CONF42

Machine Learning 2021

Multilingual Natural Language Processing using Python

Gajendra Deshpande

KLS Gogte Institute of Technology, India

https://gcdeshpande.github.io

Contents

- Introduction to NLP and concepts
- Challenges in Multi source multilingual NLP
- Tools for extracting information from various file formats
- Extract information from web pages and source code
- Methods to convert information into common language format
- Code walkthrough for multi source and multi lingual summary generation
- Conclusion and Questions

Basic Concepts of Natural Language Processing

Tokenization

- Tokenize paragraph to words and sentences

Word Embeddings

- Representation of words into vectors

Text Completion

- Predict next few words

Sentence Similarity

- Similarity score between two sentences

Normalization

- Transform text into canonical form

Transliteration

- Write Text in language A script using language B script

Translation

- Convert text in language A to language B

Phonetic Analysis

- How two characters sound

Syllabification

- Convert text to syllables

Lemmatization

- Convert words to their root form

Part-of-Speech

- Tag part-of-speech in a text

Named Entity Recognition

- Recognize entities in a text

Dependency Parsing

- Analyze the grammatical structure of a sentence.

Language Detection

- Detect the language of the text or words

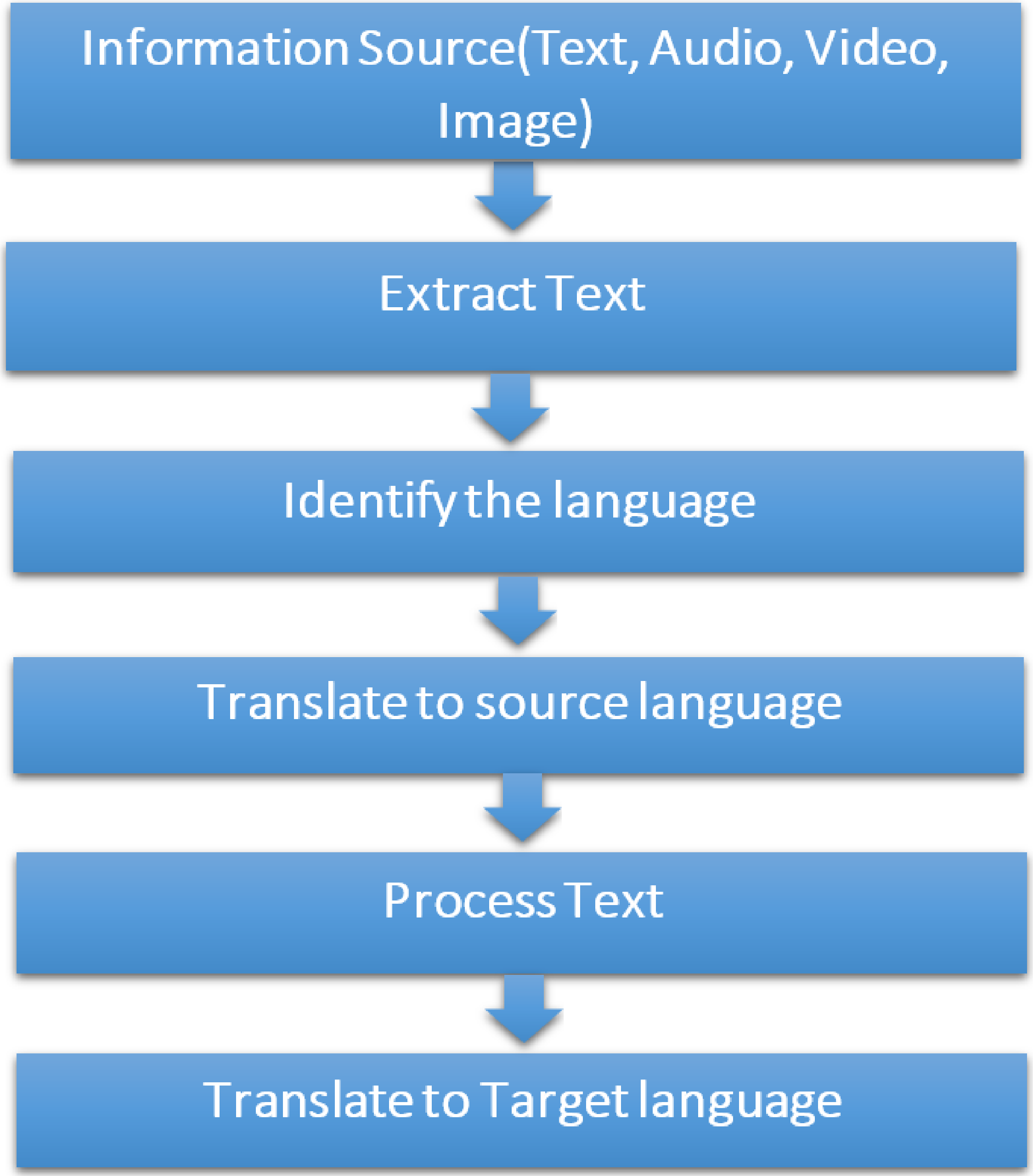
Stemming

- Remove last few characters from the word

Challenges in Multi Lingual NLP

- Language is ambiguous
- Languages have different structure, grammar and order
- Hard to deal with mixed language information
- Translation from one language to another language is not accurate
- Language semantics need to be taken into account
- Lack of libraries and features
- Many features need to hard coded

General Flow of Multi-Source and Multi Lingual Information Processing


Information Source(Text, Audio, Video, Image)
Extract Text
Identify the language
Translate to source language
Process Text
Translate to Target language

