Systematic Comparison Of Cross-Lingual Projection Techniques For Low-Density Nlp Under Strict Resource Constraints

Joshua Waxman
Graduate Center, City University of New York

Recommended Citation
https://academicworks.cuny.edu/gc_etds/395

This Dissertation is brought to you by CUNY Academic Works. It has been accepted for inclusion in All Dissertations, Theses, and Capstone Projects by an authorized administrator of CUNY Academic Works. For more information, please contact deposit@gc.cuny.edu.
SYSTEMATIC COMPARISON OF CROSS-LINGUAL PROJECTION
TECHNIQUES FOR LOW-DENSITY NLP
UNDER STRICT RESOURCE CONSTRAINTS

By

JOSHUA WAXMAN

A dissertation submitted to the Graduate Faculty in Computer Science in partial fulfillment of the requirements for the degree of Doctor of Philosophy

The City University of New York

2014
This manuscript has been read and accepted for the Graduate Faculty in Computer Science in satisfaction of the dissertation requirement for the degree of Doctor of Philosophy.

Matt Huenerfauth

Date

Chair of Examining Committee

Robert Haralick

Date

Executive Officer

Rebecca Passonneau, Columbia University

Heng Ji, Queens College, CUNY

Virginia Teller, Hunter College, CUNY

Supervisory Committee

THE CITY UNIVERSITY OF NEW YORK
Abstract

SYSTEMATIC COMPARISON OF CROSS-LINGUAL PROJECTION TECHNIQUES FOR LOW-DENSITY NLP UNDER STRICT RESOURCE CONSTRAINTS

by

JOSHUA WAXMAN

Advisor: Professor Matt Huenerfauth

The field of low-density NLP is often approached from an engineering perspective, and evaluations are typically haphazard – considering different architectures, given different languages, and different available resources – without a systematic comparison. The resulting architectures are then tested on the unique corpus and language for which this approach has been designed. This makes it difficult to truly evaluate which approach is truly the “best,” or which approaches are best for a given language.

In this dissertation, several state-of-the-art architectures and approaches to low-density language Part-Of-Speech Tagging are reimplemented; all of these techniques exploit a relationship between a high-density (HD) language and a low-density (LD) language. As a novel contribution, a testbed is created using a representative sample of seven (HD – LD) language pairs, all drawn from the same massively parallel corpus, Europarl, and selected for their particular linguistic features. With this testbed in place, never-before-possible comparisons are conducted, to evaluate which broad approach performs the best for
particular language pairs, and investigate whether particular language features should suggest a particular NLP approach.

A survey of the field suggested some unexplored approaches with potential to yield better performance, be quicker to implement, and require less intensive linguistic resources. Under strict resource limitations, which are typical for low-density NLP environments, these characteristics are important. The approaches investigated in this dissertation are each a form of *language-ifier*, which modifies an LD-corpus to be more like an HD-corpus, or alternatively, modifies an HD-corpus to be more like an LD-corpus, prior to supervised training. Each relying on relatively few linguistic resources, four variations of language-ifier designs have been implemented and evaluated in this dissertation: lexical replacement, affix replacement, cognate replacement, and exemplar replacement. Based on linguistic properties of the languages drawn from the Europarl corpus, various predictions were made of which prior and novel approaches would be most effective for languages with specific linguistic properties, and these predictions were evaluated through systematic evaluations with the testbed of languages. The results of this dissertation serve as guidance for future researchers who must select an appropriate cross-lingual projection approach (and a high-density language from which to project) for a given low-density language.

Finally, all the languages drawn from the Europarl corpus are actually HD, but for the sake of the evaluation testbed in this dissertation, certain languages are treated as if they were LD (ignoring any available HD resources). In order to evaluate how various approaches perform on an actual LD language, a case study was conducted in which part-of-speech taggers were implemented for Tajiki, harnessing linguistic resources from a related HD-language, Farsi, using all of the prior and novel approaches investigated in this dissertation.
Insights from this case study were documented so that future researchers can gain insight into what their experience might be in implementing NLP tools for an LD language given the strict resource limitations considered in this dissertation.
Acknowledgements

Completing a dissertation is hard, especially when balancing it with a full time job and childcare for two wonderful but exhausting kids but, thank-God, I have finally done it! There are a few people I would like to single out by name for the help and support they have provided through this long journey.

Foremost, I'd like to thank my advisor, Dr. Matt Huenerfauth for his advice and direction through this work. I would also like to thank my committee members, Dr. Heng Ji, Dr. Rebecca Passonneau, and Dr. Virginia Teller for their suggestions and feedback in the proposal stage, which helped clarify and focus the direction and form of this dissertation research. I would like to thank Farzona Zehni, an independent linguist whose advice regarding Farsi and Tajik and whose practical work on assembling the necessary linguistic resources proved invaluable. Thanks also to my original thesis advisor, Dr. Noemie Elhadad, of Columbia University, back when the primary focus was on building transliteration and cognition models. And thanks to Dr. William Sakas, of Hunter College, who was a committee member up until the literature survey stage, for his encouragement and suggestions.

Also, many thanks go to Michael Gordon and Jonathan Waxman for their assistance in manually aligning tagger output with the gold standard, in instances when language-ification or language-specific tokenization caused misalignments.
I would also like to express my profound gratitude to my close and extended family for their support, emotional and otherwise, through this long slog. Thanks go to my wife, Dr. Rachel Waxman, for encouraging me, putting up with me and distracting the kids as I spent hours at the computer, even as she had her own dissertation to worry about. Thanks to Meir and Eitan, for many things including letting me use the computer that otherwise could have been productively applied to such tasks as Bloons Tower Defense 4, Minecraft, Starfall, and Peep and the Big Wide World. And of course, thanks go out to my parents, Rabbi Dr. Jerry Waxman and Lorri Waxman, and to my in-laws, Rabbi Eduard and Sandra Mittelman, for all they have done for us. You are all partners in this accomplishment.

Next up, trying to resolve the cold-blooded / warm-blooded dinosaur debate!
Table of Contents

Abstract............................................................................................................................................iv
Acknowledgements..........................................................................................................................vii
Table of Contents...............................................................................................................................ix
Table of Figures..................................................................................................................................xii
Table of Tables ....................................................................................................................................xv
Chapter 1: Introduction.......................................................................................................................1
   Structure of this Dissertation.........................................................................................................2
Chapter 2: Existing Approaches to Low-Density NLP ......................................................................5
   2.1 An Overview of Low-Density NLP Approaches .....................................................................5
   2.2 Recent Work ............................................................................................................................10
   2.3 Utilizing High-Density Language Resources on Low-Density Data ......................................12
      2.3.1 POS Tagging of Dialectal Arabic: A Minimally Supervised Approach ..........................13
      2.3.2 Applying Contextual Models from Related Languages ..................................................17
      2.3.3 Projecting lexical Models Via Cognates .........................................................................19
   2.4 A Quick Summary of Low-Density NLP Approaches .............................................................22
Chapter 3: Overview of Work .........................................................................................................25
   3.1 Methodological Details ..........................................................................................................27
      3.1.1 Corpus and Corpus Construction ......................................................................................27
      3.1.2 Multiple POS sets ............................................................................................................28
Chapter 4: Implementation .............................................................................................................31
   4.1 Reimplementation of Previous Methods ...............................................................................31
   4.2 New Approaches ....................................................................................................................33
      4.2.1 Overview ..........................................................................................................................33
      4.2.2 Language-ifier ..................................................................................................................37
         4.2.4.1 General Scheme of Work on Novel Approaches ......................................................45
         4.2.4.3 HD-ifying an LD input text .......................................................................................46
         4.2.4.2 LD-ifying an HD Corpus ..........................................................................................52
Chapter 5: Language Discussion ......................................................................................................54
   5.1 Language Families ...................................................................................................................57
   Prediction [P1a, b] ..........................................................................................................................59
   Prediction [P2a, b, c, d] ...................................................................................................................59
   5.2 Constituent Word Order ..........................................................................................................61
   Prediction [P3a, b, c, d] ...................................................................................................................64
   5.3 Adjective premodifiers and postmodifiers .............................................................................66
   Prediction [P4a, b] ..........................................................................................................................72
Table of Figures

Figure 1: An overview of approaches from the literature ........................................24
Figure 2: In each case, the arrow points to the LD language and the arrow length indicates linguistic distance (roughly approximated by this author) between the languages ..........28
Figure 3: A comparison of HD-ification to Hana’s LD-ification ..................................39
Figure 4 HD-ify vs. LD-ify approaches .........................................................................45
Figure 5 The general process of HD-ification .................................................................46
Figure 6 Generating the HD-ify frequent word dictionary ..............................................47
Figure 7 L1 HD-ify process ............................................................................................47
Figure 8 Generating the HD-ify affix dictionaries .............................................................48
Figure 9 L2 HD-ify process ............................................................................................49
Figure 10 L3 HD-ify process ...........................................................................................50
Figure 11 Generating the cognate dictionaries ...............................................................51
Figure 12 L4 HD-ify process ...........................................................................................52
Figure 13 The general process of LD-ification ...............................................................52
Figure 14: Dutch is closer to German ..............................................................................52
Figure 15: Windows Interface for TreeTagger ...............................................................57
Figure 16 Relationship between Europarl languages ......................................................58
Figure 17: German and English premodifier example .....................................................68
Figure 18: English premodifier and French postmodifier example .............................69
Figure 19 English and German premodifiers in the actual corpus ...............................72
Figure 20 French postmodifiers in the actual corpus .....................................................72
Figure 21: Projecting articles from a high granularity language to a low granularity language ..................................................................................................................77
Figure 22: Projecting articles from a low granularity language to a high granularity language ................................................................. 78
Figure 23: Four cases of language-ification strategies and degrees of inflection ............ 84
Figure 24: Projection of a poorly inflected language to a richly inflected language ........ 86
Figure 25: Projection of a richly inflected language to a poorly inflected language .......... 86
Figure 26: Once again, four cases of language-ification strategies and degrees of inflection .................................................................................................................. 89
Figure 27: In each case, the arrow points to the LD language and the arrow length indicates linguistic distance (roughly approximated by this author) between the languages .............................................................................................................. 93
Figure 28: Prediction 1 results ..................................................................................... 101
Figure 29: Prediction 1 results, using LD-ify baseline .................................................. 103
Figure 30: Prediction 2 results ..................................................................................... 105
Figure 31: L4 HD-ify results ....................................................................................... 106
Figure 32: Prediction 3 results ..................................................................................... 108
Figure 33: English and German premodifiers in a sentence ........................................ 109
Figure 34: English premodifiers, French postmodifiers in a sentence ......................... 110
Figure 35: Prediction 5 results ..................................................................................... 111
Figure 36: Prediction 5 results, with 50 most frequent word replacement .................... 113
Figure 37: Projection of articles, with high and low lexical granularity ....................... 114
Figure 38: Prediction 7 results ..................................................................................... 117
Figure 39: Prediction 8 results ..................................................................................... 118
Figure 40: Prediction 9 results ..................................................................................... 119
Figure 41: HD-ify vs LD-ify results ............................................................................. 121
Figure 42: L4 HD-ify vs. LD-ify .................................................................................. 122
Figure 43: Frequent word replacement, 50...250 ....................................................... 124
Figure 44: Frequent word profile for Dutch .................................................................. 125
Figure 45: Frequent word profile for German ............................................................. 126
Figure 46: Frequent word distribution for Dutch ......................................................... 127
Figure 47: Language tree for Farsi and Tajiki ............................................................. 129
Figure 48: (Farsi, Tajiki) deltas from baseline .......................................................... 145
Figure 49: Tajiki results, prediction 1 ........................................................................ 147
Figure 50: Tajiki results, prediction 2 – H2 ................................................................. 148
Figure 51: Tajiki results, prediction 2 - L4 HD-ify against (German, Finnish) ....... 148
Figure 52: Tajiki results, prediction 2 - L4 HD-ify against (German, Dutch) and (Dutch,
German) .................................................................................................................. 149
Figure 53: Tajiki results, prediction 3 ........................................................................ 150
Figure 54: Tajiki results, prediction 4 ........................................................................ 151
Figure 55: Tajiki results, prediction 5 ........................................................................ 152
Figure 56: Tajiki results, prediction 7 ........................................................................ 153
Figure 57 Tajiki frequent word profile (left) and Farsi frequent word profile (right) ... 154
Figure 58: Tajiki frequent word replacement, 50...250 .......................................... 155
Figure 59 Frequent word profile for Dutch ............................................................... 170
Figure 60 Frequent word distribution for Dutch ....................................................... 171
Figure 61 Frequent word profile for German ............................................................. 172
Figure 62 Frequent word distribution for German ..................................................... 172
Figure 63 Frequent word profile for Finnish ............................................................. 173
Figure 64 Frequent word distribution for Finnish ..................................................... 173
Figure 65 Frequent word profile for Estonian ........................................................... 174
Figure 66 Frequent word distribution for Estonian ................................................... 174
Figure 67 Frequent word profile for English ............................................................. 175
Figure 68 Frequent word distribution for English .................................................... 175
Figure 69 Frequent word profile for French ............................................................. 176
Figure 70 Frequent word distribution for French ..................................................... 176
Table of Tables

Table 1 Constituent Word Order of Languages in Europarl.............................................63
Table 2: Adjectives as premodifiers or postmodifiers .....................................................67
Table 3: Presence and Granularity of Articles in Europarl languages.........................74
Table 4: PRO-drop and Inflection and Europarl languages..........................................81
Table 5 Results, traditional approaches ........................................................................95
Table 6 HD-ification results .........................................................................................97
Table 7: LD-ification results .......................................................................................99
Table 8 Tajiki Results, Traditional Approaches ............................................................140
Table 9: (Farsi, Tajiki) results, HD-ification, compared with other close languages ....141
Table 10: Tajiki Results, HD-ification .......................................................................144
Table 11: Tajiki results, LD-ification .......................................................................145
Chapter 1: Introduction

The state-of-the-art of NLP is quite good – for some languages. Due to the ability to harness linguistic resources, many NLP tasks, such as Machine Translation and Named Entity Recognition are possible, and perform at high levels of accuracy – at least for specific domains. This success is due in large part to the wealth of linguistic resources available for these languages. For example, large Treebanks of parsed sentences exist for English and German, which can be used to statistically train a parsing model. There are large corpora tagged for part-of-speech and for named-entities. Similarly, for machine translation, there are large bilingual corpora aligned by sentence or word, which are used to train statistical translation models. Similarly, linguists have spent many hours assembling WordNets for languages such as English and German.

In contrast, in this dissertation, we consider approaches to natural language processing of low density languages, with a specific focus on approaches of leveraging linguistic resources from other, higher-density languages. A high-density (HD) language is a natural language for which there exists a wealth of electronically available linguistic resources. A low-density (LD) language is one that lacks these resources. Since more popular languages are used by more people who are generating data, they are more likely
than less popular languages to have large electronically available corpora. Popular languages and dialects are more likely to have been the focus of attention by more researchers and therefore to have manually created linguistic resources such as Treebanks and WordNets.

The common, state-of-the-art approach to a variety of natural language tasks is statistical, and statistical methods rely on the existence of large corpora. Thus, these methods are more suited for high-density languages. For low-density languages, we cannot readily use such methods. For low-density languages, a variety of paradigms are used to develop and apply linguistic resources, depending on the amount of data available. These paradigms include:

- hand-coded rule-based solutions
- supervised learning, unsupervised learning, and active learning
- extracting the maximal data from limited data
- using tools of high-density languages on close low-density ones
- projecting linguistic resources from high-density to low-density languages

This dissertation will specifically focus on the final approach listed above, in which NLP resources for an HD language are “projected” to an LD language.

**Structure of this Dissertation**

Chapter 2 will survey several state-of-the-art architectures and approaches to low-density NLP. A challenge identified in this survey is that the evaluation of systems is typically lacks a systematic comparison; each published paper considers different architectures, given different languages, and different available resources. This haphazard nature of evaluation makes it difficult to truly evaluate which approach is truly the “best,” or which approaches are best for a given language.
To overcome this limitation, chapter 3 will discuss how, in this dissertation research, part-of-speech taggers for several languages are re-implemented – all of the techniques employ a cross-lingual projection from a high-density (HD) language to a low-density (LD) language. In this dissertation research, a testbed is created using a representative sample of seven (HD–LD) language pairs, drawn from Europarl; using this testbed, comparisons are conducted, to evaluate which approach performs best for particular language pairs with particular linguistic features.

Chapter 4 will identify and describe a set of novel cross-lingual projection approaches that fall into the category of “language-ification” techniques, in which an LD-corpus is pre-processed to appear more like an HD-corpus, or vice versa, prior to supervised training. Under a set of strict resource limitations that are considered in this dissertation, these techniques have several desirable characteristics, which are discussed in that chapter.

To make this dissertation research useful for future researchers, it is not sufficient to merely test the prior and novel approaches on a haphazard selection of languages from Europarl. Instead, Chapter 5 identifies sets of linguistic properties that specific pairs of HD–LD language pairs may possess, motivates several predictions as to how the cross-lingual approaches would perform for pairs of languages with these properties, and identifies pairs of languages in the Europarl testbed that can be used to evaluate each prediction. The aim is that future researchers who must implement an NLP tool for a specific LD language could consider these linguistic properties when selecting a cross-lingual projection approach and an appropriate HD language from which to project.

Chapter 6 describes the implementation of the testbed, the re-implementation of prior approaches and the novel language-ification techniques, and the results of the systematic
comparisons. This chapter also discusses whether each prediction from Chapter 5 was supported by the results of the evaluation.

Of course, in order to have a gold-standard against which we could evaluate our results in Chapter 6, we needed all of the languages in the testbed to actually be HD languages (so that it would be possible to rapidly obtain accurate part-of-speech tagging results for each language for evaluation purposes). In order to investigate what the experience would be for a researcher who must implement an NLP tool for an actual low-density language, Chapter 7 presents a case study that was conducted, in which part-of-speech taggers were implemented for Tajik, making use of linguistic resources from a related HD-language. The chapter documents this process so that future researchers can gain better anticipate the challenges that may arise in implementing NLP tools for an LD language, given the strict resource limitations considered in this dissertation.

Finally, Chapter 8 presents conclusions and future work.
Chapter 2: Existing Approaches to Low-Density NLP

2.1 An Overview of Low-Density NLP Approaches

A low-density language is one that lacks an abundance of linguistic resources, whether these be trained linguists or computational linguists, machine accessible dictionaries, parallel aligned bilingual corpora, tagged corpora, etc. Such a lack is problematic, given that the statistical approach that is so popular and effective today relies on the existence of large corpora. Trained linguists could assemble such resources, but that would be expensive and time-consuming, and such manpower might not be readily available for low-density languages. The alternative approach is for computational linguists to manually craft NLP systems. This is a costly and time-consuming process, and the results are in many cases not as good as statistically trained systems, if sufficiently large corpora were available.

There are several different approaches to overcoming this lack of linguistic resources. One approach is to develop ways to aid the creation of manually-created tools in shorter time. Thus, Loftsson (2007) develops a linguistic rule-based method for morphological tagging of Icelandic based on the Constraint Grammar framework, and develops it in a comparatively short time. He does this by disambiguating not only by local rules which restrict tags based on properties of words in the surrounding context, but also based on “global” heuristics which disambiguate based on syntax of non-local context. The accuracy of this tagger is comparable to a state-of-the-art statistically trained model. Thus,
such an approach can speed development time of manually crafted tools for low-density languages. Bender and Flickinger (2005) develop a framework to quickly and simply build up precise, broad-coverage grammars. They create a language-independent Core Grammar providing core functionality common to all language types. Then, they develop modules to handle specific varying behavior such as word order, types of sentential negation, and the lexicon. They develop a web-based interface for selecting the specifics of the language in question (e.g. to specify that a language is SOV). Computational linguists can then modify the generated module to further conform to the particulars of the language.

Another approach is to automatically extend the coverage of existing, though limited, manually-created deep-language resources, using minimally supervised learning. Thus, Baldwin (2005) takes a corpus of lexical items classified into lexical types, discovers morphological, syntactic, and / or ontological features of items already in the resource, and applies the same classification to lexical items which have similar features.

Yet another approach is to explore and enhance the use of alternative language models to better cope with scarcity of data. Within the field of machine translation, language models which are simple n-gram models can suffer from data scarcity, but an alternative language model proposed by Bengio et al (2003) makes use of distributed word representations, in which words are represented as a real-valued vector in a high-dimensional feature space, and train a neural probabilistic language model (NLPM), to learn the distributed representation for every word as well as the n-gram model for these distributed representations of words. In the past, these models had been difficult to apply to NLP tasks involving large vocabularies, but recently, Vaswani et al. (2013) applied more recently developed techniques to make use of such a language model practical, and applied it
successfully to parallel bitexts in Europarl. While this approach indeed looks promising, this is within the domain of machine translation rather than part of speech tagging. Further, while this approach might help with data scarcity, surely an issue for low-density languages, it is not clear that this would help with words which are entirely out of vocabulary. And finally, the approach seems orthogonal to the path we chose to explore: this is an improvement of the language model rather than of strategies of projection of HD resources, and perhaps one could even swap out one model for another and still apply all of the projection strategies we explore in this dissertation.

Still another approach is to engage in active learning, in which the model being trained participates in the learning by requesting the data that would teach it the most. Since manually tagging corpora is costly and time-consuming, such time and effort are best spent where it would do the most good. Thus, for example, Thompson et al. (1999) made use of certainty-based active learning for semantic parsing (mapping natural language questions to Prolog queries) and for information extraction. And Tang, Luo, Roukes (2002) use certainty-based active learning to train a statistical syntactic parser.

One other promising approach is projection of linguistic resources via a bridge formed by a parallel bilingual corpus. One side of this parallel word-aligned corpus is the HD language and the other is the LD language. Yarowsky, Ngai, and Wicentowski (2001) project linguistic resources by means of parallel corpora and then induce a noise-robust tagger from the projected data, for four different applications: a POS tagger, a Noun Phrase Bracketer, a Named Entity Tagger, and a Morphological Analyzer. To focus just on the POS-tagger, the HD side of the corpus is tagged, and they project the tags to the LD side. There are different scenarios of word alignment, depending on whether a word or phrase maps to
a word, phrase, or nothing, and they hand-crafted strategies to apply in each scenario. They
developed similar clever approaches to project the other linguistic resources, showing that
the approach can be applied broadly. This approach has been used successfully by Dien and
Kiem (2003) to develop a POS tagger, Phrase-Chunker, Parser, and Word-Sense
Disambiguator for Vietnamese, by directly projecting an automatically aligned Vietnamese-
English corpus and manually correcting errors. Similarly Trushkina (2006) used this method
to induce linguistic tools, such as a POS-tagger, for Afrikaans based on a trilingual Biblical
parallel corpus of Dutch, English and Afrikaans. Hwa et al. (2005) continue in this approach
of transferring linguistic resources via projection. They take syntactic dependencies from a
source language (such as English), and a word-aligned parallel corpus, and project those
dependencies to a target language (such as Spanish or Chinese).

While this is indeed seems a promising approach, it is limited in that is requires a
parallel bilingual text, which may well not exist for the LD language. It is true that there is
a probable availability of a Bible in the LD language, which can then be automatically word
aligned with an LD Bible. However, the sort of vocabulary and syntax in a Bible may well
not match that found in the target domain, such as newspaper articles or medical texts.
Because of this limitation, we do not reimplement this approach, but instead focus on
methods we can apply using the linguistic resources listed in section 4.2.1.

This requirement for an actual bilingual parallel aligned corpus can be somewhat
relaxed. Barzilay et al. (2003) worked on aligning monolingual comparable corpora, using
topic structure for paragraphs, then local alignment to find good sentence pairs. And
Rambow et al. (2006) expanded on a core translation dictionary (closed class words plus 100
most frequent words) using unsupervised training on monolingual comparable corpora. They
then expanded the lexical probabilities in their statistical model from high-density language words to all the low-density language words which appear in their translation dictionary. Using techniques and resources (monolingual comparable corpora “pseudo-bitexts”) similar to these two projects, one might be able to successfully apply projection techniques to make use of high-density NLP tools for a low-density language. However, for this dissertation research, we are not assuming even this level of linguistic resource. Rather, we explore techniques that do not require the use of bitexts or pseudo-bitexts between a high-density and low-density language.

In this survey chapter, we focus in particular on the approach of applying tools or resources developed for other, high-density languages, for a related the low-density language. Thus, Duh and Kirchhoff (2005) use a part-of-speech tagger that was developed for Standard Arabic (a high-density language) to tag low-density dialects of Arabic. Hana et al. (2004) train a Russian tagger using an annotated Czech corpus, and Hana et al. (2006) train a Portuguese tagger using an annotated Spanish corpus. Yarowsky et al. (2001) and Hwa et al. (2005) use parallel aligned corpora to project linguistic data (such as part-of-speech tags) from a high-density language to a low-density one, take steps to reduce the noise of the projected data, and then use the projected data to induce stand-alone linguistic models on the low-density language side. We believe this general approach of applying and projecting resources from high-density languages can mesh well with cognation and machine transliteration techniques (because both cognation and transliteration create bridges between lexical items across two languages).
2.2 Recent Work

Here we summarize some more recent work. We can organize recent low-density NLP research into these same broad approaches as discussed above: improving unsupervised approaches, improving the utility of limited data, and improving the efficacy of projection of resources from the HD to the LD.

First, we consider the realm of weakly supervised learning approaches, which are trained on lexicons rather than tagged corpora. Ravi and Knight (2009) and Ravi et al. (2010) introduced the idea of model minimization, approximating the minimal set of tag bigrams needed to explain the data, and employing the EM algorithm in conjunction with this. Garrete and Baldridge (2012) build on the model minimization approach, augmenting it with several heuristics and with a better HMM emission initialization. Because this approach does not work well in a low-resource setting, where most tag types are not found in the initial dictionary, Garrete and Baldridge (2013) then use label propagation and weighted model minimization techniques to create a weighted dictionary that covers the entire corpus. Other researchers have worked on expanding the scope of the lexicon. Thus, Gasser (2010) employed a web crawler and morphological analyzer to extend a lexicon by adding new roots and inferring derivational constraints that apply to known roots. The novel approaches we consider in this dissertation, however, do not rely on the existence of such an LD lexicon.

Next, we consider the realm of HD $\rightarrow$ LD linguistic projection. Some recent work has dealt with the fact that tagsets for individual languages differ in granularity, and reflect language-specific peculiarities. This can stand in the way of projecting resources from one language to another. Therefore, (Zeman and Resnik, 2008; Petrov et al., 2012; Naseem et al., 2010) develop manual mapping schemes of tags from one language to another, and Zhang et al. (2012) induce such a mapping from training data. In our work, we use a lowest
common denominator tagset for all eight languages under consideration to enable fair comparisons across languages (details in section 3.1.2).

Some researchers project tags and other linguistic data for a lexicon from the HD to the LD, based on parallel dictionaries. See for example Täckström et al. (2013), Li et al. (2012), and Das and Petrov (2011). Others make use of word-aligned corpora, where the HD has POS tags and the LD does not, to train POS taggers which emit bilingual observations. See e.g. Thu et al. (2014) and Tamura et al. (2013). In the techniques which are the focus of this dissertation, we do not assume the availability of such parallel dictionaries or corpora.

Some researchers project linguistic knowledge from multiple languages. Thus, Kim and Snyder (2013) leverage knowledge from many known languages and alphabets in order to distinguish consonants from vowels in unknown languages and alphabets, in an unsupervised manner. And Kim et al. (2011) apply a similar approach of clustering and projecting from many known languages to the task of morphological analysis. Meanwhile, Snyder et al. (2009) increase accuracy of unsupervised part-of-speech tagging on multilingual aligned untagged text (where none of the corpora possess part-of-speech tags). The approaches we consider in this dissertation, however, do not rely on the existence of parallel aligned text.

Finally, there has been some work in language-ification as a projection technique. Thus, Hana et al. (2011) transformed the tags and lexical items of a part-of-speech tagged Modern Czech corpus to resemble an Old Czech corpus and transformed an untagged Old Czech corpus to resemble Modern Czech. This research is along the lines of our language-ification approaches, but differs in that their tag and lexical language-ifications are hand-crafted by linguistic experts who know both languages in great detail.
2.3 Utilizing High-Density Language Resources on Low-Density Data

Some research on low-density language NLP involves applying tools developed for a high-density language to a related low-density language (e.g. using a Spanish part-of-speech tagger to tag a Portuguese corpus) and then improving upon the results.

This section contains discussions of part-of-speech taggers, which share some common components, though different authors use different terminology to refer to the components. Therefore, before proceeding, we will provide the definitions of terms describing three different probabilistic models, which work together:

- A **contextual model**, which is a tag sequence trigram model, \( p(t_i | t_{i-1}, t_{i-2}) \). If the tag two words ago was JJ (adjective), and the tag on the previous word was NN (noun), what is the probability that the tag of the present word is VBD (perfect verb)? What is the probability that the tag of the present word is VBP (imperfect verb)?

- An **emission probability**, which is the likelihood of a word given a tag, \( p(w_i | t_i) \). As an example from the English language, if we decide that the present tag is VBD, what is the probability that the word is “walked”? What is the probability that the word is “ate”? To calculate this, we determine how many times each word was used with each tag, out of all the total words used within this tag, throughout the entire corpus.

- A **lexical model**, which is the likelihood of a tag given a word, \( p(t_i | w_i) \). As an example from English, if the present word is “walk,” what is the probability that it is a verb (“walk to the store”) and what is the probability that it is a noun (“take a walk”)?
2.3.1 POS Tagging of Dialectal Arabic: A Minimally Supervised Approach

As noted above, if a low-density language lacks some linguistic resource, a possible solution is to simply utilize the language tools or resources developed for a closely related high-density language. This is the approach Duh and Kirchhoff (2005) take. Modern Standard Arabic (MSA) is a high-density language, but various Arabic dialects are not. Duh and Kirchhoff (2005) apply MSA resources – an MSA part-of-speech tagger and an MSA Treebank corpus – to the task of tagging low-density dialectal Arabic data. They also investigate methods of training POS-tagging models using data from several different dialects, in order to increase the size of the training data.

