Subsegmental language detection in Celtic language text

Akshay Minocha
IIIT-Hyderabad(India)
akshay.minocha@students.iiit.ac.in

Francis Tyers
UiT Norgga árktalaš universitehta
francis.tyers@uit.no

Special Thanks to,
Kevin Scannell

Introduction

We aim to perform language identification on sub segmental basis -

- Typical case is to detect the language of documents and sentences.
- We are focussing on cases where - A single sentence may have different **code switching points**

- **[en** You're a**]** **[ga** Meiriceánach, cén fáth**]** **[en** are you**]** **[ga** foghlaim Gaeilge?!**]**
- @afaltomkins **[cy** gorfod cael bach o tan**]** **[en** though init**]**
- **[en** omg**]** **[cy** mar cwn bach yn**]** **[en** black and tan**]** **[cy** a popeth,**]** **[en** even cuter!!**]**

Dataset

Simplifying the task by taking into account Celtic languages and a corresponding majority language.

Manual annotation of about 40-50 tweets for each of the three language pairs.

Pair	Language	Statistics (%)	
		Tokens	Segments
Irish—English	Irish	332	40
	English	379	42
Welsh—English	Welsh	419	64
	English	378	66
Breton—French	Breton	388	54
	French	379	53

Chunking algorithm

Require: **s** : ***sentence to chunk***

buffer = [ ] */*Undecided expanding window of chunk*/*
chunks = [ ] */*Decided labelled segment*/*
buffer language ← `LANGPREDICT`(s[0]) */*Language of first word */*
flag ← 0
for all *w* ∈ s do
  if `LANGPREDICT`(*w*)=buffer language **then**
    if flag = 1 **then**
      buffer ← buffer + [word buffer, *w*]
      flag ← 0
    else
      buffer ← buffer + [*w*]
    if `LANGPREDICT`(*w*)≠buffer language **then**
      if flag= 0 **then**
        flag ← 1
        word buffer ← w
        continue
      else
        chunks←chunks + [(buffer,buffer language)]
        buffer←[word buffer,w]
        buffer language←`LANGPREDICT`(w) */*Language of new expanding chunk */*
        flag ← 0
if length(buffer)≠0 **then**
  chunks ← chunks + [(buffer,buffer language)]

Methodology

Alphabet n-gram Approach -

- Character Language model
- Using IRSTLM we build a language model for the five languages
- For English and French - Europarl [1]
- Breton, Welsh and Irish - Corpora of text crawled from the web
- Size of the corpus from which this language model was built - 1.5 million tokens

Example - 'slainte!' would be -
{ 's', 's l', 'l a', ' ´ a i', 'i n', 'n t', 't e', 'e !', '! ´ '}

Word-based Prediction -

- Generate word lists for the languages using aspell which is widely used on Unix systems.
- Word are labeled according to their presence in the particular word list.
- In case of a confusion the word is added to the previous segment

Word-based prediction with character backoff

- Same as Word-based prediction, but in case of confusion this falls back to the Alphabet bi-gram approach.

Baseline

- Using `langid.py`[2] labeled all the lines in a particular dataset according to the majority classification

`Langid` character trigram prediction

- Trigram probabilities from `langid` were taken into account.
- All other heuristics and chunking algorithm are same as for other methods.

References

[1] Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In MT summit, volume 5, pages 79–86.

[2] Marco Lui and Timothy Baldwin. 2012. langid. py: An off-the-shelf language identification tool.

[3] Ben King and Steven P Abney. 2013. Labeling the languages of words in mixed-language documents using weakly supervised methods. In HLT-NAACL, pages 1110–1119.

Evaluation

- We followed the footsteps of ***CoNLL 2000*** shared task on language independent named entity recognition.
- Divide the text into non-overlapping segments.
- **Precision** - percentage of correctly detected phrases.
- **Recall** - number of phrases in the data that were found by the chunker.

Results

System		Irish—English		Welsh—English		Breton—French	
		Irish	English	Welsh	English	Breton	French
`baseline`	*p*	2.50	0.0	0.0	0.0	0.0	0.0
	r	2.56	0.0	0.0	0.0	0.0	0.0
`langid-3character`	*p*	5.00	14.29	0.0	21.21	1.85	20.75
	r	5.41	8.45	0.0	14.58	1.92	12.36
`wordlist`	*p*	32.50	28.57	26.69	**40.91**	57.41	33.96
	r	23.64	26.09	**26.03**	**33.75**	47.69	33.33
`character bigram`	*p*	32.50	35.71	23.44	19.70	57.41	52.83
	r	22.41	26.79	15.31	16.67	41.33	37.84
`wordlist+character bigram`	*p*	**52.50**	**50.00**	**32.81**	31.82	**70.37**	**67.92**
	r	**38.18**	**43.75**	24.14	25.61	**57.58**	**57.14**

System	Accuracy (%)		
	Irish—English	Welsh—English	Breton—French
`baseline`	42.76	42.16	44.07
`langid-3character`	57.24	45.92	43.16
`wordlist`	79.75	**74.28**	83.96
`character bigram`	81.29	65.62	76.79
`wordlist+character bigram`	**85.79**	72.40	**88.79**

Conclusions

- A very preliminary investigation into subsegment language identification in Celtic language texts.
- We would like to include supervised methods and features talked about by King and Abney (2013) [3]
- We would also like to check our methods with higher order n-grams and more options in backoff.
- Explore a lattice technique where each word is a lattice node and the inclusions of the words are done using probability.