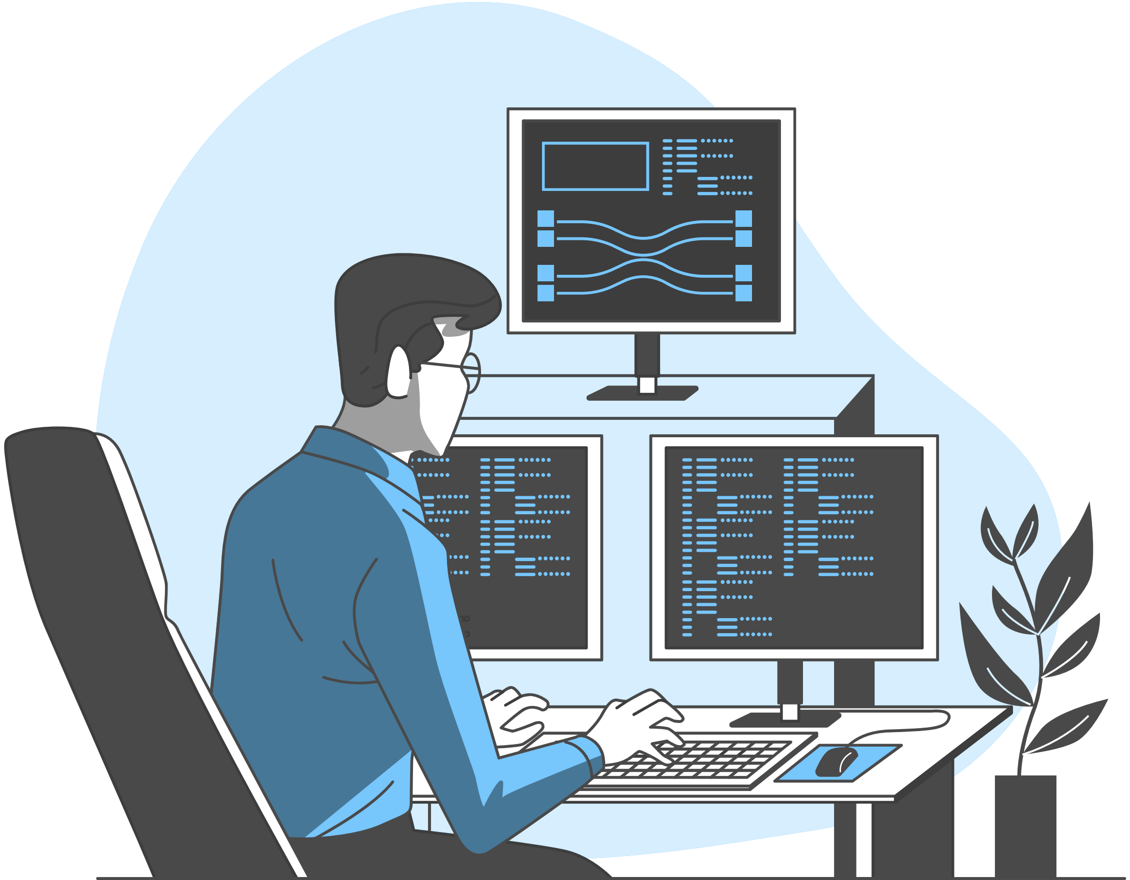
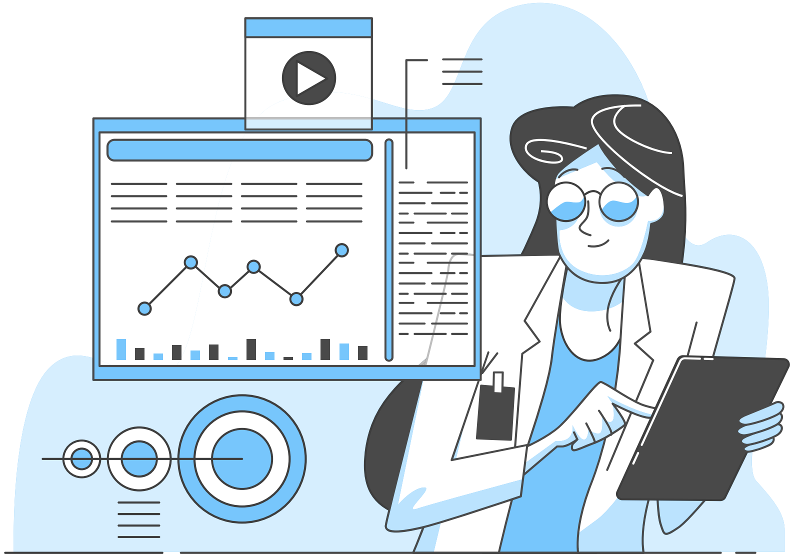