First, they built a baseline part-of-speech tagger (this approach will be referred to by the codename “D1” throughout this dissertation), by running an existing tagger for Modern Standard Arabic upon data from a corpus of Egyptian Common Arabic (ECA), a low-density dialect. This tagger generates a group of possible (and possibly inaccurate) tags for each lexical item. The ECA corpus actually carries with it morphological analysis of each word, which they use to generate a gold-standard for evaluating the performance of the MSA tagger on the ECA data. The accuracy of the baseline MSA tagger on ECA data is 62.76%.

In comparison, Brill taggers on English and MSA can achieve 97% accuracy (Khoja 2001). Still, the 62.76% accuracy is impressive, in that it is achieved using a tagger trained on data from a different Arabic dialect.

They also implement an upper baseline (which will be referred to by the codename “D2” throughout this dissertation), which is supervised training on the fully tagged LD-corpus (where they have tags they usually withhold). They also attempt unsupervised training on the LD corpus, with just the LD-lexicon (codename “D3”). And then they attempt the same unsupervised training on the LD corpus, but beginning with an “HD” lexicon,
which is the POS tags generated by running an HD morphological analyzer on the LD corpus, generating a rather noisy lexicon, with uniform distribution of tags for unrecognized words (codename “D4”).

After making their baseline as lousy and noisy as possible, in order to make their job that much harder, they explore methods of reducing this noise and improving accuracy. That is, if they can start with a unsupervised learning on noisy data drawn from a linguistic resources of a different, related language, and even so achieve the accuracy they attain, then their results are all the more impressive. In one approach (“D5”), they cluster words based on distributional criteria, select tags for each cluster whose likelihood of occurring in that cluster exceeds a certain threshold, and only allow those tags as possible tags. In another improvement (“D6”), they perform simple morphological analysis (detecting affixes) on the lexical items, and then calculate conditional probabilities of those features given specific tags. This way, they can select the more likely tag from the tagset suggested by the MSA tagger. **Combining** these two approaches and applying them on top of the baseline tagger achieves 69.83% accuracy.

Since their corpora are so small, they suggest additional methods of using corpora from other low-density dialects of Arabic. They combine the ECA corpus with a Levantine Arabic corpus. Of course, these are two different dialects, so they need to be careful what they learn from each corpus.

To calculate the probability of a word and tag co-occurring they simply combine (multiply) the probability from the contextual model and the emission probability. The probability of a sequence of words and tags is the product of each individual word and tag occurring, for each word and tag in the sequence.
In one strategy, they use linear interpolation for the contextual model. That is, they combine the probability judgments of the ECA contextual model with the probability judgments of the contextual model of a dialect out of domain (LCA). They assign different weights to data from different dialects (λ and 1-λ), choosing the weight that maximizes the likelihood of a held-out data set. As an extension of this strategy, rather than weighting each dialect uniformly, they vary the weight to the respective dialects based on the tag under consideration.

In a second strategy, they do not assign different weights to different dialects at all. Rather, they assume that the underlying tag sequence is the same across dialects, even though the emission probabilities will differ. Therefore, they train a shared contextual model but individual emission probabilities\(^{1}\). This strategy, when combined with the filtering of suggested based on affixes as described above, yields 70.88% accuracy overall.

Of course, 69.83% or 70.88% accuracy may appear low, considering that Brill taggers on English can achieve 97% accuracy or that Modern Standard Arabic taggers have also achieved 97% accuracy\(^{2}\). However, for low-density languages, the large corpora or the linguistic tools are not available to achieve such high performance. With lower (more reasonable) expectations, we can appreciate the results obtained for these low-density languages.

The picture is even better than it might seem at first. For words analyzable by the MSA tagger, their accuracy is somewhat better (about 75%), though still not approaching the ideal set by tagger performance for high-level languages. It is the unanalyzable words

---

\(^{1}\)(See Hana et al. (2006), discussed in section 4.2.2, about the possibility of making use of the lexical model of one language for another related language.)\(^{2}\) See e.g. “APT: Arabic Part-of-speech Tagger,” Shereen Khoja
(for which the MSA tagger produces no suggestions) and out-of-vocabulary words which drag the accuracy down. Of course, we want to analyze all the words, even those unanalyzable by the MSA tagger. But it is encouraging that when the MSA tagger does work, it works fairly well.

Furthermore, rather than overall performance, it is the improvement over the baseline that is key here, to demonstrate that each of these strategies for low density languages can improve performance. Thus, even with this “disappointing” performance, they demonstrated a few important points. Namely, the demonstrated that sometimes, using a language resource developed for one (high-density) language might yield some useful results for related languages and dialects. Secondly, certain strategies to refine the results and filter out noise on the low density language side can further improve the accuracy. Finally, where data is scarce but there are multiple low-density dialects available, one can make as much possible use of the resources one has by creatively combining corpora.

Within the bounds that they set, of unsupervised learning, their results seem impressive and useful. Of course, the best way to increase accuracy and performance is to use more supervised methods, but that involves a considerable investment of time and effort on the part of linguists, something they would rather avoid for these low-density languages.

One minor critique we had when considering their incorporation of an affix-based model was that this was hand-designed. This requires some expert knowledge of the language in question, and leaves room for choosing particular levels of detail to tweak results. To their credit, their affix model was remarkably simple, with a short list of prefixes and suffixes. (One based on non-concatenative morphology might have done better for Arabic.) Also, this might work well for languages such as Arabic, but not necessarily as well
for non-Semitic languages. It would be good to evaluate how well this approach works for different types of language, with differing morphology. Therefore, in this dissertation research, we re-implement this affix-model approach for all the languages in several selected language pairs, and we automatically generate the most common prefixes, without consulting any language expert².

2.3.2 Applying Contextual Models from Related Languages

As discussed in the previous section, Duh and Kirchhoff (2005) applied a Modern Standard Arabic part-of-speech tagger to a corpus of dialectal Arabic, took steps to reduce the noise in the suggested tags, and train on that data to separately induce a contextual model and the emission probabilities.

Hana et al. (2004) trained a Russian tagger using an annotated Czech corpus. They did not train on a Russian corpus at all. Rather, they assumed that the contextual model of Czech and Russian will be similar enough that the contextual model trained on Czech will suffice (we refer to this approach with codename “H1”). They also experimented with “Russifying” the Czech corpus before training. (This is a sort of language-ifier.) Thus, for example, while Czech adjectives and participles distinguish gender, Russian adjectives and participles do not, and so they stripped the gender distinction out of Czech. They hoped this would result in a marked improvement of the tagger but it did not produce much of a difference – overall accuracy improved about 1%.

The authors think of a part-of-speech tag as being very detailed. For them, a tag specifies P (part-of-speech), S (subpart of speech), g (gender), n (number), p (person) and

² To automatically generate these prefixes and suffixes, I will use Automorphology, a program by John Goldsmith which employs an unsupervised approach on an untagged corpus. It is available here: http://humanities.uchicago.edu/faculty/goldsmith/Automorphology/
various other subtags. Because of this specificity, they have 1000 different tags, and they note that with 1000 tags, there are potentially 1000\(^3\) different trigrams in a trigram model. Thus, sparsity of data is a problem. Therefore, they trained separate subtaggers on combinations of subtags. When combining related slots (such as S+n+c) and training \textit{just} on those subtags, the resulting subtagger outperformed the tagger trained on the full tags. They then combined the vote of several subtaggers for each slot, and the resulting model achieved 73.5\% accuracy on the overall tags, though with better accuracy (high 80’s to high 90’s) for each of the slots.

(In our own reimplementation of H1, tags are defined with much less granularity. This is because we need to meet the lowest common denominator, in order to compare results across many languages.)

Since Russian and Czech have different vocabularies, they were not able to use the Czech \textit{lexical model}\(^3\). Rather, for the lexical model of Russian, they built a Russian morphological analyzer and used it to analyze an untagged monolingual Russian corpus. Each word was lemmatized and then other forms were generated. (Thus, in English, “talked” would be analyzed as the verb “talk” and the lexical model would include “talk,” “talked,” “talking,” and “talks.”) Then, they use those extracted lemmas to generate all possible forms of the Russian word. For each word, they simply equally distributed the probability across all their generated entries which had that tag. This choice of equal distribution is because they are generating the words and their forms, rather than examining a large corpus, which contains the forms with certain frequency.

\(^3\) The authors refer to this as emission probabilities, but this is at odds with the more common definition of emission probability, which is probability of a word given a tag.
One limitation of Hana et al. (2004) is that their language-ifier, besides not working very well, was carefully constructed by linguists. While they took steps to restrict it, there is an aspect of “hacking” and an open-endedness to the rewrite rules one might construct. Also, there is a dependence upon linguistic knowledge. This makes it difficult to evaluate the system and compare it with others. In our own implementation of language-ifiers in this dissertation research, we take pains to avoid both of these limitations. Language-ifiers are by definition restricted – for example, a list of 250 words total for lexical replacement; and avoid dependence upon specific linguistic knowledge – for example, by choosing those 250 words not by knowledge of the most informative words, but by automatically selecting the 250 most frequent.

A second limitation of Hana et al. (2004) is that they implemented in on a specific language, and their language hack was particular to their language pair. We would not truly know how well their results compare to other approaches, since e.g. Duh (2005) applied their approach to dialects of Arabic, not to Russian and Czech. In this dissertation research, we do compare the two approaches, our re-implementing their work on the same language pair, or rather, on 20 different language pairs.

Towards the end of their 2004 work, they mention that there are some Czech-Russian cognates, such that they would like to make use of the individual lexical probabilities in such instances, but they leave that for future work.

2.3.3 Projecting Lexical Models Via Cognates

Two years later, Hana et al. (2006) attempt to do just that – make use of a lexical model (rather than just a contextual model) trained on a high-density language corpus. In this new work, the high-density language is Spanish and the low-density language is
Portuguese⁴. The languages are related, and their grammars are close enough that they simply adopt the Spanish contextual model wholesale. While in their 2004 they employed various strategies to improve the contextual model, in this case they want to focus only on projecting the lexical model via cognates, and so they chose two fairly close languages and directly use the tag sequence model from Spanish, with no further modifications.

There are several different lexical models one could use:

- The Spanish lexical model, though many Portuguese words aren’t cognates.
- The Portuguese lexical model, but they do not have this because they do not have a gold-standard tagged corpus for Portuguese
- A uniform distribution across all tags for each Portuguese word.
- The uniform distribution for Portuguese, but influenced by the Spanish lexical model specifically in the case of cognates.

For their baseline lexical model, they train entirely on Spanish and apply the model directly to Portuguese. The accuracy of this baseline part-of-speech tagger is 56.8%, which is to be expected, since many Portuguese words are not Spanish cognates. In the next experiment, they entirely discard the Spanish lexical model, but instead adopt a uniform distribution across all proposed tags for each Portuguese word. This approach makes sense if one has no idea of what the individual lexical probability of a specific Portuguese word should be. This methodology is identical to their approach in their 2004 paper, and indeed they obtain a 77.2% accuracy on full part-of-speech tags, a performance similar to their results in 2004 for Russian. Finally – and this is the novelty of their 2006 work – they

---

⁴ Though in (Feldman, Hana, Brew 2006), they apply their approach to three language pairs: Spanish-Portuguese, Spanish-Catalan, and Czech-Russian.
attempted to improve over this uniform distribution, by leveraging the Spanish lexical model in cases where the Portuguese word was a Spanish cognate. In case of a cognate, they average the uniform probability with the tag probabilities for that word from the Spanish lexical model. Thus, they do not adopt the Spanish lexical model wholeheartedly, but do use it to bias the uniform Portuguese lexical model in the direction of the Spanish (we refer to this approach with codename “H2” in this dissertation). And with this new lexical model in place, they achieve 87.6% accuracy.

In the Spanish-Portuguese work, they do not make use of subtaggers. This might be because, while they had 1000 possible tags for Czech, Portuguese has 259 tags, and Spanish has 282. Scarcity of data might still be a problem in training a trigram tag sequence model, but perhaps the reduced tagset combined with the closeness of Spanish and Portuguese made this less of an issue.

Another interesting point is that Hana et al. simply adopt the Spanish tag transition probabilities wholesale. This might work well for a language that is syntactically close to Portuguese. However, this approach may not work nearly so well when there are significant syntactic differences between the two languages that are being bridged. For example, consider applying the techniques of Hana (2006) to Yiddish and German. Yiddish differs from German in that it is V2 even in embedded clauses, that it eliminates the simple past ("I walked") and replaces it with the past participle ("I was walking"), and that nouns do not have case-marking. Furthermore, it is a different language, and so different constructions might be more or less frequently used than they are used in German. The contextual model of Yiddish would therefore be somewhat different. (Hana et al 2004) tried addressing this
problem somewhat by "Russifying" a Czech corpus and then training on it, but their attempt had little impact.

While the approach seems to perform well, we would note that it is difficult to compare their successes to the successes of other work. After all, this is not an apples-to-apples comparison. Different researches choose different languages, where particular language features may make one approach perform much better, or where there are sufficient resources for a particular approach to excel. In this dissertation research, we will address this limitation of prior evaluations by reimplmenting several techniques, and evaluating them on an identical set of languages drawn from the Europarl corpus.

2.4 A Quick Summary of Low-Density NLP Approaches

Since the above description of prior work in this chapter went into great detail, this section presents a brief overview of the most important points to come away with. Duh and Kirchhoff (2005) among other approaches, use a part-of-speech tagger which was developed for Standard Arabic (a high-density language) to tag low-density dialects of Arabic. Hana et al. (2004) train a Russian tagger using an annotated Czech corpus, and Hana et al. (2006) train a Portuguese tagger using an annotated Spanish corpus.

Briefly and broadly, prior approaches to low-density NLP can be categorized as follows:

- Apply high-density language resources to a related low-density language (Duh, 2005)

- Train low-density language tools from a limited corpus, and take steps to reduce noise and thus improve results (Duh, 2005)
• Blend results (lexical/contextual model) from a related HD-LD language pair (Duh 2005, Hana 2006)

• Change an HD-corpus into a pseudo-LD corpus, and perform supervised training on the pseudo-LD corpus (Hana 2004)

To give a slightly more detailed summary of some of these latter approaches: Duh and Kirchhoff (2005), in building a POS tagger for an LD Arabic dialect, focused on ways of making the most of limited data, as well as exploiting tools or corpora from related high-density or other low-density dialects. Thus, they performed:

• supervised training on an HD-corpus, and applied the resulting HD-tagger to an LD-corpus (D1)

• supervised training on an LD-corpus (D2)

• unsupervised training on an LD-corpus, but with an LD-lexicon (D3)

• unsupervised training on an LD-corpus, but with the lexicon generated by applying an HD morphological analyzer to the LD-corpus (D4)

• improvements upon the immediately previous by adding an affix model (D5)

• improvement upon the previous by constraining the lexicon for words unanalyzable by the HD-analyzer (D6)

In a separate experiment, which we do not reimplement in this dissertation research, the authors performed supervised training on LD-corpora from multiple dialects, under the assumption that while the lexical model will differ and needs to be trained separately, the contextual model is more or less equivalent.

Hana et al. (2004) trained an LD-tagger (for Russian) using an annotated HD corpus (Czech), assuming that the contextual models would be similar enough. Further, they
LD-ified the HD-tags, for example stripping out attributes of the (complicated) tags where the LD-language lacked those attributes. They trained the lexical model by lemmatizing an untagged LD-corpus, generating all word forms from the lemma, and imposing a uniform distribution across all such words. Hana et al. (2006) took a related HD and LD-language, adopted the contextual model wholesale, and borrowed the lexical model of the HD particularly where there was an LD cognate.

In summary, to illustrate the relationship of some of these approaches to one another, consider the following diagram.

Figure 1: An overview of approaches from the literature

A more expansive discussion of these studies is discussed in the following sections.
Chapter 3: Overview of Work

Given these diverse approaches in the literature, what is the best approach to using cross-lingual resources in NLP on low-density languages? For such languages, which lack large corpora that would enable straightforward statistical methods, there are a variety of approaches. Each of these approaches performs fairly well in the particular experiments designed to test that novel approach, on the particular language chosen to demonstrate the novel approach. But for someone starting from scratch on a new low-density language, how can one determine the best approach? It would be useful to compare apples to apples, and see how several different approaches perform on the same language.

Indeed, given that different approaches might exploit different aspects of languages or their relationships to other, high-density languages, it may well be that for one class of languages (e.g. Germanic languages), approach A might be better, while for another class of languages (e.g. Slavic languages; or languages with a close high-density language; or languages which exceed some threshold of existing linguistic resources), approach B might be better. If so, testing the approaches on a variety of language types will allow us to identify just what language features indicate that specific approaches are best. That is, there may not be one right or wrong answer – the question is what the correct approach is for the particular situation.

While there are many different NLP tasks, we will select POS-tagging as an example task for this dissertation research. We have made this choice for several reasons. (1) It is a
relatively limited task; (2) it is a task that much low-density language research has been done on, and (3) we can evaluate it readily and, hopefully, consistently.

Systematically implementing and comparing several approaches on the same set of languages may aid us in evaluating which is the "best" approach for that set of languages. Furthermore, if written correctly, these implementations will be general enough that we can readily substitute one language for another. Such a systematic testbed would be a useful contribution to the field, such that we could discover what works best for many pairs of languages.

Further, systematically implementing and comparing several approaches on the same set of languages may provide insight into just what is being done and just what is not being done. This may then reveal new approaches that have not yet been exploited.

In the second phase of this dissertation research, we explore some approaches that we believe have not been explored sufficiently, and we see how they compare to existing approaches, for several different types of languages:

Specifically, we explore how to “Language-ify” the lexical items in LD corpus to more resemble the HD before applying existing, or trained, HD-tools to it. Using limited rewrite rules, perhaps better recognition of certain lexical items will assist in the identification of surrounding words, given a functioning contextual model. Or conversely, we also explore how to language-ify the lexical items in an HD corpus to more resemble the LD before conducting supervised training on it. These approaches are discussed in greater detail in section 4.2.2.
3.1 Methodological Details

3.1.1 Corpus and Corpus Construction

Prior work had a limitation in that each group of researchers did not test on the same corpora, on the same languages, or using tagsets of similar granularity. Further, they did not attempt their approach on multiple languages, such that one approach (e.g. incorporating an affix model) may see similar results on languages with different features.

The overall goal of this dissertation research is to test each approach on several pairs of languages. Perhaps certain approaches perform better on certain pairs of languages, because of the linguistic proximity of the languages to one another, or e.g., because the morphology of the target-language lends itself to easy construction of simple linguistic tools. But given that some languages have different, larger, corpora from other languages, results may be skewed. Ideally, we would have a relatively identical corpus for each language.

The proceedings of the European Parliament (Europarl) is just such a massively parallel corpus. There is a large body of text (presently 55 million words), continuously updated, in 11 European languages. These languages are: Romance (French, Italian, Spanish, Portuguese), Germanic (English, Dutch, German, Danish, Swedish), Greek and Finnish. Unfortunately, this text is not POS-marked, and while they provide a sentence-alignment tool, the corpora are not word-aligned. But, with 11 languages, there are $11 \choose 2 = 11 \times 10 = 110$ possible language pairs from which to select, and many of the language are related to one another.

While a comprehensive approach – running all experiments on all 110 language pairs – would be ideal, the computation time to do so would be prohibitive. Instead, in chapter 5, we select seven of these pairs of languages, on the basis specific linguistic features which will allow us to explore specific linguistically-grounded predictions.
In order to provide POS tags to each corpus, we applied an existing, third-party, high-density POS tagger\(^5\) to each corpus. The resulting marked corpus is not entirely correct, and there is some noise. Still, for the purpose of the experiments conducted, we treat it as a gold standard. Later, we can rerun the experiments for individual languages on corpora hand-marked for POS, in order to assess how close our results conform to results on this less-noisy data. Still, it is useful to have this massively parallel multilingual corpus, so as to be able to assess how different approaches are best applicable to different language pairs, and to theorize about what features of these languages or language pairs make different approaches the better choice.

### 3.1.2 Multiple POS sets

Prior work was limited in that it was inconsistent in the granularity of tagsets. The more fine-grained the tagset, the more difficult the problem is. Thus, Hana (2004) had a

---

\(^5\) TreeTagger, available here: [http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/](http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/)
rather elaborate tagset, which involved many features such as person, gender, and number. Duh et al. (2005) tagged for noun, verb, determiner, and so on.

Aside from this limitation is prior work, that different languages have different POS tag sets complicates performance assessment. For example, if we apply an HD language tagger to an LD-language, the tagset applied will be that of the HD language. Yet, the gold-standard uses the LD-tagset! Therefore, we implement a mapping process for certain tagsets to certain other tagsets.

This mapping is not as straightforward as it might seem. Different tagsets for different languages have varying numbers of tags, and varying levels of granularity. One tagset might simply mark for NOUN and VERB, while another might distinguish between Pronoun, Common noun, and Proper noun. It is much easier to achieve "correct" results when the tags to be assigned are rather broad. And to assess the correctness of the results automatically by comparing to our gold-standard, as opposed to engaging in painstaking analysis by human linguists, we need to map several tags from one language's tagset to a single tag. Or in other instances, several tags to several other tags, which are collapsed into one. The result is a sort of lowest common denominator for tag assessment.

What we do is adopt the lowest-common denominator tagset. While this is an easier task, it still is a useful task within NLP. We could do this only at the assessment phase (where we tag with the full tagset where it exists for a language, but in assessment broadly lump all NOUN tags together), or else we could simplify the tagset before applying the different approaches. In practice, for the HD-ify approaches, we utilize third-party taggers, which make use of tagsets with higher granularity, and only convert the assigned tags to the lower
granularity tagset at the assessment stage. For all other approaches, we make use of the common-denominator tagset throughout.
Chapter 4: Implementation

4.1 Reimplementation of Previous Methods

We have reimplemented a representative sample of the aforementioned prior approaches, and tested these approaches on a set of language pairs, systematically. This testbed can help reveal in which situations and for which languages particular approaches perform best.

The prior approaches we have reimplemented are described in great detail in section 2.2 and 2.3, where they are referred to by the codenames D1 – D6, H1, and H2. Therefore, here we only briefly review these approaches, and describe how we have implemented them.

First, from Duh and Kirchoff (2005) there is the D1 approach: apply an HD-tagger to an LD corpus. If the languages are similar enough, perhaps the HD tagger will recognize some LD words, and perhaps the tag sequence model will be similar. While those authors treated this approach as a baseline, in this dissertation research, we achieved rather good results with this simple approach. For the third-party tagger, we used TreeTagger, because there were TreeTaggers available for all of our selected HD languages. The one exception was Finnish, but we trained a TreeTagger based on the Finnish Treebank.

There is also the D2 approach, which is supervised training on a small LD-corpus. Here, we assume we have hand-tagged a small LD corpus. We train a tagger on this LD corpus, and test on another set. In our reimplementation, all of our selected LD languages were actually HD languages, but for the purpose of our testing, we acted as if they were LD
languages without rich available resources. We did not hand-tag the LD corpus. Rather, we TreeTagged the untagged Europarl corpus for the LD language to create a training and testing corpus. We then trained on the “LD” training corpus and tagged the “LD” testing corpus.

Next, there is the D3 approach, which is **unsupervised training on the LD corpus, making use of an LD lexicon.** The idea is that we know the lexical items and the tagset, but we don't have any tagged corpus to train upon, but unsupervised training may give us correctly assigned tags. The author’s original article was ambiguous as to certain implementation details, and so we made some design decisions that we thought were sensible. We bootstrapped on a "lousy" tagging model by performing supervised training on the lexicon, which was an LD dictionary of words with all of the possible tags for each word. We generated the dictionary by applying the LD TreeTagger to the full LD corpus. There is no contextual model to be gleaned from a lexicon, as the contents of the lexicon are not sentences, and the word order doesn't matter. Rather, we conducted supervised training on 1000 sentences from the **HD** corpus, and used that HD contextual model as the basis for subsequent unsupervised learning on the untagged LD corpus.

Next, there is the D4 approach, namely **unsupervised training on the LD corpus, utilizing an HD lexicon.** This is the same approach as D3 above, but with even less information, since the lexicon is noisier. The original authors generated this lexicon by running an HD analyzer on the LD corpus. We do this by running the HD tagger on the LD corpus and collecting all tags. Now, for dialects of Arabic, this makes sense, because the dialects are so close to one another. However, the language pairs we select are actually different languages, and so the noise introduced here is actually much greater.
Next, there is the D5 approach, which is incorporating affix model into the D4 approach above. That is, instead of just using the conditional probability of the word given its tag, \( i.e. \ p(\text{word}|\text{tag}) \), for the lexical model, we multiply it by \( p(\text{prefix} \ | \ \text{tag}) \) and \( p(\text{suffix} \ | \ \text{tag}) \). The original authors made use of deep linguistic knowledge about Arabic to generate a list of all the prefixes and suffixes. Because of our design decision for this dissertation research to not assume such deep linguistic knowledge, our list of prefixes and suffixes are instead generated automatically, through unsupervised learning of the LD corpus.

Next, there is the D6 approach, which is the reduction of suggested tagset for unanalyzable words based on distributional criteria. There were certain words that were unanalyzable by the HD analyzer or by the TreeTagger. In such cases, for the D4 approach, the strategy was to assign all possible tags to such a word. In D6, we reduce the noise on the basis of distributional criteria. This involves clustering based on the surrounding tags, and then assigning only those common tags to the out-of-vocabulary words.

Next, there is the H1 approach. We trained the lexical model on an LD lexicon (mapping words to possible tags) and the contextual model on the HD corpus. Finally, there is the H2 approach, which is, in addition to H1, the adoption of lexical model from HD for cognates. We use the minimum edit distance algorithm on HD and LD wordlists, together with a threshold to reject candidate cognates that are too dissimilar.

4.2 New Approaches

4.2.1 Overview

Alongside reimplementation of these existing approaches, in this dissertation research, we have implemented several novel approaches.
Various approaches are possible, depending on the availability of particular language resources. What follows is a list of such language resources, where each approach will assume that it can start with some subset of these resources. Some of these resources are already required for the reimplementation of earlier approaches.

Some of these resources are slightly unreasonable to expect for an LD language. Others are more reasonable. It might be helpful to depict a scenario in which we have a specific goal and have certain resources. For this purpose, let us assume that we have a computational linguist “Bob,” who needs to quickly build a part-of-speech tagger for an LD language. He does not have expertise in the LD language. With additional time and additional expertise, he could develop a rather elaborate system. For example, given enough time and access to LD-language experts, he could have the experts tag an LD-corpus, and perform supervised learning on the large tagged corpus. Or, he could have the experts hand-craft linguistic rules for tagging the language. But all of those approaches would involve a good amount of time, effort, and knowledge. Such knowledge, added later, might improve performance of existing systems, but how could he get up in running with limited resources? The idea that Bob must build a system quickly and with limited resources is essential to the premise of this dissertation research, and this scenario has guided the set of techniques that have been explored.

Here are some resources which Bob might reasonably be expected to have, or could readily construct. In this listing below, the “LD” language is the one for which Bob must build a part-of-speech tagger. The “HD” language is another language (perhaps related somehow to the LD language) for which more high-density resources are available.
1. **An LD corpus, untagged.** Bob can expect such to exist. Otherwise, there is little point in developing a POS tagger for this LD language.

2. **An HD corpus, well-tagged.** This is reasonable to expect, for it is for the HD language. And where the corpus does not already exist, we can apply an existing HD tagger to an untagged corpus.

3. **Related, a good HD POS tagger.** Again, given that it is for the HD, it is reasonable for it to exist.

4. **A list of 250 very common LD words, and corresponding translations of those words in the HD language.** This is reasonable to expect, because it would be a list, on each side, of 250 words. Bob could generate this list by using a simple frequency count of the untagged LD corpus. Then, Bob could appeal to a language expert (or consult a bilingual dictionary) to generate the corresponding HD words, who could give us such a list in an hour or two. This list would include nouns, verbs and adjectives, but probably most importantly, rather common and frequent words such as determiners and pronouns.

5. **HD affix-based POS tagger/morphological analyzer.** Again, this would be extremely simplistic. The typical HD POS tagger, above, operates on a contextual model and a lexical model. But it would then fail for out of vocabulary words, as it would only rely

---

6 An alternative to this would be to generate a list of only the "important" words in the language (e.g., the determiners, pronouns, and prepositions, which are “important” in that they can provide key context in the contextual model), by consulting a simple written grammar of the language. We could then generate a mapping by consulting either a language expert or reading a simple grammar of the language. Or we could even develop a framework, in which we would put these "important" words into appropriate slots based on their role (e.g. plural definite article), and then generate the mappings automatically. While this alternative is preferable for various reasons (such as that it targets the important words that will likely differ across languages and will inform not just about themselves but their neighbors), it also entails greater linguistic knowledge than we would like to expect. Furthermore, it would be dependent on particular features of a language; for example, that it uses pronouns. For this reason, the "dumb" approach is preferable.
on the contextual model. This morphological analyzer would simply analyze the affixes of a word, and suggest possible POS tags.

6. **A list of HD affixes.** On the basis of this, Bob can use pattern matching to identify likely part-of-speech, as in (5), or to map affixes between the HD and LD. Since it is HD, it should be straightforward to come by. This would be a limited list, usually of under 50 prefixes and 50 suffixes. While one could use linguistic knowledge to identify these affixes, a better approach for our purposes does not assume such knowledge. Instead, Bob would automatically generate the most frequent affixes, using an unsupervised method on an untagged corpus.

