and positive impact, then linguists may decide to adjust their workflow to receive long-term benefits. Therefore, we explore possible implications of such adjustments by examining the impact of POS tags at very low-resource settings and examining the impact if a field project had time to tag some, but not all tokens.

6.2.1 Experimental Setup

For simple comparisons, the Transformer was chosen for both tasks. Three Transformer models were trained. The English example in (12) shows the input and output of models 1, 2, and 3. Model 1, shown in (12a), has no POS tags. Models 2 and 3 have POS tags, as shown in (12b). Model 2 has POS tags on every word, but Model 3 includes POS tags only for certain proportions of words, simulating projects unable to complete POS-tagging.

(12) a. **INPUT 1:** t a x e s

b. **INPUT 2/3 :** t a x e s N

c. **OUTPUT:** tax#levy -es#PL

All three models are trained on all the available training data. Models 1 and 2 are also trained on different proportions of training data in order to simulate very small corpora. These proportions of training data start at 1% and are gradually increased to 40% of available training data.

Even when POS tags are included in interlinear field data, it is rarely completed as Table 3.2 clearly indicates.To simulate this reality Model 3 was trained on all the available training data but the proportion of inputs with POS tags was gradually and randomly increased.

The training/development/test split is 8/1/1. All models are trained and evaluated on a 10-fold cross-validation. The folds were trained twice, once with and once without POS tags; no other changes were made to the data. All folds were evaluated on a single, consistent held-out test set. Since we wanted to simulate a realistic field situation where the system is segmenting and glossing newly transcribed but unannotated text, the test inputs do not include POS tags.

6.2.2 Segmentation and Glossing Results

POS tags have no consistent positive or negative effect on automated segmentation and glossing in low-resource settings. The overall impact of POS tags is not significant. Table 6.3 shows the differences when F_1-scores without POS tags are subtracted from the F_1-scores with POS tags,

with various amounts of training data. The largest difference is just under .1 point.

A few interesting observations can be made that should be explored with more languages. Manipuri shows the smallest differences overall; it also has the fewest POS-tagged words and the smallest tag set. The largest differences are seen in the Alas and Lamkang corpora. Alas also has a relatively small number of POS-tagged words, but it has quite a large tag set. As the size of the Alas training data increases, the impact of POS tags becomes more pronounced, suggesting that perhaps a relatively large POS tag set may have a greater effect on results in medium settings. Lamkang has the most POS-tagged words, but a sizable number were tagged as UNK. It is not clear whether the UNK tag is limited to categories that have not been fully analyzed or if it is a default tag that covers a diverse set of words. The difference made by adding POS tags all but disappears when all the Lamkang data is trained, suggesting that a smaller data set is more impacted by a large tag set or inconsistent annotations.

Overall, increasing the number of POS tags in the training data has minimal impact. Table 6.4 shows the F_1-scores when the amount of POS tags in the data is gradually increased. For example, at 30%, one random word of every three in the training data has a POS tag. In most cases, having incomplete POS-tagged data hurts performance compared to having POS tags on all words or none. The system either performs worse, or, in the case of Lezgi, makes a small improvement (.0063 points). Except for Lezgi, as more POS tags are added, the system tends to improve slightly but

Language	**1%**	**3%**	**6.5%**	**10%**	**20%**	**30%**	**40%**	**100%**
Alas	.00	.02	.02	.03	.05	.05	.04	-.09
Lamkang	.05	.08	.07	.07	.08	.08	.08	-.01
Lezgi	.03	-.01	.02	.04	.03	.03	.03	.02
Manipuri	-.01	.00	.01	.00	.00	.01	.00	.00
Natügu	.01	.03	.02	.03	.02	.03	.04	.00
Upper Tanana	-.05	.07	-.01	-.02	.00	.00	.00	.00

Table 6.3: The difference in F_1 scores with/out POS tags when training segmentation and glossing on increasing amounts of annotated data, as percentages of total available training data. Negative scores indicate that adding POS tags improves results.

never matches the best scores.

6.3 POS for Reinflection

The second study asks whether POS tags make a significant impact on learning inflectional patterns. This study replicates the CoNLL-SIGMORPHON 2018 shared task 1 of reinflection. Reinflection consists of generating unknown an inflected form, given a related inflected form $f(\ell, \vec{t}_{\gamma_1})$ and a target morphological feature vector $\vec{t}_{\gamma_2}$. Thus, it corresponds to learning the mapping $f : \Sigma^* \times \mathcal{T} \rightarrow \Sigma^*$. The goal is then to produce the inflected form $f(\ell, \vec{t}_{\gamma_2})$. An inflected form is generated when the model is given a related inflected form and the morphological features (which are essentially glosses of affixes) of the inflected form to be generated. In previous work, POS tags have been included by default as part of the morphological features. That is, they have been assumed to be helpful and to be available.

6.3.1 Experiment

Transformer models were trained on individual languages in three different data sets. The first is published Unimorph inflectional data in ten languages. The second data set consists of inflected word forms extracted from unpublished IGT in four languages; the third is the clean, or corrected, versions of the second data set. The Unimorph data was extracted from published data and is the "cleanest". Its inflected forms and morphological features were double-checked, and the forms

Language	**0%**	**1%**	**3%**	**6.5%**	**10%**	**20%**	**30%**	**40%**	**100%**
Alas	.6902	.6448	.6415	.6546	.6517	.6627	.6647	.6708	**.6968**
Lamkang	.8573	.8074	.8195	.8298	.8332	.8482	.8527	.8524	**.8645**
Lezgi	.7501	**.7564**	.7542	.7529	.7505	.7498	.7480	.7471	.7317
Manipuri	.8903	.8885	.8877	.8882	.8889	.8874	.8896	.8897	**.8921**
Natügu	.8995	.8748	.8782	.8864	.8855	.8932	.8999	.8965	**.9006**
Upper Tanana	.8073	.8112	.8086	.8098	.8094	.8073	.8080	.8077	**.8120**

Table 6.4: F_1 scores on segmenting and glossing when trained on all data with increasing percentages of words with POS tags.

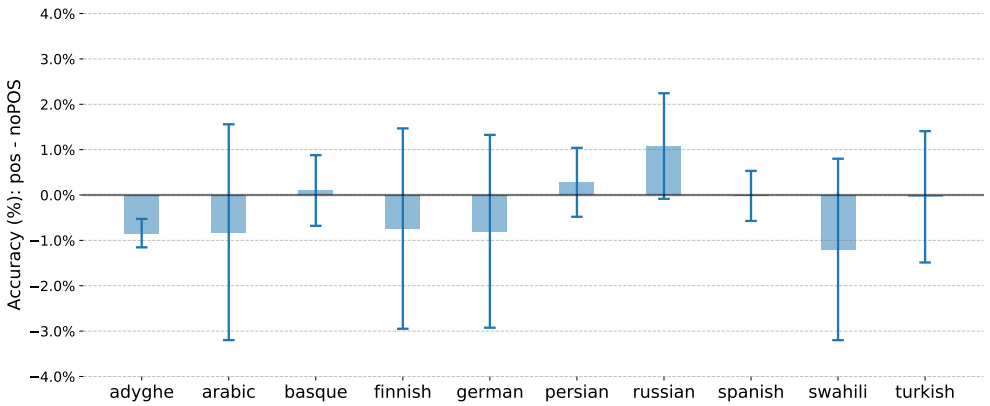

Figure 6.2: The difference in accuracy with/out POS on the reinflection task with SIGMORPHON languages. A negative score indicates that adding POS tags improves results. The bar shows the mean of the differences and lines indicate the range of the mean plus or minus the standard deviation.

provided were selected to provide a balanced picture of the language's morphological structure. The inflected forms extracted from the IGT include only inflected forms attested in original texts which are transcribed samples of natural oral speech. The noisy version was automatically grouped into paradigms based on the assumption that identical glosses of root morphemes denote the same lemma, and therefore the same morphological paradigm. The clean data was made by asking language experts to examine the noisy data and regroup paradigms when root morphemes were incorrectly glossed. The experts also corrected typos and morphological features that were incorrectly glossed.

