A Bambara Tonalization System for Word Sense Disambiguation Using Differential Coding, Segmentation and Edit Operation Filtering

Luigi (Y.-C.) Liu   Damien Nouvel

ER-TIM, INALCO, 2 rue de Lille, Paris, France

IJCNLP, 2017

Outline I

Introduction

- Bambara: african language with 4 tones (´ ` ˇ ˆ)
- Orthography: official orthography does not represent tones.
- Word sense more ambiguous: some unaccented tokens can correspond to several tonalized forms $\longrightarrow$ challenge for NLP applications.
- Goal: implemente an automatic tonalizer for Bambara
    - Improve subsequent NLP processings
    - Facilitate linguistic analysis for Bambara.

Bambara Reference Corpus I

- Bambara Reference Corpus consists 2 parts :
  - a non-disambiguated subcorpus
  - a man-annotated subcorpus

Part	Words (dist.)
Non-disamb.	2160M (58M)
Disamb.	358M (23M)

Table: Corpus statistics

Tonalization	38.73%
Other	**8.90%**
None	52.35%

Table: Annotation statistics

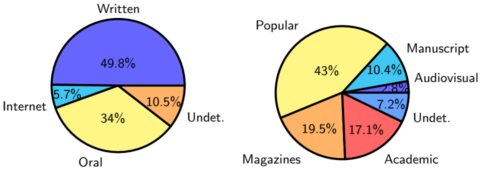

Figure: Corpus composition (medium, source)

Bambara Reference Corpus II

- Wordform annotation is done for each wordform and for 3 main features:
  - POS Tagging
  - Tone Marker Restoration
  - Gloss Assignment
- For POS tagging, we use conditional random fields (CRFs; Lafferty et al. 2001) for sequential modeling
  - Over 23 morpho-syntactic possible tags
  - Accuracy: 94% (satisfying for under-resourced language)
- For the tone marker restoration task, we considered using similar methods
  - Over 20,870 distinctive tonal forms
  - Learning is quiet inefficent due to large-scale label set
- Problem: the drawback of modeling sequences of large-scale label set (of tonal form) is the expensive computational cost needed to estimate CRF parameters.

Related Works

- Word-level modeling (Simard (1998), Tufis and Chitu (1990))
  - French Accent insertion : 2-layers Hidden Markov Model (HMM)
  - Romanian Automatic diacrtization : 3-gram tagger
- Word and character levels modeling (Elshafei et al. (2006), Scannell (2011), Nguyen et al. (2012))
  - Arabic Diacritization : 1-layer HMM
  - Uni-codification for African languages : Naive Bayes classifier.
  - Vietnamese Accent restoration : CRFs and other
- Hybrid approaches (Said et al. (2013), Metwally et al. (2016))
  - CRF + morphological analyzer
  - CRF + HMM + morphological analyzer
- Category Decomposition (Tellier et al., 2010)
  - Decompose label set in smaller pieces to train separately.
  - Result : time-wise efficiency improvement at train phase.

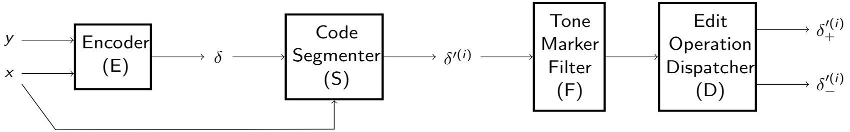

Figure: Block diagram for the proposed Bambara tonalization system at training stage

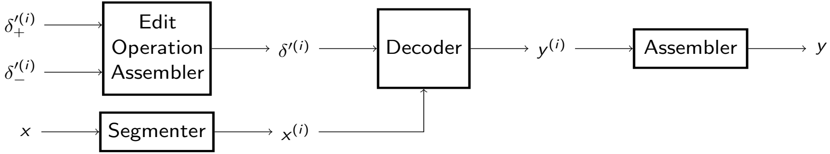

Figure: Block diagram for the proposed Bambara tonalization system at tonalization stage

Fundemental definitions I

- Discrete random variables

$$X \longrightarrow \textcolor{red}{\text{non-tonalized token}} : \textit{kelen}$$
$$Y \longrightarrow \textcolor{red}{\text{tonalized token}} : \textit{kèlen}(\text{adj. same}), \textit{kèlén}(\text{intj. already})$$
$$\Delta \longrightarrow \textcolor{red}{\text{differential code}} : (+1, 2, \acute{}\,), (+1, 2, \acute{}\,)(+1, 4, \grave{}\,)$$

- Mappings

$$\Delta = E(Y; X) \longrightarrow \textcolor{red}{\text{encoder}} \text{ function}$$
$$Y = D(\Delta; X) \longrightarrow \textcolor{red}{\text{decoder}} \text{ function}$$
$$Y = D(E(Y; X); X)$$

- Note: predict differential code Δ, recovery Y from Δ by decoder D.