7. **A list of LD affixes.** Again, using a simple unsupervised approach against an untagged corpus, this resource is within Bob’s reach.

8. **A list of LD affixes, matched with their regular HD affix replacements.** A list of HD affixes, matched with their regular LD affix replacements. The affixes of both the HD and the LD are readily discoverable from untagged corpora, as described in (6) and (7). The correspondences are between these are harder to acquire. Since we are dealing with a limited number of affixes, Bob could select a small number of LD words which contain the affix, consult a dictionary or a language expert for the HD word equivalent, and examine those word mappings to determine if any regular pattern of affix mapping occurs. The total number of affixes would be restricted, and is usually under 50.

9. **One word in each POS in the HD.** One HD noun, one HD verb, one HD adjective, etc. This list should be trivial to acquire, and should be as short as possible.

---

7 As mentioned previously, this would be done using the Automorphology program.
It is worthwhile to stress that there are certain resources which we do not assume that Bob possesses. For example, we do not assume that Bob is in possession of an LD lexicon (an LD wordlist couples with the possible parts of speech for each word). Similarly, we do not assume that Bob has access to a parallel word- or sentence-aligned corpus to project part of speech tags from the HD to the LD. It is true that, within our research, we make use of Europarl, which is an aligned corpus. However, none of our projection strategies exploit the parallel nature of this resource. We only use Europarl in order to be able to conduct and present an “apples-to-apples” comparison of projection strategies across several language pairs. Thus, Bob does not require any such parallel aligned corpus.

How we, in this dissertation research, or Bob, in his hypothetical scenario, would utilize each of these resources will be spelled out shortly.

First, though, we would like to broadly define two new approaches for making use of cross-lingual resources in NLP.

4.2.2 - Language-ifier

A language-ifier is a system that transforms some aspect of one language into that of another language. A language-ifier is not a machine translator; its much more modest aim is to modify a corpus slightly, so as to make it more palatable for use with other-language resources. As such, language-ification is by definition extremely limited in scope, and extremely straightforward and easy to implement. For example, a language-ifier may replace determiners and pronouns in Dutch for determiners and pronouns of German, but leave every other word in place. Or, it might replace the tagset of Dutch with corresponding tagset in German, in a tagged corpus. Or, it might use simple pattern matching to replace a short list
of affixes in Dutch with corresponding affixes in German. Or, it might move about words in
a Dutch corpus to correspond to word order as we would expect them in German.

The idea is to make the resulting corpus more "palatable" for use with other language
tools, or to train better tools for the other language.

Here is one practical example. As one baseline, Duh (2001) applied an HD tagger
(Modern Standard Arabic) to an LD dialect (Egyptian Arabic). Obviously, accuracy suffered
because it was a tagger for a different dialect, with a different vocabulary and (somewhat
different) tag sequence – though genetic similarity between languages, or dialects, would
decrease that gap somewhat.

But, if we language-ify the LD corpus to make it resemble an HD corpus, then
perhaps the HD POS-tagger could do a better job. If we replace the definite articles of the
LD with definite articles of the HD, the HD lexical model would recognize those HD words.
Furthermore, accurately recognizing the HD definite articles would aid in recognizing the
words in immediate context, because of the contextual model. (For example, in English, the
word after a definite or absolute determiner is probably a noun. The word after a nominative
pronoun is probably a verb.)

This dissertation research project would not be the first language-ifier, in the context
of part-of-speech tagging. Hana (2004) makes use of what we would term a language-ifier,
an LD-ifier, to transform an HD tagged corpus into something closer to an LD corpus, by
changing the tags sequences to more closely resemble those one would expect for the LD.
Then, when conducting supervised training on the pseudo-LD tagged corpus, they would
learn the contextual model for the LD, instead of that for the HD. See the right side of the
figure below.
The logical, yet unexplored approach would be to reverse the language-ifier of Hana (2004), to make an **HD-ifier** to transform an untagged LD corpus into something more similar to the HD language. We could then modify Duh's approach to improve accuracy of the tagging.

That is, if one could construct a language-ifier in one direction (HD $\rightarrow$ LD), then one could construct a language-ifier in the other direction. While Hana (2004) focused on tag sequence, a language-ifier could also focus on morphology or lexicon. And an HD-ifier (LD $\rightarrow$ HD) could then be used as a preprocessing step prior to applying the HD tagger, as in Duh.

**Both** LD-ifiers and HD-ifiers can be useful in exploiting tools and resources that exist for another language. We will elaborate below:

The process of language-ification, as we have dubbed it, is not without good precedent. Pre-processing the data to make it more “palatable” in subsequent steps has been done before – in other research contexts. While our focus is making the LD text more palatable for an HD tool, or making the lexical level of tagged HD text more similar to an
LD corpus prior to training, we see this approach used elsewhere. It seems quite common in **machine translation**, where there is a language pair in play. By moving the source closer to the target, in terms of number of words in a sentence, granularity of lexical items, and word and phrase order, alignment improves such that better word / phrase pairings occur, and the word order will more closely approximate that of the target language. There are several examples of this type of research:

- **Brown et al. (1992)** reordered French sentences and English sentences (into Intermediate French and Intermediate English) to simplify them, reduce the statistical variety, and make them more closely resemble one another. **Niessen and Ney (2004)** worked on German to English translation, and preprocessed the German source in various ways. For example, since German lexical items can be ambiguous, yet the ambiguity can often be resolved using analyzers on the German side, they expanded their vocabulary, replacing ambiguous lexical items with lexical items including markers provided by the morphosyntactic analysis. Further, since German inflects more than English, they created equivalence classes for words that would translate to the same English word, thus increasing coverage and smoothing out these irregularities. They also restructured their sentences, by changing word order in questions and for German prefix verbs, in which the prefix is detached from the verb and shifted to the end of the clause, they detect these prefixes and replace them with a single word with the prefix attached. This required linguistic knowledge of the source language and how it differed from the target language, as well as rich source-side NLP tools.

- **Collins et al. (2005)** similarly reordered German clauses to more closely resemble clause order of English, and subsequently **Wang et al. (2007)** reordered Chinese sentences to
more closely resemble English word order, prior to applying traditional machine translation techniques. This reordering required knowledge of source and target POS tags, as well as how the languages differed syntactically.

- Habash and Sadat (2006), for machine translation of Arabic to English, employed several preprocessing approaches to decrease ambiguity and word scarcity. They also used a preprocessing scheme to make the Arabic more resemble English. They split off conjunction clitics, the class of particles, the definite article and all pronominal clitics, marking them with English-like POS tags and explicitly indicating the pro-dropped word as a separate token.

- Bisazza and Federico (2009) preprocessed Turkish, an agglutinative language, using morphological segmentation schemes, prior to machine translation to English. This made the lexical granularity of Turkish more closely resemble English, reduced the total size of the training dictionary, and caused fewer words in the test set to be out of vocabulary. They note that “Morphological segmentation is highly language dependent and requires a fair amount of linguistic knowledge in its development phase,” which I think is an important point. They split off segments of words with the target language, English, in mind. Thus, for example, case endings with an English counterpart were split off, while those without an English counterpart were removed. In another interesting step, they modified the test data, replacing OOV words with the closest match in the training data.

- Hong et al. (2009) discovered that about 25% of Korean words and 21% of English sentences fail to align. These null alignments degrade the translation quality. Therefore they transform the source language sentences. If the source language is Korean, they simply remove the extra unaligned Korean words (which are e.g. case particles and final
endings). If the source language is English, they insert pseudo-words into the English sentence to match with the potential Korean function words. Furthermore, using structured parse trees, they reorder the words in accordance with rules such as moving negative markers to directly follow the verbal head.

All of this research listed above is within the realm of machine translation, where there is a source and target language, and a motivation to make one more clearly resemble the other. These pre-processing strategies could also have applications for the projection of high-density language NLP tools to low-density languages. For example, as was discussed in the literature survey at the beginning of this proposal, Yarowsky et al. (2001) and Hwa et al. (2005) project linguistic resources across parallel texts, and the latter uses mono-lingual post-projection rules to fix up difficulties caused in part by absence of such information, or items, on the source language side. Instead, one could imagine the use of pre-projection rules to make the source language more closely resemble the target language would yield better alignments.

Some low-density NLP researchers have used pre-processing techniques. As discussed in the literature survey at the beginning of this proposal, Hana et al. (2004), in working on an LD POS tagger trained on HD data, indeed tried preprocessing rules to make tags of the HD corpus more resemble the tags one would find in an LD. This required somewhat deep linguistic knowledge of how Czech (the HD) and Russian (LD) differed. Yet, their results were not encouraging.

Within this dissertation research, we consider a number of such preprocessing rules for the purpose of POS tagging. While Hana et al. (2004) focused on the tag level, these approaches focus on the lexical level. And where it seems like all of the approaches above
require somewhat deep linguistic knowledge of the source language, or of both languages such that one knows how they differ, since we are concerned with low-density languages, our own focus is on approaches which do not require such expert linguistic knowledge. Rather, L1 is simple (and possibly noisy) replacement of the most frequent 250 words, which is a simple dictionary lookup, or else limited consultation with a language expert. L2 is affix replacement, where we automatically detect affixes and replace one for the other. L3 is exemplar replacement, in which we run simple affix-based pattern matching on OOV words and replace them with an HD exemplar from the selected POS. L4 is cognate replacement, automatically constructed from separate HD and LD wordlists.

These preprocessing rules apply in both directions (LD to HD and HD to LD), and on either the test data or the training data. Thus, we can move the LD data to more closely resemble the HD, such that the performance of the HD tagger will improve. We can do the same prior to tagging, and then train on the resulting tagged corpus. We can modify a tagged HD corpus to better resemble an LD corpus, on the lexical level, such that training on the pseudo-LD corpus will yield a better LD tagger.

This dissertation research on language-ification thus differs from many other approaches to low-density POS tagging. To summarize how this dissertation research is novel, we will identify the differences between this and prior work:

- Bender and Flickinger (2005) and Loftsson (2007) developed frameworks to speed up development time and aid in the process of POS-taggers, designed by language experts.

My approach does not require linguistic expertise in the LD language.
• Thompson et al. (1999), Tang et al. (2002), Baldwin (2005), and Rambow et al. (2006) manually create the resource and combine this with a statistical approach. This once again requires linguistic expertise, to manually POS-tag a corpus.

• Yarowsky et al. (2001) and Hwa et al. (2005) project linguistic resources over bilingual parallel corpora. We don’t assume that such corpora exist for our LD language. While there is some work in utilizing monolingual comparable corpora (Barzilay (2003), Rambow et al. (2006)), we don’t even assume that level of resources.

• Hana et al. (2004) attempted some language-ification. But this required a sophisticated level of linguistic knowledge of Czech (the HD) and Russian (the LD), which our approach will not assume. The approach of those authors operated on the tag level, while our approach operates on the lexical level. Finally, their approach did not work well, while our results (presented in a later chapter) indicate that our new approach does work.

As discussed above, this dissertation research’s approach of language-ification also differs from the prior work in machine translation. This is a different domain – POS-tagging – and we are exploring the use of preprocessing techniques for low-density languages. Further, much of the exiting MT work involves changing word order or splitting off parts of words, while our approach involves making the lexemes more closely resemble the target language. Further, much of the existing MT work requires some deep linguistic knowledge, either in order to process or in order to design rules.

Thus, reordering rules to move about parts of phrases will require knowledge of the POS tags, which is the very thing we are trying to find out, and possibly require built-up parse trees. Also, in much of the work on pre-processing for machine translation, linguistic experts in these languages, who know just how the source and target languages differ, are
the ones who design the rules. For example, Bisazza and Federico (2009) needed to know Turkish quite well in order to design an approach for morphological segmentation. And they needed to know which segments corresponded to English words and while corresponded with nothing. Meanwhile, our approach assumes a very low level of linguistic knowledge of the LD language – recall the limited list of resources at Bob’s disposal in our hypothetical scenario.

4.2.4.1 General Scheme of Work on Novel Approaches

We investigate two different general techniques, LD-ification and HD-ification, in a systematic manner. If the LD language is Mohammed and the HD language resource is the mountain, then either Mohammed can come to the mountain (HD-ification), or the mountain can come to Mohammed (LD-ification).

That is, if we have an HD corpus, instead training an HD tagger on it, we first transform the lexical items of that corpus into something more like the LD. We train on that LD-ified corpus and the output is a trained “LD”-tagger. We can then use that tagger to tag LD text.
4.2.4.3 HD-ifying an LD input text

This section describes HD-ification of the LD input text prior to application of the HD tagger. The following diagram illustrates the general process of HD-ification:

That is, we assume that English is the LD language and German is the HD language. We take our English LD text and use a German HD-ifier to transform it into text that looks more like German. We then pass this modified text as input to the 3rd party German HD tagger. That German HD tagger outputs the tags that it assigns to the modified corpus.

We apply four different HD-ification strategies in an attempt to transform the LD corpus, in order to make it more palatable to the HD tagger:

1. Gloss replacement of the 250 most frequently occurring LD words with their HD equivalents (L1)
2. Replacement of longest recognized affixes with their HD equivalents (L2)
3. Gloss replacement unrecognized words with a HD exemplar, after affix-based projection to an HD tagged corpus (L3)
4. Gloss replacement of LD words with their automatically selected cognate (L4)

In order to perform L1 HD-ify, we need a dictionary that maps the 250 most frequent LD words to their HD equivalents. To construct this dictionary, we use a computer program...
that takes an untagged LD corpus as input and outputs an LD wordlist, in descending order of frequency. Then, a human expert translates each LD word into its HD equivalent.

Figure 6 Generating the HD-ify frequent word dictionary

With this LD → HD dictionary resource in place, L1 HD-ification will take any LD input text, consult the dictionary and perform gloss replacement of every LD word in the dictionary. Finally, the HD tagger can be used to tag the HD-ified text.

Figure 7 L1 HD-ify process

In order to perform L2 HD-ify, we need an affix dictionary consisting of common LD prefixes and suffixes mapped to their HD equivalents. To generate this dictionary, we start with an untagged LD corpus and use the work of Goldsmith (2001), namely the Automorphology program, which uses unsupervised learning of the morphology of a natural
language. Then, we use a computer program to extract from the LD corpus a list of ten LD words with that affix. A human translates those words to the HD, and determines if there is any regular mapping from an LD affix to an HD affix. If so, that affix mapping is added to the affix dictionary.

Another resource which L2 HD-ify uses is a large untagged HD corpus. The strategy is as follows: For each word in the LD corpus, utilize every possible prefix and suffix mapping to generate a series of candidate HD words. (This includes: (a) replacing neither the prefix nor the suffix, (b) replacing only the prefix, (c) replacing only the suffix, and (d) trying prefixes and suffixes of varying lengths.) Then, select the candidate word that appears most frequently in the untagged HD corpus. The goal of this strategy is to not accidentally move words out-of-vocabulary via overeager affix replacement.
In order to perform L3 HD-ify, we utilize the following resources:

1. the same affix dictionaries as were generated for L2 HD-ify,
2. a tagged HD corpus, and
3. an unambiguous HD exemplar (human generated, for each part-of-speech).

The strategy is as follows: Consult the HD tagger (or an HD corpus) to discover which words are out-of-vocabulary. For those words, take the longest prefix and suffix and use those to project to the HD corpus, via the affix dictionaries. Rather than looking for a single target HD word, in the hopes that projection hit an actual word, consider every HD word which possesses that HD prefix and suffix, and find the majority part of speech. Then, using the exemplar list, replace the original out-of-vocabulary LD word with the HD exemplar. Finally, tag using the HD tagger.
The effect of this strategy is to pre-tag those out-of-vocabulary words based on their affixes (selected the tag from the corresponding combined affixes in the HD corpus), and then allow the normal HD tagger to operate upon the rest of the words.

In order to perform L4 HD-ify, we need a dictionary that maps LD words to their HD cognate. We do not develop a complicated, and language-dependent, cognition model in order to identify LD words and their cognates. (Recall that our goal in this dissertation is to investigate quick strategies that use minimal resources.) Instead, we use minimum edit distance to identify cognates. The cost function for this minimum edit distance is as follows: substitutions of a letter for itself costs 0; of a vowel for a different vowel, 0.25; otherwise, a cost of 1. This is because vowel substitutions are common in cognates across several languages, and so we would like to favor substitutions vowels to the alternative. As for insertions and deletions, the cost of inserting or deleting a consonant is 1; for a vowel, if the previous letter was a vowel, the cost is 0.25; and a 1 otherwise. Once again, this is because often a word will have two vowels where its cognate has one, and we would like to penalize such a vowel insertion or deletion less than inserting or deleting a vowel elsewhere.
Not every word has a true cognate, and so we imposed a threshold to attempt to separate out the real cognates. That is, we normalized the minimum edit distance by dividing it by the average length of the source and target words, and then took only those greater than the threshold of 0.5.

To build this cognition dictionary, we used as input large untagged corpora for the LD dictionary and the HD dictionary. Precomputing each of these cognates takes a long time, but by splitting the source corpus into pieces, we took advantage of parallel processing.

![Figure 11 Generating the cognate dictionaries](image.png)

With this LD → HD cognate dictionary resource in place, L4 HD-ification will take any LD input text, consult the dictionary and perform gloss replacement of every LD word in the dictionary with its HD cognate. Finally, the HD tagger tags the HD-ificied text.
4.2.4.2 LD-ifying an HD Corpus

The previous section discussed HD-ification, transforming the untagged LD input text so that it would resemble the HD language and tagger. Here, we discuss the reverse approach, LD-ification, which is transforming a tagged HD corpus so that it resembles the LD language. Training on such an LD-ified corpus would produce a tagger more capable of tagging the LD language. The following diagram illustrates the general process of LD-ification:

Figure 13 The general process of LD-ification

The HD-ification approaches required the construction of limited linguistic resources, and the LD-ification approaches require this as well. In general, the resources to be constructed here are the reverse of the resources required for HD-ification.
Thus, while L1 HD-ify required a dictionary of the most frequent LD words and their corresponding translation in the HD language, L1 LD-ify required a dictionary of the most frequent HD words and their corresponding translation in the LD language. For L2 LD-ify, we ran an untagged HD corpus through Automorphology and extracted affixes, selected translations of HD words with those affixes, and looked for regular patterns of HD affix \( \rightarrow \) LD affix. For L4 LD-ify, we produced a cognate dictionary mapping HD words to their LD cognate (which is not the same as LD words to their HD cognates).
Chapter 5: Language Discussion

As noted previously, Europarl is a parallel corpus, the proceedings of the European Parliament. It is untagged, but we can use third-party tools (TreeTagger) to introduce a somewhat noisy tagset, to create a tagged corpus for further experimentation. It is positive that the texts are parallel, for it leads to better language-to-language comparison, with corpora of the same size in the same domain, with similar content. Europarl is available in 21 languages, though some languages cover more years of the Parliament Proceedings. The approaches considered in this thesis involve creating and exploiting connections between related languages. In theory, we could consider every single possible language pair, where $21 \times 20 = 420$. Besides taking an inordinate amount of time, and cause us to drown in data, it would not yield us much more informative data than if we carefully selected our language pairs.

Therefore, our approach will be to consider linguistic features of the possible languages, and select those language pairs which enable us to analyze several predictions about how the properties of the pair of languages might relate to the relative success of various approaches. We would expect certain language-ification approaches to work better on specific language pairs over others. A full discussion of language-ification approaches and how they might relate to language features will follow but, by way of illustration, Dutch and German are both West Germanic languages, a branch of Germanic languages, which is a branch of Indo-European languages. Meanwhile, Finnish is Uralic, rather than Indo-
European language. We might therefore expect greater language similarities between Dutch and German, and thus a shorter language bridge to cross, either before or after language-ification strategies are employed.

![Figure 14: Dutch is closer to German](image)

We consider specifically those language features which we believe would have an impact on our approaches. In brief, these approaches include (a) applying a 3rd-party HD tool to an LD-language, (b) modifying the LD language to make it more palatable to the HD tool, and (c) modifying an HD corpus to better approximate a LD corpus, and training a tagging model on that LD corpus. A tagging model encompasses a lexical model (word given tag) and a contextual model (tag given previous tag(s)), and similarity or dissimilarity of the linguistic features of two languages can impact the bridging of one or both of these models.

Some linguistic similarities will affect the lexical model more. For example, a large overlap in the vocabulary of two languages would mean a greater number of in-vocabulary words, and thus a more effective application of the HD lexical model. A large number of cognates shared between two languages might mean that a cognate replacement strategy will bring more words in vocabulary than if the two languages did not share those cognates.

Some linguistic similarities will affect the contextual model more. For example, if within two languages, adjectives regularly precede nouns, then the contextual model of the HD language might be more effectively applied to the LD language than if the adjective – noun order differed. And if a replacement strategy would, for instance, aid an HD tagger in
identifying adjectives LD adjectives, then this might mean that the nouns, which regularly follow, will be correctly tagged. However, if the two languages in a pair do not share this regular adjective and noun order, then the HD contextual model cannot be brought to bear to inform about the following noun.

The linguistic features we would like to consider are:

1. Language family – section 5.1
2. Constituent Word order (SOV, SVO, etc.) – section 5.2
3. Do adjectives appear before or after the nouns they modify (noun phrase word order) – section 5.3
4. The presence or absence of definite and indefinite articles – section 5.4
5. The lexical granularity of definite and indefinite articles – section 5.5
6. Is the language pro-drop? – section 5.6
7. How richly is it inflected? – section 5.7

There will naturally be some overlap between these linguistic features. For instance, pro-drop languages are usually richly inflected while non-pro-drop languages are not. And languages within the same language family are likely to share many linguistic features. Still, we would like to consider each of these language features separately, and discuss how they might impact our language-ification approaches.

These linguistic features we would like to consider will help us identify specific language pairs to study. Of course, there are two other practical considerations which will also restrict the languages we may consider, namely which languages appear in Europarl and which languages may be tagged by TreeTagger. The 21 Europarl languages are: French, Italian, Spanish, Portuguese, Romanian, English, Dutch, German, Danish, Swedish,
Bulgarian, Czech, Polish, Slovak, Slovene, Finnish, Hungarian, Estonian, Latvian, Lithuanian and Greek.

Also, recall that Europarl is untagged, so we need third-party taggers to tag it. Thus, we are restricted to those relatively “HD” languages for which a 3rd party tagger exists. To maintain consistency across languages, I chose TreeTagger as that 3rd party tagger. Of those 21 Europarl languages, there were nine for which there also existed TreeTagger parameter files: French, Italian, Spanish, Portuguese, English, Dutch, German, Bulgarian, Slovak, and Estonian. To these, we added a parameter file for Finnish, by training it on the Finnish Treebank.

5.1 Language Families

Languages are often grouped by linguists into language families. These reflect historical development of these languages. But languages within the same language family are likely to share a great number of linguistic features. For instance, two closely related

Figure 15: Windows Interface for TreeTagger

Germanic languages are likely to share quite a number of cognates, due to shared vocabulary. They are likely to share syntactic features as well, such as constituent word order, morphology, and presence of articles. We would then expect that the closer the relationship between languages, the better fit we would achieve in transferring both the *lexical* and *contextual* model, both with and without language-ifications.

The hierarchical tree in Figure 16 (below) demonstrates the relationship between all the languages which occur in the Europarl corpus. Those leaf notes colored in green are the ones for which there is a TreeTagger.
On the basis of this tree, we select the following language pairs, with German being the LD language in each case:

1. (Dutch, German) as very closely related Germanic languages
2. (English, German) as less closely related Germanic languages
3. (French, German) as quite unrelated languages, where the first of the pair is not a Germanic language at all.

In each case, the second item in the language pair is German, for purposes of better comparison.

Prediction [P1a, b]

We predict that simple application of HD taggers to LD text will perform best for pairs of languages with phylogenetic or historical cultural connections.

More concretely, we predict the following, about the D1 approach, which is simply applying the HD tagger to the LD language.

A. An HD Dutch tagger tagging German will have greater accuracy than an HD English tagger tagging German.

B. An HD English tagger tagging German will have greater accuracy than an HD French tagger tagging German.

Prediction [P2a, b, c, d]

We would also predict that low-density NLP projection approaches based on cognate-identification techniques will perform best for pairs of languages with phylogenetic or historical cultural connections.
One of our language-ification strategies, L4 HDify, identifies shared cognates within a language pair and, prior to employing the HD tagger to tag an LD sentence, replaces each instance of an LD word with its cognate, if a cognate exists.

Another of our language-ification strategies, L4 LDify, operates in similar manner but in the opposite direction. It identifies shared cognates within a language pair and then modifies an HD corpus, replacing HD words with their cognates, if a cognate exists. A TreeTagger is trained on that modified LDified corpus and used to tag the LD language.

These language-ification based on cognates operate primarily on the **lexical model**, through reduction of noise by bringing out-of-vocabulary (OOV) words into the vocabulary. However, once words are brought in vocabulary, there is a greater opportunity for the contextual model to work effectively as well. (Once we know a word is an adjective, the contextual model can figure out that the next word is likely to be a noun.)

Additionally, one of the prior approaches we reimplemented was H2. Recall that the H2 strategy, adopts wholesale the HD language’s contextual model. For the lexical model, it takes the average of the uniform LD lexical model (all tags assigned a word in a simple dictionary) and the HD lexical model for that word, if it is a cognate.

We expect that between closely related languages, there would be a greater number of shared cognates. Furthermore, we expect that the cognates, where automatically detected, would be of higher quality (less noisy) in more closely related languages, wherever a cognition strategy is used. Therefore, we predict that for the cognate language-ification approaches (L4 HD-ify and LD-ify), there would be a greater increase over the baseline for closely related languages than for distant languages. And for the same reason, we predict that H2, as well, would be more successful if the languages are closer.
We therefore select the following language pairs to test this prediction:

1. (Dutch, German) as very closely related Germanic languages
2. (German, Dutch) as very closely related Germanic languages
3. (Finnish, German) unrelated languages

We predict that there would be a greater number of cognates between Dutch and German than between Finnish and German in their respective vocabularies; and further than slight morphological differences between Dutch and German can be bridged by cognition in a way that Finnish and German might not.

(A note, for the sake of clarity: for the language-ification strategies, we consider the difference from the baseline approach, and refer to this as delta.)

Therefore, we would make the following concrete predictions:

A. L4 HD-ification on (Dutch, German) would be more effective at bridging the language gap – that is, it would have an increase over the baseline – than L4 HD-ification on (Finnish, German). That is, delta L4 HD-ify on (Dutch, German) > delta L4 HD-ify on (Finnish, German)

B. L4 HD-ify (German, Dutch) > delta L4 HD-ify on (Finnish, German)

C. H2 on (Dutch, German) would be more accurate than H2 on (Finnish, German)

D. H2 on (German, Dutch) would be more accurate than H2 on (Finnish, German)

5.2 Constituent Word Order

Languages differ in word order. By this, we mean whether the arrangement in a phrase is Subject Object Verb (SOV), Subject Verb Object (SVO), or some other permutation. This can have an impact on the contextual model, or p(t | previous tags). Languages with identical word order can be more successfully bridged without modification
to the tag sequence. Depending on the granularity of the tag set, this could have greater or lesser impact. For instance, say a tag set distinguishes between nouns in the nominative case (subjects) and nouns in the accusative case (objects). Then, the contextual model will separately contain $p(\text{verb} | \text{nominative_noun})$ and $p(\text{verb} | \text{accusative_noun})$, which captures, to an extent, the constituent word order. If a pair of languages shares word order, then their contextual models will resemble one another in this regard. Even with lesser granularity, where we simply consider nouns and verbs, there can be greater similarity between the contextual models. For instance, $p(\text{PERIOD} | \text{verb})$ vs. $p(\text{PERIOD} | \text{noun})$, or $p(\text{noun} | \text{verb})$.

It would, therefore, be instructive to consider the cases of languages pairs which are similar to each other, and differ from each other, in this particular language feature. We probably don’t need to consider all permutations of (Subject, Verb, Object) order when selecting language pairs. Rather, it is a question of linguistic similarity in this feature.

There are two definitions of constituent word order. One is the surface word order, and one is a conceptual word order. For instance, a language might be underlyingly SOV, but movement can rearrange the surface word order. While it might be of interest to consider the base structure in creating a contextual model, in practice we consider, and statistically train on, the surface word order. Different base word orders can still have some sort of impact. Thus, German is fundamentally SOV, but has V2 in main clauses (due to movement) and SOV in subordinate clauses. This will differ from a language which in fundamentally SVO, which would have SVO in subordinate clauses as well.

---

9 Or, $p(\text{article} | \text{verb})$. These cases are just for the purpose of illustration.
The following chart considers all the languages in the Europarl corpus, and the particulars of this language feature.