For the Unimorph data, the original SIGMORPHON training/validation/test splits were kept. The prepared *medium* setting of 1,000 training examples was used. This setting was chosen because of the three possible settings (100, 1k and 10k), it is the closest in size to number of inflected word forms extracted from the four IGT corpora, which provided between 600 and 3,000 training examples. An 8/1/1 training/development/test split was used for the IGT data.

6.3.2 Reinflection Results

Five reinflection models with random seeds were trained on each data set. All models were trained twice, once with and once without POS tags on the input. Crosswise pairs were compared by subtracting the results with POS tags from the results without POS, giving 25 accuracy scores

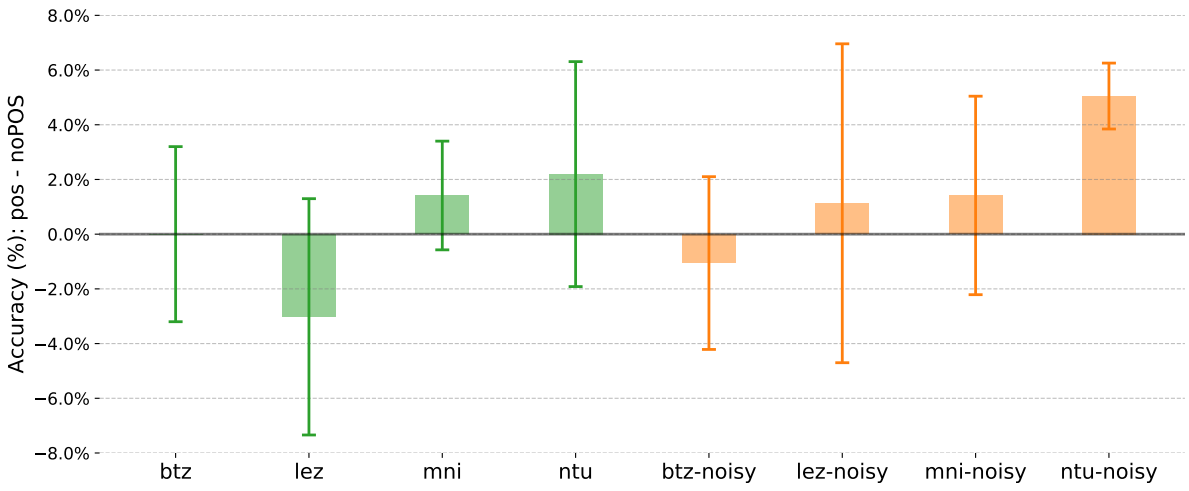

Figure 6.3: The difference in accuracy with/out POS on the reinflection task with cleaned and noisy field data. Negative scores indicate that adding POS tags improve results. The bar shows the mean of the differences and line indicates the range of the mean plus or minus the standard deviation.

per language.

The range of differences shows that POS tags do not have a consistently positive or negative impact. Only two languages show a clear tendency to be impacted in one way. In Natügu, POS tags improve accuracy while in Adyghe, they decrease accuracy. Figures 6.2 and 6.3 show the average and range of differences between the two. The average difference in accuracy on any data set is rarely more than 1 percentage point. As the data becomes less polished, the impact of POS tags increases slightly, and the range of differences grows noticeably. The largest average difference (5 percentage points) is seen in the noisy data from field IGT. This indicates that time invested in polishing existing IGT data may give a better return than time spent on POS-tagging. For the SIGMORPHON languages, the largest mean difference is barely over 2 points and for the clean IGT-extracted data the largest mean difference is about 3 points.

6.4 Discussion

The number of languages we used is not large, but a few general observations can be made. For both tasks, the impact made by the presence or absence of POS tags is minimal. Still, best results with a small corpus are achieved when either all or no tokens are POS-tagged, at least for

segmentation and glossing. This suggests that having a completely tagged corpus is better than an incompletely tagged corpus, so perhaps limited annotation time might be better spent on more segmentation and glossing.

The size or specificity of the tag set may make a difference in the impact of POS tags. When comparing the tag sets in the SIGMORPHON data and the IGT from fieldwork, the difference in the number of lexical categories is significant. The SIGMORPHON data sets have at most three: noun (N), verb (V), and adjective (ADJ). The IGT corpora have larger tag sets; for example, they may have tags for both finite verb forms (VF) and non-finite forms (VNF). The smallest IGT tag set has six categories (Manipuri). That is twice as many POS tags as the SIGMORPHON languages, but still much smaller than the other corpora, which have over 20 unique tags. However, the difference in results cannot be definitely attributed to tag set size. The IGT tag sets are larger because the goal of descriptive work is to discover fine-grained categories, whereas the Unimorph data use more broad categories which are common for language learning material or general dictionaries. Similar fine-grained distinctions appear in the Penn TreeBank tag set and are presumably useful for NLP tasks. Future work could re-tag IGT with more broad categories to test how the size and specificity of POS tags on small corpora impact these tasks. This could be a fruitful area of research because it might help us predict the usefulness of another linguistic category: the category of morphemes. Morpheme-level categories are like POS tags but are tagged for individual morphemes. Interestingly, morpheme categories generally take higher priority than word-level tags in documentary and descriptive linguistics and are therefore more often available in field data.

Finally, although a consistent impact by POS tags cannot be seen on morphological analysis across corpora, some corpora did see a slight impact from the presence or absence of POS tags. Sometimes better results were had by removing POS tags, sometimes by adding them. Reinflection in Adyghe and the "clean" version of Lezgi data tend to benefit by removing POS tags while Persian, Russian, and the noisy version of Natügu generally have more accurate results with POS tags. In segmentation and glossing, Alas and Lamkang show in some settings nearly .1 points difference when POS tags are added and removed, respectively. With these trends, a more interesting question for

these corpora becomes "When are POS tags helpful?" and this should be explored further.

6.5 Conclusion

This chapter investigated the usefulness of POS tags for morphological reinflection because field data does not often include POS annotation but NLP traditionally places high priority on POS-tagging. Surprisingly, POS tags are not beneficial to recent state-of-the-art systems during morphological analysis. This is a useful discovery for researchers who wish to optimize their inflection systems. We conclude that the presence or absence of POS tags does not have a significant impact on two morphological analysis tasks: segmentation and glossing, or reinflection. For computation morphological analysis no clear advantage is gained by tagging low-resource data with POS labels. In segmentation and glossing, the greatest average difference is a loss of .09 F_1-score when a large POS tag set is added to a small field corpus. In reinflection, the overall tendency, though slight, is that accuracy decreases when POS tags are added. The greatest average difference is 1.2 points of accuracy for published data, 2.2 points for unpublished "clean" data, and 5 points for unpublished noisy data.