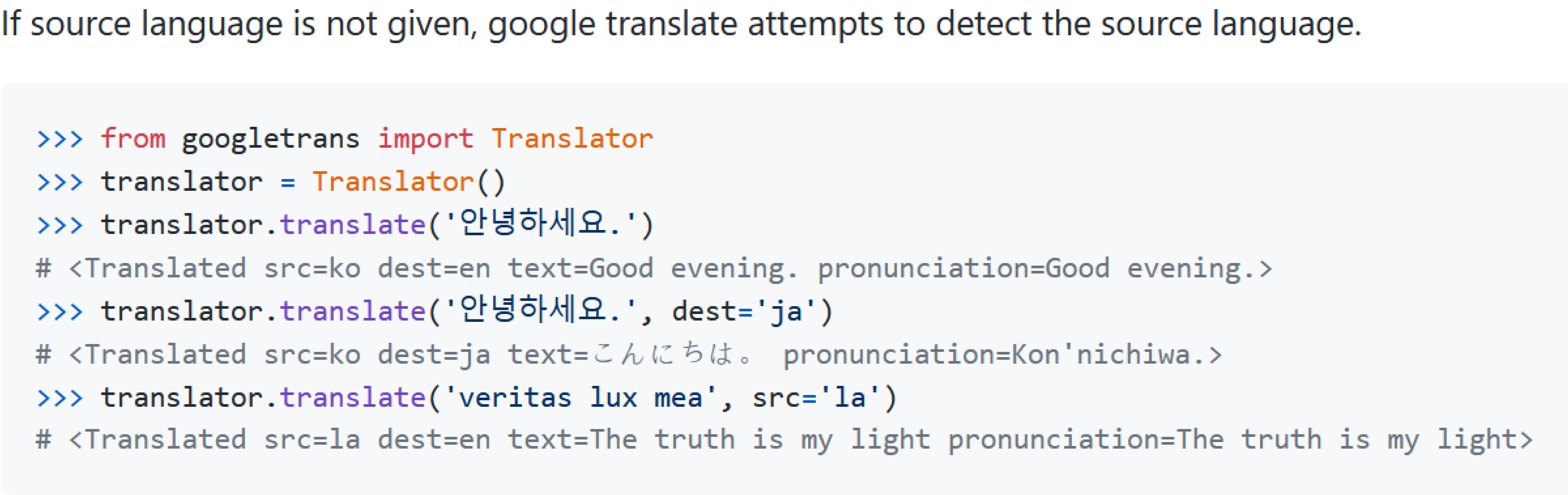
Googletrans 3.0.0

- Googletrans is a **free** and **unlimited** python library that implemented Google Translate API. This uses the Google Translate Ajax API to make calls to such methods as detect and translate.
- Compatible with Python 3.6+.
- Fast and reliable - it uses the same servers that translate.google.com uses
- Auto language detection
- Bulk translations
- Customizable service URL
- HTTP/2 support

Source: https://pypi.org/project/googletrans/

Googletrans 3.0.0

pip install googletrans

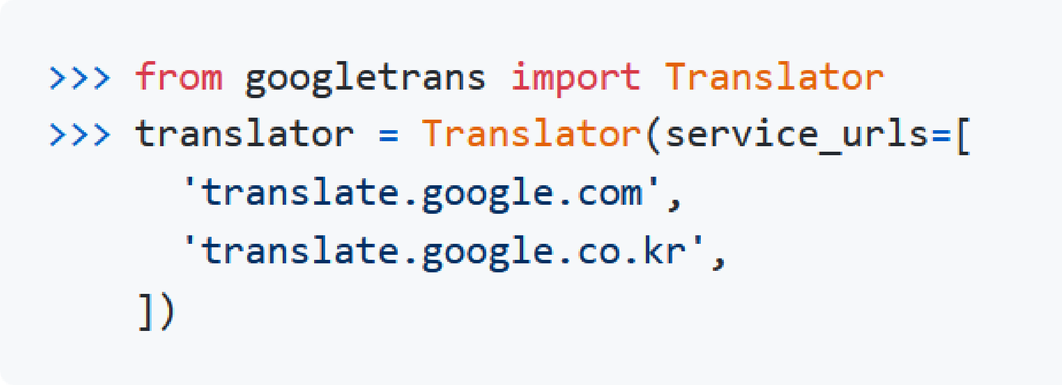


If source language is not given, google translate attempts to detect the source language.

```
>>> from googletrans import Translator
>>> translator = Translator()
>>> translator.translate('안녕하세요.')
# <Translated src=ko dest=en text=Good evening. pronunciation=Good evening.>
>>> translator.translate('안녕하세요.', dest='ja')
# <Translated src=ko dest=ja text=こんにちは。 pronunciation=Kon'nichiwa.>
>>> translator.translate('veritas lux mea', src='la')
# <Translated src=la dest=en text=The truth is my light pronunciation=The truth is my light>
```

Googletrans 3.0.0

You can use another google translate domain for translation. If multiple URLs are provided, it then randomly chooses a domain.

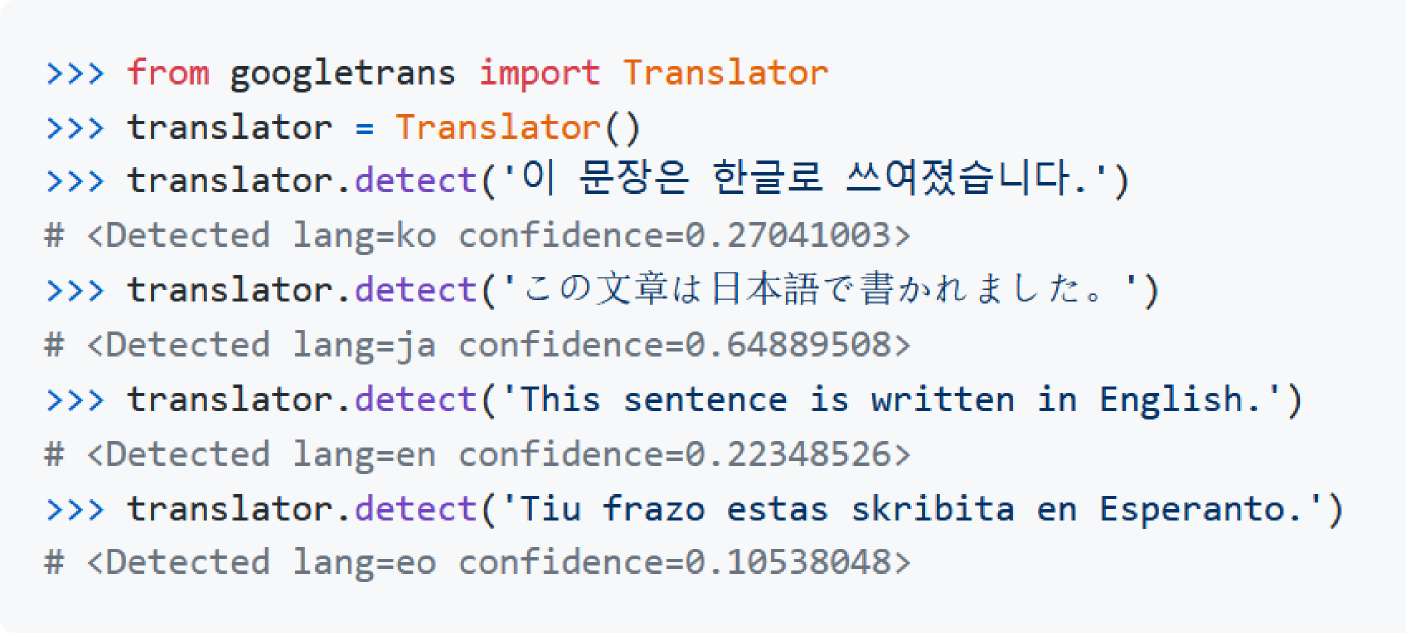


```
>>> from googletrans import Translator
>>> translator = Translator(service_urls=[
      'translate.google.com',
      'translate.google.co.kr',
    ])
```