Table 1 Constituent Word Order of Languages in Europarl

<table>
<thead>
<tr>
<th>Europarl Language</th>
<th>Constituent Word Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulgarian</td>
<td>Stylistically, 80.5% are SVO. Yet, with the DOC (direct object clitic), all other word orders are allowed. So, free word order.</td>
</tr>
<tr>
<td>Czech</td>
<td>Free, because inflected forms indicate the syntactic relations. Basic word order: SVO. But objective word order, subjective word order can cause a reordering. VSO for questions.</td>
</tr>
<tr>
<td>Danish</td>
<td>SVO; has inversion in questions, with verb placed before the subject.</td>
</tr>
<tr>
<td>Dutch</td>
<td>SVO in main clause, SOV in subordinate clauses</td>
</tr>
<tr>
<td>German</td>
<td>SVO in main clause, SOV in subordinate clauses</td>
</tr>
<tr>
<td>Greek</td>
<td>SVO, but more relaxed than in English, due to richer morphology.</td>
</tr>
<tr>
<td>English</td>
<td>SVO</td>
</tr>
<tr>
<td>Spanish</td>
<td>SVO</td>
</tr>
<tr>
<td>Estonian</td>
<td>V2. SVO in main clauses, for neutral word order. Used to have SOV a lot in early 20th century in embedded clauses, but nowadays only in minority. So originally might have been SOV. This would not really impact us much in terms of tagging. Because it is highly inflective, also has fairly free word order.</td>
</tr>
<tr>
<td>Modern Finnish</td>
<td>SVO. Proto-Uralic was SOV. But because it is highly inflected, fairly loose about placement of adverbials, for instance.</td>
</tr>
<tr>
<td>French</td>
<td>SVO for nouns, SOV for pronouns.</td>
</tr>
<tr>
<td>Hungarian</td>
<td>Neutral word order is SVO. But pragmatic word order due to it being a topic prominent language, so sometimes roughly described as a free word order.</td>
</tr>
<tr>
<td>Italian</td>
<td>SVO. But other orders, like OVS, SVO, VSO, for emphasis or stylistic concerns.</td>
</tr>
<tr>
<td>Lithuanian</td>
<td>Main word order: SVO, but relatively free, because highly declined.</td>
</tr>
<tr>
<td>Latvian</td>
<td>Main word order is SVO, but relatively free word order.</td>
</tr>
<tr>
<td>Polish</td>
<td>SVO</td>
</tr>
<tr>
<td>Portuguese</td>
<td>SVO</td>
</tr>
<tr>
<td>Romanian</td>
<td>SVO</td>
</tr>
<tr>
<td>Slovak</td>
<td>SVO, relatively free</td>
</tr>
<tr>
<td>Slovene</td>
<td>SVO, relatively free</td>
</tr>
<tr>
<td>Swedish</td>
<td>SVO, more rigid word order</td>
</tr>
</tbody>
</table>

Thus, there seem to be 4 options amongst the languages in Europarl.

---

10 So, Leafgren (2002). According Dyer (1992), SVO is statistically most common, and also stylistically neutral.

11 See Elhala (1995)
1. **French** looks interesting because of the different of SOV for pronouns and SVO for nouns.

2. **German and Dutch**, as SVO in main clause, SOV in subordinate clause.

3. **English** as generally SVO, with fixed word order.

4. **Finnish and Estonian**, which are SVO, but with a lot of flexibility due to inflection.

We would want to compare languages which were similar to each other, and languages which diverge from each other.

**Prediction [P3a, b, c, d]**

We predict that Low-density NLP projection approaches based on simple adoption of the HD contextual model will be most effective for pairs of languages that are closest in their fundamental word order (e.g., SVO, SOV, etc.).

That is, one of the approaches re-implemented in this dissertation is called H1. It builds it lexical model on a Low-density language dictionary of words to tags, assigning equal probability to each tag. So, e.g., if the word “bank” appears in the dictionary as a noun and a verb, then the lexical model considers noun and verb as equally likely tags for “bank”. Since this lexical model is provided for each LD language, we can more or less **cancel out** the lexical model when comparing different languages. So drastic differences in vocabulary between the HD and LD languages simply do not matter. Meanwhile, the contextual model is adopted wholesale from the HD language. Therefore, similarities in contextual models would greatly affect H1.

Therefore we select the following language pairs when considering H1:
1. (German, Dutch) as languages with very close constituent word order: with SVO in main clause, SOV in subordinate clause.

2. (German, Finnish) as languages with more distant constituent word order. While both are SVO, Finnish has relatively free word order.

3. (Estonian, Finnish) as languages with very close constituent word order: with SVO as the neutral word order but generally relatively free word order due to high inflection.

4. (English, German) as languages which are close but different. Both are Germanic languages and have SVO in general, but German differs by having SOV in the subordinate clause.

With these languages chosen, our predictions about the relative accuracy of the H1 strategy is as follows:

A. We predict that H1 on (German, Dutch) would be more accurate that H1 on (German, Finnish). German and Dutch both have a fixed word order, have SVO in the main clause, and have SOV in the subordinate clause. Finnish, meanwhile, always has SVO as the neutral order, and has relatively free word order.

B. We also predict that H1 on (Estonian, Finnish) would be more accurate that H1 on (German, Finnish). Estonian and Finnish both have SVO as the neutral order, and have relatively free word order. German, meanwhile, differs as described above.

C. We predict that H1 on (German, Dutch) would be more accurate than H1 on (English, German). While German, Dutch and English all have SVO in the main clause, only German and Dutch have SOV in the subordinate clause.
D. We predict that H1 on (English, German) would be more accurate than H1 on (German, Finnish). Even though English word order is different from German in its subordinate clause, it still more closely resembles German in the main clause and in its fixed word order.

While this lends insight to how H1 might perform, we expect that the impact of the language relatedness to the contextual model would be felt even when applying various simple or more complex projections as well. They would provide a boost, in that the language pair starts with contextual models that are similar. And then the lexical projections would bring words into vocabulary, allowing the contextual model to operate more effectively with the new data it has.

### 5.3 Adjective premodifiers and postmodifiers

Another language feature that might impact contextual models, and efforts to apply them across languages, is whether the adjective typically precedes the noun or follows it. This can impact whether a determiner will predict a noun or an adjective, and whether an adjective will predict a noun or vice versa. Where target and source language share this same order, the contextual model will transfer much more readily. Here is a survey of the languages in the Europarl corpus, in terms of whether adjectives are premodifiers (coming before the noun) or postmodifiers (coming after the noun).
<table>
<thead>
<tr>
<th>Europarl Language</th>
<th>Adjective Pre- or Post-Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulgarian</td>
<td>premodifier</td>
</tr>
<tr>
<td>Czech</td>
<td>premodifier</td>
</tr>
<tr>
<td>Danish</td>
<td>premodifier</td>
</tr>
<tr>
<td>Dutch</td>
<td>premodifier</td>
</tr>
<tr>
<td>German</td>
<td>premodifier</td>
</tr>
<tr>
<td>Greek</td>
<td>premodifier (In NT Greek, attributive before the noun, restrictive after the noun.)</td>
</tr>
<tr>
<td>English</td>
<td>premodifier</td>
</tr>
<tr>
<td>Spanish</td>
<td>before or after, depending on purpose</td>
</tr>
<tr>
<td>Estonian</td>
<td>premodifier</td>
</tr>
<tr>
<td>Modern Finnish</td>
<td>premodifier</td>
</tr>
<tr>
<td>French</td>
<td>depends on type and meaning; but most are postmodifiers</td>
</tr>
<tr>
<td>Hungarian</td>
<td>attributively, before the noun; predicatively, after the noun</td>
</tr>
<tr>
<td>Italian</td>
<td>in general, postmodifier. But certain common adjectives come before.</td>
</tr>
<tr>
<td>Lithuanian</td>
<td>premodifier</td>
</tr>
<tr>
<td>Latvian</td>
<td>premodifier</td>
</tr>
<tr>
<td>Polish</td>
<td>postmodifier</td>
</tr>
<tr>
<td>Portuguese</td>
<td>postmodifier</td>
</tr>
<tr>
<td>Romanian</td>
<td>postmodifier</td>
</tr>
<tr>
<td>Slovak</td>
<td>premodifier</td>
</tr>
<tr>
<td>Slovene</td>
<td>premodifier</td>
</tr>
<tr>
<td>Swedish</td>
<td>premodifier</td>
</tr>
</tbody>
</table>

Table 2: Adjectives as premodifiers or postmodifiers

From this table, we select two language pairs on the basis of their adjective-noun order. We select a language pair which is alike and one which is dissimilar in this regard. Alas, those that are dissimilar also are from more distant language families, such that it is difficult to test just the effect of this one language feature.

1. (English, German) as (premodifier, premodifier)

2. (French, German) as (mostly postmodifier, premodifier)

We would expect that language pairs which are alike in positioning their adjectives and nouns would possess a greater similarity in their contextual models that languages which are dissimilar.
We therefore predict that Low-density NLP projection approaches based on simple adoption of the HD contextual model will be most effective for pairs of languages that are closest in their noun-adjective order (e.g., premodifier, postmodifier, mixed).

Consider the following LD English sentence and its HD German equivalent:

The quick brown fox jumped over the lazy dog.

Der schnelle braune Fuchs sprang über den faulen Hund.

Figure 17: German and English premodifier example

The word order throughout the sentence is identical, from the word the until the period. And so, the two adjectives quick and brown precede the noun fox, just as the two adjectives schnelle and braune precede the word Fuchs. Likewise, lazy precedes dog, just as faulen precedes Hund.

Because of sentences like this, the following trigrams of tags might be expected to occur often in both English and German:

1. # ARTICLE ADJECTIVE – # the quick
2. ARTICLE ADJECTIVE ADJECTIVE – the quick brown
3. ADJECTIVE ADJECTIVE NOUN – quick brown fox
4. ADJECTIVE NOUN VERB – brown fox jumped
5. PREPOSITION ARTICLE ADJECTIVE – over the lazy
6. ARTICLE ADJECTIVE NOUN – the lazy dog
7. ADJECTIVE NOUN PERIOD – lazy dog.
(We didn’t list *fox jumped over* and *jumped over the*, which are the only trigrams which don’t include an adjective.) Each listed trigram will occur, with some strong possibility, in both the German and English contextual model. Each listed trigram is only possible because adjectives are premodifiers of nouns. If a language had adjectives as postmodifiers of nouns only, then not a single one of these trigrams would be possible, for the following reasons:

1. Not possible since only nouns would follow articles, not adjectives.
2. Not possible for the same reason.
3. Not possible because plus the noun cannot follow the adjective.
4. Not possible because the noun cannot follow the adjective; we would expect the verb to come after the adjective instead.
5. Not possible because the adjective cannot immediately follow the article
6. Not possible for the same reason.
7. Not possible because the noun cannot follow the adjective.

Indeed, consider the same LD English sentence when translated to French, where the adjective-noun order depends on type and meaning:

**The quick brown fox jumped over the lazy dog.**

**Le renard brun rapide saute par dessus le chien paresseux.**

*Figure 18: English premodifier and French postmodifier example*

In this example, none of the seven trigrams listed above occur. Instead, the corresponding French trigrams trigrams are:
1. # ARTICLE NOUN– #Le renard
2. ARTICLE NOUN ADJECTIVE – Le renard brun
3. NOUN ADJECTIVE ADJECTIVE – renard brun rapide
4. ADJECTIVE ADJECTIVE VERB – brun rapide saute
5. PREPOSITION ARTICLE NOUN– par-dessus le chien
6. ARTICLE NOUN ADJECTIVE – le chien paresseux
7. NOUN ADJECTIVE PERIOD – chien paresseux.

Of these French trigrams, many of these would not be possible in language where
the adjective is solely a premodifier:

1. Possible.
2. Not possible because the adjective cannot follow the noun
3. Not possible for the same reason
4. Not possible because a noun should intervene before the verb
5. Possible.
6. Not possible because the adjective cannot follow the noun
7. Not possible for the same reason

Note that French adjectives are not strictly postmodifiers. Consider these
counterexamples:

- un grand homme – a great man
- un petit homme – a little man
- un bon appétit – a good appetite
And examples of this sort can bring many trigrams of tags into the contextual model, even though at a reduced probability. Still, we see from this discussion that this simple difference of premodifier vs. postmodifier thus can have a profound effect on the contextual model.

We will note here that this example sentence of “The quick brown fox…” is merely illustrative, but it reflects actual adjective-noun order as it appears in the Europarl corpus. For example, consider the following Europarl sentence. The German and English do not produce every trigram described above, but do produce some:

**German**: Hier spielt auch die Frage der humanitären Organisationen hinein, deren Tätigkeit allem Anschein nach häufig durch intolerantes Verhalten der Kriegführenden beeinträchtigt wird, die zynisch versuchen, Zeit zu gewinnen, um ihre Siege zu zementieren oder um Vergeltungsaktionen gegen die gefährdete Bevölkerung zu begünstigen.

**English**: The question of humanitarian organisations comes in here too, as their scope for action is frequently affected by the intolerable behaviour of certain parties to conflicts who are cynically trying to win time to seal their victories or to take retaliatory action affecting populations at risk.

**French**: La question des organisations humanitaires entre ici en jeu, leur action semble fréquemment souffrir de comportements intolérables de la part des belligérants, qui cherchent à gagner du temps de manière cynique pour asseoir leurs victoires ou promouvoir des actions de représailles contre les populations à risque.

The following figure demonstrates adjectives act as premodifiers in the English and German corpora:
The question of humanitarian organisations … frequently affected by the intolerable behaviour …

… die Frage der humanitären Organisationen … nach häufig durch intolerantes Verhalten …

Figure 19 English and German premodifiers in the actual corpus

Meanwhile, the following figure demonstrates how French adjectives act as postmodifiers, and the mismatch in the contextual model that results:

The question of humanitarian organisations … frequently affected by the intolerable behaviour …

La question des organisations humanitaires … fréquemment souffrir de comportements intolérables …

Figure 20 French postmodifiers in the actual corpus

Prediction \[P4a, b\]

Therefore, we predict that language pairs which share noun-adjective order will benefit more from wholesale adoption of the HD contextual model, and will thus have greater accuracy than language pairs which don’t.

More concretely, with the two language pairs selected above:

A. We predict that H1 on (English, German) would be more accurate than H1 on (French, German).

B. We predict that D1 on (English, German) would be more accurate than D1 on (French, German).
5.4 The Presence of Definite and Indefinite Articles

This is another difference between languages - whether the languages use a definite and indefinite article for nouns. This feature will have an impact primarily on the contextual model since, for instance, nouns and adjectives might be predicted by the presence of such an article. In language pair (a, b), there are three configurations:

1. both a and b use articles
2. neither a and b use articles
3. one language in the pair uses articles

Considering case (1), there should be a closer match in the contextual models of such a language pair than case (3). Considering case (2), absence of the article might make for a close match, depending on other linguistic features (e.g., will a noun often follow an adjective?), and so it might make for a better match than in (3).

While the impact might be primarily on the contextual model, there will also an impact within the lexical model. The definite and indefinite articles are, after all, lexical items. And articles are frequent and fairly unambiguous. Simple language-ifications (namely L1, for most frequent words, or L4, if they are related languages and therefore likely have articles which are cognates) should vastly improve the lexical model of the target language, for those frequent articles.

The table that follows considers each of the languages in the Europarl corpus, in terms of presence, absence, and granularity of articles.
Table 3: Presence and Granularity of Articles in Europarl languages

<table>
<thead>
<tr>
<th>Europarl Language</th>
<th>Definite and Indefinite Articles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulgarian</td>
<td>no indefinite article. a suffix –ът/-ът at the end of nouns for an indefinite article(^{12}).</td>
</tr>
<tr>
<td>Czech</td>
<td>no definite or indefinite articles</td>
</tr>
<tr>
<td>Danish</td>
<td>indefinite, prepositive as separate word, definite article, postpositive -en suffix, three separate forms, for common, neuter, plural</td>
</tr>
<tr>
<td>Dutch</td>
<td>definite and indefinite articles. total #: 3. de, het, een</td>
</tr>
<tr>
<td>German</td>
<td>prepositive as separate word for both definite and indefinite articles. A lot of different ones, based on case.</td>
</tr>
<tr>
<td>Greek</td>
<td>definite and indefinite articles, marked for case, number, gender</td>
</tr>
<tr>
<td>English</td>
<td>definite, indefinite articles: total of 2</td>
</tr>
<tr>
<td>Spanish</td>
<td>definite, indefinite article, marked for number and gender, for a total of 8</td>
</tr>
<tr>
<td>Estonian</td>
<td>None. Though the demonstrative is making inroads into becoming the definite article in Estonian(^{13}).</td>
</tr>
<tr>
<td>Modern Finnish</td>
<td>None</td>
</tr>
<tr>
<td>French</td>
<td>marked for number and gender, definite, indefinite</td>
</tr>
<tr>
<td>Hungarian</td>
<td>definite and indefinite. but not marked for anything</td>
</tr>
<tr>
<td>Italian</td>
<td>definite, marked for gender and number; indefinite, marked for gender</td>
</tr>
<tr>
<td>+Lithuanian</td>
<td>None</td>
</tr>
<tr>
<td>Latvian</td>
<td>None</td>
</tr>
<tr>
<td>Polish</td>
<td>None</td>
</tr>
<tr>
<td>Portuguese</td>
<td>definite and indefinite, marked for gender and number</td>
</tr>
<tr>
<td>Romanian</td>
<td>definite and indefinite, marked for gender, number, case. the definite is attached to the end of the word as an enclitic</td>
</tr>
<tr>
<td>Slovak</td>
<td>None</td>
</tr>
<tr>
<td>Slovene</td>
<td>None</td>
</tr>
<tr>
<td>Swedish</td>
<td>an indefinite article for singular, as separate word preceding the noun. definite article as a suffix, marked for number and gender.</td>
</tr>
</tbody>
</table>

---

\(^{12}\) This suffix would impact the identification of nouns. On the lexical level, this might interfere with identification of cognates. Plus, it increases the sparsity of the data. On the contextual level, it is difficult to exploit, since the granularity of the tagset does not include definiteness as a feature of nouns. Perhaps a plausible language-ification would strip off these definite articles for parallel languages which lack the definite article, or would place a dummy definite article immediately before for languages which possess it. (This is not one of the language-ifications proposed, though.)

\(^{13}\) See [http://ee-translations.com/Documents/cafslFIN.pdf](http://ee-translations.com/Documents/cafslFIN.pdf)
We would like to select representative language pairs for each of the three configurations discussed above, based on presence or absence of definite and articles. The three language pairs we choose are:

1. (Dutch, German), in which both languages use definite and indefinite articles\(^{14}\)
2. (Estonian, Finnish), in which neither language uses articles.
3. (German, Finnish) in which only the first language uses articles.

Most of the languages considered in the prior section on constituent word order, are of type (1).

**Prediction \([\text{P5a, b, c, d}]\)**

We predict that language pairs in which both languages have articles would benefit more from gloss replacement of frequent words than would languages pairs in which only one language, or both languages, lack articles.

Given these language pair choices, I would therefore expect the following:

A. We predict that \(\Delta L_1\) LD-ify on (Dutch, German) > \(\Delta L_1\) LD-ify on (German, Finnish)\(^{15}\).

B. We predict that \(\Delta L_1\) LD-ify on (Dutch, German) > \(\Delta L_1\) LD-ify on (Estonian, Finnish), simply because there are more of these fairly frequent words which are now being accurately tagged for (Dutch, German).

---

\(^{14}\) There is a difference in granularity of these articles, but given the low granularity of our tagset, this should not be a concern.

\(^{15}\) Note that since Finnish is the target language and lacks regular articles, the problem discussed below in section 5.5, of many LD articles being out-of-vocabulary, is not a confounding issue here.
C. We predict that delta L1 HD-ify on (Dutch, German) > delta L1 HD-ify on (German, Finnish).

D. We predict that delta L1 HD-ify on (Dutch, German) > delta L1 LD-ify on (Estonian, Finnish),

5.5 The Granularity of Definite and Indefinite Articles

While the previous section considered the mere presence or absence of articles in a language, this section considers the granularity of those articles. In some languages, articles may be marked for case, gender, and number. Markings for case (in articles and nouns), in particular, would aid greatly within the contextual model. For instance, the accusative case would indicate an impending accusative noun, as well as a likely end of sentence. If both languages in a language pair make use of such case markings and transfer a more precise contextual model. If one does while the other does not, then it depends on the directionality of transfer. If the target language does not possess differentiation for case, then we can strip out such information. If the target language does possess differentiation for case, but the source language does not, it is non-trivial to reconstruct this information.

We inspected the tags assigned by TreeTagger to languages with high lexical granularity of articles (e.g. German), and noted that unfortunately the tag granularity does not match the lexical granularity (they are simply marked as DET or ARTICLE), such that such an investigation does not seem possible.

We would, however, expect that this difference in lexical granularity could have a significant impact on the lexical model, particularly in the LDification approach. Consider the following instructive example of article mapping from a language with high lexical granularity for articles (German) to one with low lexical granularity for articles (English).
Figure 21: Projecting articles from a high granularity language to a low granularity language

If employing an HD-ification strategy, say we take a German input sentence, replace all instances of German *dem*, *der* and *den* with English *the*, and then tag with an HD English tagger. This strategy will succeed, because *the* is in-vocabulary in the HD lexical model, whereas *dem*, *der* and *den* were not.

However, if employing an LD-ification strategy, then a different mapping is in play. Recall that the LDification approach is to take the HD English tagged corpus, conduct gloss-replacements of certain English words for their German equivalents, train a tagger on that modified corpus, and then use that corpus on German input sentences. But consider the ambiguity of the mapping:
Here, the English word *the* could theoretically map to German *dem, der* or *den*. And we can only choose a single German determiner to replace the English determiner. If we select *der*, then the word *der* will be in-vocabulary in the newly trained tagger, but *dem* and *den* will be out-of-vocabulary. (A more complex replacement model could anticipate this problem and solve it by alternating through the mappings, replacing *the* with *dem, der*, and *den*. Or it could duplicate the sentence in the corpus, in order to give each determiner a chance. However, in the simple one-to-one mapping assumed for our language-ifications, this difference in lexical granularity of determiners will present a problem.)

Furthermore, consulting the same table of languages above and selecting languages for their lexical granularity of articles, I would choose the following three languages pairs:

1. (English, German), in which the HD language (English) has a low lexical granularity of articles and the HD language (German) has a high lexical granularity of articles.

2. (German, Dutch), in which the HD language (German) has a high lexical granularity of articles and the LD language (Dutch) has a low lexical granularity of articles.
**Prediction [P6a, b, c]**

Thus, we predict that gloss replacement of frequent words, in an LD-ification strategy, would have greater success if the HD language of the language pair has a low lexical granularity of articles and the LD language has a high lexical granularity of articles. Furthermore, we predict that this difference in success of gloss-replacement of frequent words is limited to this specific scenario. The gloss replacement of frequent words, in an HD-ification strategy, generally performs quite well, and should perform quite well in this instance as well. Further, if the article granularities of the HD and LD languages were reversed, with the HD language having the higher lexical granularity of articles, then HD-ify and LD-ify strategies should be comparable.

Thus, we would make the following concrete predictions:

A. We predict that \( \delta L1 \text{ LD-ify on (German, Dutch)} > \delta L1 \text{ LDify on (English, German)} \).

B. We predict that \( \delta L1 \text{ LD-ify on (German, Dutch)} \) would be more or less comparable to \( \delta L1 \text{ HD-ify on (German, Dutch)} \) over its HD baseline.

C. We predict that \( L1 \text{ LDify on (English, German)} \) would compare unfavorably to \( L1 \text{ HDify on (English, German)} \).

### 5.6 PRO-Dropness

Pro-dropness (also known as the NULL subject phenomenon), is the tendency to omit pronouns. There are four criteria of a PRO-drop language, as discussed by Geeslin (1999):
1. Overt subjects (where they are pronouns) are optional
2. Expletives (such as “it” and “there”) do not exist
3. Wh-movement does not leave a that-trace
4. In many such languages, there is subject-verb inversion

This will have an obvious impact on the **contextual** model of a language, but also on the **lexical** model, and well as efforts to import a model from one language to another. In terms of the contextual model, to take one example, if overt pronouns are optional, then this could impact probabilities such as $p(\text{VERB} \mid \text{PRONOUN})$. And if expletives do not exist, then we will not see expletives represented in the contextual model. Indeed, all these criteria result in different surface words or word orders, which will affect the contextual model. In terms of the lexical model, certain lexical items will not exist. Where the lexical item does exist, the decreased incidence of certain pronouns will change $p(\text{word} \mid \text{tag})$. Therefore, transferring a contextual or lexical model from a PRO-drop language to a non-PRO-drop language, or vice versa, may not be as successful as transferring between two languages which are identical in their PRO-dropness or lack thereof.

One could conceivably design a language-ifier (an LD-ifier) to specifically target this divergence. For instance, inserting or removing expletives, inverting subjects and verbs, or inserting or removing subject pronouns in some HD POS-tagged corpus, prior to training, based on syntactic analysis and grammatical rules determining where they belong. This would require too much linguistic knowledge, however. We are trying to keep our language-ifiers as simple as possible.

Given the simpler language-ification techniques, we would expect that language pairs which are both pro-drop or are both not pro-drop would attain greater accuracy than a
language pair which differed in its pro-dropness. Therefore, to test this hypothesis, we should select a language pair with the following features, where the pair is (HD source, LD destination):

1. (PRO-drop, PRO-drop); or (not PRO-drop, not PRO-drop)
2. (not PRO-drop, PRO-drop); or (PRO-drop, not PRO-drop)

The following table describes the PRO-dropness of languages in Europarl. The same table describes whether a language is inflected and to what degree. Inflection is actually related to PRO-dropness, as PRO-drop languages are typically highly inflected, while non-PRO-drop languages are not.

Table 4: PRO-drop and Inflection and Europarl languages

<table>
<thead>
<tr>
<th>Language</th>
<th>Pro-drop?</th>
<th>Inflection</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulgarian</td>
<td>Yes</td>
<td>Yes. Verbs for person, number, gender, tense, mood, voice</td>
</tr>
<tr>
<td>Czech</td>
<td>Yes</td>
<td>Yes, heavily. Verbs for number, gender, voice, tense,</td>
</tr>
<tr>
<td>Danish</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>German</td>
<td>No</td>
<td>Moderately</td>
</tr>
<tr>
<td>Greek</td>
<td>Yes</td>
<td>Fully inflected</td>
</tr>
<tr>
<td>English</td>
<td>No</td>
<td>Just for number and tense</td>
</tr>
<tr>
<td>Spanish</td>
<td>Yes</td>
<td>number and gender of noun</td>
</tr>
<tr>
<td>Estonian</td>
<td>explicit in written, dropped in spoken</td>
<td>Yes</td>
</tr>
<tr>
<td>Finnish</td>
<td>only explicit when need to be inflected</td>
<td>Yes</td>
</tr>
<tr>
<td>French</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Hungarian</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Italian</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Lithuanian</td>
<td>Yes</td>
<td>Yes (highly)</td>
</tr>
<tr>
<td>Latvian</td>
<td>Yes</td>
<td>Yes (moderately)</td>
</tr>
<tr>
<td>Dutch</td>
<td>No</td>
<td>Poorly inflected</td>
</tr>
<tr>
<td>Polish</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Portuguese</td>
<td>Yes</td>
<td>Yes (highly)</td>
</tr>
<tr>
<td>Romanian</td>
<td>Yes</td>
<td>Yes (definiteness, number, gender, case)</td>
</tr>
<tr>
<td>Slovak</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Slovene</td>
<td>Yes</td>
<td>Yes (nouns for case and number)</td>
</tr>
<tr>
<td>Swedish</td>
<td>No</td>
<td>Weak. Lost most of it</td>
</tr>
</tbody>
</table>

On the basis of this table, we select the following languages pairs:
1. (Finnish, German) as (PRO-drop, non-PRO-drop)
2. (English, German) as (non-PRO-drop, non-PRO-drop)

**Prediction [P7]**

We predict that approaches involving wholesale adoption of the HD contextual model (H1) will be most effective for pairs of languages in which both are PRO-drop or both are not PRO-drop, but that if there is a divergence in this language feature, then such an approach will not be as successful.

More concretely, with the language pairs selected, we predict that H1 on (English, German) would have greater accuracy than H1 on (Finnish, German).

**5.7 Richness of Inflection**

Inflection is the modification of a word to reflect different grammatical categories, and can includes conjugation of verbs and declension of nouns, adjectives, and pronouns. Words can be marked for gender, number, case, mood, voice, and so on. Whether a language is weakly or strongly inflected has repercussions for its contextual and lexical models. In terms of the **contextual** model, if the granularity of the POS tag set is rich, then these would be represented in the model. And, a noun or pronoun with nominative case could indicate the beginning of a sentence (which might be followed by a verb) while a noun or pronoun with accusative case could appear at the end of a sentence (which might be followed by a period). The granularity of the part of speech tags are not that rich. In terms of the **lexical** model, greater inflection means more lexical items, and therefore increased scarcity of the data.

In terms of transferring contextual models or lexical models from a source to a target language, divergences in levels of inflection can cause problems. This were previously
discussed in section 10.3, in relation to the lexical granularity of articles, but it now applies to other word classes such as nouns and adjectives as well.

In terms of the contextual model, our shared simplified tagset does not distinguish between different cases of nouns and adjective. That is, German is a richly inflected language but the tags assigned by TreeTagger did not account for these different cases. We trained the Finnish TreeTagger and did distinguish in the tagset between different cases. But because the German tagset and other tagsets to not make such a distinction, the shared simplified tagset does not make such a distinction. Since H1 makes use of only the shared simplified tagset, we cannot simply use H1 to assess the impact of richer or poorer inflected tagsets. Therefore we will unfortunately have to leave such an investigation for future work.

However, we can still focus on the lexical granularity that such inflection entails and consider the impact this would have on the lexical model and language-ification strategies. One can imagine at least four different cases involving language-ification strategies and degree of inflection:
Figure 23: Four cases of language-ification strategies and degrees of inflection

In case (A), the HD language (and corpus) is richly inflected, and LD-ification will have the effect of removing inflection information.

In case (B), the HD language (and corpus) is poorly inflected, and LD-ification will have the effect of adding (arbitrarily chosen) inflection information.

Cases (C) and (D) are same configurations as (A) and (B) but with the arrows reversed; we are HD-ifying LD text prior to tagging it with the HD tagger.

In case (C), the HD language is richly inflected, and HD-ification of the LD text will have the effect of adding (arbitrarily chosen) inflection information.
In case (D), the HD language is poorly inflected, and so HD-ification of the LD text will have the effect of removing inflection information.

We shall focus first on LD-ification, that is, cases (A) and (B). We would expect that case (A) would be a more successful language-ification strategy than case (B).

For instance, if nouns in the target LD language are inflected for case, number, and gender, while nouns in the source language are not, then how will a LD-ification proceed? If LD-ifying a tagged HD corpus, we cannot know which target case, etc., of the richly inflected LD language to use. Should we use the plural feminine accusative case, or the singular masculine nominative case? We cannot easily determine that from the poorly inflected HD corpus, and we are only making use of simple language-ification strategies.