We hypothesize that POS tags do not have a significant impact on these tasks because the information provided by POS tags is implicitly learned. These are not the only tasks where POS tags could be leveraged for low-resource languages and these conclusions should not be taken as a definitive statement regarding the impact of POS tags in other NLP tasks with low-resource languages, particularly ones that are more syntactic or semantic in nature. However, it does bring into question whether the development of POS taggers and POS tagging should be given the priority they have assumed. This needs to be methodically tested.

Future work should explore how other NLP tasks are impacted by POS tags. The results might influence workflow priorities for documentary and descriptive linguists who want to receive/give benefit from/to NLP. When a sophisticated POS tag set and POS taggers are available for a language, then leveraging POS tags is a trivial matter. However, as NLP expands into a broader range of languages, the usefulness of POS tags may become an important question because documentary and

descriptive linguistics does not currently place a high priority on lexical categories. Discovering a language's lexical categories requires a detailed understanding of the language's syntax–something linguists do not always possess in the initial stages of describing a new language.

Chapter 7

Conclusion

The need to address language endangerment by increasing documentary and descriptive data is great. Integrating machine learning into the process is perhaps our best hope of addressing this with reasonable speed and accuracy. Systems that are reasonably effective in limited data settings already exist, as do archived corpora of documentary and descriptive data that could be used to train those systems. However, the two have barely been leveraged to benefit work in NLP and linguistics. That is why this dissertation explored how such integration might affect current workflows and lead to new methods.

In this dissertation, I examined methods for training machine learning systems on limited data and integrating those systems into the documentation and description of endangered languages, with a specific focus on morphological analysis. Unlike previous studies, I involved more than one or two languages, experimenting with corpora from nine typologically diverse languages. The research results can be described as follows. First, various methods for automating morpheme segmentation and glossing were compared. Treating the two tasks as sequential steps, where a separate machine learning model is trained on each step, does slightly better at both tasks than treating segmentation and glossing as a single, joint step. This indicates that linguists who integrate machine learning may want to plan their project to focus first on segmentation, and after segmentation, to use an accurate segmentation model to assist with glossing. Whether a segmentation model should be trained on surface segmentation rather than canonical, or underlying, segmentation is unclear, and what difference there is between the two segmentation strategies nearly disappears when the amount

of training data is doubled. When comparing feature-based and deep learning models, it turns out that the older feature-based models do better or at least as well as the newer, state-of-the-art deep learning models. Feature-based models outperform deep learning models by as much as .3 F_1-score. Second, a new task was presented for learning morphological paradigms and automatically generating new morphological resources, called: IGT-to-paradigms (IGT2P). Human annotation was integrated into IGT2P and demonstrated a small amount of additional annotation can increase the model's accuracy by 2% to 16%. This implies that machine learning can be used not only for repetitive annotation tasks but also for more complex descriptive work. Third, the presence or absence of POS tags when performing joint segmentation and glossing as well as paradigm induction was examined and shown to have minor impact on either task. The largest average difference across six languages was a mere .09 improvement in F_1-scores by removing POS tags. This indicates that NLP may benefit from documentary linguistic data even when customarily expected kinds of annotation do not yet exist.

The results have implications in several areas. The research provided more usable data in several under-documented languages, leveraged field data for NLP, and tested machine learning model efficacy with limited and noisy data—often the only data available for endangered or low-resource languages. It explored murky issues derived from different approaches in NLP and in linguistics. The integration of machine learning for learning morphological paradigms indicates that NLP tools and methods can be used to build and test hypotheses during structural analysis of a previously undescribed language. The potential to use machine learning to advance linguistic analysis in more languages, in turn, could support the refining of linguistic theories. Furthermore, the methods explored in this research can be applied to any low-resource language, and by extension, to low-resource genres or domains in high-resource languages. Although I have focused mainly on the implications for endangered languages, even very stable languages such as Manipuri [mni] with a million or more speakers have limited annotated resources available and the increase of linguistic data and analytical knowledge will support NLP expansion into those languages and could benefit communities who wish to create language learning or literacy materials.

The three studies described in the previous three chapters discussed future work specific to each study, but this work as a whole has natural future directions that should be considered. All three studies are presented as static experiments, where machine learning is trained on manually annotated data and the model's performance is measured by a single cycle of training and testing. This static approach gives the impression that the real-life application would be limited to spitting out results of a certain accuracy which linguists can accept as is without further interaction or consideration of machine learning integration or the possibility of continued automated assistance. If accurate analysis and annotation is to be completed, then the model's output must be manually vetted and corrected. This situation should not be the end of the story.

The practical implications of integrating machine learning into documentary and descriptive linguistics should be explored by a more realistic iterative cycle of machine learning and manual annotation. Such a human-in-the-loop cycle, sometimes known as active learning, re-trains models on small sets of new manual annotations. For example, a first model could be trained on initial human annotation with the most effective methods as described in this dissertation, but instead of simply returning the model's predictions to the annotators for correcting, a human-in-the-loop approach would be introduced. Another machine learning system would learn to strategically select small (∼50-100) batches of tokens from the predictions. These would be presented with context to the annotator for correcting or vetting. The first model would be then retrained.

A key question is how to select the data to be annotated so that the model will improve very quickly with least manual effort. Tokens for additional annotation should be selected so that they are most informative about the language's structure. Selecting informative tokens to be annotated would allow the model to improve quickly with only small batches of new annotated data. Selection methods based on computational and linguistic motivations need to be tested and compared. Common active learning selection strategies do not integrate linguistic knowledge. Linguists who have begun to analyze a language have gained linguistic knowledge about the languages. Human-in-the-loop experimentation should integrate this knowledge into the strategies for selecting the most informative tokens to be annotated before re-training. For example, a study might compare generic

selection strategies such as uncertainty sampling against strategies based on linguistic factors, such as selecting longer words (assuming they are more morphologically complex and therefore more informative), and then compare against a combination of methods, such as selecting words with substrings found on words about which the machine prediction has high uncertainty.

An iterative approach of annotating, training, strategically selecting new tokens for annotation, and retraining should be done at all stages of the documentary and descriptive workflow, but I will outline here how it might be done with IGT2P (cf. Chapter 5). Active learning would involve first training a model on partially completed inflectional paradigm tables that were filled from forms attested in documentary data, as described in Chapter 5, then comparing results after retraining the model once on data selected by different methods. These methods could be: 1) selecting word forms from the model's predictions that it had least "confidence" about, 2) randomly selecting word forms, 3) selecting a random number of word forms but making sure they are evenly distributed among tables, or 4) selecting word forms so that each table has the same minimum number of forms per table. This same experiment could be conducted for joint segmentation and glossing. The first two methods would remain the same, but other methods could be 3) selecting a few less frequent word types, 4) selecting word forms based on statistically rare letter sequence combinations (in an attempt to identify rare morphological combinations), or some other method motivated by linguistic facts of the language such as 5) annotating only nouns in Lezgi which are most likely to have irregular allomorphy. Each strategy could be compared on immediate improvement after one batch of annotation and then after multiple runs in order to determine short-term and long-term effect on the performance curve.