Predicting Morphological Inflections

By: Anirudh, Max, Amy

CS 159 - NLP

Table of Contents



01
Problem Statement

Problem Statement

We're using a (non-neural network) approach to determine inflections given lemmas in 70+ languages. We repurpose a grapheme to phoneme model (G2P, Alegria and Etxeberria) to do this.

Sub-question: How does the length of the string of morphological features affect the performance of the G2P model? Do different compressions have an effect on the quality of output?

grapheme - written form

phoneme - spoken form

lemmas - the base form of a word (For ex: break is the lemma of broken)

Motivation

- Different languages have different inflections, which include plural forms, tense, superlatives, and more.
  Ex: English (pass/passed) and Spanish (hombre/hombres)
- Reconstruct a model that can accurately inflect a word for a variety of languages based on labelled examples
- An example from the Estonian dataset:

`kuulma  kuuldav V.PTCP;PASS;PRS`

lemma   Inflected form (gold)   Morphological features

Project Background

- This project is based on solving a 2018 shared task from CoNLL: **“Universal Morphological Reinflection”**
- There are two main approaches to this task: using neural networks or not using neural networks
    - Non-NN approaches are easier to implement and analyze
    - We want to see how much we can infer from a non-NN approach

Related Work

UZH (Makarov & Ruzsics & Clematide)

Best-performing submission

Uses an encoder decoder model with copy mechanism and neural state-transition system

CRF (Ling Liu & Lingshuang Jack Mao)

Non-neural network solution

Uses a linear-chain conditional random field model to map input characters sequences to output characters sequences and develop features useful for predicting inflection

G2P (Alegria and Etxeberria)

Uses grapheme to phoneme conversion tools for morphological reinflection by replacing the phoneme labels with the desired word forms (inflections)

Current Model

Problem: Reconstructing Alegria & Etxeberria's approach for the 2016 task (10 languages) doesn't work for the 2018 task (100 languages)

- Formatting of the 2016 data did not match the formatting of the 2018 data, which is why we could not use their original compression method
- *Does the length of the string of morphological features contribute negatively to the performance of our model? How can we improve upon our metrics using different compression methods?*

[1] A Simple Proposal: Grapheme-to-Phoneme for Inflection by Inaki Alegria, Izaskun Etxeberria

Methods

Problem: Condense the morphological features into a short string mostly made up of the lemma to make the lemma easier to identify

Solution: Test three different approaches to string compression of morphological features to improve our results

Note: In each case, we concatenate the compressed string onto the beginning and end of the lemmas to create the final training instance (Alegria and Etxeberria)

1. No compression
2. Unicode mapping
3. Buckets

Data Set

76 different languages (filtered out)

- 10,000 training instances
- 1000 testing instances (sometimes 100)
- labelled

Training:	dear	dhear tú	V;2;SG;PST;IND
Testing:	bibe	an bibe	N;NOM;SG;DEF

...

How does G2P work?

Training:	grapheme	→	phoneme
Predictions:	grapheme	→	predicted phoneme/no pronunciation

Unformatted Data:	dysgu	dysgit	V;LIT;2;SG;IND;IPFV
	(lemma)	(inflection)	(morphological features)

Formatted training data:	<features><lemma><features>	<inflection>
Formatted testing data:	<features><lemma><features>	?

...

Our Three String Compression Strategies

Original:

`interpretoj   interpretoni  V;2;PL;IND;PRS`

1. Use the morphological features from our data set as is (no compression)

   `V;2;PL;IND;PRSinterpretojV;2;PL;IND;PRS  interpretoni`

2. Mapping each morphological feature to a UNICODE character

   `VκıĳķinterpretojVκıĳķ   interpretoni`

3. Fix a number of characters for morphological features, and truncate any other features that come after. Repeat Lemma

   ...

   `V2PLINinterpretojiinterpretijii..V2PLIN    interpretoni`

Progress...