Here we consider a practical example from the Europarl corpus. The English word “new” is the 76th most frequent word in our English corpus, and is mapped to German “neu”. However, due to the rich inflection of German, and depending on whether this adjective modifies a masculine, feminine, neutral, or plural noun, as well as whether the adjective is in the nominative, genetive, dative, or accusative case, there are actually several German words meaning “new”. Examples from the German corpus include:

vom Juni 1999 **neu** geregelt

zur Schaffung **neuer** Arbeitsplätze

des **neuen** Jahres das Angebot

als die **neue** Kommission im Amt war

eines Jahrhunderts zu **neuen** Leben

ein gutes **neues** Jahr und Millennium
Some of these lexical items are overloaded – for instance, “neuer” is the masculine nominative, the feminine genitive and dative, and the plural genitive. Still, when transforming the tagged English corpus into a tagged German corpus, only one of these will be selected, namely “neu”. The remaining German words will remain out-of-vocabulary. This multiple mapping may be illustrated by the following figure:

Figure 24: Projection of a poorly inflected language to a richly inflected language

As one may readily observe, the words replaced in an L1 LD-ification strategy would be an arbitrary choice among candidates, and all the other candidates would remain out of vocabulary.

Figure 25: Projection of a richly inflected language to a poorly inflected language
However, in the opposite direction, if the HD language is richly inflected while the LD language is not, then LD-ifying the HD corpus simply involves stripping off the case. There will be fewer LD words than HD words, and so all LD words would be in vocabulary:

We therefore predict that L1 LD-ify on a language pair which is (richly inflected, poorly inflected) would have a greater increase over its baseline than a language pair which is (poorly inflected, richly inflected).

Another LD-ification strategy is L2 LD-ify. Here, we take a tagged HD corpus, perform gloss replacement of the HD affixes with their LD equivalents, and train on the resulting LD-ified corpus. This replacement is done automatically, with no attention paid to whether the resulting word is actually an LD word. In some cases, we might arrive at an actual LD word, and bring the entire LD word into vocabulary. But aside from this, since TreeTagger builds affix trees, this is bringing the affixes and associated tags into vocabulary.

Once again, we predict that affix replacement which are many-to-one (many HD affixes become a single LD affix) would perform better over its baseline than affix replacement which is one-to-many (one HD affix which could ambiguously be assigned to several different LD affixes, but of which we can only select one).

Another LD-ification strategy is L4 LD-ify. Here, we take a tagged HD corpus, perform gloss replacement of the recognized HD cognates with their LD equivalents, and train on the resulting LD-ified corpus. This cognate detection is done automatically, with no attention paid to whether the words are actually cognates of one another, such that we might be introducing noise. Once again, if the LD language is richly inflected language while the HD language is poorly inflected, then there might be multiple candidate cognate words, while one of those candidates can be selected. Indeed, these cognate candidates might well
be the words with various case endings, where only one can be selected. Meanwhile, in the opposite direction, for any HD word, the possible valid LD replacements are more limited.

Once again, we predict that cognate replacements that are many-to-one would perform better over its baseline than cognate replacement that is one-to-many.

To test this out, we consult the preceding table, and we select the following two language pairs, which are reflections of one another:

1. (German, Dutch) as (richly inflected, poorly inflected)
2. (Dutch, German) as (poorly inflected, richly inflected)

**Prediction [P8a, b, c]**

With these two language pairs selected, we can test out the prediction regarding the LD-ification cases. Namely, that LD-ification of a richly inflected language towards a poorly inflected language will be more successful than LD-ification of a poorly inflected language towards a richly inflected language.

We make the following concrete predictions:

A. We predict that \( \Delta L_1 \) LD-ify for (German, Dutch) > \( \Delta L_1 \) LD-ify on (Dutch, German) over its baseline.

B. We predict that \( \Delta L_2 \) LD-ify for (German, Dutch) > \( \Delta L_2 \) LD-ify on (Dutch, German).

C. We predict that \( \Delta L_4 \) LD-ify for (German, Dutch) > \( \Delta L_4 \) LD-ify on (Dutch, German).
Figure 26: Once again, four cases of language-ification strategies and degrees of inflection

Turning now towards the HD-ification strategy, namely cases (C) and (D), in which we reverse the direction of the arrows from cases (A) and (B), we would expect a reversal of the relative success of language pairs as well.

That is, while for LD-ification, (A) > (B), for HD-ification, (C) < (D). Put yet another way, in general, language-ification in the direction of the poorly inflected language should be more successful than language-ification in the direction of the richly inflected language.

The issue of concern for LD-ification was words remaining out-of-vocabulary. That is not an issue for HD-ification, since any of the (arbitrarily chosen) gloss replacements can,
with equal likelihood, be in vocabulary. Rather, the issue of concern for HD-ification is noise.

That is, if the HD language is richly inflected while the LD language is poorly inflected, then there is more opportunity for noise. L2, L3, and L4 gloss replacements are performed automatically, and the noisier the target space, the less likely we are to arrive at the correct replacement. And since we can only select one word here as the replacement, which the 3rd party HD tagger then simply treats using its lexical model, a wrong choice can be disastrous for that particular word, and then the surrounding context. (This is not relevant for L1, in which the gloss replacements of frequent words were generated by a human being. Those words will likely be in-vocabulary, and will be in the correct part of speech. The out-of-vocabulary issue is relevant only to LD-ification.) Meanwhile, in the direction of the poorly inflected language, the same inflection (or no inflection) may be shared across different parts of speech, so a wrong choice is both less likely and less disastrous.

For HD-ification, we will select the same two language pairs as we did above for LD-ification:

1. (German, Dutch) as (richly inflected, poorly inflected)
2. (Dutch, German) as (poorly inflected, richly inflected)

**Prediction [P9a, b, c]**

With these two language pairs selected, we can test out the prediction regarding the HD-ification cases. Namely, that HD-ification of a richly inflected language towards a poorly inflected language will be more successful than HD-ification of a poorly inflected language towards a richly inflected language.
More concretely, we make the following three predictions about HD-ification between languages differing in their level of inflection:

A. We predict that delta L2 HD-ify for (Dutch, German) > delta L2 HD-ify on (German, Dutch).

B. We predict that delta L3 HD-ify for (Dutch, German) > delta would improve over its baseline more than L3 HD-ify on (German, Dutch) over its baseline.

C. We predict that L4 HD-ify (Dutch, German) would improve over its baseline more than L4 HD-ify on for (German, Dutch) over its baseline.

5.8 Language Pair Selections

We have thus considered linguistic features of the candidate languages available in the Europarl corpus. We have considered, and predicted, how these absence, presence, or disparity of these features in a language pair might impact the success of a given strategy. Finally, we have selected languages and language pairs in such manner that we can test our predictions.

The following are our language pair choices, together with the sections in which we discuss why we selected the particular language pair. Note that a language pair might well satisfy criteria listed in other sections, but we did not present it as an exemplar of that language pair:

1. (Dutch, German) – 5.1 [P1a, P2a, c], 5.4 [P5a-d], 5.7 [P8a-c; P9a-c]
2. (German, Dutch) – 5.1 [P2b, d], 5.2 [P3a, c], 5.5 [P6a, b], 5.7 [P8a-c, P9a-c]
3. (Finnish, German) – 5.1 [P2a-d], 5.6 [P7]
4. (German, Finnish) – 5.2 [P3a, b, d], 5.4 [P5a, c]
5. (Estonian, Finnish) – 5.2 [P3b], 5.4 [P5b, d]
6. (French, German) – 5.1 [P1b], 5.3 [P4a, b]

7. (English, German) – 5.1 [P1a, b], 5.2 [P3c, d], 5.3 [P4a, b], 5.5 [P6a, c], 5.6 [P7]
Chapter 6: Results Discussion

The consideration in the previous chapter was of the linguistic features of the languages in Europarl and how they might affect the success of various projection and tagging strategies. This consideration led us to select representative language pairs for which we could make concrete predictions and, with specific language pairs chosen, to make concrete predictions about the relative success of language projection strategies. For example, since adjectives are premodifiers of nouns in English and German while adjectives are mostly postmodifiers of nouns in French, we predict that H1 of (English, German) > H1 (French, German).

Figure 27: In each case, the arrow points to the LD language and the arrow length indicates linguistic distance (roughly approximated by this author) between the languages

To enable the testing of our various linguistically driven predictions, we selected the following seven language pairs. In each instance, the first language in the language pair is the HD and the second is the LD.
1. (Dutch, German)
2. (German, Dutch)
3. (Finnish, German)
4. (German, Finnish)
5. (Estonian, Finnish)
6. (French, German)
7. (English, German)

These particular language pair choices, when considered as a whole, have some useful features:

i) German is the LD target of several HD languages as differing distances.

ii) (German, Dutch) and (Dutch, German) form a case of closely related languages where each language can function as the HD and the LD in an LD-ification and HD-ification strategy.

iii) (German, Finnish) and (Finnish, German) form a case of unrelated languages where each language can function as the HD and the LD in an LD-ification and HD-ification strategy.

iv) Finnish can serve as an LD target for both a close language, Estonian, and for a distant language, German.

With those languages and language pairs selected, we then set out to the task of assembling the linguistic resources required to comprehensively test every possible combination of language pair and tagging strategy. Even if, e.g., we wouldn’t expect any interesting result from applying L2 HDify to (German, Finnish), we still assembled the
necessary resource, by taking an automatically generated list of Finnish affixes and analyzing our Finnish → German frequent word dictionary to see if we could spot regular and unambiguous affix substitutions.

### 6.1 Results Tables

We implemented each projection and tagging strategy in Python, building upon the Natural Language Toolkit, which is also implemented in Python. Then, we ran experiments for each combination of strategy and language pair. The results of these extensive experiments are in the table below.

Table 5 details the results of traditional approaches run on all of our selected language pairs:

<table>
<thead>
<tr>
<th>APPROACHES TABLE: RESULTS (as % correct)</th>
<th>Dutch/ German</th>
<th>French/ German</th>
<th>English/ German</th>
<th>Finnish/ German</th>
<th>German/ Dutch</th>
<th>German/ Finnish</th>
<th>Estonian/ Finnish</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Traditional Approaches</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>D1 -- Apply 3rd party HD tagger to LD text</td>
<td>43.5%</td>
<td>35.1%</td>
<td>36.6%</td>
<td>40.6%</td>
<td>35.5%</td>
<td>51.8%</td>
<td>59.5%</td>
</tr>
<tr>
<td>D2 -- Upper Baseline, apply 3rd party LD tagger to LD text</td>
<td></td>
<td>96.6%</td>
<td></td>
<td></td>
<td>96.2%</td>
<td></td>
<td>91.2%</td>
</tr>
<tr>
<td>D3 -- Unsupervised training on LD</td>
<td>24.3%</td>
<td>24.3%</td>
<td>24.3%</td>
<td>24.3%</td>
<td>25.2%</td>
<td>43.6%</td>
<td>42.7%</td>
</tr>
<tr>
<td>D4 -- Unsupervised training on LD</td>
<td>22.1%</td>
<td>20.8%</td>
<td>20.8%</td>
<td>20.7%</td>
<td>21.7%</td>
<td>41.7%</td>
<td>42.4%</td>
</tr>
<tr>
<td>D5 -- Same as D4, but filter for</td>
<td>21.9%</td>
<td>21.7%</td>
<td>21.7%</td>
<td>21.2%</td>
<td>21.7%</td>
<td>42.0%</td>
<td>42.0%</td>
</tr>
<tr>
<td>D6 -- Same as D4, but incorporate affix</td>
<td>22.1%</td>
<td>20.7%</td>
<td>20.8%</td>
<td>20.7%</td>
<td>22.2%</td>
<td>44.3%</td>
<td>44.3%</td>
</tr>
<tr>
<td>H1 -- using LD lexicon, uniform lexical</td>
<td>82.1%</td>
<td>77.6%</td>
<td>79.3%</td>
<td>77.9%</td>
<td>84.9%</td>
<td>62.2%</td>
<td>83.8%</td>
</tr>
<tr>
<td>H2 -- same as H1, but for cognates, average with HD’s lexical model</td>
<td>80.5%</td>
<td>77.3%</td>
<td>74.2%</td>
<td>57.3%</td>
<td>82.1%</td>
<td>44.2%</td>
<td>70.9%</td>
</tr>
</tbody>
</table>

Table 5 Results, traditional approaches

To quickly recap, for each of these columns, the first listed language is the HD and the second listed language is the LD. Thus, for the Dutch/ German column, the HD is Dutch and the LD is German. The rows D1 through D6 represent the techniques described in Duh and Kirchoff (2005).
1. D1, simple application of an HD tagger to an LD text, is also the baseline for our language-ification strategies.

2. D2 is supervised training on an LD tagged corpus, and forms an upper baseline for supervised approaches. (Some cells in row D3 are merged because it is only the LD language that matters, with the HD not entering the picture at all.)

3. D3 is unsupervised training on an LD corpus, and forms an upper baseline for unsupervised approaches. In order to attempt to bootstrap this unsupervised learning, we began with the contextual model of the HD language and an LD lexicon.

4. D4 is unsupervised training on an LD corpus, using noisy lexicon matching lexical items from the LD with tags assigned by an HD analyzer.

5. D5 is an attempt to reduce the noise of D4 based on distributional criteria, prior to unsupervised training.

6. D6 is also unsupervised training, but incorporates an affix model.

In general, the unsupervised models did not converge, perhaps because of the noise of initial tagging, an unsatisfactory initial starting configuration, or because the connection between languages in the language pair is not the same as between two dialects or Arabic. That is, unsupervised training via EM does not find good HMM POS-taggers. See Ali (2008) and Johnson (2007). But where Duh and Kirchoff (2005) succeeded, it was in two dialects of Arabic, such that there were far fewer unanalyzable or out-of-vocabulary words than when dealing which two distinct languages.

The rows for H1 and H2 represent the techniques described in Hana et al (2004) and (2006), which are both supervised approaches. H1 is the wholesale adoption of the HD contextual model paired with a lexical model formed based on an LD dictionary, with equal
weight assigned to each possible tag assigned to a word in the dictionary. H2 is the same, except that for recognized cognates, this weight is averaged with the weights assigned in the lexical model of the HD. Because our cognate detection was performed automatically rather than manually, rather than introducing clarity into the weights in the LD model (to say, e.g. that word X is almost always a determiner and only very occasionally a pronoun), this instead introduced noise due to false cognates (saying e.g., that word X is often an adjective, even though in reality it is an adverb). This then pulled down the overall tagging accuracy.

Table 6 details the results of HD-ification approaches run on all of our selected language pairs.

<table>
<thead>
<tr>
<th>HD-ification Approaches</th>
<th>(Dutch, German)</th>
<th>(French, German)</th>
<th>(English, German)</th>
<th>(Finnish, German)</th>
<th>(German, Dutch)</th>
<th>(German, Finnish)</th>
<th>(Estonian, Finnish)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 3rd party tagger</td>
<td>43.5%</td>
<td>35.1%</td>
<td>36.6%</td>
<td>40.6%</td>
<td>35.5%</td>
<td>51.8%</td>
<td>59.5%</td>
</tr>
<tr>
<td>L1 – Hdfy – frequent word replacement</td>
<td>25.8%</td>
<td>32.6%</td>
<td>29.6%</td>
<td>12.3%</td>
<td>35.1%</td>
<td>7.7%</td>
<td>6.8%</td>
</tr>
<tr>
<td>L2 – Hdfy – affix replacement</td>
<td>1.2%</td>
<td>0.8%</td>
<td>0.5%</td>
<td>0.0%</td>
<td>0.8%</td>
<td>-7.6%</td>
<td>1.1%</td>
</tr>
<tr>
<td>L3 – Hdfy – exemplar replacement</td>
<td>3.1%</td>
<td>1.6%</td>
<td>-8.8%</td>
<td>0.7%</td>
<td>1.0%</td>
<td>-0.2%</td>
<td>1.8%</td>
</tr>
<tr>
<td>L4 – Hdfy</td>
<td>26.4%</td>
<td>-1.9%</td>
<td>-2.0%</td>
<td>-1.9%</td>
<td>6.8%</td>
<td>0.1%</td>
<td>-4.2%</td>
</tr>
</tbody>
</table>

Table 6 HD-ification results

The top row, labeled Baseline 3rd party tagger, is a repetition of the D1 row from Table 5, and represents simple application of the 3rd party HD tagger to the LD language. The remaining rows show the delta – the change from that Baseline – due to application of a gloss replacement technique.
1. In general, for the HD-ify case, L1 (replacement of most frequent LD words based on a manually-constructed dictionary) worked the best at bringing LD words into vocabulary for the HD tagger to tag them.

2. L2 (specifically for words unrecognized by the HD tagger, replacement of longest recognized LD prefix and suffix for HD equivalent, specifically where the resulting word was a known HD word), comparatively had limited impact.

3. L3 (making use of the same prefix and suffix dictionaries, recognizing longest prefix and suffix, projecting to the HD prefix and suffix, finding the most common tag for that combination, and selecting an exemplar – an HD word unambiguously tagged with this majority tag) also had somewhat limited impact, but often performed slightly better than LD.

4. L4 (replacement of automatically recognized cognates) generally performed poorly for distant languages, for which there were fewer true cognates and more noise in the replacement list. However, for Dutch and German, this strategy performed nicely.
Table 7 details the results of LD-ification approaches run on all of our selected language pairs.

<table>
<thead>
<tr>
<th>LD-ification Approaches</th>
<th>Dutch/ German</th>
<th>French/ German</th>
<th>English/ German</th>
<th>Finnish/ German</th>
<th>German/ Dutch</th>
<th>German/ Finnish</th>
<th>Estonian/ Finnish</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 3rd party tagger</td>
<td>50.0%</td>
<td>34.9%</td>
<td>36.4%</td>
<td>33.3%</td>
<td>33.7%</td>
<td>42.2%</td>
<td>60.7%</td>
</tr>
<tr>
<td>L1 LDify</td>
<td>14.3%</td>
<td>0.4%</td>
<td>-1.1%</td>
<td>-3.5%</td>
<td>32.3%</td>
<td>8.1%</td>
<td>3.2%</td>
</tr>
<tr>
<td>L2 – Ldify – affix replacement</td>
<td>-20.30%</td>
<td>-10.80%</td>
<td>-11.70%</td>
<td>-7.80%</td>
<td>9.80%</td>
<td>-1.20%</td>
<td>5.10%</td>
</tr>
<tr>
<td>L4 – Ldify – cognate replacement</td>
<td>-7.50%</td>
<td>-4.4%</td>
<td>-4.9%</td>
<td>-0.1%</td>
<td>4.1%</td>
<td>3.7%</td>
<td>-2.2%</td>
</tr>
</tbody>
</table>

Table 7: LD-ification results

The Baseline 3rd party tagger in this LD-ify table is different than the Baseline 3rd party table for the HD-ify table. Since in the LD-ify case, we are training on a much smaller corpus (rather than the original, inaccessible corpus initially used to construct the HD tagger) and using the simplified shared tagset, the LD baseline is also trained on that smaller corpus. The remaining rows show the delta – the change from that Baseline – due to application of a gloss replacement technique.

1. L1 (gloss replacement of frequent words), where it is successful, is here the clear winner, though in some places the technique fails.

2. L2 (replacement of longest HD prefix and suffix for the LD equivalent, regardless of whether the resulting word is in vocabulary), often fails. However, unlike in the HD case, here it occasionally succeeds.

3. L4 (cognate replacement) is not as successful in the LD-ify case as in the HD-ify case.
6.2 Revisiting the Predictions

That was a quick introduction to the rows and columns of the results tables. However, in the previous chapter we had made general and concrete predictions, driven by linguistic features of the language pair. With these results tables in place, we can now reexamine the predictions we made in the previous chapter and discover if and how they were borne out.

At the end of each reexamination of our predictions, we will consider our results can serve as a guideline, when approaching a completely new language pair, for researchers who wish to use the techniques in this dissertation.

A caveat is in order here. While we believe that our results were quite interesting and indicative of how various strategies might perform given linguistic features of a language pair, we realize that our study is by no means comprehensive. We have, after all, only considered seven language pairs. Application of these approaches to many more language pairs in concert with consideration of the linguistic features of those language pairs would add a greater level of confidence to our results.

Also, for the sake of clarity, we will repeat the subheadings of chapter 5 here, with identical numbering, so that the linguistic motivators for the concrete predictions may be made clearer, and so that the reader may flip back to the relevant subsection of the previous chapter to read more about the topic. Thus, for example, section 6.2.1, Language Families Predictions, corresponds to section 5.1, Language Families, and section 6.2.2, Constituent Word Order Predictions, corresponds to section 5.2, Constituent Word Order.
6.2.1 Language Families Predictions

Prediction [P1a, b]

You may recall that our first prediction was that simple application of an HD tagger to LD text (D1 strategy) will perform best for pairs of languages with phylogenetic or historical cultural connections.

We had selected (Dutch, German), (English, German) and (French, German) as our language pairs, where in each successive language pair, the phylogenetic distance was greater.

Upon examining the relevant cells in the D1 row, we discover that our predictions have been borne out. This is:

1. D1 (Dutch, German) [43.5%] > D1 (English, German) [36.6%].
2. D1 (English, German) [36.6%] > D1 (French, German) [35.1%].

Thus, it does indeed appear that a closer relationship between paired languages predicts greater accuracy in even simple application of the HD tagger to the LD language.

However, looking through the rest of the data, we discovered an interesting apparent counterpoint, namely using Finnish as the HD and once again German as the LD. Finnish is
even more distant than French, so we would expect the accuracy to be lower. However, In this case, the accuracy was 40.6%, which while lower than Dutch as the HD, was higher than both English as French as the LD!

Upon investigation, we discovered that this was due to noise. Namely, we had trained this TreeTagger on the Finnish TreeBank, and we discovered that the Finnish TreeBank incorporated a few foreign language snippets in it. For example, there were a few instances of the following text: *Nationale Maatschappij der Belgische Spoorwegen*, which is the National Railway Company of Belgium:

- **Nationale**: N Nom Sg Cap
- **Maatschappij**: NON-TWOL Cap
- **der**: N ART Forgn
- **Belgische**: NON-TWOL Cap
- **Spoorwegen**: NON-TWOL Cap

While *der* was marked as a *Foreign* word, to cut down on the number of tags, I had taken only the concatenation of the first two tag elements, N_ART. And so, *der* was recognized as a determiner.

And there was, e.g., the German text *Landesversicherungsanstalt für das Saarland*, National Insurance Institute for Saarland marked as follows:

- **Landesversicherungsanstalt**: NON-TWOL Cap
- **für**: NON-TWOL
- **das**: N ART Forgn
- **Saarland**: N Prop Nom Sg Cap
And so, *das* was also recognized as a determiner. Thus, a few important and frequent German words, such as the German determiners, were in-vocabulary, something which was not the case for the other language TreeTaggers. Looking at the actually assigned tags, these German determiners were indeed being tagged accurately, something which would not occur had there not been foreign language text within the Finnish TreeBank.

For the LD-ification baseline, we had used the already created 3rd party HD taggers in order to tag a limited corpus within their own language, and then trained a new TreeTagger upon that limited corpus. A fortunate side effect of this would be to exclude any German words from the LD-ification baseline Finnish tagger, because no German sentences appear in the Europarl Finnish corpus upon which we trained. Looking at that lower row in the table, we indeed see all the relative accuracies which we would expect, in descending order based on proximity to German, namely that Dutch [50.0%] > English [36.4%] > French [34.9%] > Finnish [33.3%].

![Apply HD Tagger (trained on limited data) to German LD Language](image)

**Figure 29: Prediction 1 results, using LD-ify baseline**

**Bob using Prediction 1**

Of what use is this to “Bob,” who hopes to build a tagging solution for the LD language? Practically, it means that when seeking an HD language from which to project, it
might pay to seek out the HD language with the closest connections phylogenetic or historical cultural connections to the LD language. This is, after all, the starting point for any projection strategy, whether simple or complex.

**Prediction [P2a, b, c, d]**

As you may recall, the second prediction was that low-density NLP projection approaches based on cognate-identification techniques (H2 and L4 HD-ify) would perform best for pairs of languages with phylogenetic or historical cultural connections. Such language pairs could be expected to have more true cognates, and so cognate-identification replacement would be more common and more accurate.

We had selected the following language pairs to test this prediction: (Dutch, German), (German, Dutch), and (Finnish, German), where Dutch and German are very closely related Germanic languages and Finnish and German are unrelated languages.

Concretely, we predicted that H2 of (German, Dutch) and H2 of (Dutch, German) would each outperform H2 of (Finnish, German), as a tagging approach. Further, we predicted that HD-ification strategy of L4 on (German, Dutch) and (Dutch, German) would each be more successful than L4 on (Finnish, German).

Recall that the way we measure the success of a language-ification strategy is as delta from the baseline of that particular language pair, so the delta of HD-ify L4 (Dutch, German) is measured as compared with the HD-ify baseline of (Dutch, German).

These concrete predictions were borne out. Specifically,

A. Delta L4 HD-ify on (Dutch, German) [+24.6%] > Delta L4 HD-ify on (Finnish, German) [-1.9%]
B. Delta L4 HD-ify on (German, Dutch) [+6.8%] > L4 HDification on (Finnish, German) [-1.9%]

C. H2 on (Dutch, German) [80.5%] > H2 on (Finnish, German) [57.3%]

D. H2 on (German, Dutch) [82.1] > H2 on (Finnish, German) [57.3%]

Figure 30: Prediction 2 results

In general when considering L4 HDify on all language pairs, it was only German and Dutch, as closely related languages, for which cognate replacement made noticeable improvements. For the remainder of language pairs, cognate replacement made negligible improvement or caused harm. Recall that cognate detection was performed automatically, and replacement of a word by a false cognate introduces noise, which could harm the tagging effort.
Bob using Prediction 2

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? Practically, it means that if Bob plans to make use of simple cognation identification and gloss replacement to bridge the linguistic gap between his HD and LD, then it pays to seek out the HD language with the closest connections phylogenetic or historical cultural connections to the LD language.

Notably, the languages had to be extremely close in order for a low-quality cognate identification tool to help. If Bob wishes to use a cognate replacement strategy for more distant languages, then he will need to invest more time in developing the cognate tool – perhaps something hand-crafted after studying the two languages to see in particular how cognates transform from one language to the other; or perhaps something statistically trained on true cognate lists. Otherwise, using this strategy for other language-pairs would not be a good idea, and Bob should seek out a different language-ification strategy.
6.2.2 Constituent Word Order Predictions

Prediction [P3a, b, c, d]

As you may recall, the third prediction was that Low-density NLP projection approaches based on simple adoption of the HD contextual model (H1) will be most effective for pairs of languages that are closest in their fundamental word order (e.g., SVO, SOV, etc.).

We selected four language pairs:

1. (German, Dutch) which both are SVO in the main clause and SOV in the subordinate clause
2. (German, Finnish) which greatly differ, since Finnish has relatively free word order
3. (Estonian, Finnish) which both have relatively free word order
4. (English, German) where English has SVO in both the main and subordinate clause.

Concretely, we predicted that the closer the constituent word order of the language paired with German, the more successful the approach would be. Therefore H1 on (German, Dutch) > H1 on (English, German) > H1 on (German, Finnish). Further, the closer the constituent word order of the language paired with Finnish, the more successful the approach would be. Therefore, H1 on (Estonish, Finnish) > H1 on (German, Finnish).

These predictions were in fact borne out.
Figure 32: Prediction 3 results

A. H1 on (German, Dutch) [84.9%] > H1 on (German, Finnish) [62.2%]

B. H1 on (Estonian, Finnish) [83.8%] > H1 on (German, Finnish) [62.2%]

C. H1 on (German, Dutch) [83.8%] > H1 on (English, German) [79.3%]

D. H1 on (English, German) [79.3%] > H1 on (German, Finnish) [62.2%]

Also as mentioned there, while this comparison of H1 results is instructive for someone contemplating using the H1 approach, with its particular demands of linguistic resources, it is also instructive for someone contemplating various language-ification approaches as well. That is, the H1 results reveal the similarity of the contextual models of the language pair (since the LD lexical model based on an LD dictionary serves as a sort of oracle and may be considered of more-or-less equal use in each case), and that similarity might contribute to greater overall success.

**Bob using Prediction 3**

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? Practically, it means that if Bob wishes to use the H1 approach (or the H2
approach, which is built upon it), then he would do well to seek out an HD language which
shares constituent word order with his LD language.

Also, while this is by no means absolute, for the reasons discussed above, even when
contemplating other projection approaches (such as L1), it may pay to seek out an HD
language which resembles the LD language in constituent word order.

Bob might also extrapolate to other language features, other than constituent word
order, which would impact the contextual model of a language, and seek an HD language
which resembles the LD language in that regard.

6.2.3 Adjective premodifers and postmodifers Predictions

Prediction [P4a, b]

As you may recall, the fourth prediction was that language pairs which share noun-
adjetive order will benefit more from wholesale adoption of the HD contextual model (H1,
D1), and will thus have greater accuracy than language pairs which don’t.

To test this prediction, we selected (English, German) and (French, German).
Adjectives in both German and English are premodifiers (in that they come before the noun),
while they are (mostly) postmodifers in French. Consider, again, the following sample
sentence, in English, German, and French:

The quick brown fox jumped over the lazy dog.

Der schnelle braune Fuchs sprang über den faulen Hund.

Figure 33: English and German premodifiers in a sentence
Therefore, we would expect that, at least in this regard, the contextual models of German and English would be similar, while the contextual models of German and French would be different, and therefore H1 and D1 on (English, German) would have greater accuracy than H1 and D1 on (French, German).