Eventually, the active learning, or human-in-the-loop experimentation, will of course need to involve actual human annotation. However, that introduces complications related to levels of expertise, and various speeds of annotation, and expense of hiring annotators. Therefore, the first experiments should use pre-annotated data as a stand-in for iterative human annotation. Once the most effective strategy, or combination of strategies, is found, then human annotation should be introduced to study the effect of human pace and (in)accuracy.

Finally, a human-in-the-loop approach should be tested *in situ*, as it would be used in a documentary and descriptive project, perhaps during fieldwork, so that its effect on the workflow of a field project can be observed and the methods adjusted as practical demands may require. Since linguistics degree programs still rarely include computer programming skills, this would require a usable software interface. This requirement should motivate more interdisciplinary and collaborative work. It could be an excellent chance to introduce experts in Human Computer Interaction or User Experience design who could build and test an interface for fieldwork.

This work could impact the workflow of language documentation and description. Linguists who wish to benefit from automated assistance may want to consider how to adjust their research design. NLP scientists who wish to expand into low-resource languages may want to consider what kind of data they can expect when working with available data. Annotation quality determines how machine learning models perform, perhaps as much as the amount of training data.

In conclusion, this dissertation has demonstrated the potential to advance the science of linguistics through the integration of NLP machine learning systems. It has shown that effective machine learning methods can be integrated into morpheme segmentation and glossing and into morphological paradigm learning. Effective methods are shown to be those that do not depend on conventional expectations or traditional research designs of linguistics or NLP, including expectations of surface or canonical segmentation, joint or sequential approaches to segmentation and glossing, assumptions about the superiority of deep learning models, the expense of human annotation, or the necessity of POS tags.

Bibliography

2020. Catalogue of Endangered Languages. University of Hawaii at Manoa.

Jade Z. Abbott and Laura Martinus. 2018. Towards neural machine translation for African languages. arXiv:1811.05467 [cs, stat].

A.K. Abdulaev and I. K. Abdullaev. 2010. Cezyas folklor/Dido (Tsez) folklore/Didojskij (cezskij) fol´klor. "Lotos", Leipzig–Makhachkala.

Farrell Ackerman, James P. Blevins, and Robert Malouf. 2009. Parts and wholes: Implicative patterns in inflectional paradigms. In James P. Blevins and Juliette Blevins, editors, Analogy in Grammar: Form and Acquisition, pages 54–82. Oxford University Press.

Oliver Adams. 2017. Automatic understanding of unwritten languages. PhD Thesis, University of Melbourne.

Roee Aharoni and Yoav Goldberg. 2017. Morphological inflection generation with hard monotonic attention. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2004–2015, Vancouver, Canada. Association for Computational Linguistics.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2014. Semi-supervised learning of morphological paradigms and lexicons. In Proceedings of 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 569–578. Association for Computational Linguistics.

Malin Ahlberg, Markus Forsberg, and Mans Hulden. 2015. Paradigm classification in supervised learning of morphology. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1024–1029. Association for Computational Linguistics.

Mohammad Abdullah Al Mumin, Md Hanif Seddiqui, Muhammed Zafar Iqbal, and Mohammed Jahirul Islam. 2019. Neural machine translation for low-resource English-Bangla. Journal of Computer Science, 15(11).

Antonios Anastasopoulos. 2019. Computational Tools for Endangered Language Documentation. Ph.D. thesis, University Of Notre Dame.

Antonios Anastasopoulos, David Chiang, and Long Duong. 2016. An unsupervised probability model for speech-to-translation alignment of low-resource languages. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1255–1263, Austin, Texas. Association for Computational Linguistics.

Antonios Anastasopoulos and Graham Neubig. 2019. Pushing the limits of low-resource morphological inflection. 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 984–996, Hong Kong, China. Association for Computational Linguistics.

Ebrahim Ansari, Zdeněk Žabokrtský, Mohammad Mahmoudi, Hamid Haghdoost, and Jonáš Vidra. 2019. Supervised Morphological Segmentation Using Rich Annotated Lexicon. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), pages 52–61, Varna, Bulgaria. INCOMA Ltd.

Eric Auer, Albert Russel, Han Sloetjes, Peter Wittenburg, Oliver Schreer, S. Masnieri, Daniel Schneider, and Sebastian Tschöpel. 2010. ELAN as flexible annotation framework for sound and image processing detectors. In European Language Resources Association LREC 2010: Proceedings of the 7th International Language Resources and Evaluation, pages 890–893. European Language Resources Association.

David Baines. 2018. An overview of FieldWorks and related programs for collaborative lexicography and publishing online or as a mobile app. In Proceedings of the XVIII Euralex International Congress, Ljubliana, Slovenia. Ljubliana University Press.

Jason Baldridge and Dan Garrette. 2013. Learning a part-of-speech tagger from two hours of annotation. In Proceedings of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT-13), pages 138–147. Association of Computational Linguistics.

Jason Baldridge and Miles Osborne. 2008. Active learning and logarithmic opinion pools for HPSG parse selection. Natural Language Engineering, 14(2):191–222.

Jason Baldridge and Alexis Palmer. 2009. How well does active learning actually work? Time-based evaluation of cost-reduction strategies for language documentation. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 296–305.

Peter Baumann and Janet Pierrehumbert. 2014. Using resource-rich languages to improve morphological analysis of under-resourced languages. In Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014).

Emily M. Bender. 2014. Language CoLLAGE: Grammatical Description with the LinGO Grammar Matrix. In Proceedings of the Ninth International Conference of Language Resources and Evaluation (LREC-2014), pages 2447–2451.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. Journal of Machine Learning Research, 3:1137–1155.

Toms Bergmanis, Katharina Kann, Hinrich Schütze, and Sharon Goldwater. 2017. Training data augmentation for low-resource morphological inflection. Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection, pages 31–39.

Elan van Biljon, Arnu Pretorius, and Julia Kreutzer. 2020. On optimal transformer depth for low-resource language translation. arXiv:2004.04418 [cs].

Steven Bird and David Chiang. 2012. Machine translation for language preservation. In Proceedings of COLING 2012, pages 125–134, Mumbai.

Brenda H. Boerger, Sarah Ruth Moeller, Will Reiman, and Stephen Self. 2016. Language and Culture Documentation Manual. Leanpub.

Claire Bowern. 2008. Linguistic fieldwork: A practical guide. Palgrave Macmillan, Houndmills, Basingstoke, Hampshire [England]; New York.

Sylvia M. Broadbent. 1964. The Southern Sierra Miwok Language. University of California Press.

Lyle Campbell, Nala Huiying Lee, Eve Okura, Sean Simpson, and Kaori Ueki. 2013. New knowledge: Findings from the Catalogue of Endangered Languages. Presented at the 3rd International Conference on Language Documentation and Conservation (ICLDC).