Tools: Open source tools WSFT, Phonetisaurus and Docker

1. Wrote and used scripts to organize and reformat hundreds of files and folders with different language training and testing data
2. Got G2P running on Docker using a Docker image
3. We trained a **model on each language of our training set and test on its corresponding test data.**
4. Repeated the above for three compression strategies
5. Wrote and ran evaluation scripts

Data Analysis

Evaluation Metrics

Total Accuracy:

$$\frac{\text{Correct predictions}}{\text{Total test instances}}$$

Levenshtein distance:

Minimum number of character changes to get from one string to another

...

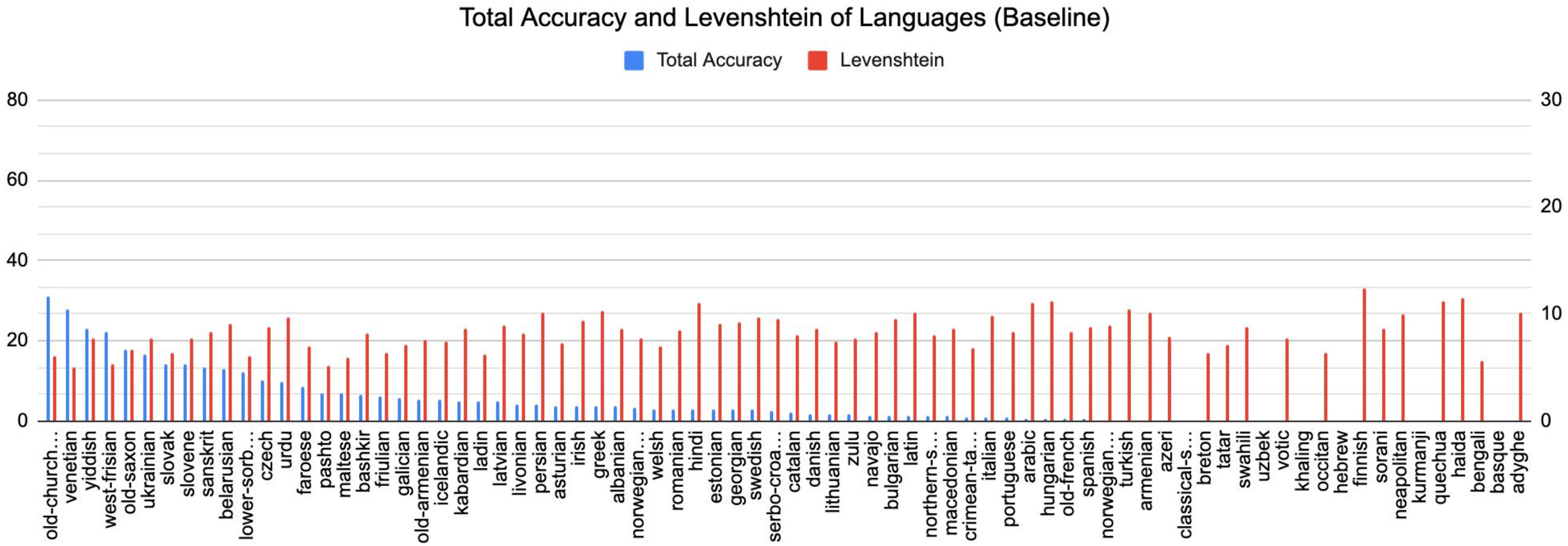
Results

	Total accuracy (%)	Avg. Levenshtein
Baseline	4.66	7.66
No compression	6.60	3.39
Unicode	**9.01**	**3.26**
Buckets	6.13	10.51