Unofficial, unstable, The maximum character limit on a single text is 15k.

Use Google’s official Translate API

The detect method, as its name implies, identifies the language used in a given sentence.



```
>>> from googletrans import Translator
>>> translator = Translator()
>>> translator.detect('이 문장은 한글로 쓰여졌습니다.')
# <Detected lang=ko confidence=0.27041003>
>>> translator.detect('この文章は日本語で書かれました。')
# <Detected lang=ja confidence=0.64889508>
>>> translator.detect('This sentence is written in English.')
# <Detected lang=en confidence=0.22348526>
>>> translator.detect('Tiu frazo estas skribita en Esperanto.')
# <Detected lang=eo confidence=0.10538048>
```

SpeechRecognition : Extract text from an audio file

- Library for performing speech recognition, with support for several engines and APIs, online and offline.
- Speech recognition engine/API support:
  - CMU Sphinx (works offline)
  - Google Speech Recognition
  - Google Cloud Speech API
  - Wit.ai
  - Microsoft Bing Voice Recognition
  - Houndify API
  - IBM Speech to Text
  - Snowboy Hotword Detection (works offline)

Source: https://pypi.org/project/SpeechRecognition/

SpeechRecognition : Extract text from an audio file

```
pip3 install SpeechRecognition pydub

import speech_recognition as sr

filename = “filename.wav"

# initialize the recognizer
r = sr.Recognizer()

# open the file
with sr.AudioFile(filename) as source:
    # listen for the data (load audio to memory)
    audio_data = r.record(source)
    # recognize (convert from speech to text)
    text = r.recognize_google(audio_data)
    print(text)
```

Source: https://www.thepythoncode.com/article/using-speech-recognition-to-convert-speech-to-text-python

pytesseract: Extract text from an image file

- Python-tesseract is an optical character recognition (OCR) tool for python. That is, it will recognize and “read” the text embedded in images.
- Python-tesseract is a wrapper for Google’s Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported by the Pillow and Leptonica imaging libraries, including jpeg, png, gif, bmp, tiff, and others. Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file.

Source: https://pypi.org/project/pytesseract/

pytesseract: Extract text from an image file

```
import pytesseract

pytesseract.pytesseract.tesseract_cmd = r‘tesseract path‘

print(pytesseract.image_to_string(r‘Image Path'))
```

eautifulsoup4: Extract information from a webpage

- Beautiful Soup is a library that makes it easy to scrape information from web pages. It sits atop an HTML or XML parser, providing Pythonic idioms for iterating, searching, and modifying the parse tree.

Source: https://pypi.org/project/beautifulsoup4/

beautifulsoup4: Extract information from a webpage

```
import requests

from bs4 import BeautifulSoup

URL = 'https://www.monster.com/jobs/search/?q=Software-Developer&where=Australia'

page = requests.get(URL)

soup = BeautifulSoup(page.content, 'html.parser')
```

```
<div id="ResultsContainer">
        <!-- all the job listings -->
</div>
```

```
results = soup.find(='ResultsContainer')

print(results.prettify())
```