Erwin Chan. 2006. Learning probabilistic paradigms for morphology in a latent class model. In Proceedings of the Eighth Meeting of the ACL Special Interest Group on Computational Phonology and Morphology, SIGPHON '06, pages 69–78. Association for Computational Linguistics.

Shobhana Lakshmi Chelliah. 1997. A Grammar of Meithei, volume 17 of Mouton Grammar Library. Mouton de Gruyter, Berlin.

Greville G. Corbett. 2013. The unique challenge of the Archi paradigm. In Proceedings of the 37th Annual Meeting of the Berkeley Linguistics Society:Special Session on Languages of the Caucasus, pages 52–67, Berkeley, CA.

Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. Machine learning, 20(3):273–297.

Ryan Cotterell and Georg Heigold. 2017. Cross-lingual character-level neural morphological tagging. Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 748–759.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Arya D. McCarthy, Katharina Kann, Sabrina J. Mielke, Garrett Nicolai, Miikka Silfverberg, David Yarowsky, Jason Eisner, and Mans Hulden. 2018a. The CoNLL–SIGMORPHON 2018 shared task: Universal morphological reinflection. In Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 1–27, Brussels. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017a. CoNLL-SIGMORPHON 2017 shared task: Universal morphological reinflection in 52 languages. In Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection, pages 1–30, Vancouver. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Arya D. McCarthy, Katharina Kann, Sebastian Mielke, Garrett Nicolai, Miikka Silfverberg, David Yarowsky, Jason Eisner, and Mans Hulden. 2018b. The CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection. In Proceedings of the CoNLL SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 1–27, Brussels. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, Géraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sandra Kübler, David Yarowsky, Jason Eisner, and Mans Hulden. 2017b. CoNLL-SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection in 52 Languages. In Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection, Vancouver. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016a. The SIGMORPHON 2016 shared Task—Morphological reinflection. In Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 10–22, Berlin, Germany. Association for Computational Linguistics.

Ryan Cotterell, Christo Kirov, John Sylak-Glassman, David Yarowsky, Jason Eisner, and Mans Hulden. 2016b. The SIGMORPHON 2016 shared task—morphological reinflection. In Proceedings of the 14th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 10–22.

Ryan Cotterell, Thomas Müller, Alexander M. Fraser, and Hinrich Schütze. 2015. Labeled morphological segmentation with semi-markov models. In CoNLL, pages 164–174.

Ryan Cotterell, John Sylak-Glassman, and Christo Kirov. 2017c. Neural graphical models over strings for principal parts morphological paradigm completion. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 759–765, Valencia, Spain. Association for Computational Linguistics.

Ryan Cotterell, Tim Vieira, and Hinrich Schütze. 2016c. A Joint Model of Orthography and Morphological Segmentation. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 664–669, San Diego, California. Association for Computational Linguistics.

Andrew Cowell and Alonzo Moss. 2008. The Arapaho Language. University Press of Colorado.

Christopher Cox. 2010. Probabilistic tagging of minority language data: a case study using Qtag. In S.Th. Gries, S. Wulff, and M. Davies, editors, Corpus linguistic applications: current studies, new directions., pages 213–231. Rodopi, Amsterdam:.

Christopher Cox, Gilles Bouliame, and Jahangir Alam. 2019. Taking aim at the transcription bottleneck: Integrating speech technology into language documentation and conservation. Presented at the 6th International Conference on Language Documentation and Conservation (ICLDC).

Mathias Creutz and Krista Lagus. 2002. Unsupervised discovery of morphemes. In Proceedings of the ACL-02 workshop on Morphological and phonological learning-Volume 6, pages 21–30. Association for Computational Linguistics.

Mathias Creutz and Krista Lagus. 2007. Unsupervised models for morpheme segmentation and morphology learning. ACM Trans. Speech Lang. Process., 4(1):3:1–3:34.

William Croft. 2000. Parts of speech as language universals and as language-particular categories. In Petra M. Vogel and Bernard Comrie, editors, Approaches to the Typology of Word Classes, number 23 in Empirical Approaches to Language Typology [EALT], pages 65–102. De Gruyter Mouton.

Ewa Czaykowska-Higgins. 2009. Research models, community engagement, and linguistic fieldwork: Reflections on working within Canadian indigenous communities. Language Documentation & Conservation, 3(1):15–50.

Guy De Pauw. 2012. Resource-Light Bantu Part-of-Speech Tagging. In Proceedings of the workshop on Language technology for normalisation of less-resourced languages (SALTMIL8/AfLaT2012), Istanbul, Turkey. European Language Resources Association (ELRA).

Charles Donet. 2014. The Importance of Verb Salience in the Followability of Lezgi Oral Narratives. MA thesis, Graduate Institute of Applied Linguistics, Dallas, TX.

Markus Dreyer and Jason Eisner. 2011. Discovering morphological paradigms from plain text using a dirichlet process mixture model. In Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 616–627. Association for Computational Linguistics.

Gregory Druck, Gideon Mann, and Andrew Mccallum. 2007. Reducing annotation effort using generalized expectation criteria. Technical report, University of Massachusetts Amherst, Dept. of Computer Science.

Kevin Duh, Paul McNamee, Matt Post, and Brian Thompson. 2020. Benchmarking neural and statistical machine translation on low-resource African languages. In Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), Marseille, France.

Long Duong. 2017. Natural language processing for resource-poor languages. PhD Thesis, University of Melbourne.

Long Duong, Antonios Anastasopoulos, David Chiang, Steven Bird, and Trevor Cohn. 2016. An attentional model for speech translation without transcription. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 949–959, San Diego, California. Association for Computational Linguistics.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 845–850, Beijing, China. Association for Computational Linguistics.

Greg Durrett and John DeNero. 2013. Supervised learning of complete morphological paradigms. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1185–1195, Atlanta, Georgia. Association for Computational Linguistics.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig, editors. 2020. Ethnologue: Languages of the World, twenty-third edition. SIL International, Dallas, Texas.

Ramy Eskander, Smaranda Muresan, and Michael Collins. 2020. Unsupervised cross-lingual part-of-speech tagging for truly low-resource scenarios. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 4820–4831. Association for Computational Linguitsics.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. LIBLINEAR: A library for large linear classification. Journal of machine learning research, 9(Aug):1871–1874.

Manaal Faruqui, Yulia Tsvetkov, Graham Neubig, and Chris Dyer. 2016. Morphological inflection generation using character sequence to sequence learning. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 634–643, San Diego, California. Association for Computational Linguistics.

Paul Felt. 2012. Improving the Effectiveness of Machine-Assisted Annotation. MA thesis, Brigham Young University.

Ben Foley, Josh Arnold, Rolando Coto-Solano, Gautier Durantin, T. Mark Ellison, Daan van Esch, Scott Heath, František Kratochvíl, Zara Maxwell-Smith, David Nash, Ola Olsson, Mark Richards, Nay San, Hywel Stoakes, Nick Thieberger, and Janet Wiles. 2018. Building speech recognition systems for language documentation: The CoEDL endangered language pipeline and inference system. In Proceedings of the 6th International Workshop on Spoken Language Technologies for Under-resourced Languages (SLTU 2018).

