
Romanagari Detection in Twitter

14 Oct 2015

— Hrishikesh Terdalkar - Shubhangi Agarwal —

Motivation

- Why Twitter?
- Most NLP techniques deal with English text only
- Tweets are often of the form:

"Yeh kaisi field placings lagayi hain? Powerplay mein

slip? Via @ARangarajan1972 #IndvsPak "

Romanagari = Noise

Goal

- Collect and create a quality tweet-dataset containing Romanagari words
- Romanagari Text Detection
- (possibly) Translate to English language

Languages Targeted

- Hindi

Steps

Create a dictionary of Romanagari words



Detect Romanagari text mixed with English text



Translate to English

Sounds easy?!

Challenges

1. Data Collection

- i. Search terms
- ii. Noise (different languages)
- iii. Disambiguation (polysemy in Hindi and English)

2. Detect and differentiate between English and Romanagari text

- i. Phonetic typing
- ii. SMS language
- iii. Spelling errors
- iv. Disambiguation

Challenges

3. Handle commonly occurring inflections in the social media text
 - i. whatttttt!, whennnn?!, kyunnnn?!
 - ii. mann, bool, bol

4. And many more (yet to be encountered)

Approach

1. Data

- i. Collection
 - Frequent Romanagari words
 - Tweepy
 - SMS language
- ii. Synthetic Generation

2. Language detection/correction

- i. Tools available (PyEnchant, langid, langdetect, guess-language etc)

3. Almost phonetic representations

- i. Metaphone
 - ii. Double Metaphone
 - iii. Soundex
- Also used for Romanagari text detection

Strategies

- Find frequently used Romanagari words in tweets/social media.
(Different from “most frequent” Hindi words from other corpora such as books / wiki)
- Try to obtain annotated-datasets from social media such as facebook from existing papers and frequency analysis on this smaller “spoken-hindi” dataset.
- Context analysis (if possible)
 - n-grams

So far..

- Python
- Twitter collection
 - ◆ most frequent hindi words as FILTER
 - ◆ low success rate on tweets + lot of noise
 - ◆ explored synthetic generation ^[3]
- Exploration of existing classifiers
 - ◆ *PyEnchant*: a spellchecking library for Python based on Enchant
 - ◆ *langdetect*: python implementation of “language-detection” Java library
 - ◆ *langid*: language identification, n-gram, 97 languages, scores for multiple languages
- Soundex / Metaphone Experiments

Soundex vs Double Metaphone

"kyun"

```
>>> s("kyun")
'K50000'
>>> s("kyunn")
'K50000'
>>> s("kyunnnnn")
'K50000'
```

```
>>> doublemetaphone("kyun")
('KN', '')
>>> doublemetaphone("kyunn")
('KN', '')
>>> doublemetaphone("kyunnnnn")
('KNNN', '')
```

"haan"

```
>>> s("haan")
'H50000'
>>> s("haaaannn")
'H50000'
```

```
>>> doublemetaphone("haan")
('HN', '')
>>> doublemetaphone("haaaannn")
('HNN', '')
```

"what"

```
>>> s("what")
'W30000'
>>> s("wwhaaattt")
'W30000'
```

```
>>> doublemetaphone("what")
('AT', '')
>>> doublemetaphone("wwhaaattt")
('TT', '')
```

Soundex vs Double Metaphone

"burp"

```
>>> s("burp")  
'B61000'  
>>> s("burrpppp")  
'B61000'
```

```
>>> doublemetaphone("burp")  
( 'PRP', '' )  
>>> doublemetaphone("burrpppp")  
( 'PRRPP', '' )
```

"lol"

```
>>> s("lol")  
'L40000'  
>>> s("lollll")  
'L40000'
```

```
>>> doublemetaphone("lol")  
( 'LL', '' )  
>>> doublemetaphone("lollll")  
( 'LLL', '' )
```

"boom"

```
>>> s("booooooooooom")  
'B50000'  
>>> s("boom")  
'B50000'  
>>> s("boon")  
'B50000'
```

```
>>> doublemetaphone("booooooooooom")  
( 'PM', '' )  
>>> doublemetaphone("boom")  
( 'PM', '' )  
>>> doublemetaphone("boon")  
( 'PN', '' )
```

"ah / oh"

```
>>> s("ah")  
'A00000'  
>>> s("aaahhh")  
'A00000'  
>>> s("ohhh")  
'000000'  
>>> s("ooohhh")  
'000000'
```

```
>>> doublemetaphone("ah")  
( 'A', '' )  
>>> doublemetaphone("aaahhh")  
( 'A', '' )  
>>> doublemetaphone("ohhh")  
( 'A', '' )  
>>> doublemetaphone("ooohhh")  
( 'A', '' )
```