Source: https://realpython.com/bonus/beautiful-soup/

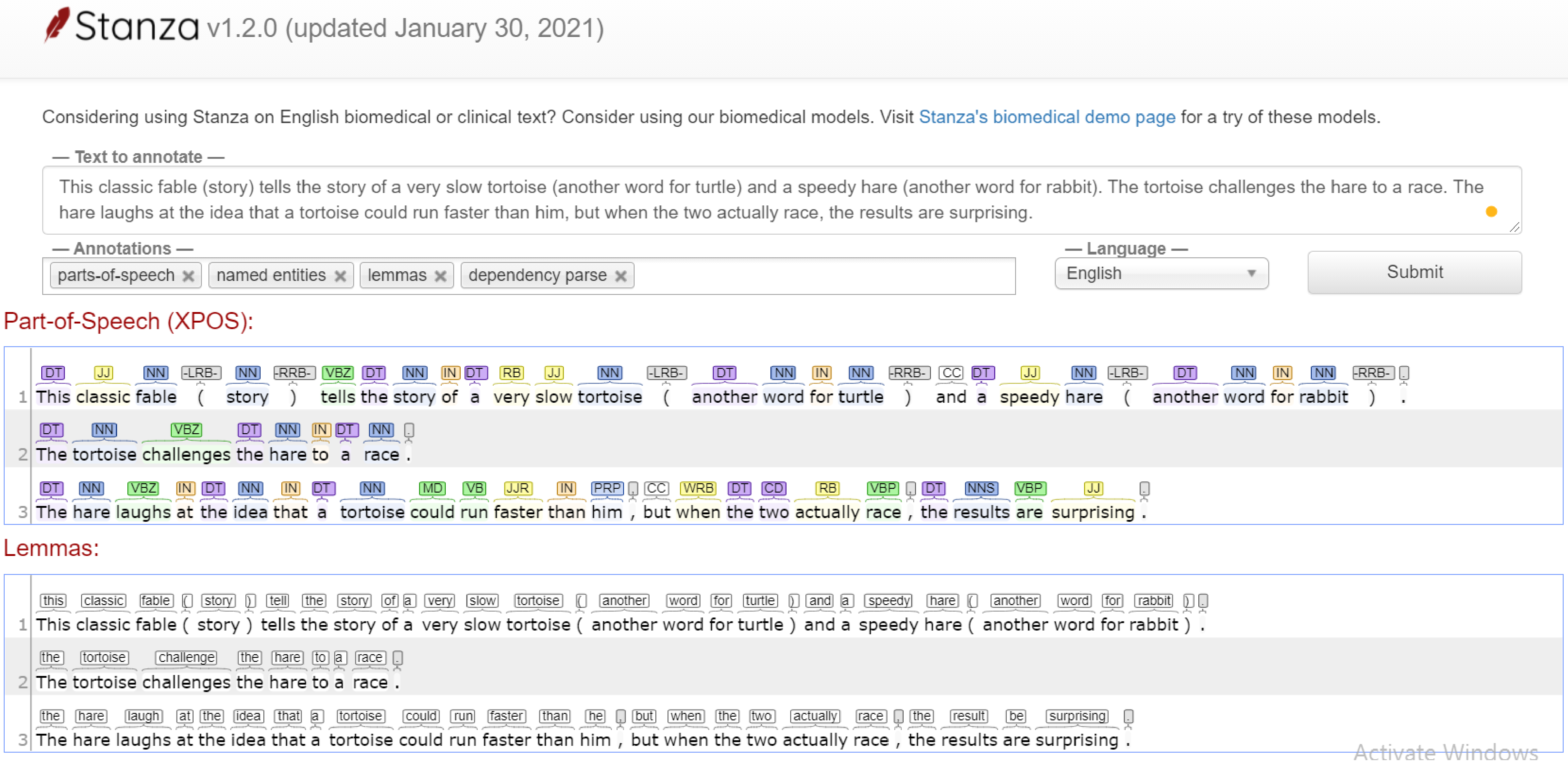
Stanza –Python NLP Package for many human languages

- Stanza is a collection of accurate and efficient tools for many human languages in one place. Starting from raw text to syntactic analysis and entity recognition, Stanza brings state-of-the-art NLP models to languages of your choosing
- Native Python implementation requiring minimal efforts to set up;
- Full neural network pipeline for robust text analytics, including tokenization, multi-word token (MWT) expansion, lemmatization, part-of-speech (POS) and morphological features tagging, dependency parsing, and named entity recognition;
- Pretrained neural models supporting 66 (human) languages;
- A stable, officially maintained Python interface to CoreNLP.

Source: https://stanfordnlp.github.io/stanza/

Live Demo: https://stanza.run

Code Examples:https://github.com/gcdeshpande/IndicNLP/tree/main/stanza



Stanza v1.2.0 (updated January 30, 2021)

Considering using Stanza on English biomedical or clinical text? Consider using our biomedical models. Visit Stanza's biomedical demo page for a try of these models.

— Text to annotate —

This classic fable (story) tells the story of a very slow tortoise (another word for turtle) and a speedy hare (another word for rabbit). The tortoise challenges the hare to a race. The hare laughs at the idea that a tortoise could run faster than him, but when the two actually race, the results are surprising.

— Annotations —
parts-of-speech ✕ named entities ✕ lemmas ✕ dependency parse ✕

— Language —
English ▾

Part-of-Speech (XPOS):

1
DT JJ NN -LRB- NN -RRB- VBZ DT NN IN DT RB JJ NN -LRB- DT NN IN NN -RRB- CC DT JJ NN -LRB- DT NN IN NN -RRB- .
This classic fable (story) tells the story of a very slow tortoise (another word for turtle) and a speedy hare (another word for rabbit) .

2
DT NN VBZ DT NN IN DT NN .
The tortoise challenges the hare to a race .

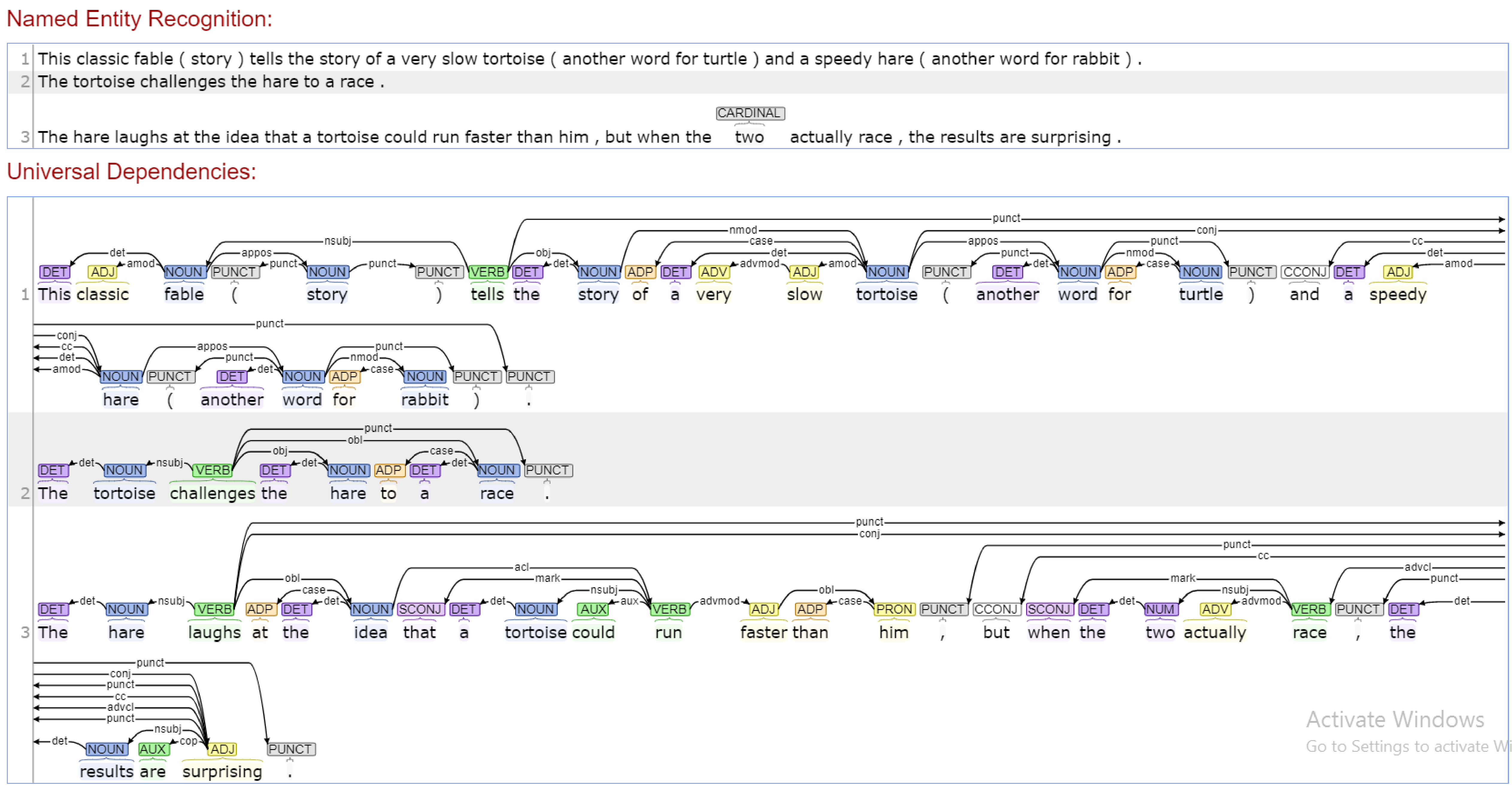
3
DT NN VBZ IN DT NN IN DT NN MD VB JJR IN PRP , CC WRB DT CD RB VBP , DT NNS VBP JJ .
The hare laughs at the idea that a tortoise could run faster than him , but when the two actually race , the results are surprising .