Markus Forsberg and Mans Hulden. 2016. Learning transducer models for morphological analysis from example inflections. In Proceedings of the SIGFSM Workshop on Statistical NLP and Weighted Automata, pages 42–50. Association for Computational Linguistics.

forthcoming. Proceedings of the 18th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology. Online.

Ryan Georgi. 2016. From Aari to Zulu: Massively Multilingual Creation of Language Tools using Interlinear Glossed Text. PhD Thesis, University of Washington.

David Gil. 2005. Word order without syntactic categories: How Riau Indonesian does it. In Andrew Carnie, Sheila Ann Dooley, and Heidi Harley, editors, Verb First: On the Syntax of Verb-initial Languages, volume 73 of Linguistik Aktuell/Linguistics Today, pages 243–264. John Benjamins Publishing.

Yoav Goldberg. 2017. Neural Network Methods for Natural Language Processing. Number 37 in Synthesis Lectures on Human Language Technologies. Morgan & Claypool.

John Goldsmith. 2001. Unsupervised learning of the morphology of a natural language. Computational linguistics, 27(2):153–198.

John Goldsmith, Jackson Lee, and Aris Xanthos. 2017. Computational learning of morphology. Annual Review, 3.

Jiatao Gu, Hany Hassan, Jacob Devlin, and Victor O.K. Li. 2018. Universal neural machine translation for extremely low resource languages. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 344–354, New Orleans, Louisiana. Association for Computational Linguistics.

Harald Hammarström and Lars Borin. 2011. Unsupervised learning of morphology. Computational Linguistics, 37(2):309–350.

Zellig Harris. 1955. From phoneme to morpheme. Language, 31(2):190–222.

Martin Haspelmath. 1993. A grammar of Lezgian. Mouton de Gruyter, Berlin; New York.

Luheng He, Julian Michael, Mike Lewis, and Luke Zettlemoyer. 2016. Human-in-the-loop parsing. In EMNLP, pages 2337–2342.

Nikolaus P. Himmelmann. 1998. Documentary and descriptive linguistics. Linguistics, 36:161–195.

Gary Holton, Kavon Hooshiar, and Nicholas Thieberger. 2017. Developing collection management tools to create more robust and reliable linguistic data. In 2nd Workshop on Computational Methods for Endangered Languages.

Max Planck Institute. 2008. The Leipzig Glossing Rules: Conventions for interlinear morpheme-by-morpheme glosses.

Huiming Jin, Liwei Cai, Yihui Peng, Chen Xia, Arya McCarthy, and Katharina Kann. 2020. Unsupervised morphological paradigm completion. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 6696–6707. Association for Computational Linguistics.

Michael Job, editor. 1994. The indigenous languages of the Caucasus. Volume 3: The North East Caucasian languages. Part 1, volume 3. Caravan Books, Delmar, N.Y.

Katharina Kann, Samuel R. Bowman, and Kyunghyun Cho. 2020a. Learning to learn morphological inflection for resource-poor languages. Proceedings of the AAAI Conference on Artificial Intelligence, 34(05):8058–8065. Number: 05.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017a. Neural multi-source morphological reinflection. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers, pages 514–524, Valencia, Spain. Association for Computational Linguistics.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2017b. One-shot neural cross-lingual transfer for paradigm completion. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1993–2003, Vancouver, Canada. Association for Computational Linguistics.

Katharina Kann, Ryan Cotterell, and Hinrich Schütze. 2016. Neural multi-source morphological reinflection. Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection.

Katharina Kann, Arya D. McCarthy, Garrett Nicolai, and Mans Hulden. 2020b. The SIGMORPHON 2020 Shared Task on Unsupervised Morphological Paradigm Completion. In Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 51–62, Online. Association for Computational Linguistics.

Katharina Kann and Hinrich Schütze. 2016. Single-model encoder-decoder with explicit morphological representation for reinflection. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 555–560, Berlin, Germany. Association for Computational Linguistics.

Lauri Karttunen and Kenneth R. Beesley. 2005. Twenty-five years of finite-state morphology. In Inquiries Into Words, a Festschrift for Kimmo Koskenniemi on his 60th Birthday, pages 71–83. CSLI publications.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings.

Christo Kirov, Ryan Cotterell, John Sylak-Glassman, Geraldine Walther, Ekaterina Vylomova, Patrick Xia, Manaal Faruqui, Sebastian Mielke, and Arya D McCarthy. 2016. UniMorph 2.0: Universal morphology. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018), European Language Resources Association (ELRA), pages 1868–1873.

Amit Kirschenbaum, Peter Wittenburg, and Gerhard Heyer. 2012. Unsupervised morphological analysis of small corpora: First experiments with Kilivila. Language Documentation & Conservation Special Publication, 3:25–31.

Oskar Kohonen, Sami Virpioja, and Krista Lagus. 2010. Semi-supervised learning of concatenative morphology. In Proceedings of the 11th Meeting of the ACL Special Interest Group on Computational Morphology and Phonology, pages 78–86. Association for Computational Linguistics.

Michael Krauss. 1992. The world's languages in crisis. Language, 68(1):4–10.

Michael Krauss. 2007. Keynote–mass language extinction and documentation: The race against time. In O. Miyaoka, O. Sakiyama, and M.E. Krauss, editors, The Vanishing Languages of the Pacific Rim, Oxford linguistics, pages 3–24. OUP Oxford, New York.

Steven Krauwer. 2003. The basic language resource kit (BLARK) as the first milestone for the language resources roadmap. In Proceedings of SPECOM 2003, pages 8–15.

Julian Kupiec. 1992. Robust part-of-speech tagging using a hidden Markov model. Computer Speech & Language, 6(3):225–242.

John Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional Random Fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the 18th International Conference on Machine Learning, pages 282–289.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. Nature, 521(7553):436–444.

Haley Lepp, Olga Zamaraeva, and Emily M. Bender. 2019. Visualizing inferred morphotactic systems. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), pages 127–131, Minneapolis, Minnesota. Association for Computational Linguistics.

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.

Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. Mathematical Programming, 45(1-3):503–528.

Ling Liu, Ilamvazhuthy Subbiah, Adam Wiemerslage, Jonathan Lilley, and Sarah Moeller. 2018. Morphological reinflection in context: CU boulder's submission to CoNLL-SIGMORPHON 2018 shared task. In Proceedings of the CoNLL SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 86–92. Association for Computational Linguistics.

Olga Lovick. 2020. A Grammar of Upper Tanana, Volume 1: Phonology, Lexical Classes, Morphology. University of Nebraska Press, Lincoln.

Friederike Lupke. 2010. Data collection methods for field-based language documentation. Language Documentation and Description, 7:55–104.

Peter Makarov and Simon Clematide. 2018a. Imitation learning for neural morphological string transduction. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2877–2882, Brussels, Belgium. Association for Computational Linguistics.

Peter Makarov and Simon Clematide. 2018b. UZH at CoNLL-SIGMORPHON 2018 shared task on universal morphological reinflection. In Proceedings of the CoNLL SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 69–75. Association for Computational Linguistics.

Peter Makarov, Tatiana Ruzsics, and Simon Clematide. 2017. Align and copy: UZH at SIGMORPHON 2017 shared task for morphological reinflection. In Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection, pages 49–57. Association for Computational Linguistics.