Results

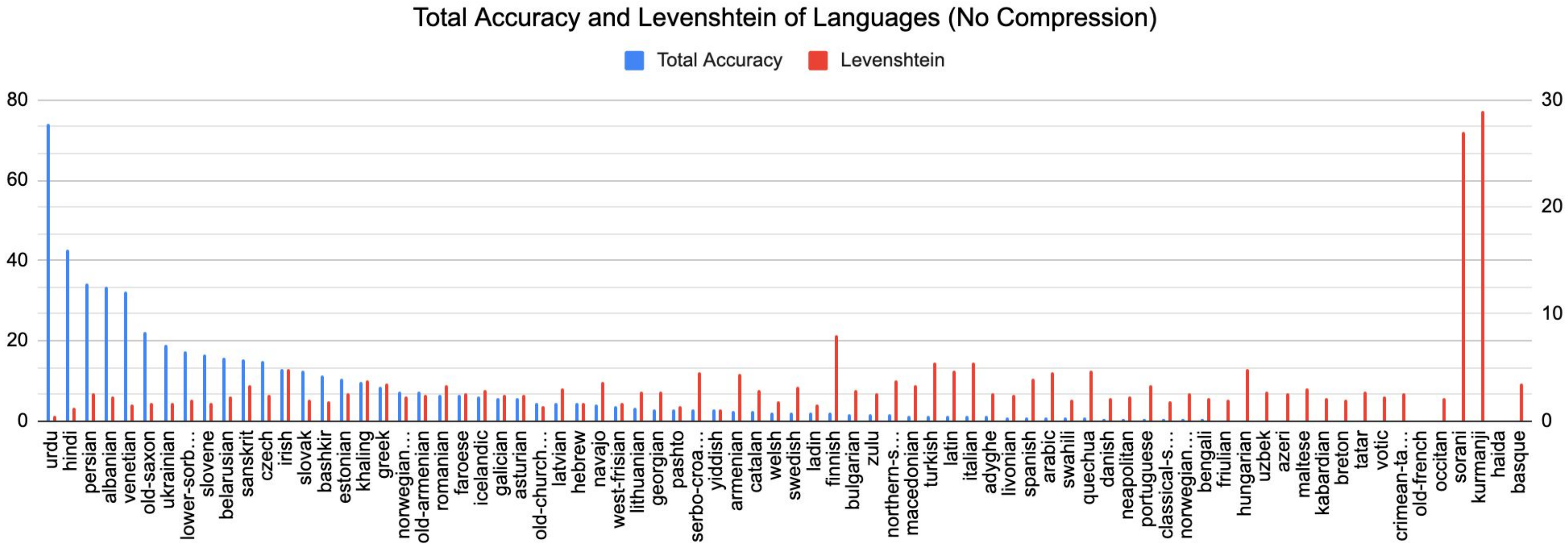
	Total accuracy (%)	Avg. Levenshtein
Baseline	4.66	7.66
No compression	6.60	3.39
Unicode	**9.01**	**3.26**
Buckets	6.13	10.51