Lemmas:

1
this classic fable (story) tell the story of a very slow tortoise (another word for turtle) and a speedy hare (another word for rabbit) .
This classic fable (story) tells the story of a very slow tortoise (another word for turtle) and a speedy hare (another word for rabbit) .

2
the tortoise challenge the hare to a race .
The tortoise challenges the hare to a race .

3
the hare laugh at the idea that a tortoise could run faster than he , but when the two actually race , the result be surprising .
The hare laughs at the idea that a tortoise could run faster than him , but when the two actually race , the results are surprising .



Named Entity Recognition:

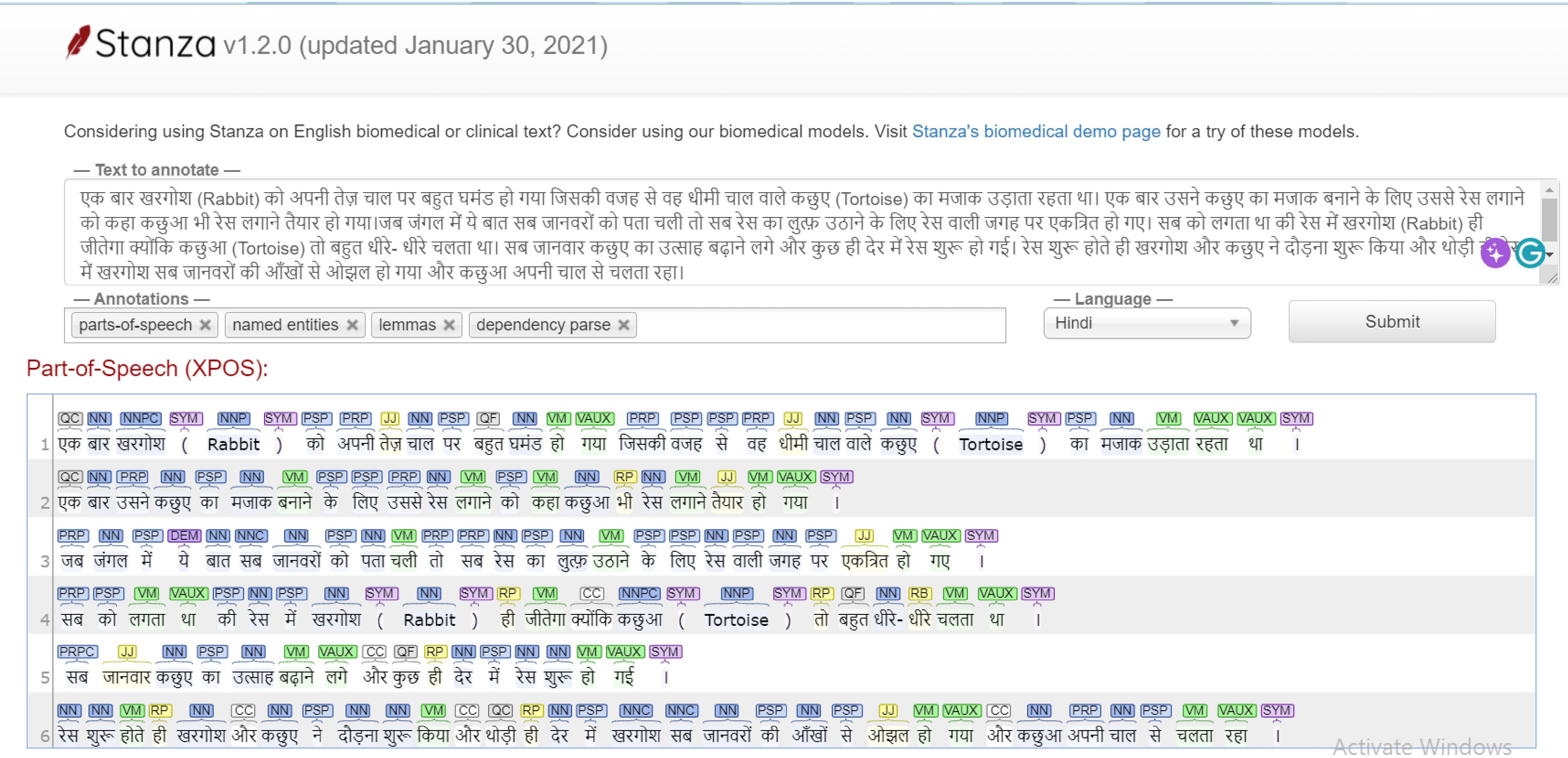
1 This classic fable (story) tells the story of a very slow tortoise (another word for turtle) and a speedy hare (another word for rabbit) .