Robert Malouf. 2016. Generating morphological paradigms with a recurrent neural network. San Diego Linguistic Papers, 6:122–129.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of english: The penn treebank. Comput. Linguist., 19(2):313–330.

Laura Martinus and Jade Z. Abbott. 2019. A focus on neural machine translation for African languages. ArXiv, abs/1906.05685.

Arya D. McCarthy, Ekaterina Vylomova, Shijie Wu, Chaitanya Malaviya, Lawrence Wolf-Sonkin, Garrett Nicolai, Christo Kirov, Miikka Silfverberg, Sabrina J. Mielke, Jeffrey Heinz, Ryan Cotterell, and Mans Hulden. 2019. The SIGMORPHON 2019 shared task: Morphological analysis in context and cross-lingual transfer for inflection. In Proceedings of the 16th Workshop on Computational Research in Phonetics, Phonology, and Morphology, pages 229–244, Florence, Italy. Association for Computational Linguistics.

Angelina McMillan-Major. 2020. Automating gloss generation in interlinear glossed text. In Proceedings of the Society for Computation in Linguistics, volume 3.

Jeffrey C. Micher. 2017. Improving coverage of an Inuktitut morphological analyzer using a segmental recurrent neural network. In Proceedings of the 2nd Workshop on the Use of Computational Methods in the Study of Endangered Languages, Honolulu. Association for Computational Linguistics.

Alice Millour and Karën Fort. 2019. Unsupervised data augmentation for less-resourced languages with no standardized spelling. In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), pages 776–784, Varna, Bulgaria. INCOMA Ltd.

Sarah Moeller and Mans Hulden. 2018. Automatic glossing in a low-resource setting for language documentation. In Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages, pages 84–93. Association for Computational Linguistics.

Sarah Moeller and Mans Hulden. 2021. Integrating Automated Segmentation and Glossing into Documentary and Descriptive Linguistics. In Proceedings of the Workshop on Computational Methods for Endangered Languages, volume 1, pages 86–95, Honolulu, HI.

Sarah Moeller, Ghazaleh Kazeminejad, Andrew Cowell, and Mans Hulden. 2018. A neural morphological analyzer for Arapaho verbs learned from a finite state transducer. In Proceedings of the Workshop on Computational Modeling of Polysynthetic Languages, pages 12–20. Association for Computational Linguistics.

Sarah Moeller, Ghazaleh Kazeminejad, Andrew Cowell, and Mans Hulden. 2019. Improving low-resource morphological learning with intermediate forms from finite state transducers. In Proceedings of the Workshop on Computational Methods for Endangered Languages.

Sarah Moeller, Ling Liu, Changbing Yang, Katharina Kann, and Mans Hulden. 2020. IGT2P: From Interlinear Glossed Texts to Paradigms. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processi, pages 5251–5262. Association for Computational Linguistics.

Christian Monson, Jaime Carbonell, Alon Lavie, and Lori Levin. 2007. ParaMor: Finding paradigms across morphology. In Advances in Multilingual and Multimodal Information Retrieval, Lecture Notes in Computer Science, pages 900–907. Springer, Berlin, Heidelberg.

Taesun Moon, Katrin Erk, and Jason Baldridge. 2009. Unsupervised morphological segmentation and clustering with document boundaries. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2, pages 668–677. Association for Computational Linguistics.

Thomas Müller, Helmut Schmid, and Hinrich Schütze. 2013. Efficient higher-order CRFs for morphological tagging. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pages 322–332.

Garrett Nicolai, Colin Cherry, and Grzegorz Kondrak. 2015. Inflection generation as discriminative string transduction. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 922–931, Denver, Colorado. Association for Computational Linguistics.

Garrett Nicolai, Kyle Gorman, and Ryan Cotterell, editors. 2020. Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology. Association for Computational Linguistics, Online.

Garrett Nicolai and Grzegorz Kondrak. 2017. Morphological analysis without expert annotation. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, 2:211–216.

Åshild Næss and Brenda H. Boerger. 2008. Reefs–santa Cruz as Oceanic: Evidence from the verb complex. Oceanic Linguistics, 47:185–212.

Naoaki Okazaki. 2007. Crfsuite: A fast implementation of Conditional Random Fields (CRFs). http://www.chokkan.org/software/crfsuite/.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In Proceedings of NAACL-HLT 2019: Demonstrations.

Alexis Palmer, Taesun Moon, Jason Baldridge, Katrin Erk, Eric Campbell, and Telma Can. 2010. Computational strategies for reducing annotation effort in language documentation. Linguistic Issues in Language Technology, 3(4):1–42.

Alexis Mary Palmer. 2009. Semi-automated annotation and active learning for language documentation. PhD thesis, University of Texas at Austin.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A Universal Part-of-Speech Tagset. In Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), pages 2089–2096, Istanbul, Turkey. European Language Resources Association (ELRA).

Hoifung Poon, Colin Cherry, and Kristina Toutanova. 2009. Unsupervised morphological segmentation with log-linear models. In Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 209–217. Association for Computational Linguistics.

Martin Popel and Ondřej Bojar. 2018. Training tips for the Transformer model. The Prague Bulletin of Mathematical Linguistics, 110(1).

Lance A. Ramshaw and Mitchell P. Marcus. 1999. Text chunking using transformation-based learning. In Natural language processing using very large corpora, pages 157–176. Springer.

Gisa Rauh. 2010. Syntactic Categories: Their Identification and Description in Linguistic Theories. Oxford University Press.

Gisa Rauh, Jens Fleischhauer, Anja Latrouite, and Rainer Osswald. 2016. Linguistic categories and the syntax-semantics interface: Evaluating competing approaches. In Explorations of the Syntax-Semantics Interface, pages 15–55. düsseldorf university press, Düsseldorf.

Sally Rice and Dorothy Thunder. 2017. Community-based corpus-building: Three case studies. Presented at the 5th International Conference on Language Documentation and Conservation (ICLDC).

Brian Roark and Richard William Sproat. 2007. Computational approaches to morphology and syntax. Oxford University Press.

Chris Rogers. 2010. Review of Fieldworks Language Explorer (FLEx) 3.0. Language Documentation & Conservation, 4.

Teemu Ruokolainen, Oskar Kohonen, Kairit Sirts, Stig-Arne Grönroos, Mikko Kurimo, and Sami Virpioja. 2016. A comparative study of minimally supervised morphological segmentation. Computational Linguistics, 42(1):91–120.

Teemu Ruokolainen, Oskar Kohonen, Sami Virpioja, and Mikko Kurimo. 2013. Supervised morphological segmentation in a low-resource learning setting using Conditional Random Fields. In CoNLL, pages 29–37.

Tanja Samardzic, Robert Schikowski, and Sabine Stoll. 2015. Automatic interlinear glossing as two-level sequence classification. In Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH), Beijing, China. Association for Computational Linguistics.

Frank Seifart, Nicholas Evans, Harald Hammarström, and Stephen C. Levinson. 2018. Language documentation twenty-five years on. Language, 94(4):e324–e345.