	Predictions made (%)
Baseline	71.7
No compression	73.2
Unicode	74.1
Buckets	81.2



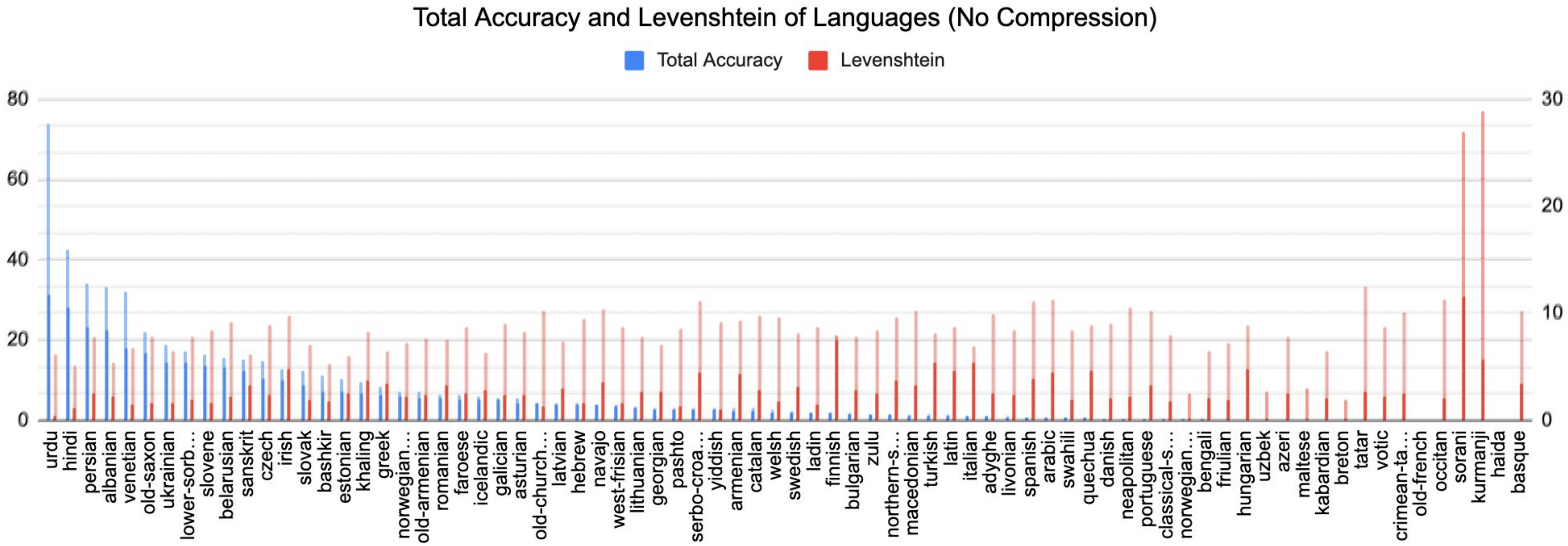
Average total accuracy: 4.655263158%

Average levenshtein: 7.659736842

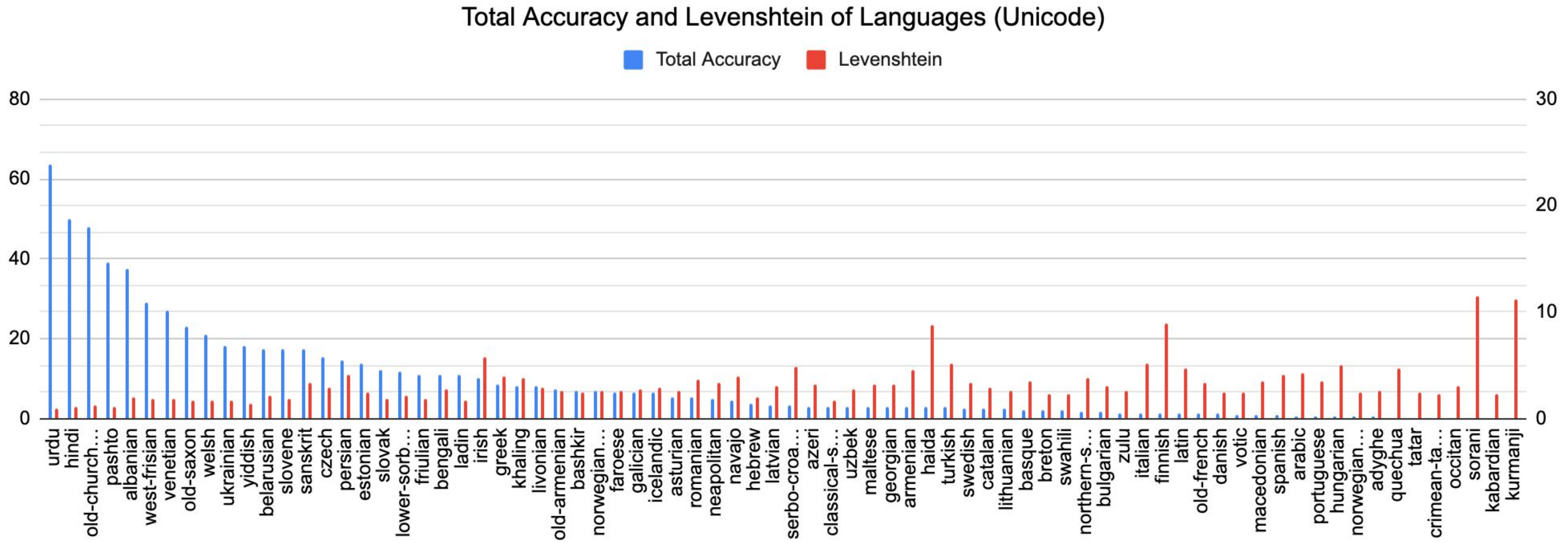


Average total accuracy: 6.796052632%

Average levenshtein: 3.386578947

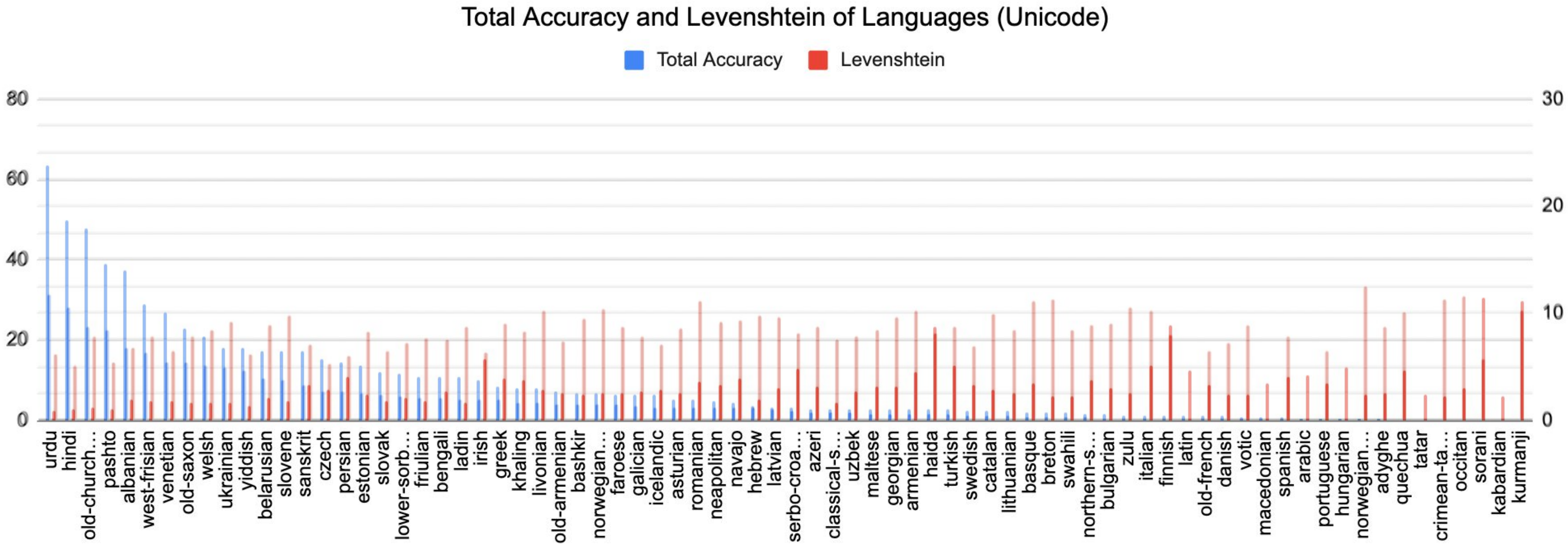


Average total accuracy: 6.796052632%