This was indeed the case:

A. H1 on (English, German) [79.3%] > H1 on (French, German) [77.6%].

B. D1 on (English, German) [36.6%] > D1 on (French, German) [35.1%].

**Bob using Prediction 4**

Indeed, this is another example of a shared language feature, like constituent word order, which would lead to a closer contextual model shared between the HD and the LD.

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? Practically, it means that if Bob wishes to use H1, or start off on stronger contextual-model footing with a language-ification approach, he may wish to consider features such as adjective-noun order.
6.2.4 Article Presence Predictions

Prediction [P5a, b, c, d]

As you may recall, the fifth prediction was that language pairs in which both languages have articles would benefit more from gloss replacement of frequent words (L1 LD-ify, L1 HD-ify) than would languages pairs in which only one language, or both languages, lack articles.

This expectation was based on the idea that languages pairs in which both languages regularly make use of articles would have a closer shared contextual model than a language pair in which one of the languages lacked articles. Furthermore, since articles are rather frequently occurring lexical items, any language-ification which brought those articles into vocabulary would have greater success.

To test this, we selected three language pairs: (Dutch, German), where both languages have articles; (Estonian, Finnish), where both languages lack articles; (German, Finnish), where the HD has articles and LD does not.

Our concrete predictions were borne out:

![Figure 35: Prediction 5 results](image-url)
A. Delta L1 LDify on (Dutch, German) [+ 14.3%] > L1 LDify on (German, Finnish) [+ 8.1%].

B. Delta L1 LDify on (Dutch, German) [+ 14.3%] > L1 LDify on (Estonian, Finnish) [+ 3.2%]. Part of this may be attributable to lack of articles in Estonian and Finnish. I would also attribute this, though, to the fact that (Estonian, Finnish) started out as a much better match at it LDify baseline (at 60.7%, prior to any language-ification). Meanwhile, (Dutch, German) started out at only 50.0%.

C. Delta L1 HDify on (Dutch, German) [+ 28.5%] > L1 LDify on (German, Finnish) [+ 7.7%].

D. Delta L1 LDify on (Dutch, German) [+ 14.3%] > L1 LDify on (Estonian, Finnish) [+ 6.8%].

We note that, in languages which make use of them, we would expect articles to be among the most frequent words in a language. This is because there are so few of them, compared to nouns or adjectives, and they are required in so many contexts. Elsewhere in this study we consider just how many words need to be replaced in an L1 approach, as well as the distribution of different parts of speech in the most frequent X words of each language. But we might well expect that this boost would take hold with even a minimal amount of replacements – for instance, of the 50 most frequent LD words.
Figure 36: Prediction 5 results, with 50 most frequent word replacement

Indeed, this seems to be the case: The improvement over the baseline for (Dutch, German), for just 50 words, is [+ 20.8%]. Compare that with the improvement over the baseline for (Dutch, German), replacing 250 words, at [+ 25.8%]. Just the first 50 words took us most of the way.

**Bob using Prediction 5**

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? If both his LD language and his available HD language employ articles, then L1 might well be a useful strategy to select. And, if he is employing L1 as a strategy and his LD language has articles, he would do well to select an HD language with articles as well.

Furthermore, since we see that gloss-replacement of articles (and other most frequent occurring words, such as conjunctions) can have make a drastic improvement over the baseline, even if Bob intends to employ some other strategy, in addition to that strategy, he might engage a language expert for the minimally intensive task of translating just the LD articles into the HD and just replace those.
6.2.5 Article Granularity Predictions

Prediction [P6a, b, c]

While the previous prediction considered the mere presence or absence of articles in a language, this prediction considers granularity of those articles.

As you may recall, the sixth prediction was that gloss replacement of frequent words (L1), in an LD-ification strategy, would have greater success if the HD language of the language pair has a low lexical granularity of articles and the LD language has a high lexical granularity of articles. Furthermore, that this difference in success of gloss-replacement of frequent words is limited to this specific scenario. The gloss replacement of frequent words, in an HD-ification strategy, generally performs quite well, and should perform quite well in this instance as well. Further, if the article granularities of the HD and LD languages were reversed, with the HD language having the higher lexical granularity of articles, then HD-ify and LD-ify strategies should be comparable.

We used these illustrations to demonstrate the difficulties involved in L1 LD-ify as opposed to L1 HD-ify.

![Diagram](image)

Figure 37: Projection of articles, with high and low lexical granularity

If German has more articles than English, then German LD-ifying an English HD-corpus would bring only one German article into vocabulary. But, English LD-ifying a German HD corpus would bring all (one) of the English articles into vocabulary. And
regardless, HD-ifying some LD text prior to tagging would work well regardless of the
lexical granularity of articles, because there would be gloss replacements for every LD
article, and whatever HD article was chosen to replace it, it would indeed be an article.

We selected the following language pairs, based on lexical granularity of articles:
(English, German) as an example of (low granularity, high granularity) and (German, Dutch)
as an example of (high granularity, low granularity).

Our concrete predictions were borne out:

A. Delta L1 LD-ify on (German, Dutch) [+32.3%] > delta L1 LD-ify on (English,
   German) [-1.1%].

B. Delta L1 LDiffy on (German, Dutch) [+32.3%] is approximately the same as delta
   L1 HD-ify on (German, Dutch) [35.1%].

C. However, as predicted, delta L1 LD-ify on (English, German) [-1.1%] < delta L1
   HD-ify on (English, German) [+29.6%], and is in fact much smaller.

These results make sense because, as previously discussed, the greater lexical
granularity of articles in German as opposed to English means that, in the LD-ify direction,
many German articles remain out of vocabulary. This manifests itself in a disparity between
the results of L1 LD-ification of the respective language pairs (that is, result A). This
problem is only present in the LD-ify direction because only there do the German articles
remain out of vocabulary. Therefore, where there is no change in the lexical granularity of
articles within a language pair, as in (German, Dutch), the LD-ify result will approximately
match that of the HD-ify result (that is, result B). But where there is a change in the lexical
granularity of articles within a language pair, HD-ify will succeed to a much higher degree
than LDiffy (that is, result C).
Bob using Prediction 6

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? If he is considering LD-ify approaches, then he should carefully consider the relative lexical granularity of articles in his language. If his LD has high lexical granularity of articles, then his HD should as well. If there is this lexical granularity disparity in the direction of the HD, then he should consider using an HD-ify approach instead.

Alternatively, he should realize that this is going to present a problem for his LD-ification approach and take steps to address it. For example, he might build a dictionary of (the fewer) HD articles mapped to the (many) LD article equivalents, and then when performing gloss replacement in the HD corpus, iterate through the alternatives in order to allow each of the LD articles a chance to appear in the LD-ified corpus. Or alternatively, with a little language-specific knowledge, he can consider placement of the article in the sentence to guess at whether to use the article with the nominative or accusative case. Regardless, he should realize that he might need to design his solution around this difficulty.

Furthermore, while this was studied for articles, the same might be said for other parts of speech with differing levels of lexical granularity between languages, and so Bob might keep that in mind as well.

6.2.6 PRO-Dropness Predictions

Prediction [P7]

As you may recall, the seventh prediction was that approaches involving wholesale adoption of the HD contextual model (H1) will be most effective for pairs of languages in which both are PRO-drop or both are not PRO-drop, but that if there is a divergence in this language feature, then such an approach will not be as successful.
In fact, this prediction was borne out:

H1 on (English, German) [79.3%] > H1 on (Finnish, German) [77.9%].

![Figure 38: Prediction 7 results](image)

**Bob using Prediction 7**

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? If he is using an H1 (or H2) approach, then it would be better to select an HD language which is similar in PRO-dropness. Further, it might be a good idea to do this if using language-ification strategies in general, because what this reveals is that there is indeed an impact felt in the contextual model as a result of this difference.

**6.2.7 Richness of Inflection**

**Prediction [P8a, b, c]**

As you may recall, our eighth prediction was that, assuming a difference in level of inflection between the HD and the LD languages, LD-ification approaches would be more successful if the HD inflection level was greater than the LD inflection level.
We select the following two language pairs, which are reflections of one another:

1. (German, Dutch) as (richly inflected, poorly inflected)
2. (Dutch, German) as (poorly inflected, richly inflected)

Our concrete predictions were borne out:

A. Delta L1 LDify for (German, Dutch) [+32.3%] > delta L1 LDify on (Dutch, German) [+14.3%].
B. Delta L2 LDify for (German, Dutch) [+9.8%] > delta L2 LDify on (Dutch, German) [-20.3%].
C. Delta L4 LDify for (German, Dutch) [+4.1%] > delta L4 LDify on (Dutch, German) [-7.5%].

![Level of Inflection and Language-ification](image)

**Figure 39: Prediction 8 results**

**Bob using Prediction 8**

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? If he is considering an LD-ification approach, whether of the simplistic variety as
explored in this dissertation or the more complex variety, he would do well to consider the level of inflection of his LD and his HD, because gloss replacement which projects from the corpus of poorly inflected language to a richly inflected one does not perform well.

**Prediction [P9a, b, c]**

As you may recall, the ninth prediction was that, assuming a difference in level of inflection between the HD and the LD languages, HD-ification approaches would be more successful if the HD inflection level was **lower** than the LD inflection level. (That is, in general, for both HD-ify and LD-ify, it is better if the language-ification is in the direction of the lower level of inflection.)

We selected the same two language pairs as in the previous prediction. Our concrete predictions were borne out:

**Figure 40: Prediction 9 results**

A. Delta L2 HD-ify for (Dutch, German) [+ 1.2%] > delta L2 HD-ify on (German, Dutch) [+ 0.8%].

B. Delta L3 HD-ify for (Dutch, German) [+ 3.1%] > delta L3 HD-ify on (German, Dutch) [+ 1.0%].
C. Delta L4 HDify for (Dutch, German) [+ 26.4%] > delta L4 HDify on (German, Dutch) [+ 6.8%].

**Bob using Prediction 9**

Of what use is this to Bob, who hopes to build a tagging solution for the LD language? If he is considering an HD-ification approach, whether of the simplistic variety as explored in this dissertation or the more complex variety, he would do well to consider the level of inflection of his LD and his HD, because in the case of HD-ification, gloss replacement which projects from poorly inflected language to a richly inflected one does not perform well.

### 6.3 HD-ification vs. LD-ification

Within this study we considered the effects of language-ification, but in two opposite directions: HD-ification of the LD text followed by tagging by a 3\textsuperscript{rd} party HD tagger; and LD-ification of a large tagged HD corpus, followed by supervised training of that LD-ified corpus and then application of that trained tagger to an LD text.

If Bob reads this study, which over-arching approach should he select, HD-ify or LD-ify? I believe that the answer depends in part on which language-ification approach, L1, L2, L3, or L4, he intends to use.
For L1, the overall scores for HD-ify were higher than for LD-ify, but this is not necessarily as meaningful as it seems. Recall that for the HD-ification approaches, the 3rd party HD tagger might have been trained on a very large, expert-tagged HD corpus. For the LD approaches, we did not have access to that original HD corpus, and so we used the HD tagger to tag HD text in a somewhat smaller the Europarl corpus, used the smaller shared tagset, and trained on that. As a result, the LD-ification corpora are (a) somewhat noisier, being the result of machine tagging; (b) with less lexical HD words; and (c) with less granularity in their tagsets.

If Bob were to attempt LD-ification, he would start with a large manually-tagged HD corpus, and so might well be on more equal footing with the HD-ification approaches.

Still, L1 LD-ification failed in a number of cases, for reasons discussed above (e.g. prediction 6). While an analysis of the linguistic features of the language pair would serve
to identify such cases such that Bob can avoid LD-ification or engineer around it, it still does seem that L1 HD-ify is the surer bet.

![Figure 42: L4 HD-ify vs. LD-ify](image)

In terms of L4 as well, it seems that the HD-ify approach was the better approach. In general, L4 failed, or only yielded a rather small improvement, due to the noise inherent in the process of automatic cognate identification. Yet, where the languages were close enough, such as (Dutch, German) and (German, Dutch), there was the possibility of significant improvement over the baseline.

### 6.4 Frequent Word Replacement – How many words?

One language-ification approach that seemed to consistently succeed in rather dramatic fashion was the L1 HD-ify approach, which was gloss replacement of the most frequent words in the LD for their HD equivalents. However, left unspecified was just how many of those most frequent words to replace? If one replaced 0 words prior to tagging with the HD tagger, then that is the same as the HD-ify baseline. At the other extreme, if the LD language has a vocabulary of 10,000 words and the most frequent 10,000 LD words are
replaced, then this is substitution of every single LD word. How many words will we replace in this strategy?

However, we would expect the law of diminishing returns to apply here. The most frequent words – determiners, pronouns, modal verbs, and so on – will probably be quite frequent, meaning that relatively few lexical items will be represented a large number of times in the LD text. Therefore, in terms of the impact on the lexical model, gloss replacements of just these words would have a great impact. Further, we might also expect many of these frequent words to have tags which are somewhat out of the ordinary, and thus have a greater impact on the contextual model as well. For instance, if an article were brought into-vocabulary, then not only would the article be tagged correctly but, due to strongly encoded trigrams in the contextual model, the tagger might more accurately predict that the (unknown or ambiguous) word which followed was a noun or adjective, as opposed to a conjunction. Meanwhile, bringing a noun into-vocabulary would not necessarily tell us as much about the word which followed, since many different types of words appear in the immediate context of nouns.
We therefore ran the L1 HD-ify experiment on each language pair multiple times, with a frequent wordlist in increments of fifty. That is, we replaced the 0, 50, 100, 150, 200, and 250 most frequent words in the LD language, and say how accurate the tagging was. The results are summarized in the following chart:

![Chart showing frequent word replacement](image)

**Figure 43**: Frequent word replacement, 50...250

Some language pairs seem to have a rather large delta with only 50 replacements, and so we investigated further, staging experiments with 25, 10, 5, or even 2 replacements. For example, (German, Dutch) increased from 35.5% to 64.5% (a delta of 29%) after replacing just the 50 most frequent words. Subsequent investigation revealed that by replacing just two Dutch words (*de* $\rightarrow$ *die*, meaning *the*, and *van* $\rightarrow$ *von*, meaning *of*), accuracy increased to 46.9% (a delta of 11.4%). By replacing just another three words (*het* $\rightarrow$ *die*, meaning *the*; *en* $\rightarrow$ *und*, meaning *and*; and *een* $\rightarrow$ *ein* meaning *a*), accuracy increased to 53.6% (a delta of 18.11%).
You may recall that we had selected seven language pairs, and had taken them from a set of six languages. Those languages were: German, Dutch, English, French, Finnish, and Estonian. For each of these six languages, we analyzed just what parts of speech occur in each the frequent wordlists of each size. For example, consider the following table / chart, for Dutch:

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>4</td>
<td>17</td>
<td>27</td>
<td>40</td>
<td>46</td>
</tr>
<tr>
<td>ADVERB</td>
<td>13</td>
<td>30</td>
<td>43</td>
<td>57</td>
<td>69</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>8</td>
<td>15</td>
<td>20</td>
<td>20</td>
<td>23</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>NOUN</td>
<td>15</td>
<td>33</td>
<td>62</td>
<td>79</td>
<td>105</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>14</td>
<td>23</td>
<td>23</td>
<td>27</td>
<td>31</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>15</td>
<td>29</td>
<td>32</td>
<td>36</td>
<td>39</td>
</tr>
<tr>
<td>VERB</td>
<td>9</td>
<td>15</td>
<td>23</td>
<td>33</td>
<td>42</td>
</tr>
</tbody>
</table>

Figure 44: Frequent word profile for Dutch

If you turn your head sideways, the bar graphs in each column (50 most frequent words, 100, etc.) correspond to the distribution of tags for those frequent words. Considering only the 50 most frequent words, only 5 of those words are articles, compared with 15 pronouns. This stays the case even when we consider the 150 most frequent words – 5 are articles, compared with 62 pronouns. For the most part, the general form of these bar graphs in this particular remains the same, though nouns do quickly outpace pronouns.

This frequent word profile might lend some quick visual insight into the lexical / contextual models of a language as well as perhaps its similarity to another language. In turn, this might conceivably help someone predict in advance how well a language-ification attempt will work for a pair of languages. To illustrate this, consider the contours of the bar
charts for Dutch, in the chart above, and compare them with the contours of the bar chart for German:

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>3</td>
<td>11</td>
<td>17</td>
<td>24</td>
<td>31</td>
</tr>
<tr>
<td>ADVERB</td>
<td>10</td>
<td>24</td>
<td>35</td>
<td>46</td>
<td>57</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>4</td>
<td>14</td>
<td>19</td>
<td>22</td>
<td>25</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>NOUN</td>
<td>6</td>
<td>18</td>
<td>30</td>
<td>52</td>
<td>79</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>16</td>
<td>21</td>
<td>25</td>
<td>29</td>
<td>32</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>18</td>
<td>31</td>
<td>40</td>
<td>48</td>
<td>54</td>
</tr>
<tr>
<td>VERB</td>
<td>10</td>
<td>16</td>
<td>27</td>
<td>34</td>
<td>39</td>
</tr>
</tbody>
</table>

Figure 45: Frequent word profile for German

The overall numbers for each of these parts of speech might also tell us something about their respective lexical granularity, or level of inflection, which as we saw earlier is useful in making predictions as to how various language projection strategies might perform.

It is also worth noting that these tables only count parts of speech within the gloss-replacement wordlist. That is, the five Dutch articles might be the most frequent words in a Dutch corpus, but it also counts in the above reckoning as 5. Meanwhile, the fifteen nouns would count as 15, even though when looking at how often they actually appear in a corpus, all these nouns altogether occur less frequently than the five articles. Because of this, we decided to also capture the relative frequency of these replaced words, by part-of-speech. For example, here is such a chart for Dutch:
We see here that when replacing only 50 most frequent words, the five articles comprise a good 25% of replaced words in the corpus, compared with just 5% for the fifteen nouns. Even as the dictionary gets larger, to 250 words, the articles make up 18%, even though few to no articles were added. Meanwhile, a large number of nouns were added to the dictionary by the time it is 250 words, but each noun had low frequency, such that altogether it only comprises 14%.

If you recall, Dutch had a rather dramatic rise in accuracy even when replacing relatively few words. Perhaps these tables and charts can help account for this, in that so many of these articles and prepositions, comprising such a high percent of the corpus words, were brought into vocabulary early on even within to list of 50 most frequent words. And since they comprise such a high percentage of the overall corpus, they also have the most dramatic impact on tagging accuracy.

We computed these metrics for all languages we studied, but our aim here is only to define the metrics we used and note how they might prove useful. All the charts and tables for all six languages may be found in Appendix B.

![Figure 46: Frequent word distribution for Dutch](image)

<table>
<thead>
<tr>
<th></th>
<th>Dutch</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>50</td>
</tr>
<tr>
<td>ADJECTIVE</td>
<td>0%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>7%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>25%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>17%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>5%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>23%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>20%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>9%</td>
</tr>
</tbody>
</table>
Chapter 7: Projecting from Farsi to Tajiki: A Case Study

Until this point in our study, we have only considered pseudo-LD languages. By this, we mean that our concern was in language-ification approaches, and one language in a pair was designated as the low density language (for which we withheld linguistic resources) while the other language in the pair was designated as the high-density language, for which a wealth of linguistic resources (e.g. a large tagged corpus or a third-party tagger) was available. However, neither of the languages in the pair was, strictly speaking, truly low-density. In order to be able to evaluate how well different strategies performed, and in order to compare equitably across different language pairs, we selected languages which were in the Europarl corpus and for which there existed a TreeTagger. In this way, we could tag the untagged Europarl text and create a somewhat noisy gold standard for our designated low-density language.

Now, in this chapter, we consider the case of tagging an actual low-density language, Tajik, using a related high-density language, Farsi. That is, there are three main distinct varieties or dialects of Persian: Iranian Persian (Farsi), Afghani Persian (Dari) and Tajiki Persian (Tajik).
Farsi is written in an extended Arabic script and has many linguistic resources associated with it. Tajiki, meanwhile, used to be written in the same extended Arabic script but is presently written in an extended Cyrillic script. It also does not have many linguistic resources.

Tajiki and Farsi are quite related. They are mutually intelligible, though due to pronunciation differences, written Tajiki is much more intelligible than spoken Tajiki to a Farsi speaker. Tajiki and Farsi have a large overlapping vocabulary, but there are still differences. For example\textsuperscript{16}, the word \textit{kalaan} means “big” in both Farsi and Tajiki, but in Farsi, it is not commonly used, and the word \textit{bozorg} is preferred. So too as part of larger words. Farsi for grandfather is \textit{pedarbozorg} (father-big), while in Tajiki it is \textit{pedarkalaam}. Farsi for grandmother is \textit{maadarbozord} (mother-big), while in Tajiki it is \textit{maadarkalaam}. To grow up in Farsi is \textit{bozorg shodan}, while in Tajiki it is \textit{kalaam shodan}. Tajiki maintains

\textsuperscript{16} See discussion here: http://polyglotclub.com/language/tajik/post/14897
a number of archaic Persian words in its vocabulary. Tajiki vocabulary are also has many loanwords and loan translations from Uzbek (a Turkic language) and Russian.

Farsi and Tajiki grammar also differ. Tajiki retains certain archaic grammatical constructions which have been dropped in other Persian languages. One prominent example of how Tajiki differs from classical Persian\(^{17}\) is in the construction of the present progressive tense (“I am writing a letter”). In Tajiki, one expresses this as the present progressive participle ending with –а (thus, навишта) and the cliticized verb form –аст, 'to be' (thus, истода-ам). The full sentence would be Ман макуб навишта истода-ам, meaning “I letter write first-person-singular-be-participle”). In contrast, Farsi would use the verb dar (‘to have’) followed by a conjugated verb in either the simple present tense, the habitual past tense, or the habitual past perfect tense.

In terms of morphology, both Farsi and Tajiki use the direct object marker ro, but Tajiki has it as a suffix while Farsi has it as a stand-alone morpheme. Both Farsi and Tajiki have the izofat, a grammatical particle which links two words together, but in Farsi it is spoken but not written. For more on the many differences between these two Persian dialects in vocabulary, syntax, morphology, and phonology, see Beeman (2005) and Aliev and Okawa (2010).

We selected Tajiki as a low-density language for a few reasons. Firstly, it was quite dissimilar to many of the other languages discussed so far, and so it would be interesting to know how the different projection strategies would fare. Secondly, Farsi and Tajiki were closer than any of my other language pairs, and are indeed both dialects of Persian. We expect some interesting results from certain language-ification strategies. For instance, there

\(^{17}\) From Wikipedia: http://en.wikipedia.org/wiki/Tajik_language
is a rather large overlap in vocabularies, so even baseline application of a Farsi tagger to a Tajiki corpus should have a measure of success; and since a large number of true cognates exist, cognate detection and replacement might succeed much more than it did for any of my other language pairs. Thirdly, Tajiki is written in a different script than Farsi, and the problem of cross-orthographic projection is one we would like to explore. Finally, this author lives and works in Flushing, where there is a large Tajiki and Bukharian (Judeo-Tajik) community, and so we thought to make use of a local resource.

7.1 Construction of Tajiki linguistic resources

Our initial plan was to hire Tajiki speakers or Bukharian speakers who were not linguists (e.g. college students, neighbors), teach them how to identify parts of speech with examples taken from English and from a tagged Farsi corpus, and ask them to translate and tag a Tajiki corpus we had assembled. By having two or more people working on tagging the same text, and investigating and resolving instances of disagreement, we could assemble a small Tajiki corpus to use as a gold standard, for the sake of evaluating how well different projection strategies performed.

However, we soon revised that plan. We approached a Bukharian-speaking colleague for recommendations in who could assist in this task, and his referral brought us to someone who not only could speak Tajiki and Farsi, but was also a linguistic expert, who had developed a Farsi to Tajiki intensive Bridge course and had helped develop a multi-volume college textbook on the Tajiki language.

We asked the expert to construct three linguistic resources:

1. A tagged Tajiki corpus, to serve as a gold standard

2. A Tajik → Farsi frequent word dictionary
3. A Farsi $\rightarrow$ Tajik frequent word dictionary

By far, the most time, attention, and labor-intensive resource to assemble was the tagged Tajiki corpus. However, we would stress that this represents an effort that Bob would not need to undertake. We only needed this hand-tagged Tajiki corpus as a gold standard, in order to evaluate how well the various strategies were performing.

We first gathered many news articles from the TojNews\textsuperscript{18} website, spanning from October 11\textsuperscript{th} to October 17\textsuperscript{th}, 2011. This gave us an untagged corpus. We tokenized the text and placed each lexical item on its own line in an Excel spreadsheet and pretagged what we could ourselves (e.g. numbers and punctuation). This yielded a corpus of 14,132 words.

We decided to utilize the simplified joint tagset used for the gold standard for all the other languages, since this would aid in comparison to other language pairs. After discussion with the language expert, we added an additional part-of-speech, CompoundVerb, in which otherwise each word would be a different part or speech but doesn’t deliver the semantic meaning of the verb.

The effort to build this gold standard took several months, for a few reasons. The language expert was working on this part-time; our respective schedules and workloads did not always match up; and we had numerous discussions regarding the best way to handle ambiguous tagging cases. While for all of these ambiguous cases, the language expert certainly could have given a part-of-speech tagging which was linguistically correct, there was a secondary goal which was to make the tags conform as closely as possible to the tags in our chosen Farsi corpus, which was to serve as the basis for our Farsi TreeTagger. In this

\textsuperscript{18} http://www.tojnews.tj/
way, we could try to minimize noise due to divergent decisions of human taggers of Farsi and Tajiki.

In contrast, the effort to develop the Tajiki \(\rightarrow\) Farsi frequent word dictionary took considerably less time and effort. We used a computer program to generate the 250 most frequent words from a Tajiki corpus and asked the language expert to give the following for each word: English equivalent (for our own knowledge, since it might be helpful for analysis and our own understanding of the tagged corpus), part of speech (for the analysis of part-of-speech composition of the most common words), and the Farsi equivalent (which was the only part actually needed for the L1 strategy). We did the equivalent and asked for the equivalent for the Farsi \(\rightarrow\) Tajiki frequent word dictionary. (We had considered building this resource ourselves using online and print dictionaries of Persian and Tajiki, using English as a pivot language, but we opted instead to ask the expert.)

One consideration, which likely helped the generation of these frequent word dictionaries, is that the words in Farsi and Tajiki were so similar, and in most cases were cognates. There were a few complications and design decisions even here. For one example, which the language expert raised, both Farsi and Tajiki make use of the izofat, a grammatical particle which links two words together. In Farsi the izofat is pronounced but not written. In Tajiki, it is also written. Therefore, when translating from Tajiki to Farsi, the specific word is unambiguous and one can select the appropriate Farsi translation. However, when translating from Farsi to Tajiki, how does one know whether the word should be the one with the izofat or the one without? For instance, مورد can be read as "маврид" with a meaning of [case, instance, occasion] or as "мавриди" (with the bolded letter as the izofat, an unstressed enclitic vowel pronounced /i/), where it is a part of phrase "дар мавриди" with
meaning of [about, in the case of]. The former is a noun, while the latter is a complex preposition or conjunction, and these are used more frequently. When looking at a word in context, within a Farsi corpus, the specific meaning is readily apparent. But here we are dealing with word in isolation, so it is unclear which translation to select.

We decided that, since the point of this dictionary resource was replacement of the most frequent word, it made sense to select, in each instance, the Tajiki equivalent which would be the more common. Either choice would lead to some incorrect replacement and thus noise, but this approach would minimize the noise and maximize the words in the Persian corpus being brought correctly into vocabulary.

**Construction of Farsi TreeTagger**

On the Farsi side, we needed one linguistic resource, namely a TreeTagger. All of our other HD languages in our study had TreeTaggers, and so a Farsi TreeTagger was called for in order to perform a proper comparison.

We surveyed the various Farsi part-of-speech tagged corpora which were available, and selected the Bijankhan corpus19. There are two versions of this corpus: the original corpus, as created in Oroumchian et al (2006), with a tagset granularity of 550 tags and a processed version of the corpus, more suitable for NLP tasks, as created in Amiri et al. (2007), with tagset granularity of 40 distinct tags. Based on conversations with the author of TreeTagger and past experience struggling to create a TreeTagger for Finnish with a high tagset granularity, we came to understand that one cannot create a TreeTagger with as many as 550 distinct tags. Further, many of the TreeTaggers for other languages discussed so far

have a tagset granularity of approximately 40, and so the latter version of the corpus is the right choice. These 40 tags and their meaning are listed in Appendix A.

One obvious difficulty with creating a Farsi TreeTagger with the intent to use it on Tajiki is that Farsi and Tajiki are written in different scripts. The Bijankhan corpus is written using a Persian Arabic script, while the Tajiki corpus is in an extended Cyrillic script. If so, every single Tajiki word would be out of vocabulary. To bridge this orthographic gap, we could:

1. Convert the Tajiki corpus from Cyrillic → Arabic script and then tag
2. Convert the Bijankhan corpus from Arabic script → Cyrillic
3. Romanize both the Tajiki corpus (Cyrillic → Latin) and Bijankhan corpus (Arabic → Latin) and have them meet in the middle

We decided upon the third approach for the sake of simplicity. Romanized text would work more straightforwardly with third party tools like Automorphology (used to discover Farsi or Tajiki affixes). Each character in a string would take up a single byte (as an ASCII character) rather than multiple bytes (as a Unicode character), and this would work better with e.g. the Python cognate identification code that was written. Furthermore, meeting in this common-denominator middle of Roman characters could help overcome some of the ambiguous mappings between the two scripts.