2 The tortoise challenges the hare to a race .

3 The hare laughs at the idea that a tortoise could run faster than him , but when the two actually race , the results are surprising .

Universal Dependencies:

1
DET ADJ NOUN PUNCT NOUN PUNCT VERB DET NOUN ADP DET ADV ADJ NOUN PUNCT DET NOUN ADP NOUN PUNCT CCONJ DET ADJ
This classic fable (story) tells the story of a very slow tortoise (another word for turtle) and a speedy
NOUN PUNCT DET NOUN ADP NOUN PUNCT PUNCT
hare (another word for rabbit) .
det amod nsubj appos punct obj nmod case advmod conj cc
2
DET NOUN VERB DET NOUN ADP DET NOUN PUNCT
The tortoise challenges the hare to a race .
det nsubj obj obl case punct
3
DET NOUN VERB ADP DET NOUN SCONJ DET NOUN AUX VERB ADJ ADP PRON PUNCT CCONJ SCONJ DET NUM ADV VERB PUNCT DET
The hare laughs at the idea that a tortoise could run faster than him , but when the two actually race , the
NOUN AUX ADJ PUNCT
results are surprising .
det nsubj obl case acl mark aux advmod punct conj cc advcl cop



Stanza v1.2.0 (updated January 30, 2021)

Considering using Stanza on English biomedical or clinical text? Consider using our biomedical models. Visit Stanza's biomedical demo page for a try of these models.

— Text to annotate —

एक बार खरगोश (Rabbit) को अपनी तेज़ चाल पर बहुत घमंड हो गया जिसकी वजह से वह धीमी चाल वाले कछुए (Tortoise) का मजाक उड़ाता रहता था। एक बार उसने कछुए का मजाक बनाने के लिए उससे रेस लगाने को कहा कछुआ भी रेस लगाने तैयार हो गया।जब जंगल में ये बात सब जानवरों को पता चली तो सब रेस का लुत्फ़ उठाने के लिए रेस वाली जगह पर एकत्रित हो गए। सब को लगता था की रेस में खरगोश (Rabbit) ही जीतेगा क्योंकि कछुआ (Tortoise) तो बहुत धीरे- धीरे चलता था। सब जानवार कछुए का उत्साह बढ़ाने लगे और कुछ ही देर में रेस शुरू हो गई। रेस शुरू होते ही खरगोश और कछुए ने दौड़ना शुरू किया और थोड़ी ही देर में खरगोश सब जानवरों की आँखों से ओझल हो गया और कछुआ अपनी चाल से चलता रहा।