Average levenshtein: 3.386578947

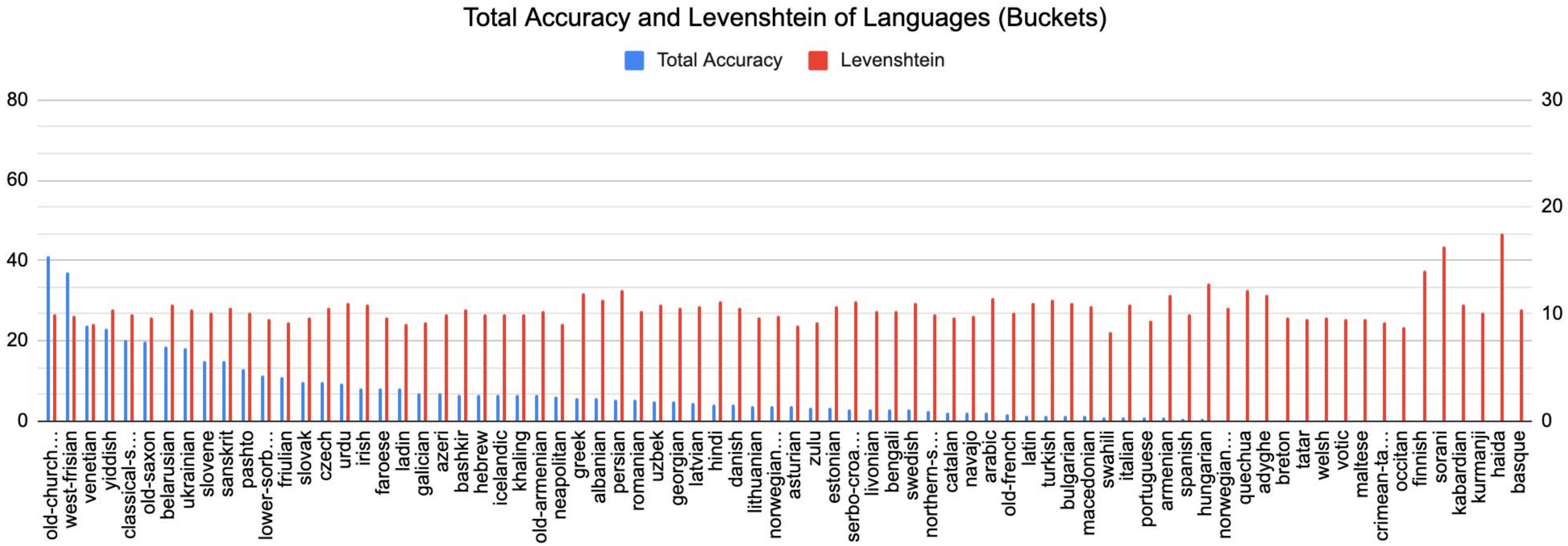


Average total accuracy: 9.010526316%

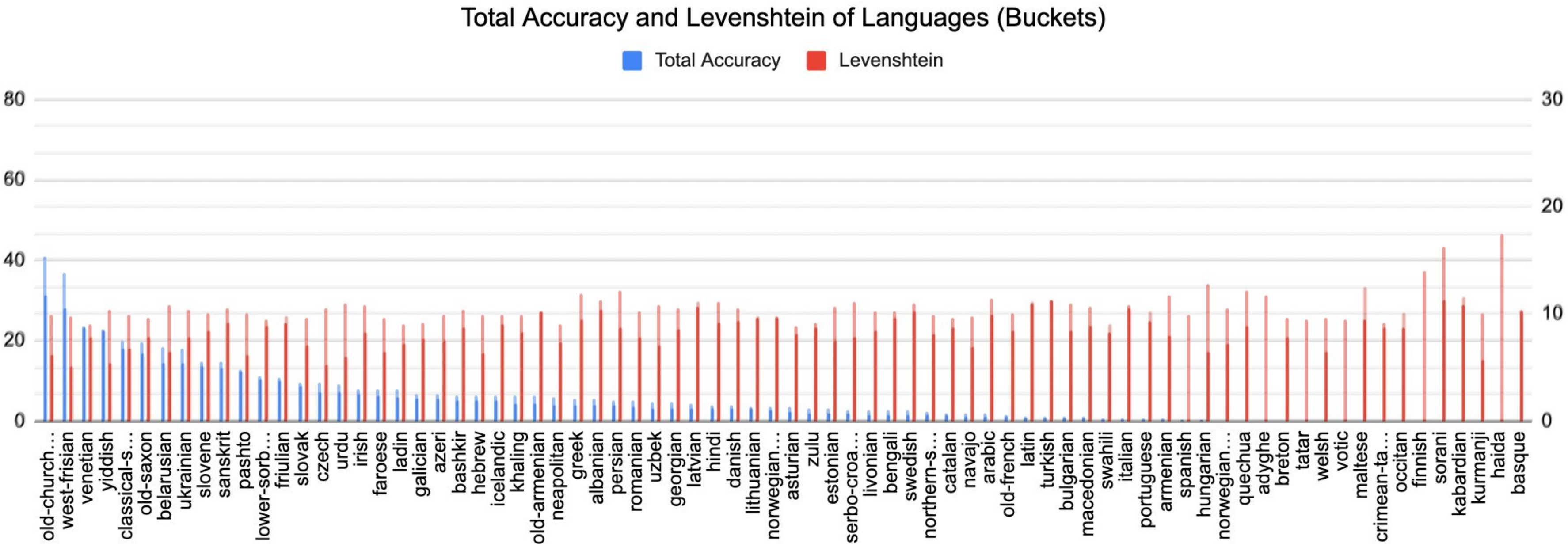
Average levenshtein: 3.260394737



Average total accuracy:	9.010526316%
Average levenshtein:	3.260394737



Average total accuracy: 6.127631579%
Average levenshtein: 10.51421053



Average total accuracy:	6.127631579%
Average levenshtein:	10.51421053

Results

	Total accuracy (%)	Avg. Levenshtein
Baseline	4.66	7.66
No compression	6.60	3.39
Unicode	**9.01**	**3.26**
Buckets	6.13	10.51

Each method is a tradeoff of two factors:

- Compression
- Information lost

Next Steps

1. Potentially consider other compression methods
2. Investigate why the model does so well on Urdu and Hindu
3. Finish writing our paper

Thank you!

Questions?