Tajiki, written in extended Cyrillic script, has both capital and lowercase letters, while Farsi, written in Persian Arabic script, does not. So, for example, there is both a capital Д and a lowercase д, which make the sound /d/, and which map to ژ in the Persian Arabic script. This capitalization could be helpful on the Tajiki side, in that capital letters which do not appear at the beginning of a sentence would indicate a proper noun (even if the word
were out of vocabulary), and could disambiguate between a capitalized word (noun) and a lowercase word (say, and adjective). However, in a resource-light environment, with little to no language-specific code, there is no real way to know whether a particular .twitch in the Bijankhan corpus should be mapped to a capital Д or a lowercase д. Indeed, the noun tags in the Bijankhan corpus are N_PL and N_SING to distinguish between plural and singular, but there is no distinction made between proper and common nouns, so we could not decide on that basis. In this case, Romanizing the Bijankhan corpus would map the twitch to a d, and at the same time Romanizing the Tajiki corpus would map both the capital Д and the lowercase д to a d. Therefore whichever way (capital or lowercase) the word appeared in the original Tajiki corpus, it would not be in-vocabulary to the Farsi tagger.

Tajiki as written reflects pronunciation while Farsi, for borrowings from Arabic words, maintains the distinct Arabic letters even where they are not pronounced differently. Persian as pronounced lacks phonemes such as interdentals and emphatic alveolars. Written Farsi words will maintain those distinct letters while written Tajiki will not. Thus, for example, Tajiki will have capital З and lowercase з, with a phonetic value of /z/, and the written Farsi equivalent might be any of the following case-insensitive letters: ض (Ḍād), ط (Ẓāʾ), ذ (Ḏāl) , and ز (Zayin). Transliteration of Farsi → Tajiki would then be straightforward, with all these Arabic letters mapping to the lowercase Cyrillic з. However, in the opposite direction, Tajiki → Farsi, it is not as straightforward. Should the з be mapped to ﺿ (Ḍād), ط (Ẓāʾ), ذ (Ḏāl) , or ز (Zayin)?

Megerdoomian and Parvaz (2008) attempt to deal with ambiguous mappings such as this. With an eventual goal of making use of Farsi machine translation tools, they build a grammar which compiles to a finite-state-transducer to transliterate Tajiki words to Farsi.
This transducer overproduces, but they then scan through HD Farsi resources to find the produced word that is in-vocabulary.

Our Romanization approach deals with this problem in a different fashion (and can indeed do so because we are in control of the production of the Farsi tagger). The Cyrillic З and з are mapped to a z, and the Persian Arabic ض (Ḍād), ط (Ẓā’), ذ (Ḍāl), and ز (Zayin) are mapped to a z. This admittedly introduces some noise, but does not require extensive access to Farsi resources at the transliteration step.

Another difficulty in transliterating Farsi, in Persian Arabic script, to Tajiki, in Cyrillic script, is that often the Persian Arabic will not represent certain vowels. For example, the izofat discussed above is written in Tajiki as и and is pronounced as /e/. In Farsi, this is a diacritic and is most often not actually written, and so the word would appear without the izofat in the Bijanhkan corpus. With some deep linguistic knowledge of Persian, one could theoretically examine the assigned tags (to see e.g. if it is assigned N_SING for NOUN SINGULAR or PP for Prepositional Phrase) or examine the immediate lexical context (to see if e.g. it appears immediately after the Farsi word در (Romanized dar and Cyrillic դար). This deep linguistic knowledge and analysis is, however, what Bob is at least initially trying to avoid. The Romanization approach would not help here, and there will be a mismatch both from the Farsi side and from the Tajiki side.

Similarly, there is not always a straightforward mapping of Tajiki vowels to Farsi vowels. For instance, some vowels (/a/, /e/ and /o/) appear as diacritics but sometimes these diacritics are absent. The /o/ sound might be encoded in Farsi as an alef, an ayn, or as nothing at all. The /a/ sound might be encoded in Farsi as an alef (!) in word-initial position, a heh (‘) in word-final position, and not written at all when appearing in the middle of the word. One
could carefully catalogue all the ways vowels appear, depending on context, and engineer a transliteration solution to deal with many of these cases. Our hypothetical computational linguist “Bob” does not do this, and the Romanization approach will yield mismatches here as well.

After studying how Farsi and Tajiki mapped to one another in a somewhat ambiguous manner, we constructed Arabic→Roman and Cyrillic→Roman transliteration dictionaries where the unambiguous letters would meet in the middle and the ambiguous letters would also meet in the middle. We then Romanized our Tajiki corpus and the Bijankhan corpus.

We then preprocessed the Romanized Bijankhan corpus to make it acceptable input to the train-treetagger program. These preprocessing transformations included such actions as replacing contiguous spaces with a single tab, replacing the corpus-particular sentence delimiter (#, tagged with DELM) with period (tagged with SENT), replacing the DELM tag for punctuation with a PUNCT character, dealing with multiple words assigned a single tag, and removing numbers and their tags from the lexicon file. All of these steps were implemented as regular expression replacements, which we carefully documented and saved, so that we could repeat these steps as needed (say, if we discovered at a late stage that the Romanization missed or mismatched some character). Earlier, we had also written a Python script to automatically build the additional lexicon and taglist files needed as input to train-treetagger (used for the LD-ify strategies and for building the Finnish TreeTagger as well), and this same Python script worked here as well.

In this manner, we built the (Romanized) Farsi TreeTagger.
7.2 Farsi and Tajiki cognation detection

One Python script needed particular tweaking: that for cognate detection. For other language pairs, the cognition detection model ignored capitalization. Here, since we utilized certain Roman capital letters as opposed to their lowercase equivalents to capture specific Tajiki and Farsi letters. For instance, we encoded Tajiki ғ and Farsi گ (/g/) as a lowercase g and we encoded Tajiki ғ and Farsi ٔ (/ʁ/, that is /gh/) as a capital G. The cognition detection should not conflate the two. Furthermore, we used certain consonantal characters (e.g. capital R) to represent Tajiki (e.g. ی) and Farsi (e.g. ی and یَ) vowels, and the cognition detection assigns a lower cost to vowel substitution, deletion, and insertion than for consonantal substitution, deletion and insertion. Also, the Farsi ھ (h) when appearing in word-final position often functions to indicate the presence of a vowel. Therefore these particular vowels needed to be added to the vowel list.

Finally, for other language pairs, in order to reduce noise and eliminate spurious cognate suggestions, insertion and deletion of a vowel incurred a cost of 0.25 when that vowel followed another vowel and 1.0 when that vowel was word-initial or followed a consonant. In this way, if ae in the source language mapped to a in the target language, that substitution only incurred a cost of 0.25, but arbitrary insertion of word-initial vowels or vowels between consonants incurred a real insertion cost. In the Tajiki and Farsi case, however, the izofat often appears after a consonant, and this should not be penalized; and the Farsi diacritics which often do not appear, but which follow consonants, corresponding to Tajiki vowels, should not incur a cost of 1. Therefore, we modified the cost function so as to assign a cost of 0.25 in all instances.
7.3 *Farsi and Tajiki affix dictionaries*

Other linguistic resource we needed to build were the prefix and suffix dictionaries for Tajiki $\rightarrow$ Farsi. We ran the Romanized corpus of Tajiki through Automorphology to generate a list of prefixes and suffixes. Then, we consulted the L1 frequent word dictionaries for Tajiki $\rightarrow$ Farsi which our linguistic expert had created. Since the Farsi and Tajiki words in this dictionary were in most cases similar, we could find identify the regular target prefixes and suffixes. There was indeed some ambiguity as to this mapping, and so, for L2 approaches, we selected the most frequently occurring target affix while, for L3, we allowed it to function as a multimap. The other linguistic resources we needed to build were the prefix and suffix dictionaries in the opposite direction, for Farsi $\rightarrow$ Tajiki. We used a similar identical process to what was described above, but using the Romanized Farsi corpus and the frequent word dictionary for Farsi $\rightarrow$ Tajiki.

7.4 *Results of experiments on (Tajiki, Farsi)*

We then ran all the experiments on the (Farsi, Tajiki) language pair. Here is a table of the results for our reimplementation of the traditional approaches:

<table>
<thead>
<tr>
<th>APPROACHES TABLE: RESULTS (as % correct)</th>
<th>Farsi / Tajik</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Traditional Approaches</strong></td>
<td></td>
</tr>
<tr>
<td>D1 -- Apply 3rd party HD tagger to LD text</td>
<td>61.2%</td>
</tr>
<tr>
<td>D2 -- Upper Baseline, apply 3rd party LD tagger to LD text</td>
<td>98.2%</td>
</tr>
<tr>
<td>D3 -- Unsupervised training on LD corpus</td>
<td>42.0%</td>
</tr>
<tr>
<td>D4 -- Unsupervised training on LD corpus, using noisy tags from 3rd</td>
<td>31.7%</td>
</tr>
<tr>
<td>D5 -- Same as D4, but filter for distributional criteria</td>
<td>36.0%</td>
</tr>
<tr>
<td>D6 -- Same as D4, but incorporate affix probabilities</td>
<td>32.8%</td>
</tr>
<tr>
<td>H1 -- using LD lexicon, uniform lexical model</td>
<td>87.5%</td>
</tr>
<tr>
<td>H2 -- same as H1, but for cognates, average with HD's lexical model</td>
<td>79.4%</td>
</tr>
</tbody>
</table>

Table 8 Tajiki Results, Traditional Approaches

The D1 baseline, applying the HD Farsi tagger to the LD Tajiki text had an accuracy of 61.2%, which was higher than any other language pair, including close languages such as (Dutch, German) [43%] and (Estonian, Finnish) [59.9%]. Even so, only 30.9% of the Tajik
corpus was in-vocabulary of Farsi. The H1 approach for (Farsi, Tajiki) as well had an accuracy of 87.5%, which was higher than for any other considered language pair, including close languages such as (Dutch, German) [82/1%] and (Estonian, Finnish) [83.8%]. We note that even though H1 is so high, at 87.5%, it is still not as high as D2, which is a Tajiki tagger tagging Tajiki, at 98.2%.

Here is the table with results for our novel HD-ification approaches, as they worked for (Farsi, Tajiki), juxtaposed with the results of two other close language pairs. The first row is the baseline and the numbers in subsequent rows represent the delta from that baseline:

<table>
<thead>
<tr>
<th>HD-ification Approaches</th>
<th>(Dutch, German)</th>
<th>(Estonian, Finnish)</th>
<th>(Farsi, Tajiki)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 3rd party tagger</td>
<td>43.5%</td>
<td>59.5%</td>
<td>61.2%</td>
</tr>
<tr>
<td>L1 – Hdify – frequent word replacement</td>
<td>25.8%</td>
<td>6.8%</td>
<td>20.3%</td>
</tr>
<tr>
<td>L2 – Hdify – affix replacement</td>
<td>1.2%</td>
<td>1.1%</td>
<td>7.2%</td>
</tr>
<tr>
<td>L3 – Hdify – exemplar replacement</td>
<td>3.1%</td>
<td>1.8%</td>
<td>7.6%</td>
</tr>
<tr>
<td>L4 – Hdify</td>
<td>26.4%</td>
<td>-4.2%</td>
<td>17.4%</td>
</tr>
</tbody>
</table>

Table 9: (Farsi, Tajiki) results, HD-ification, compared with other close languages

In general, all four of the HD-ification approaches performed quite well, something that was not the case for other language pairs. It is true that for L1 and L4, that delta for (Dutch, German) is larger than the delta for (Farsi, Tajiki). However, that is in large part because the baseline for (Dutch, German) performed poorly, comparatively. If we consider the overall accuracy for (Farsi, Tajiki), we see that after L1 was 81.5% and after L4 was 78.6%, much higher than the overall accuracy for (Dutch, German). This makes sense
because L1 is gloss replacement of frequent words and, because of the large shared vocabulary, many of these Tajiki words were already in-vocabulary. The same is true for L4, which brings words in-vocabulary via cognition detection. A good amount of these cognates were actually absolutely identical and were thus in-vocabulary, and so while the deltas for L1 and L4 was indeed quite impressive, they still did not have as much of an impact as for a language pair in which there was less of a shared vocabulary.

In the general case of language pairs, L2 and L3 HD-ify, which are affix-based strategies, did not show much improvement. However, there was indeed significant improvement for these affix-based approaches for (Farsi, Tajik). This is partly because besides the regular affix interchanges we might find between any close languages, these affix replacements also supplement deficiencies in the simple transliteration scheme, which would bring many slight differences in spelling back into alignment. For instance, the Romanized Tajiki suffix -oti commonly corresponds to the Romanized Farsi suffix –at. That is, in many words, the Tajiki ى, as the izofat, Romanized as i, did not appear in the Romanized Farsi. And the Romanized Tajiki letter o, from the Tajiki Cyrillic ơ, in such contexts often was written in Farsi as one of the letters which corresponded with Romanized a. For another example, the Romanized Tajiki prefix mo- commonly corresponds to the Romanized Farsi prefix m-. That is, the o in Tajiki is written as a diacritic that does not regularly appear in this context in Farsi. By effectively removing the o in this context, we move a large number of Tajiki words into-vocabulary.

In retrospect, the affix replacement could have performed even better. When discovering the Tajiki prefixes, we fed a relatively small corpus into the Automorphology program, and so only 4 Tajiki prefixes and 35 Tajiki suffixes were discovered. Analysis of
a larger corpus may have yielded many additional affixes, and the regular Farsi replacements
would have helped bridge the transliteration gap even more.

Furthermore, this only addresses missing or mismatched vowel letters when they
appear near the beginning or end of word. If the missing vowels appear in the center of the
word, then this affix-based approach would be to no help.

If we wanted to develop a general morphology-based approach for
language-ification, which would also, as it happens, address these missing vowels in the
middle of the words, then perhaps the following would have been more successful: to
determine, either via careful linguistic analysis or by unsupervised detection methods such
as in Automorphology, the consonant-verb pattern in use through the entire word. For
example, the Romanized Tajiki word manzili could have the pattern CaCCiCi, or if certain
consonants (say the initial m) serve morphological purposes and often repeat across words,
those letters could be part of the pattern, such as maCCiCi. And then, when the
corresponding Farsi word, mnzl, is found to be mCCC, then that could serve as a
language-ification rule.

It might also be instructive to see just what percentage of the corpus is in-vocabulary.
This measure is of recognized words in corpus / total size of corpus, meaning that repeated
words count for both the numerator and the denominator. A word being in-vocabulary does
not necessarily mean that it is being recognized correctly, just that the word in its current
form is recognized by the tagger.
The 61.2% baseline is that high even with only 30.9% of words in vocabulary. The L1 approach targets the most frequently occurring words and perhaps functionally important words, so even though only 55.3% of the corpus are in vocabulary, it has a greater delta than L4, for which 87.5% of words are in-vocabulary. Additionally, these words replaced via L1 are given by a human expert, so these are in most cases the correct in-vocabulary words, while the words replaced by L4 are given by a computer program, and so the replacement is noisy, and the new in-vocabulary words might in many cases be incorrect. L2 brings into-vocabulary a small number of words via affix replacement (only 0.2%), but that is sufficient for a significant improvement in tagging accuracy. L3 has 100% words in-vocabulary only in the trivial sense, in that an exemplar is suggested for (almost) every unknown word, from a best-guess based on affixes.

Here is the table with results for our novel LD-ification approaches as they worked for (Farsi, Tajiki). The first row is the baseline and the numbers in subsequent rows represent the delta from that baseline. These results are coupled with the percentage of words in the corpus which were in-vocabulary, for each approach:

<table>
<thead>
<tr>
<th>HD-ification Approaches</th>
<th>(Farsi, Tajiki)</th>
<th>In vocabulary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 3rd party tagger</td>
<td>61.2%</td>
<td>30.9%</td>
</tr>
<tr>
<td>L1 – Hdify – frequent word replacement</td>
<td>20.3%</td>
<td>55.3%</td>
</tr>
<tr>
<td>L2 – Hdify – affix replacement</td>
<td>7.2%</td>
<td>31.1%</td>
</tr>
<tr>
<td>L3 – Hdify – exemplar replacement</td>
<td>7.6%</td>
<td>100.0%</td>
</tr>
<tr>
<td>L4 – Hdify</td>
<td>17.4%</td>
<td>87.5%</td>
</tr>
</tbody>
</table>

Table 10: Tajiki Results, HD-ification
Here are some interesting points regarding the relative success of approaches. Firstly, of the three approaches, the comparative success as measured as delta for the baseline L1 > L4 > L2. This was the case for both HD-ify approaches and LD-ify approaches. This can be understood as frequent human-expert crafter replacements > comprehensive but automatic whole word cognate replacement > automatic partial word replacement.

<table>
<thead>
<tr>
<th>LD-ification Approaches</th>
<th>Farsi / Tajiki</th>
<th>In-vocabulary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline 3rd party tagger</td>
<td>67.5%</td>
<td>30.9%</td>
</tr>
<tr>
<td>L1 – Ldify – frequent replacement</td>
<td>8.9%</td>
<td>39.3%</td>
</tr>
<tr>
<td>L2 – Ldify – affix replacement</td>
<td>4.8%</td>
<td>38.1%</td>
</tr>
<tr>
<td>L4 – Ldify – cognate replacement</td>
<td>8.8%</td>
<td>78.8%</td>
</tr>
</tbody>
</table>

Table 11: Tajiki results, LD-ification

In terms of which to prefer, LD-ify or HD-ify, as in the general language pair case, it again seems that HD-ify had greater success. We should acknowledge, though, that the
LD-ify baseline is somewhat higher than the HD-ify baseline, such that the overall accuracy HD-ify and LD-ify is closer than may appear.

7.5 Predictions

Given that Farsi and Tajiki are extremely close languages – indeed, both are dialects of Persian – Farsi is an obvious language choice for the HD from which to project. Despite the difficulties that arise from cross-orthographic projection, one would not expect Russian (also written in a Cyrillic script, and sharing a bit of vocabulary) to produce better results than Farsi. Similarly, if Dari (Afghani Persian, written in a Persian-Arabic script) were the LD, then Tajiki as an HD (if it were high-density) would likely still be a better match than Arabic. Therefore, Bob does not really need to think deeply about language features, or refer to many of the predictions to make his choice of language from which to project. Still, it may be useful to consider if, and how, these predictions played out in the case of (Farsi, Tajiki). Further, it may suggest useful refinements to the basic language-ification approaches based on the particular features of Farsi and Tajiki. Also, if a prediction speaks to how one strategy (HD-ify, LD-ify, L1, L4, etc.) may perform, then Bob may indeed find it useful.

7.5.1 Prediction [P1a, b]

You may recall that our first prediction was that simple application of an HD tagger to LD text (D1 strategy) would perform best for pairs of languages with phylogenetic or historical cultural connections. In this case, Farsi and Tajiki have a rather strong phylogenetic connection, more so than any other language pair, even the extremely close ones. As a result, we would expect that D1 on (Farsi, Tajiki) > D1 on (Dutch, German), and
certainly that D1 on (Farsi, Tajiki) > D1 on (French, German). In fact, this is the case, as is illustrated by the following chart:

![Chart showing D1 results for Farsi/Tajiki, Dutch/German, and French/German]

**Figure 49: Tajiki results, prediction 1**

### 7.5.2 Prediction [P2a, b, c]

As you may recall, the second prediction was that low-density NLP projection approaches based on cognate-identification techniques (H2 and L4 HD-ify) would perform best for pairs of languages with phylogenetic or historical cultural connections. Such language pairs could be expected to have more true cognates, and so cognate-identification replacement would be more common and more accurate.

As a result, we would expect that H2 on (Farsi, Tajiki) > H2 on (Finnish, German). In fact, this is the case, as is illustrated by the following chart:
We would further expect that the delta of HD-ify L4 on (Farsi, Tajiki) > delta of HD-ify L4 on (Finnish, German). In fact, this is the case, as is illustrated by the following chart:

In a previous chapter, we had compared (Dutch, German) and (German, Dutch), as close languages, with those of (Finnish, German). Since Farsi and Tajiki are even closer, we might expect the delta for (Farsi, Tajiki) to be even greater. In fact, the situation is a bit more complicated than that, as the following chart illustrates:
As discussed above, this is because, even though cognation is working in each of these language pairs to bring words into-vocabulary, for (Farsi, Tajiki), a good deal of the cognates are identical and thus are already in-vocabulary. Still, we do say that cognation replacement is working, and indeed quite well, for this language pair.

7.5.3 Prediction \([P3a, b, c, d]\)

As you may recall, the third prediction was that Low-density NLP projection approaches based on simple adoption of the HD contextual model (H1) will be most effective for pairs of languages that are closest in their fundamental word order (e.g., SVO, SOV, etc.).

Tajiki and Farsi both have SOV word order. Meanwhile, while English and German are both SVO, in the subordinate clause German has SOV while English has SVO. As a result, we would expect that H1 on (Farsi, Tajiki) > H1 on (English, German). In fact, this is the case, as is illustrated by the following chart:
7.5.4 Prediction \([P4a, \ b]\)

As you may recall, the fourth prediction was that language pairs which share noun-adjective order will benefit more from wholesale adoption of the HD contextual model (H1, D1), and will thus have greater accuracy than language pairs which don’t.

In Farsi, adjectives are generally postmodifiers. Where they come before the noun, they are joined by an enclitic unstressed vowel, the izofat. Recall that this izofat is generally not written in Farsi. An examination of the gold standard Tajiki corpus reveals the same pattern of nouns and adjectives, which is readily apparent since the izofat is written. Meanwhile, French and German differ in their noun-adjective order. Therefore, we would expect that H1 on (Farsi, Tajiki) > H1 on (French, German) and that D1 on (Farsi, Tajiki) > D1 on (French, German). In fact, this is the case, as is illustrated by the following chart:
7.5.5 Prediction \([P5a, b, c, d]\)

As you may recall, the fifth prediction was that language pairs in which both languages have articles would benefit more from gloss replacement of frequent words (L1 LD-ify, L1 HD-ify) than would languages pairs in which only one language, or both languages, lack articles.

Our example of a language pair in which both languages make regular use of articles was (Dutch, German). Meanwhile, Farsi and Tajiki lack a definite article, and the indefinite article occurs as an enclitic attached to the noun or adjective, rather than as a separate word. As such, from the perspective of the contextual and lexical model, and for any whole-word-based language-ification effort, these articles don’t exist.

As a result, we would expect that delta L1 LD-ify on (Dutch, German) > delta L1 LD-ify on (Farsi, Tajiki), and also that delta L1 HD-ify on (Dutch, German) > delta H1 LD-ify on (Farsi, Tajiki). In fact, this is the case, as is illustrated by the following chart:
As you may recall, the sixth prediction had to do with the effects of divergent granularity of articles on LD-ification depending on whether the source or target language of a language-ification had a richer or poorer level of articles.

This is then not relevant to the (Farsi, Tajiki) language pair. As just discussed, Farsi and Tajiki have no definite article and the indefinite article is enclitic, attached to the noun or adjective, rather than standing as a separate word. The granularity of both is the same (or, non-existent), and so no meaningful predictions can be made here for LD-ification, with either of the languages as source or target.

As you may recall, the seventh prediction was that approaches involving wholesale adoption of the HD contextual model (H1) will be most effective for pairs of languages in
which both are PRO-drop or both are not PRO-drop, but that if there is a divergence in this language feature, then such an approach will not be as successful.

Both Farsi and Tajiki are PRO-drop languages. In contrast, Finnish is PRO-drop while German is not. Therefore, we might expect that H1 on (Farsi, Tajiki) > H1 on (Finnish, German). In fact, this is the case, as is illustrated by the following chart:

![Figure 56: Tajiki results, prediction 7](image)

7.5.8 Prediction [P8, P9]

As you may recall, our eighth prediction was that, assuming a difference in level of inflection between the HD and the LD languages, LD-ification approaches would be more successful if the HD inflection level was greater than the LD inflection level. And the ninth prediction was that, assuming a difference in level of inflection between the HD and the LD languages, HD-ification approaches would be more successful if the HD inflection level was
lower than the LD inflection level. (That is, in general, for both HD-ify and LD-ify, it is better if the language-ification is in the direction of the lower level of inflection.)

Since there is no difference in level of inflection between Farsi and Tajiki, neither of these predictions were relevant.

### 7.6 Frequent word profile for Farsi and Tajiki

Here is the frequent-word profile of Tajiki.

<table>
<thead>
<tr>
<th>Tajiki</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
<th>Farsi</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>8</td>
<td>13</td>
<td>15</td>
<td>21</td>
<td>28</td>
<td>ADJECTIVE</td>
<td>4</td>
<td>7</td>
<td>14</td>
<td>21</td>
<td>25</td>
</tr>
<tr>
<td>ADVERB</td>
<td>3</td>
<td>5</td>
<td>7</td>
<td>11</td>
<td>12</td>
<td>ADVERB</td>
<td>2</td>
<td>2</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>ARTICLE</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>CONJUNCTION</td>
<td>3</td>
<td>6</td>
<td>6</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>NOUN</td>
<td>20</td>
<td>46</td>
<td>73</td>
<td>95</td>
<td>126</td>
<td>NOUN</td>
<td>12</td>
<td>33</td>
<td>57</td>
<td>89</td>
<td>123</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>7</td>
<td>9</td>
<td>13</td>
<td>17</td>
<td>18</td>
<td>PREPOSITION</td>
<td>9</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>5</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>13</td>
<td>PRONOUN</td>
<td>7</td>
<td>11</td>
<td>12</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>VERB</td>
<td>10</td>
<td>21</td>
<td>35</td>
<td>43</td>
<td>45</td>
<td>VERB</td>
<td>10</td>
<td>22</td>
<td>31</td>
<td>35</td>
<td>37</td>
</tr>
</tbody>
</table>

Figure 57 Tajiki frequent word profile (left) and Farsi frequent word profile (right)

As we consider the 50, 100, etc., most frequent words of the language, how many of them are nouns, adjectives, adverbs, and so on? Similarity of such profiles between languages in a language pair reveals something about the similarity between contextual and lexical models, and well as might help us intuit just how many of the more important words are added, at which stage, which in turn might indicate the amount of effort Bob should put in towards building a frequent-word dictionary.

If you turn your head sideways, the bar graphs in each column (50 most frequent words, 100, etc.) correspond to the distribution of tags for those frequent words. The pattern
of relative tag frequency emerges early, and stays more or less constant. If we compare this frequent-word profile to that of Farsi, we can see that its profile is quite similar to that of Tajiki.

Finally, we consider the results of replacing that number of Tajiki words for their Farsi equivalents, and see that even at 50 words, the slope has more or less flattened out.

![Graph showing the comparison of Farsi and Tajiki frequent word replacement](image)

**Figure 58: Tajiki frequent word replacement, 50...250**

The main impact of the L1 HD-ification happened with just a 50 word replacement. Even if Bob does not wish to invest much in building an L1 dictionary, a small dictionary of just 50 words would have helped.
Chapter 8: Conclusion

In this chapter we take stock of what we have learned from this study and this series of experiments. Earlier in this dissertation, we introduced a hypothetical computational linguist “Bob” who needed to implement an NLP tool, as quickly as possible, under very tight resource constraints. If Bob were to read this dissertation, what practical advice would he come away with? If a researcher wanted to continue along the path set out by this study, what has already been established? What are the contributions of this work, both in terms of ideas and in terms of practical resources?

8.1 Practical advice

In terms of practical advice that Bob can take away from this, first and foremost, there is the fact that L1 HD-ify is surprisingly effective. In many cases this is true even when replacing only a very small number of words; this is especially notable in that Bob does not need to invest much time, effort, or resources to assembling this. Whatever other projection approaches Bob uses, he should consider incorporating L1 HD-ify alongside it.

Secondly, if Bob wants to make use of cognition to project linguistic resources from the HD to the LD, then a simple cognate-identification algorithm, such as the one we used in this dissertation, will likely not suffice. The identification algorithm, even with the threshold in place, introduced a lot of false cognates. This noise actually led to a reduction in accuracy from H1 to H2, because of the introduction of noise. And L4 approaches sometimes worked quite well, but not consistently: the significant improvements were seen
in L4 HD-ify for (Dutch, German), L4 LD-ify for (German, Dutch), and L4 LD-ify and HD-ify for (Farsi, Tajiki).

Based on this pattern, it seems that for this simplistic cognation projection to work, the languages need to be closely related to one another in surface form. Also, it seems that inflection and overloaded or close morphology can easily derail the cognate detection algorithm. Finally, cognate projection in the direction of the language with less inflection is more successful.

Bob’s primary goals were to accomplish as much as possible while utilizing very limited resources: if Bob might need to implement NLP tools for multiple low-density languages, this would imply that he should not spend a lot of time implementing techniques that were not language-pair specific. However, if Bob wishes to make use of cognate projection techniques, he will likely need to obtain a better list of cognates. He might hire a language expert to produce such a list, or to correct a computer-generated list. Alternatively, he might start with a cognate list and train a cognation model upon it.

Thirdly, Bob should note that simplistic affix-based projection techniques are not successful in the general case for either LD-ification or HD-ification. While there were a few successes (see e.g. L2 LD-ify of (Estonian, Finnish) and (German, Dutch)), in the general there was little improvement. What seems necessary for the simplistic affix replacement to work is the existence of regular and unambiguous affix correspondences, which often did not exist between languages in the language pair.

Our unlikely hope that L2 HD-ification would prove somewhat effective, using simple affix replacement to hit upon cognates, turned out to be overly optimistic. While it
worked in the case of (Farsi, Tajiki), those were very close languages. Perhaps a less simplistic affix-replacement approach might perform better in the general case.

Fourthly, if Bob has to choose between HD-ify and LD-ify, he should prefer the HD-ify approach, which generally performed better.

In addition to those high-level “take away” lessons that Bob can learn from this dissertation, the results chapter also contained several linguistically-driven predictions which might prove useful to Bob, and so we will summarize and reflect upon them here. When setting out to project from an HD language to particular LD language, Bob can theoretically select from a large number of HD languages, and he can select from a number of different projection approaches. Some basic linguistic analysis of the LD language and the candidate HD languages can help guide him in these selections.