— Annotations —

parts-of-speech ✕ named entities ✕ lemmas ✕ dependency parse ✕

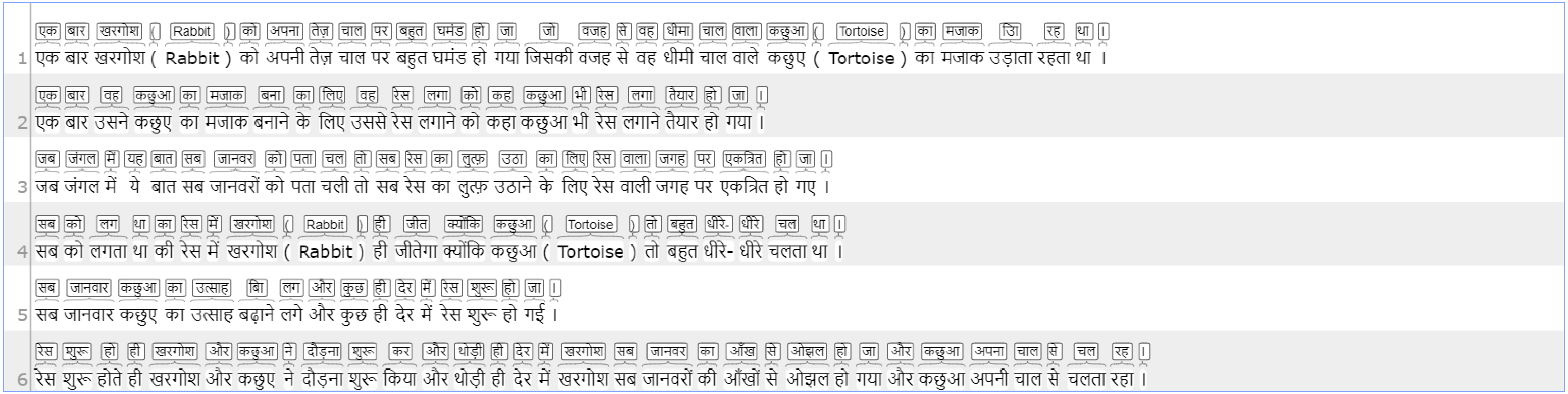
— Language —

Hindi ▾

Part-of-Speech (XPOS):

```
  QC  NN  NNPC  SYM NNP    SYM PSP PRP  JJ   NN  PSP QF   NN   VM VAUX PRP   PSP PSP PRP JJ   NN  PSP  NN   SYM NNP      SYM PSP NN   VM    VAUX  VAUX SYM
1 एक बार खरगोश (   Rabbit  )   को  अपनी तेज़ चाल पर  बहुत घमंड हो गया  जिसकी वजह से वह  धीमी चाल वाले कछुए (  Tortoise  )  का  मजाक उड़ाता रहता  था   ।

  QC  NN  PRP   NN   PSP NN    VM    PSP PSP  PRP  NN  VM    PSP VM  NN    RP  NN  VM    JJ     VM VAUX SYM
2 एक बार उसने कछुए का  मजाक बनाने के  लिए उससे रेस लगाने को  कहा कछुआ भी रेस लगाने तैयार हो गया  ।

  PRP NN   PSP DEM NN  NNC NN      PSP NN  VM  PRP PRP NN  PSP NN   VM    PSP PSP NN  PSP  NN   PSP JJ      VM VAUX SYM
3 जब जंगल में  ये  बात सब जानवरों को  पता चली तो  सब  रेस का  लुत्फ़ उठाने के  लिए रेस वाली जगह पर  एकत्रित हो गए   ।

  PRP PSP VM   VAUX PSP NN  PSP NN    SYM NN     SYM RP VM    CC    NNPC SYM NNP      SYM RP QF   NN    RB  VM   VAUX SYM
4 सब  को  लगता था   की  रेस में  खरगोश (   Rabbit )   ही जीतेगा क्योंकि कछुआ (   Tortoise )   तो बहुत धीरे- धीरे चलता था   ।

  PRPC JJ     NN   PSP NN    VM    VAUX CC  QF  RP NN  PSP NN  NN   VM VAUX SYM
5 सब   जानवार कछुए का  उत्साह बढ़ाने लगे  और कुछ ही देर में  रेस शुरू हो गई  ।

  NN  NN   VM   RP NN    CC  NN   PSP NN    NN   VM   CC  QC    RP NN  PSP NNC   NNC NN      PSP NN    PSP JJ   VM VAUX CC  NN   PRP  NN  PSP VM    VAUX SYM
6 रेस शुरू होते ही खरगोश और कछुए ने  दौड़ना शुरू किया और थोड़ी ही देर में  खरगोश सब  जानवरों की  आँखों से  ओझल हो गया और कछुआ अपनी चाल से  चलता रहा  ।
```

Lemmas:



1 एक बार खरगोश (Rabbit) को अपना तेज़ चाल पर बहुत घमंड हो जा जो वजह से वह धीमा चाल वाला कछुआ (Tortoise) का मजाक उिा रह था ।
एक बार खरगोश (Rabbit) को अपनी तेज़ चाल पर बहुत घमंड हो गया जिसकी वजह से वह धीमी चाल वाले कछुए (Tortoise) का मजाक उड़ाता रहता था ।

2 एक बार वह कछुआ का मजाक बना का लिए वह रेस लगा को कह कछुआ भी रेस लगा तैयार हो जा ।
एक बार उसने कछुए का मजाक बनाने के लिए उससे रेस लगाने को कहा कछुआ भी रेस लगाने तैयार हो गया ।

3 जब जंगल में यह बात सब जानवर को पता चल तो सब रेस का लुत्फ़ उठा का लिए रेस वाला जगह पर एकत्रित हो जा ।
जब जंगल में ये बात सब जानवरों को पता चली तो सब रेस का लुत्फ़ उठाने के लिए रेस वाली जगह पर एकत्रित हो गए ।

4 सब को लग था का रेस में खरगोश (Rabbit) ही जीत क्योंकि कछुआ (Tortoise) तो बहुत धीरे- धीरे चल था ।
सब को लगता था की रेस में खरगोश (Rabbit) ही जीतेगा क्योंकि कछुआ (Tortoise) तो बहुत धीरे- धीरे चलता था ।

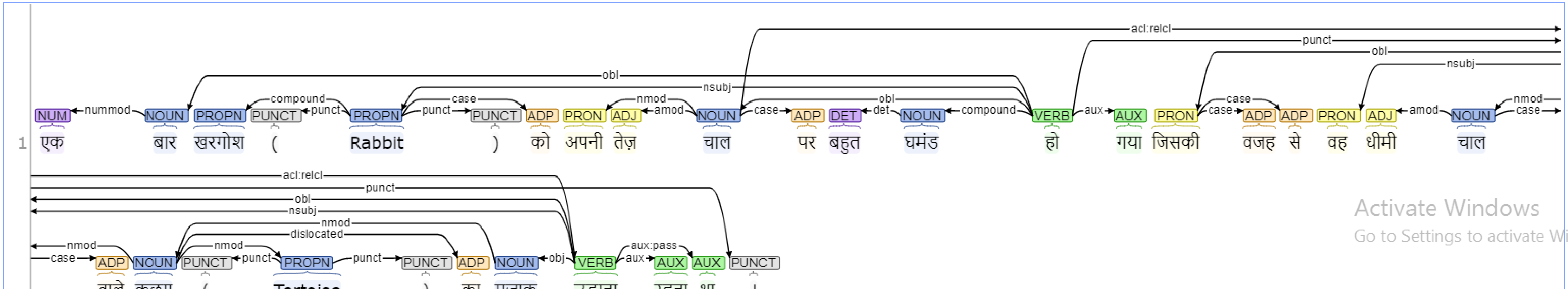
5 सब जानवर कछुआ का उत्साह बिा लग और कुछ ही देर में रेस शुरू हो जा ।
सब जानवार कछुए का उत्साह बढ़ाने लगे और कुछ ही देर में रेस शुरू हो गई ।

6 रेस शुरू हो ही खरगोश और कछुआ ने दौड़ना शुरू कर और थोड़ी ही देर में खरगोश सब जानवर का आँख से ओझल हो जा और कछुआ अपना चाल से चल रह ।
रेस शुरू होते ही खरगोश और कछुए ने दौड़ना शुरू किया और थोड़ी ही देर में खरगोश सब जानवरों की आँखों से ओझल हो गया और कछुआ अपनी चाल से चलता रहा ।

Named Entity Recognition:

NER is not available for this language at this time.