Tonalization as Edit Operation I

- Code Δ can be
  - either $\varnothing$ (when $X = Y$, 52.35%)
  - or a concatenation of codewords like $\sigma_1\sigma_2\sigma_3, \ldots$
- A codeword σ is a triplet (m, p, c) containing
  - m: operation type ($+1$ for insertion, -1 for deletion)
  - p: position for operation, a positive integer
  - c: character (if insertion), $c \in \Omega$
- Encoder $E(y; x) =$ Applying W.-F. algorithm[1] (Wagner and Fischer, 1974) on (x, y) to produce the code δ
- Decoder $D(\delta; x) =$ Applying edit operations in δ on x to get tonalized token y

[1] In this article, we apply Wagner-Fischer algorithm in its special case where there are only 2 available edit operations against 3 edit operations including the substitution as in its general case.

Segmentation

- To facilite learning processing, segmentation is introduced to divide data pair (x, δ) to train in several segments of data pair $(x^{(i)}, \delta^{(i)})$ where i is segment id.
- Learning on segments of data pair is easier beacuse that there is less edit operations to predict and this facilite our tonalization modeling.
- The segmentation mode w
  - $w = -1$ indicates a syllabification (by morphological parser)
  - $w = 0$ for no segmentation
  - $w > 0$ specifies a w-width regular segmentation[2].

[2] A regular segmenter forms a segment of every w successive characters, from left to right (i.e. in direction of writing of Bambara), in its input string. By exception, the last segment at output contains the rest of the string which has not yet been segmented so that we allow it to be equal or shorter than a segment of w characters.

Edit Operation Filtering

- Annotators also introduce : typographic, orthographic corrections.
- Focus on tonalization operations $\longrightarrow$ filtering on edit operations.
- Tone Marker Filtering: for each position of input string,
  - Remove all insertions except for tone markers
  - Keep only the 1st of tone insertions
  - Keep only the 1st of tone deletions
- Edit operation dispatcher F_m: it gives from input code δ_{in} a sub-sequence composed of operations of type m, $m = -1, +1$
- If δ_{in} is a filtered result, inverse mapping from $\{F_{-1}(\delta_{in}), F_{+1}(\delta_{in})\}$ to δ_{in} exists. What we call edit operation assembler.

Experiment Result I

- About half (52.35%) of tokens in BRC do not need any tone markers

Sys. \ w	-1 (Syll.)	1	2	3	4	0
Majority vote	**0.843**					
S ∘ E	**0.923**	0.915	**0.922**	**0.922**	0.917	0.893
time	101.63	25.52	42.03	235.35	378.37	2683.72
D ∘ F ∘ S ∘ E	**0.923**	0.912	**0.923**	**0.923**	0.918	0.893
time	19.88	17.62	13.17	15.67	19.62	261.83

Table: Accuracy for our system trained with four different system configurations and eight segmentation modes (p = 50%)

Experiment Result II

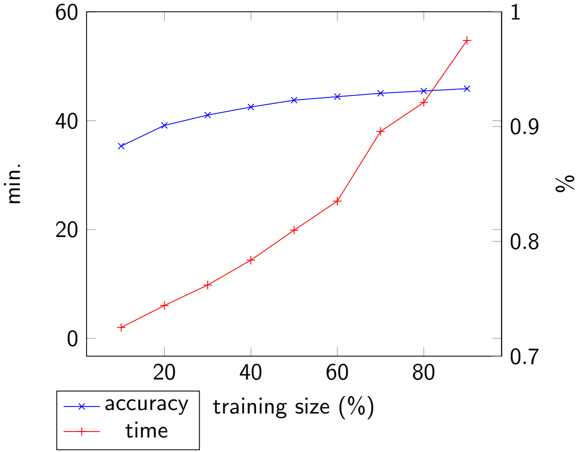

Figure: Accuracy and time of training (configurated as $D \circ F \circ S \circ E$ using syllabification) with respect to different training size 90%-10%

Experiment Result III

Error Type	Ratio
Tone Only	**58.52%**
Position Only	1.17%
Tone and Position	0.023%
Silence	**40.08%**

Table: Error dist. by type for insertion opt. with p = 50%, system = $D \circ F \circ S \circ E$

		Predicted			
		´	`	ˆ	ˇ
Actual	´	0.9541	**0.0438**	0.0021	0.0000
	`	**0.0841**	0.9141	0.0015	0.0003
	ˆ	0.0035	**0.0322**	0.9643	0.0000
	ˇ	0.0000	**0.0952**	0.0000	0.9048

Table: Confusion matrix on prediction of tone markers

Conclusion

- Differential encoder :
  - Reduce entropy of labels to be predicted, make CRF learning efficient
  - Allow to implement tone marker filter, edit operation decomposition
- Segmentation :
  - Increase tonalization accuracy
  - Greatly reduce training time
- Tone marker filter :
  - Normalize the tonalized token
  - Lead to reduce training time
- Edit operation decomposition unit (dispatcher) :
  - Split the tokens in insertion and deletion of tone markers
  - Allows to accelerate furthermore the training time reduction

Perspectives

- Take into account more linguistic information for bambara
- Generalization for other languages like French, Arabic, Yoruba, etc.
- Avaliable ressources and tools :
  - Bambara Reference Corpus (French) : `http://cormand.huma-num.fr/index.html`
  - Tonalizer - CRF-based Tone Reconstitution Tool (English): `https://github.com/vieenrose/tonalizer`