Tweet Collection

The image shows a code editor with a Python script named `streaming_simple.py` and a file explorer on the right. The script is designed to collect tweets from a Twitter stream and save them to a file.

```
1 from tweepy.streaming import StreamListener
2 from tweepy import OAuthHandler
3 from tweepy import Stream
4 import json
5 from auth import TwitterAuth
6 import numpy
7
8 #Very simple (non-production) Twitter stream example
9 #1. Download / install python and tweepy (pip install tweepy)
10 #2. Fill in information in auth.py
11 #3. Run as: python streaming_simple.py
12 #4. It will keep running until the user presses ctrl+c to exit
13 #All output stored to output.json (one tweet per line)
14 #Text of tweets also printed as recieved (see note about not doing this in production (final) code
15
16 class StdOutListener(StreamListener):
17
18     #This function gets called every time a new tweet is received on the stream
19     def on_data(self, data):
20         #Just write data to one line in the file
21         fhOut.write(data)
22
23         #Convert the data to a json object (shouldn't do this in production; might slow down and miss tweets)
24         j=json.loads(data)
25
26         #See Twitter reference for what fields are included -- https://dev.twitter.com/docs/platform-objects/tweets
27         text=j["text"] #The text of the tweet
28         print(text) #Print it out
29
30     def on_error(self, status):
31         print("ERROR")
32         print(status)
33
34 if __name__ == '__main__':
35     try:
36         #Create a file to store output. "a" means append (add on to previous file)
37         fhOut = open("data/output.json", "a")
38
39         #Create the listener
40         l = StdOutListener()
41
42         auth = OAuthHandler(TwitterAuth.consumer_key, TwitterAuth.consumer_secret)
43         auth.set_access_token(TwitterAuth.access_token, TwitterAuth.access_token_secret)
44
45         #Connect to the Twitter stream
46         stream = Stream(auth, l)
47
48         #Terms to track
49         # words = numpy.loadtxt('hindi.txt', dtype=str, delimiter=' ', usecols = (0,))
50         term_file = open("terms2.txt", "r")
51         words = term_file.read().splitlines()
52         stream.filter(track= words)
53
54         #Alternatively, location box for geotagged tweets
55         #stream.filter(locations=(-0.530, 51.322, 0.231, 51.707))
56
57     except KeyboardInterrupt:
58         #User pressed ctrl+c -- get ready to exit the program
59         pass
```

The file explorer on the right shows the following files and folders:

Name	Size	Type	Date Modified
data		Folder	14/10/15 4:47 AM
raw		Folder	14/10/15 3:36 AM
sample		Folder	14/10/15 5:20 AM
auth_example.py	362 bytes	py File	14/10/15 2:49 AM
auth.py	520 bytes	py File	14/10/15 2:14 AM
auth.pyc	622 bytes	pyc File	14/10/15 2:49 AM
data2metions_retweet_network.py	2 KB	py File	14/10/15 1:12 AM
data2spreadsheet.py	3 KB	py File	14/10/15 1:12 AM
hindi_100.txt	1 KB	txt File	14/10/15 3:46 AM
hindi.txt	1 KB	txt File	14/10/15 3:46 AM
LICENSE	17 KB	File	14/10/15 1:12 AM
nlp_functions.sh	141 bytes	sh File	14/10/15 3:41 AM
README.md	5 KB	md File	14/10/15 1:12 AM
search_generic.py	2 KB	py File	14/10/15 1:12 AM
streaming_simple.py	1 KB	py File	14/10/15 4:46 AM
streaming.py	4 KB	py File	14/10/15 1:12 AM
terms.txt	696 bytes	txt File	14/10/15 4:28 AM

The console shows the following output:

```
Python 1 04:16:47
JITNA TWEETNA HAI AAJ HI TWEET LO PHIR
RT @sabtv: Mahavinashini ne aazad kiya jin Fitoori ko. Agar apko maangni hoti usse ek wish, to kya hoti woh? #Baalveer http://t.co/xM82FQ0a..
abang gw tersayang nih @jinyngd_cmm dia yg bisa banget gw rasain perhatiannya kya abang rl lah :) walau gw kya orng bego mt tpe a off rp
I took a pic with the jeepers creepers looking shit now my ass gonna be possessed https://t.co/5KpGSqo0hg
How I Manage 5 Kids and a Growing Business: ...risks I take are smart ones.2. Don't work from home.When we fir... http://t.co/ijj4DFM
RT @MamaMia_bgr: Kenapa ya dari semua pasien aku kalo pulang cium kening trus selalu bilang "JANGAN NAKAL YA" kya serasa bocah c serasa p..
RT @cheenee_bruce: Wow naman this vid clip made my day!Hhhmm naicp ko sa knya kya tlga galing ung "MAALDEN KITA"#ALDUBWayBackHc https://-
RT @TeamMaIden: With great power comes great responsibility. Kng nasan man si M&P;A 2day is bcos of us. Kya dpt naiintndhan ny AdminKendz
...
FIR against Union Minister Giriraj Singh for poll malpractice http://t.co/uDAxKPz1et
#JeetegaNitishJhumeGadDesh http://t.co/ZLNDLrn0FX
exit
>>> Use exit() or Ctrl-D (i.e. EOF) to exit
```

Plan

- Better dataset collection strategies
- Better synthetic generation than mentioned in [3]
- Perform experiments to test feasibility of Soundex/Metaphone for Hindi
- Pre-processing tweets followed by language identifiers with modification
- Compose a list of Hindi-specific disambiguation rules
- Detect Romanagari words
- Annotate / Attach English meaning to Romanagari words

References

1. Barman, Utsab, et al. "Code Mixing: A Challenge for Language Identification in the Language of Social Media." *EMNLP 2014* (2014): 13.
2. Gella, Spandana, Jatin Sharma, and Kalika Bali. "Query word labeling and back transliteration for indian languages: Shared task system description." *FIRE Working Notes* (2013).
3. Gella, Spandana, Kalika Bali, and Monojit Choudhury. "'ye word kis lang ka hai bhai?' Testing the Limits of Word level Language Identification."
4. Das, Amitava, and Björn Gambäck. "Identifying Languages at the Word Level in Code-Mixed Indian Social Media Text." *Proceedings of the 11th International Conference on Natural Language Processing, Goa, India*. 2014.
5. Han, Bo, and Timothy Baldwin. "Lexical normalisation of short text messages: Makn sens a# twitter." *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011.
6. Proceedings of Social india 2014
7. Tweepy: <https://github.com/tweepy/tweepy>
8. Chaware, Sandeep, and Srikantha Rao. "Rule-Based Phonetic Matching Approach for Hindi and Marathi." *Computer Science & Engineering*, 1.3 (2011), AIRCC

Questions?



Thank You