Universal Dependencies:

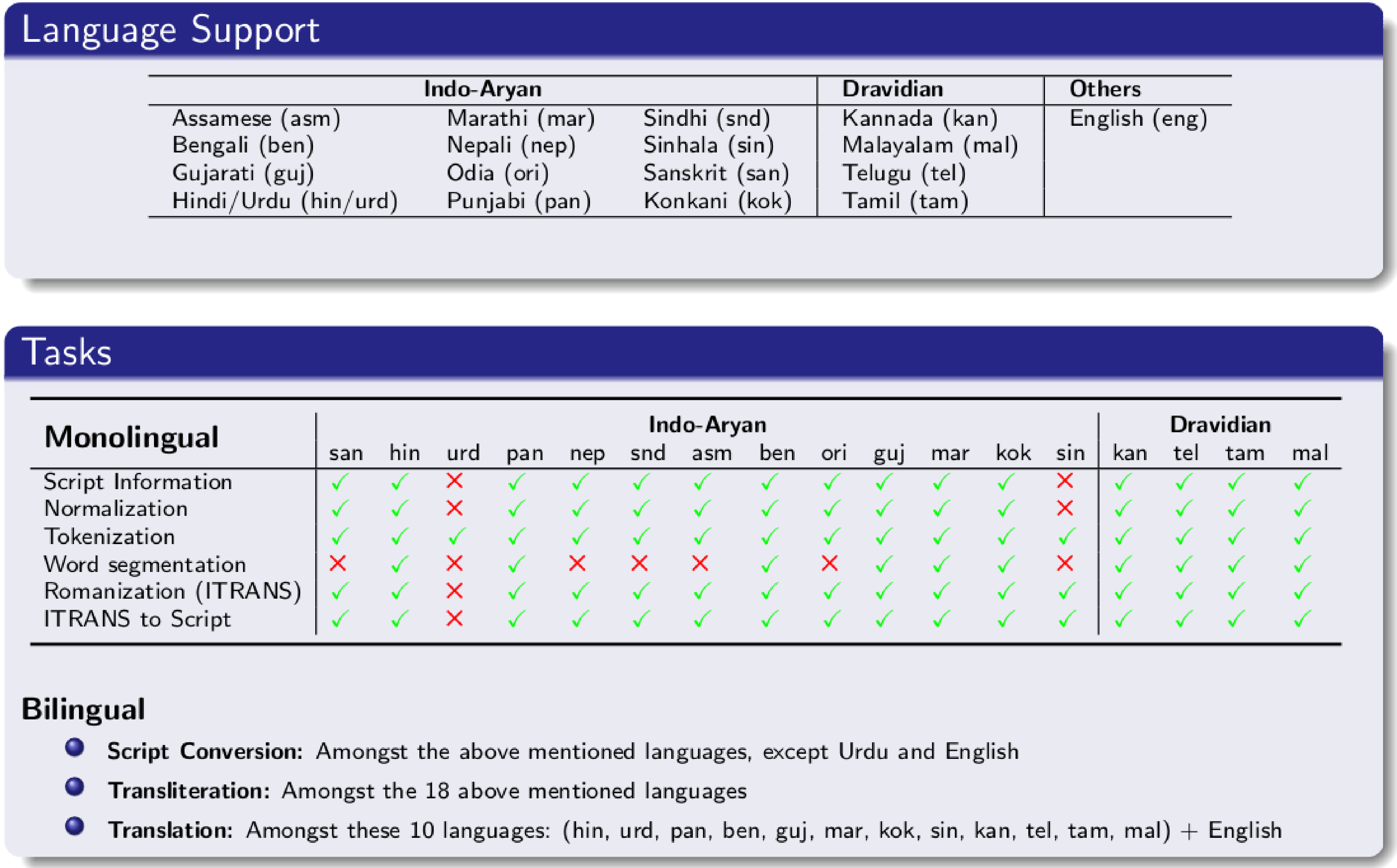


iNLTK: Natural Language Toolkit for Indic Languages

- Created by Gaurav Arora
- iNLTK aims to provide out of the box support for various NLP tasks that an application developer might need for Indic languages.
- Supports native languages and code mixed langauges
- inltk is currently supported only on Linux and Windows 10 with Python >= 3.6

Source: https://inltk.readthedocs.io/en/latest/
https://github.com/goru001/inltk

Indic-nlp-library features



Language Support

Indo-Aryan			Dravidian	Others
Assamese (asm)	Marathi (mar)	Sindhi (snd)	Kannada (kan)	English (eng)
Bengali (ben)	Nepali (nep)	Sinhala (sin)	Malayalam (mal)	
Gujarati (guj)	Odia (ori)	Sanskrit (san)	Telugu (tel)	
Hindi/Urdu (hin/urd)	Punjabi (pan)	Konkani (kok)	Tamil (tam)	

Tasks

Monolingual	Indo-Aryan													Dravidian			
	san	hin	urd	pan	nep	snd	asm	ben	ori	guj	mar	kok	sin	kan	tel	tam	mal
Script Information	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓
Normalization	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓
Tokenization	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Word segmentation	✗	✓	✗	✓	✗	✗	✗	✓	✗	✓	✓	✓	✗	✓	✓	✓	✓
Romanization (ITRANS)	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ITRANS to Script	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Bilingual

- **Script Conversion:** Amongst the above mentioned languages, except Urdu and English
- **Transliteration:** Amongst the 18 above mentioned languages
- **Translation:** Amongst these 10 languages: (hin, urd, pan, ben, guj, mar, kok, sin, kan, tel, tam, mal) + English

Source: https://anoopkunchukuttan.github.io/indic_nlp_library/

Indic-nlp-library features

- Created by Anoop Kunchukuttan
- The goal of the Indic NLP Library is to build Python based libraries for common text processing and Natural Language Processing in Indian languages. Indian languages share a lot of similarity in terms of script, phonology, language syntax, etc. and this library is an attempt to provide a general solution to very commonly required toolsets for Indian language text

Source: https://anoopkunchukuttan.github.io/indic_nlp_library/

http://nbviewer.ipython.org/url/anoopkunchukuttan.github.io/indic_nlp_library/doc/indic_nlp_examples.ipynb

Polyglot NLP Library

- Polyglot is a natural language pipeline that supports massive multilingual applications.
- Developed by Rami Al-Rfou
- Features
  - Tokenization (165 Languages)
  - Language detection (196 Languages)
  - Named Entity Recognition (40 Languages)
  - Part of Speech Tagging (16 Languages)
  - Sentiment Analysis (136 Languages)
  - Word Embeddings (137 Languages)
  - Morphological analysis (135 Languages)
  - Transliteration (69 Languages)

Source: https://pypi.org/project/polyglot/
https://github.com/aboSamoor/polyglot

Summary

- Performing NLP tasks on multiple human languages at a time is hard especially when the text includes mixed languages
- The information need to extracted from multiple sources and multiple languages and should be converted to common language.
- Multi lingual NLP helps to generate output in a target language.
- There are various libraries offering different features. Not a single library offers all features.