Abhishek Sharma, Ganesh Katrapati, and Dipti Misra Sharma. 2018. IIT(BHU)–IIITH at CoNLL–SIGMORPHON 2018 shared task on universal morphological reinflection. In Proceedings of the CoNLL–SIGMORPHON 2018 Shared Task: Universal Morphological Reinflection, pages 105–111, Brussels. Association for Computational Linguistics.

Steven Shearing, Christo Kirov, Huda Khayrallah, and David Yarowsky. 2018. Improving low resource machine translation using morphological glosses. In Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Papers). Association for Machine Translation in the Americas.

Miikka Silfverberg and Mans Hulden. 2018. An encoder-decoder approach to the paradigm cell filling problem. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2883–2889. Association for Computational Linguistics.

Gary F. Simons and M. Paul Lewis. 2013. The world's languages in crisis: A 20-year update. In Elena Mihas, Bernard Perley, Gabriel Rei-Doval, and Kathleen Wheatley, editors, Responses to Language Endangerment: In honor of Mickey Noonan. New directions in language documentation and language revitalization, number 142 in Studies in Language Companion Series, pages 3–20. John Benjamins, Amsterdam.

Peter Smit, Sami Virpioja, Stig-Arne Grönroos, and Mikko Kurimo. 2014. Morfessor 2.0: Toolkit for statistical morphological segmentation. In Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 21–24, Gothenburg, Sweden. Association for Computational Linguistics.

Benjamin Snyder and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In Proceedings of ACL-08: HLT, pages 737–745.

Radu Soricut and Franz Och. 2015. Unsupervised morphology induction using word embeddings. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1627–1637.

Richard Sproat. 1992. Morphology and Computation. MIT Press.

Akhilesh Sudhakar and Anil Kumar Singh. 2017. Experiments on morphological reinflection: CoNLL-2017 shared task. Proceedings of the CoNLL SIGMORPHON 2017 Shared Task: Universal Morphological Reinflection, pages 71–78.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2818–2826.

Terrence D Szymanski. 2012. Morphological Inference from Bitext for Resource-Poor Languages. PhD Thesis, University of Michigan.

Harimohon Thounaojam and Shobhana L. Chelliah. 2007. The Lamkang language: Grammatical sketch, texts and lexicon. Linguistics of the Tibeto-Burman Area, 30(1):1–212.

Kristina Toutanova and Mark Johnson. 2008. A Bayesian LDA-based Model for Semi-Supervised Part-of-speech Tagging. In Proceedings of Proceedings of the 31st International Conference on Neural Information Processing System. MIT Press.

Rosa Vallejos. 2014. Integrating language documentation, language preservation, and linguistic research: Working with the Kokamas from the Amazon. Language Documentation & Conservation, 8:38–65.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems, pages 6000–6010, Long Beach, California, USA. Curran Associates Inc.

Sami Virpioja, Ville Turunen, Sebastian Spiegler, Oskar Kohonen, and Mikko Kurimo. 2011. Empirical comparison of evaluation methods for unsupervised learning of morphology. TAL, 52(2):45–90.

Ekaterina Vylomova, Jennifer White, Elizabeth Salesky, Sabrina J. Mielke, Shijie Wu, Edoardo Ponti, Rowan Hall Maudslay, Ran Zmigrod, Joseph Valvoda, Svetlana Toldova, Francis Tyers, Elena Klyachko, Ilya Yegorov, Natalia Krizhanovsky, Paula Czarnowska, Irene Nikkarinen, Andrej Krizhanovsky, Tiago Pimentel, Lucas Torroba Hennigen, Christo Kirov, Garrett Nicolai, Adina Williams, Antonios Anastasopoulos, Hilaria Cruz, Eleanor Chodroff, Ryan Cotterell, Miikka Silfverberg, and Mans Hulden. 2020. The SIGMORPHON 2020 shared task 0: Typologically diverse morphological inflection. In Proceedings of the 17th SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology.

Linlin Wang, Zhu Cao, Yu Xia, and Gerard de Melo. 2016. Morphological segmentation with window LSTM neural networks. In Association for the Advancement of Artificial Intelligence (AAAI), pages 2842–2848.

David Allen Wax. 2014. Automated Grammar Engineering for Verbal Morphology. Thesis, University of Washington.

Tony Woodbury. 2003. Defining documentary linguistics. Language Documentation and Description, 1:35–51.

Shijie Wu and Ryan Cotterell. 2019. Exact hard monotonic attention for character-level transduction. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1530–1537, Florence, Italy. Association for Computational Linguistics.

Shijie Wu, Ryan Cotterell, and Mans Hulden. 2021. Applying the Transformer to character-level transduction. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics.

Fei Xia, William D. Lewis, Michael Wayne Goodman, Glenn Slayden, Ryan Georgi, Joshua Crowgey, and Emily M. Bender. 2016. Enriching a massively multilingual database of interlinear glossed text. Language Resources and Evaluation, 50(2):321–349.

David Yarowsky and Grace Ngai. 2001. Inducing Multilingual POS Taggers and NP Bracketers via Robust Projection Across Aligned Corpora. In Second Meeting of the North American Chapter of the Association for Computational Linguistics.

David Yarowsky and Richard Wicentowski. 2000. Minimally supervised morphological analysis by multimodal alignment. In Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, pages 207–216, Hong Kong. Association for Computational Linguistics.

Zhong Zhou, Lori S. Levin, David Mortensen, and Alex Waibel. 2020. Using interlinear glosses as pivot in low-resource multilingual machine translation. arXiv: Computation and Language.

Ran Zmigrod, Sabrina J. Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual data augmentation for mitigating gender stereotypes in languages with rich morphology. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 1651–1661, Florence, Italy. Association for Computational Linguistics.

Appendix A

Details of IGT2P

	Transf	**+aug**	**+uninfl**	**+both**	**mono**	**+aug**	**+uninfl**	**+both**
arp clean	10:55:55	11:46:45	14:55:17	9:51:25	2:02:02	2:15:51	3:00:02	2:14:14
arp noisy	6:36:37	6:18:37	10:16:38	6:42:19	2:42:41	2:46:29	4:03:22	3:14:27
ddo clean	1:54:09	1:57:28	3:57:43	3:58:00	0:09:56	0:10:42	0:18:54	0:15:04
ddo noisy	1:51:07	1:56:24	3:23:37	3:47:12	0:08:34	0:10:59	0:20:54	0:19:41
lez clean	0:29:05	0:37:26	1:03:58	1:02:38	0:00:20	0:01:53	0:02:02	0:04:21
lez noisy	0:32:02	0:37:22	0:56:55	0:59:00	0:00:29	0:01:40	0:01:52	0:02:27
mni clean	1:15:06	1:16:19	2:12:52	2:05:02	0:03:56	0:04:42	0:08:17	0:10:11
mni noisy	1:16:59	1:18:55	2:13:06	2:14:21	0:04:32	0:08:41	0:07:20	0:08:09
ntu clean	1:09:01	0:58:37	1:28:45	1:29:39	0:02:19	0:03:34	0:02:40	0:05:53
ntu noisy	1:00:25	1:01:40	1:36:53	1:38:05	0:02:22	0:03:59	0:03:08	0:05:09

Table A.1: **Details on Computing.** Training time of our models. All models have been trained on an NVIDIA GP102 [TITAN Xp] GPU.