Thus according to prediction [P1], pairs of languages with phylogenetic or historical cultural connections perform better for D1 (simple application of the HD tagger to the LD language). So, as one would intuit, if Bob wants to tag Yiddish, he would be better off using a German tagger than an English tagger, and he should prefer it, even if he is less familiar with German or will have greater difficulties obtaining a tagger.

It is likely, though, that he will want to improve over this baseline tagger with various language-ification approaches, and we should point out that depending the particular language features and the particular approach, the delta from baseline could conceivably compensate from this starting at a lower accuracy level. (For instance, practically, if the LD and HD languages are very close but due to independent development, the HD differs by lacking articles, a more distant HD language could perform better after L1 HD-ify.)
Further according to prediction [P2], cognate-based language-ification approaches work best on such language pairs with phylogenetic or historical cultural connections; so, if Bob has such a language pair, it makes sense to consider such an approach. And if he is considering such an approach, he would do well to select an appropriate language pair.

According to [P3], linguistic features which impact word order or phrase order will impact the efficacy of projecting the HD contextual model. Thus, Bob might consider constituent word order (SOV, SVO, etc.) of the languages in his selected language pair, especially if he is using a strategy such as H1, which adopts wholesale the HD contextual model. So too [P4], he might consider whether adjectives are premodifiers or postmodifiers of nouns, as this similarity or dissimilarity would be reflected in the contextual models of the two languages. This would be noticed when employing such approaches as H1 or D1, and would be a contributing factor for other projection approaches as well (though other features such as similar morphological patterns or shared vocabulary might have greater impact in those cases). So too [P7], he might consider whether both of his selected languages are PRO-drop, as similarity in this feature will yield better results in strategies like H1.

According to [P5], gloss replacement of frequent words – L1 – is especially effective where articles are present in both languages of the pair. Thus, if a language lacks or rarely employs articles, Bob should know this beforehand, and this might steer him towards a different approach. This is especially true if the size of his frequent word wordlist is small, such as 50 words. If Bob uses the L1 approach for such a language pair, then he should invest in assembling a larger wordlist. Conversely, if both languages in the pair have articles, then Bob might expect some dramatic results even with a small wordlist, or a wordlist consisting of just the articles.
Following prediction [P6], languages can differ in the lexical granularity of their articles, and this can get in the way of LD-ification approaches (particularly L1, but conceivably others as well). If Bob has selected an HD with a lower lexical granularity than the LD, then he should either favor the HD-ify approach, or take steps to work around this issue – for example, by creating a dictionary which is a multimap and iterating through the various LD options when replacing articles in the HD corpus; or, by exploiting language-specific linguistic knowledge to select the appropriate article based on context. He might consider the same for other parts of speech as well (e.g. pronouns), where there is a difference in lexical granularity between the two languages.

According to predictions [P8] and [P9], if two languages differ in their level of inflection, language-ification in the direction of the poorly inflected language is more successful than in the direction of the richly inflected language. Thus, if Bob has already selected his language pair, he might use this to choose whether HD-ify or LD-ify is the right approach.

Finally, as the Tajiki case study illustrated, even where there are orthographic distinctions between the two languages (e.g. Farsi and Tajiki, Tajiki and Bukhori, German and Yiddish, or Arabic and Malti), simple transliteration approaches can be effective, with the simple language-ification approaches helping to bridge the rest of the way.

8.2 Future work

If Margaret, a computational linguist, were to read this dissertation, and wanted to continue along the same lines, how might she do so? We note a few areas of future research that are ripe for further study. Firstly, in this dissertation, we generated the frequent word profiles for each language, showing the distribution of parts of speech for the most frequent
(50, 100, 150, 200, 250) words – these may be found in appendix B – and suspect that they might prove useful in predicting how well L1 approaches could work, ahead of time. It is still unclear precisely how, but future research might reveal some useful patterns.

Secondly, it would be interesting to see how well cognate-based language-ification approaches work with a really good list of cognates, as opposed to the noisy list which we generated. As mentioned earlier, simplistic approaches such as minimum-edit-distance with a threshold do identify cognates, but also misidentify words as cognates, and are readily derailed by inflection and overloaded or close morphology. Thus, a better approach might be to use a human-generated list, or use a human to double-check the computer-generated list.

Alternatively, Margaret could obtain a small list of cognates and use it to train a good cognition model. A simple one might be a transliteration model (similar to a translation model and based on source letter, destination letter, and context). Based on the particular weaknesses we observed in our simplistic cognition detection, across several language pairs, we might point out specific places where this might cognition model might be supplemented. For instance, if an HD language uses an –s suffix for both noun plurals and present tense verbs, but the LD language uses different suffixes for each, the cognition model might confuse the two. However, on the HD language side of the training cognate list, it is reasonable to have the part(s)-of-speech. And later, when performing cognate detection or generation, it is reasonable to know the HD part of speech. Training separate cognition models for each part-of-speech might then be more effective than training a single cognition model for the entire language pair.
Thirdly, cognation detection often worked well for the less frequent words – e.g. long adjectives and nouns, where the core of the word was basically the same but there were morphological or inflectional differences, or slight differences in vowels or consonants. Many of the important and frequent function words, even where cognates, were not identified correctly, because they were short words and there were other similar words to which they match. Since cognition dictionaries and frequent word dictionaries target different set of words, it would be interesting to explore the using a frequent word dictionary first, and only using the cognition dictionary for the remainder of the words.

Fourthly, there are a few ways we could improve the affix replacement approach, without relying too much on LD linguistic knowledge. The Automorphology program was good at identifying, in an unsupervised manner, many of the prefixes and suffixes of a single language based on an untagged corpus. There was, of course, noise, such as parts of the stem incorporated as part of the affix. For instance, rather than pre-, there might be pred- and prec-. However, we actually have (on the HD side) the parts-of-speech, and this classification might be harnessed to better identify the prefixes. Also, perhaps using a small corpus of the language pair (either as an aligned corpus, aligned wordlist, or two unaligned corpora), we could better identify these affixes or even the appropriate affix replacements.

Fifthly, there are other language-ification approaches which we have not explored. For instance, we discussed whether adjectives were premodifiers or postmodifiers. One possible, though somewhat language-specific, LD-ification could be based on reordering of adjectives and nouns in a tagged corpus. If these sorts of differences in the contextual model could be detected via supervised learning (on a small tagged LD corpus and a small tagged
HD corpus) and then automatically applied, then it might be a useful approach of interest to Bob.

Another possible language-ification could be splitting off, or joining up, clitics (e.g. the direct object marker *ro*, which is a suffix in Tajiki and a stand-alone-word in Farsi). In this study, we kept word-boundaries as sacrosanct. We would need to consider how language-specific this sort of language-ification is.

Sixthly, while we believe that our results were quite interesting and indicative of how various strategies might perform given linguistic features, we realize that our study was by no means comprehensive. We have, after all, only considered seven language pairs. Future research, which would apply the approaches to many more languages, would add a greater level of confidence to our results.

Finally, it would be interesting to explore how these approaches carry over to other computational linguistic tasks. We had considered part-of-speech tagging (and, in the case of Tajiki, transliteration and part-of-speech tagging). But this was intended as an example task. It would be of interest to see explore how these approaches fare in machine-translation, or in named entity recognition.

### 8.3 Contributions

Besides what is discussed in the previous sections, what are the contributions of this work, both in terms of ideas and in terms of practical resources?

First, there is the idea of investigating an approach across multiple language pairs, in order to investigate how an approach performs in the general case, rather than the particular language-pair selected by the researcher. Often a language pair is selected because of a
unique feature it possesses, and so the approach might not carry over to another language pair of interest to Bob or Margaret.

Along similar lines, there is the actual comparison, side-by-side, of existing approaches, on the same set of languages and the same corpora, such that they can be more readily assessed in terms of performance and effort expended to implement. This way, one can select an approach to implement or extend when faced with the same challenge for a new LD language.

Second, there is the fact that in general, L1 HD-ify has a dramatic impact on accuracy over the D1 approach, and generating these frequent word dictionaries is a relatively non-intensive task. If so, perhaps other researchers should use this as a baseline, when considering some other novel approach.

Thirdly, there is the idea of language-ifications, in the direction of the HD or in the direction of the LD, and that deliberately simple and language-ification approaches can aid in projecting resources or taggers from one language to another. Relatedly, what sort of language-ifications work, across different language pairs.

Fourthly, there is the actual code and testbed we have developed, by which researchers can quickly consider some new language pair or approach, and compare the results against other language pairs and approaches in a systematic manner. Much of this code is written in Python, with hooks into the Natural Language Toolkit (NLTK). Also, the output of much are our code consists of tagged corpora, which can be readily fed as input into 3rd party programs, such as TreeTagger. This code will be provided upon request.

Fifthly, there is the actual taggers developed. In the case of HD-ification, this entails language-ification of the test corpus and running through an existing HD tagger; in the case
of LD-ification, there are actual trained TreeTaggers for each of the LD languages. Admittedly, most of these taggers are for so-called LD languages which are really HD, such that anyone wishing to actually tag the language would instead turn to an existing HD tagger. However, these produced taggers might be useful for the sake of comparison with other projection approaches. They will be available for download at the LATLab website\textsuperscript{20}.

In a few cases, though, the taggers we built would indeed be quite useful for other researchers. When looking to tag Finnish, we discovered that there was no TreeTagger available. It took a great deal of effort and aggravation to get it working properly, but in the end, we successfully produced a TreeTagger for Finnish, based on an extremely large tagged Finnish corpus, namely Finnish Treebank. This Finnish TreeTagger now resides on the TreeTagger website, so that others who wish to tag Finnish, or to tag the Europarl corpus to compare results with that of other languages in that parallel language corpus, can simply download the parameter file and do so. Likewise, the Farsi TreeTagger that we trained upon the Bijankhan corpus may be useful for someone who wishes to tag Farsi.

Sixthly, Tajiki is a genuine low-density language, we assembled a number of linguistic resources which could prove useful for someone performing computational linguistics tasks on this language. These resources include the Romanization dictionaries for Tajiki and Farsi, which allow the languages to meet in the middle; also, the affix dictionaries and cognate dictionaries mapping between these two languages. These resources will also be available for download at the LATLab website.

Additionally, there are the three linguistic resources assembled by the Farsi and Tajiki language expert: (a) the tagged Tajiki gold standard corpus, with 14,132 tokens; (b)

\textsuperscript{20} http://latlab.cs.qc.cuny.edu/
the frequent word dictionary for Tajiki, which maps Tajiki words to their Farsi and English equivalents, and lists their part-of-speech; and (c) the frequent word dictionary for Farsi, which maps Farsi words to their Tajiki and English equivalents, and lists their part of speech. Because so many of the words in (a) and (b) were cognates of one another, besides being useful for L1 projection, these resources might prove useful for the training of transliteration and cognition models. Further, because these frequent-word dictionaries contain the English equivalent, these resources open up the possibility of projection using English as either the HD language or, perhaps more usefully, as a pivot.

Finally, there are the actual Tajiki taggers we produced. There is a pure Tajiki tagger, trained on our somewhat limited Tajiki corpus. Because the corpus size was so small, there may be many genuine Tajiki words which would be out of vocabulary (though TreeTagger does consider the affixes as well). However, there are also the many projected Farsi \(\rightarrow\) Tajiki taggers we produced, which have a broader Persian-based vocabulary and which perform fairly well in the general case.

### 9.4 Summation

In sum, in this dissertation research, we have considered and implemented approaches to low-density NLP that make use of cross-lingual projection techniques, under strict resource constraints. Rather than selecting a particular language or language pair as single exemplar for evaluation, we selected a number of (HD, LD) language pairs, in order to perform a systematic comparison. We selected those particular (HD, LD) language pairs on the basis of interesting linguistic features, and explored how those features might predict the relative success or failure of given linguistic-projection strategies. Finally, because our selected LD languages were really HD languages, we conducted a case study on an actual
LD language, Tajiki, projecting from its close HD relative, Farsi. From our perspective, several of our results look promising, and we hope that they can advance the field of low-density NLP.
Appendix A

This appendix contains the part-of-speech tags for the Bijankhan corpus, specifically the one with a tagset granularity of 40. It is copied directly from the webpage\(^1\), and is provided here to provide the reader with a sense of just what information is and is not available in this tagged Farsi corpus.

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJ</td>
<td>Adjective, General</td>
</tr>
<tr>
<td>ADJ_CMPR</td>
<td>Adjective, Comparative</td>
</tr>
<tr>
<td>ADJ_INO</td>
<td>Past Participle</td>
</tr>
<tr>
<td>ADJ_ORD</td>
<td>Adjective, Ordinal</td>
</tr>
<tr>
<td>ADJ_SIM</td>
<td>Adjective, Simple</td>
</tr>
<tr>
<td>ADJ_SUP</td>
<td>Adjective, Superlative</td>
</tr>
<tr>
<td>ADV</td>
<td>Adverb, General</td>
</tr>
<tr>
<td>ADV_EXM</td>
<td>Adverb, Exemplar</td>
</tr>
<tr>
<td>ADV_I</td>
<td>Adverb, Question</td>
</tr>
<tr>
<td>ADV_NEGG</td>
<td>Adverb, Negation</td>
</tr>
<tr>
<td>ADV_NI</td>
<td>Adverb, Not Question</td>
</tr>
<tr>
<td>ADV_TIME</td>
<td>Adverb, Time</td>
</tr>
<tr>
<td>AR</td>
<td>Arabic Word</td>
</tr>
</tbody>
</table>

\(^1\) [http://ece.ut.ac.ir/dbrg/bijankhan/Corpus/Bijankhan-tagset-description.txt](http://ece.ut.ac.ir/dbrg/bijankhan/Corpus/Bijankhan-tagset-description.txt)
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CON</td>
<td>Conjunction</td>
</tr>
<tr>
<td>DEFAULT</td>
<td>Default</td>
</tr>
<tr>
<td>DELM</td>
<td>Delimiter</td>
</tr>
<tr>
<td>DET</td>
<td>Determiner</td>
</tr>
<tr>
<td>IF</td>
<td>Conditional</td>
</tr>
<tr>
<td>INT</td>
<td>Interjection</td>
</tr>
<tr>
<td>MORP</td>
<td>Morpheme</td>
</tr>
<tr>
<td>MQUA</td>
<td>Modifier of Quantifier</td>
</tr>
<tr>
<td>MS</td>
<td>Mathematic Symbol</td>
</tr>
<tr>
<td>N_PL</td>
<td>Noun, Plural</td>
</tr>
<tr>
<td>N_SING</td>
<td>Noun, Singular</td>
</tr>
<tr>
<td>NN</td>
<td>Number</td>
</tr>
<tr>
<td>NP</td>
<td>Noun Phrase</td>
</tr>
<tr>
<td>OH</td>
<td>Oh Interjection (حرف تدا)</td>
</tr>
<tr>
<td>OHH</td>
<td>Oh noun (منادی)</td>
</tr>
<tr>
<td>P</td>
<td>Preposition</td>
</tr>
<tr>
<td>PP</td>
<td>Prepositional Phrase</td>
</tr>
<tr>
<td>PRO</td>
<td>Pronoun</td>
</tr>
<tr>
<td>PS</td>
<td>Pseudo-Sentence</td>
</tr>
<tr>
<td>QUA</td>
<td>Quantifier</td>
</tr>
<tr>
<td>SPEC</td>
<td>Specifier</td>
</tr>
<tr>
<td>V_AUX</td>
<td>Verb, Auxiliary</td>
</tr>
<tr>
<td>V_IMP</td>
<td>Verb, Imperative</td>
</tr>
<tr>
<td>V_PA</td>
<td>Verb, Past Tense</td>
</tr>
<tr>
<td>V_PRE</td>
<td>Verb, Predicative</td>
</tr>
<tr>
<td>V_PRS</td>
<td>Verb, Present Tense</td>
</tr>
<tr>
<td>V_SUB</td>
<td>Verb, Subjunctive</td>
</tr>
</tbody>
</table>
Appendix B

This appendix contains the frequent word profiles that were discussed at the tail end of chapter 6. For each of the seven languages we have studied, we have computed two different metrics, which describe the distribution of parts of speech within the most frequent 50 (or 100, 150, 200, 250) words in that language’s corpus.

The first metric considers the 50 most frequent words and their tags, and counts how many of those tags are NOUN, ADJECTIVE, and so on. This count is unweighted, meaning that if the word “the” appears one million times and the word “and” appears only a quarter of a million times, then it still counts as a single DETERMINER and a single CONJUNCTION. For the sake of illustration, we present again the chart for Dutch:

<table>
<thead>
<tr>
<th>Dutch</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>4</td>
<td>17</td>
<td>27</td>
<td>40</td>
<td>46</td>
</tr>
<tr>
<td>ADVERB</td>
<td>13</td>
<td>30</td>
<td>43</td>
<td>57</td>
<td>69</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>8</td>
<td>15</td>
<td>20</td>
<td>20</td>
<td>23</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0</td>
<td>3</td>
<td>5</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>NOUN</td>
<td>15</td>
<td>33</td>
<td>62</td>
<td>79</td>
<td>105</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>14</td>
<td>20</td>
<td>23</td>
<td>27</td>
<td>31</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>15</td>
<td>29</td>
<td>32</td>
<td>36</td>
<td>39</td>
</tr>
<tr>
<td>VERB</td>
<td>9</td>
<td>15</td>
<td>23</td>
<td>33</td>
<td>42</td>
</tr>
</tbody>
</table>

Figure 59 Frequent word profile for Dutch
If you turn your head sideways, the bar graphs in each column (50 most frequent words, 100, etc.) correspond to the distribution of tags for those frequent words. Considering only the 50 most frequent words, only 5 of those words are articles, compared with 15 pronouns. This stays the case even when we consider the 150 most frequent words – 5 are articles, compared with 62 pronouns. For the most part, the general form of these bar graphs in this particular remains the same, though nouns do quickly outpace pronouns.

The second metric we compute is the weighted distribution. When looking at the actual corpus, now that we are considering these 50 words, what percentage of the corpus is covered by the 50 most frequent NOUNS, ADJECTIVES, and so on? Again for the sake of illustration, we present the results for Dutch:

<table>
<thead>
<tr>
<th>Dutch</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>0%</td>
<td>2%</td>
<td>3%</td>
<td>4%</td>
<td>4%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>7%</td>
<td>9%</td>
<td>9%</td>
<td>9%</td>
<td>10%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>25%</td>
<td>21%</td>
<td>20%</td>
<td>19%</td>
<td>18%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>17%</td>
<td>15%</td>
<td>14%</td>
<td>14%</td>
<td>13%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>5%</td>
<td>9%</td>
<td>12%</td>
<td>12%</td>
<td>14%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>23%</td>
<td>22%</td>
<td>20%</td>
<td>20%</td>
<td>19%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>20%</td>
<td>20%</td>
<td>19%</td>
<td>19%</td>
<td>18%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>9%</td>
<td>9%</td>
<td>10%</td>
<td>10%</td>
<td>11%</td>
</tr>
</tbody>
</table>

Figure 60 Frequent word distribution for Dutch

Based on this chart, we can readily see that even though, when considering 50 frequent words, there were only 5 articles, they covered 25% of the corpus.

The charts for the remaining six languages are presented here without further comment.
German:

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>German</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADJECTIVE</td>
<td>3</td>
<td>11</td>
<td>17</td>
<td>24</td>
<td>31</td>
</tr>
<tr>
<td>ADVERB</td>
<td>10</td>
<td>24</td>
<td>35</td>
<td>46</td>
<td>57</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>8</td>
<td>8</td>
<td>9</td>
<td>9</td>
<td>9</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>4</td>
<td>14</td>
<td>19</td>
<td>22</td>
<td>25</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0</td>
<td>2</td>
<td>4</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>NOUN</td>
<td>6</td>
<td>18</td>
<td>30</td>
<td>52</td>
<td>79</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>16</td>
<td>21</td>
<td>25</td>
<td>29</td>
<td>32</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>18</td>
<td>31</td>
<td>40</td>
<td>48</td>
<td>54</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>VERB</td>
<td>10</td>
<td>16</td>
<td>27</td>
<td>34</td>
<td>39</td>
</tr>
</tbody>
</table>

Figure 61: Frequent word profile for German

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>German</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADJECTIVE</td>
<td>1%</td>
<td>1%</td>
<td>1%</td>
<td>2%</td>
<td>2%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>4%</td>
<td>5%</td>
<td>6%</td>
<td>7%</td>
<td>7%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>34%</td>
<td>28%</td>
<td>26%</td>
<td>25%</td>
<td>24%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>9%</td>
<td>11%</td>
<td>11%</td>
<td>11%</td>
<td>10%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>3%</td>
<td>5%</td>
<td>7%</td>
<td>9%</td>
<td>11%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>26%</td>
<td>24%</td>
<td>23%</td>
<td>22%</td>
<td>21%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>12%</td>
<td>14%</td>
<td>14%</td>
<td>14%</td>
<td>14%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>11%</td>
<td>11%</td>
<td>11%</td>
<td>11%</td>
<td>11%</td>
</tr>
</tbody>
</table>

Figure 62: Frequent word distribution for German
Finnish:

<table>
<thead>
<tr>
<th>Finnish</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>3</td>
<td>6</td>
<td>12</td>
<td>16</td>
<td>19</td>
</tr>
<tr>
<td>ADVERB</td>
<td>14</td>
<td>27</td>
<td>45</td>
<td>54</td>
<td>71</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>11</td>
<td>15</td>
<td>18</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>NOUN</td>
<td>7</td>
<td>16</td>
<td>26</td>
<td>39</td>
<td>53</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>3</td>
<td>7</td>
<td>15</td>
<td>22</td>
<td>29</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>10</td>
<td>23</td>
<td>35</td>
<td>45</td>
<td>48</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERB</td>
<td>5</td>
<td>17</td>
<td>24</td>
<td>32</td>
<td>40</td>
</tr>
</tbody>
</table>

Figure 63 Frequent word profile for Finnish

<table>
<thead>
<tr>
<th>Finnish</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>2%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>16%</td>
<td>16%</td>
<td>16%</td>
<td>16%</td>
<td>17%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>2%</td>
<td>1%</td>
<td>1%</td>
<td>2%</td>
<td>2%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>34%</td>
<td>29%</td>
<td>26%</td>
<td>24%</td>
<td>23%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>11%</td>
<td>12%</td>
<td>12%</td>
<td>13%</td>
<td>14%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>3%</td>
<td>4%</td>
<td>6%</td>
<td>6%</td>
<td>7%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>14%</td>
<td>16%</td>
<td>17%</td>
<td>18%</td>
<td>17%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>18%</td>
<td>19%</td>
<td>19%</td>
<td>19%</td>
<td>18%</td>
</tr>
</tbody>
</table>

Figure 64 Frequent word distribution for Finnish
Estonian:

<table>
<thead>
<tr>
<th>Category</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>7</td>
<td>14</td>
<td>18</td>
<td>21</td>
<td>29</td>
</tr>
<tr>
<td>ADVERB</td>
<td>14</td>
<td>23</td>
<td>38</td>
<td>54</td>
<td>64</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>7</td>
<td>10</td>
<td>15</td>
<td>16</td>
<td>16</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>1</td>
<td>1</td>
<td>4</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>NOUN</td>
<td>9</td>
<td>24</td>
<td>39</td>
<td>55</td>
<td>76</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>7</td>
<td>15</td>
<td>21</td>
<td>32</td>
<td>36</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>15</td>
<td>19</td>
<td>24</td>
<td>30</td>
<td>36</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERB</td>
<td>2</td>
<td>10</td>
<td>19</td>
<td>23</td>
<td>28</td>
</tr>
</tbody>
</table>

Figure 65 Frequent word profile for Estonian

<table>
<thead>
<tr>
<th>Category</th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>0%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
<td>4%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>10%</td>
<td>10%</td>
<td>12%</td>
<td>13%</td>
<td>13%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>1%</td>
<td>1%</td>
<td>1%</td>
<td>1%</td>
<td>1%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>25%</td>
<td>21%</td>
<td>20%</td>
<td>18%</td>
<td>17%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>11%</td>
<td>15%</td>
<td>16%</td>
<td>17%</td>
<td>19%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>6%</td>
<td>8%</td>
<td>8%</td>
<td>9%</td>
<td>9%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>22%</td>
<td>20%</td>
<td>18%</td>
<td>18%</td>
<td>17%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>13%</td>
<td>13%</td>
<td>14%</td>
<td>13%</td>
<td>13%</td>
</tr>
</tbody>
</table>

Figure 66 Frequent word distribution for Estonian
### English:

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>5</td>
<td>18</td>
<td>29</td>
<td>46</td>
<td>61</td>
</tr>
<tr>
<td>ADVERB</td>
<td>9</td>
<td>23</td>
<td>36</td>
<td>46</td>
<td>53</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>6</td>
<td>11</td>
<td>14</td>
<td>17</td>
<td>21</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0</td>
<td>5</td>
<td>7</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>NOUN</td>
<td>5</td>
<td>21</td>
<td>42</td>
<td>69</td>
<td>99</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>10</td>
<td>15</td>
<td>18</td>
<td>18</td>
<td>21</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>13</td>
<td>26</td>
<td>30</td>
<td>34</td>
<td>35</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>VERB</td>
<td>11</td>
<td>21</td>
<td>37</td>
<td>54</td>
<td>66</td>
</tr>
</tbody>
</table>

**Figure 67** Frequent word profile for English

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJECTIVE</td>
<td>3%</td>
<td>4%</td>
<td>5%</td>
<td>6%</td>
<td>6%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>3%</td>
<td>5%</td>
<td>6%</td>
<td>6%</td>
<td>6%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>22%</td>
<td>18%</td>
<td>17%</td>
<td>16%</td>
<td>15%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>12%</td>
<td>11%</td>
<td>11%</td>
<td>10%</td>
<td>10%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>2%</td>
<td>6%</td>
<td>8%</td>
<td>9%</td>
<td>11%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>29%</td>
<td>27%</td>
<td>25%</td>
<td>24%</td>
<td>23%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>13%</td>
<td>14%</td>
<td>14%</td>
<td>13%</td>
<td>13%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>15%</td>
<td>15%</td>
<td>15%</td>
<td>16%</td>
<td>15%</td>
</tr>
</tbody>
</table>

**Figure 68** Frequent word distribution for English
### French:

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>French</td>
<td>50</td>
<td>100</td>
<td>150</td>
<td>200</td>
<td>250</td>
</tr>
<tr>
<td>ADJECTIVE</td>
<td>12</td>
<td>18</td>
<td>27</td>
<td>33</td>
<td>42</td>
</tr>
<tr>
<td>ADVERB</td>
<td>8</td>
<td>20</td>
<td>23</td>
<td>28</td>
<td>36</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>6</td>
<td>8</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>4</td>
<td>8</td>
<td>11</td>
<td>12</td>
<td>13</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>NOUN</td>
<td>13</td>
<td>27</td>
<td>55</td>
<td>86</td>
<td>115</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>15</td>
<td>18</td>
<td>20</td>
<td>24</td>
<td>25</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>14</td>
<td>27</td>
<td>30</td>
<td>34</td>
<td>41</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>VERB</td>
<td>4</td>
<td>11</td>
<td>19</td>
<td>22</td>
<td>26</td>
</tr>
</tbody>
</table>

Figure 69: Frequent word profile for French

<table>
<thead>
<tr>
<th></th>
<th>50</th>
<th>100</th>
<th>150</th>
<th>200</th>
<th>250</th>
</tr>
</thead>
<tbody>
<tr>
<td>French</td>
<td>50</td>
<td>100</td>
<td>150</td>
<td>200</td>
<td>250</td>
</tr>
<tr>
<td>ADJECTIVE</td>
<td>1%</td>
<td>2%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
</tr>
<tr>
<td>ADVERB</td>
<td>4%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
<td>5%</td>
</tr>
<tr>
<td>ARTICLE</td>
<td>23%</td>
<td>20%</td>
<td>19%</td>
<td>19%</td>
<td>19%</td>
</tr>
<tr>
<td>CONJUNCTION</td>
<td>7%</td>
<td>7%</td>
<td>7%</td>
<td>7%</td>
<td>7%</td>
</tr>
<tr>
<td>INTERJECTION</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>NOUN</td>
<td>5%</td>
<td>8%</td>
<td>11%</td>
<td>11%</td>
<td>11%</td>
</tr>
<tr>
<td>PREPOSITION</td>
<td>39%</td>
<td>34%</td>
<td>32%</td>
<td>32%</td>
<td>32%</td>
</tr>
<tr>
<td>PRONOUN</td>
<td>17%</td>
<td>18%</td>
<td>17%</td>
<td>17%</td>
<td>17%</td>
</tr>
<tr>
<td>SYMBOL</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>VERB</td>
<td>4%</td>
<td>5%</td>
<td>6%</td>
<td>6%</td>
<td>6%</td>
</tr>
</tbody>
</table>

Figure 70: Frequent word distribution for French
Bibliography


Hadi Amiri, Hosein Hojjat, Farhad Oroumchian, “Investigation on a Feasible Corpus for Persian POS Tagging”, *12th international CSI computer conference*, Iran, 2007


Young-Bum Kim and Benjamin Snyder, “Unsupervised Consonant-Vowel Prediction over Hundreds of Languages”, *Proceedings of the Association for Computational Linguistics*, 2013


John R. Leafgren, “Register, Mode, and Bulgarian Object Placement”, presented at the Balkan Conference, 2002


G. Mann and D. Yarowsky, “Multipath translation lexicon induction via bridge languages,” *Proceedings of NAACL 2001*


Sujith Ravi and Kevin Knight, “Minimized models for unsupervised part-of-speech tagging”, *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and


Ashish Vaswani, Yinggong Zhao, Victoria Fossum, and David Chiang, “Decoding with large-scale neural language models improves translation”, *Proceedings of the Empirical Methods on Natural Language Processing*, 2013

