ϵ-extension Hidden Markov Models and Weighted Transducers for Machine Transliteration

Balakrishnan Vardarajan
Dept. of Electrical and Computer Engineering
Johns Hopkins University
bvarada2@jhu.edu

Delip Rao
Dept. of Computer Science
Johns Hopkins University
delip@cs.jhu.edu

Abstract

We describe in detail a method for transliterating an English string to a foreign language string evaluated on five different languages, including Tamil, Hindi, Russian, Chinese, and Kannada. Our method involves deriving substring alignments from the training data and learning a weighted finite state transducer from these alignments. We define an ϵ-extension Hidden Markov Model to derive alignments between training pairs and a heuristic to extract the substring alignments. Our method involves only two tunable parameters that can be optimized on held-out data.

1 Introduction

Transliteration is a letter by letter mapping of one writing system to another. Apart from the obvious use in writing systems, transliteration is also useful in conjunction with translation. For example, machine translation BLEU scores are known to improve when named entities are transliterated. This engendered several investigations into automatic transliteration of strings, named entities in particular, from one language to another. See Knight and Graehl(1997) and later papers on this topic for an overview.

Hidden Markov Model (HMM) (Rabiner, 1989) is a standard sequence modeling tool used in various problems in natural language processing like machine translation, speech recognition, part of speech tagging and information extraction. There have been earlier attempts in using HMMs for automatic transliteration. See (Abdul Jaleel and Larkey, 2003; Zhou et al., 2008) for example. In this paper, we define an ϵ-extension Hidden Markov Model that allows us to align source and target language strings such that the characters in the source string may be optionally aligned to the ϵ symbol. We also introduce a heuristic that allows us to extract high quality sub-alignments from the ϵ-aligned word pairs. This allows us to define a weighted finite state transducer that produces transliterations for an English string by minimal segmentation.

The overview of this paper is as follows: Section 2 introduces ϵ-extension Hidden Markov Model and describes our alignment procedure. Section 3 describes the substring alignment heuristic and our weighted finite state transducer to derive the final n-best transliterations. We conclude with a result section describing results from the NEWS 2009 shared task on five different languages.

2 Learning Alignments

The training data $\mathcal{D}$ is given as pairs of strings $(\mathbf{e}, \mathbf{f})$ where $\mathbf{e}$ is the English string with the corresponding foreign transliteration $\mathbf{f}$. The English string $\mathbf{e}$ consists of a sequence of English letters $(e_1, e_2, \ldots, e_N)$ while $\mathbf{f} = (f_1, f_2, \ldots, f_M)$.

We represent $\mathcal{E}$ as the set of all English symbols and $\mathcal{F}$ as the set of all foreign symbols.[1] We also assume both languages have a special null symbol ϵ, that is $\epsilon \in \mathcal{E}$ and $\epsilon \in \mathcal{F}$.

Our alignment model is a Hidden Markov Model $\mathcal{H}(X, Y, \mathcal{S}, \mathbf{T}, \mathbf{P}_s)$, where

- X is the start state and Y is the end state.
- $\mathcal{S}$ is the set of emitting states with $S = |\mathcal{S}|$. The emitting states are indexed from 1 to S. The start state X is indexed as state 0 and the end state Y is indexed as state $S + 1$.
- $\mathbf{T}$ is an $(S + 1) \times (S + 1)$ stochastic matrix with $\mathbf{T} = [t_{ij}]$ for $i \in \{0, 1, \ldots, S\}$ and $j \in \{1, 2, \ldots, S + 1\}$.

[1] Alphabets and diacritics are treated as separate symbols.

Proceedings of the 2009 Named Entities Workshop, ACL-IJCNLP 2009, pages 120–123,
Suntec, Singapore, 7 August 2009. ©2009 ACL and AFNLP

- $\mathbf{P}_s = [p_{ef}]$ is an $|\mathcal{E}| \times |\mathcal{F}|$ matrix of joint emission probabilities with $p_{ef} = P(e, f|s)$ $\forall s \in \mathcal{S}$.

We define $\tilde{\mathbf{s}}$ to be an ϵ-extension of a string of characters $\mathbf{s} = (c_1, c_2, \ldots, c_k)$ as the string obtained by pumping an arbitrary number of ϵ symbols between any two adjacent characters c_l and c_{l+1}. That is, $\tilde{\mathbf{s}} = (d_{i_1}, \ldots, d_{i_2}, \ldots, d_{i_k})$ where $d_{i_j} = c_j$ and $d_l = \epsilon$ for $i_m < l < i_{m+1}$ where $1 \leq l < k$. Observe that there are countably infinite ϵ-extensions for a given string $\mathbf{s}$ since an arbitrary number of ϵ symbols can be inserted between characters c_m and c_{m+1}. Let $\mathcal{T}(\mathbf{s})$ denote the set of all possible ϵ-extensions for a given string $\mathbf{s}$.

For a given pair of strings $(\mathbf{u}, \mathbf{v})$, we define a *joint* ϵ-extension of $(\mathbf{u}, \mathbf{v})$ as the pair $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ s.t. $\tilde{\mathbf{u}} \in \mathcal{T}(\mathbf{u})$ and $\tilde{\mathbf{v}} \in \mathcal{T}(\mathbf{v})$ with $|\tilde{\mathbf{u}}| = |\tilde{\mathbf{v}}|$ and $\nexists i$ s.t. $\tilde{u}_i = \tilde{v}_i = \epsilon$. Due to this restriction, there are finite ϵ-extensions for a pair $(\mathbf{u}, \mathbf{v})$ with the length of $\tilde{\mathbf{u}}$ and $\tilde{\mathbf{v}}$ bounded above by $|\mathbf{u}| + |\mathbf{v}|$. [2] Let $J(\mathbf{u}, \mathbf{v})$ denote the set of all joint ϵ-extensions of $(\mathbf{u}, \mathbf{v})$.

Given a pair of strings $(\mathbf{e}, \mathbf{f})$ with $\mathbf{e} = (e_1, e_2, \ldots, e_N)$ and $\mathbf{f} = (f_1, f_2, \ldots, f_M)$, we compute the probability $\alpha(\mathbf{e}, \mathbf{f}, s')$ that they are transliteration pairs ending in state s' as

$$\alpha(\mathbf{e}, \mathbf{f}, s') = \sum_{(\tilde{\mathbf{e}}, \tilde{\mathbf{f}}) \in J(\mathbf{e}, \mathbf{f})} \sum_{0 = s_0, \ldots, s_{|\tilde{\mathbf{e}}|} = s'} t_{0, s_1} \prod_{i=1}^{|\tilde{\mathbf{e}}|} t_{s_i, s_{i+1}} P(\tilde{e}_i, \tilde{f}_i | s_i)$$

In order to compute the probability $Q(\mathbf{e}, \mathbf{f})$ of a given transliteration pair, the final state has to be the end state $S + 1$. Hence

$$Q(\mathbf{e}, \mathbf{f}) = \sum_{s=1}^{S} \alpha(\mathbf{e}, \mathbf{f}, s) t_{s, S+1} \qquad (1)$$

We also write the probability $\beta(\mathbf{e}, \mathbf{f}, s')$ that they are transliteration pairs starting in state s' as

$$\beta(\mathbf{e}, \mathbf{f}, s') = \sum_{(\tilde{\mathbf{e}}, \tilde{\mathbf{f}}) \in J(\mathbf{e}, \mathbf{f})} \sum_{s' = s_0, \ldots, s_{|\tilde{\mathbf{e}}|+1} = S+1} t_{s_0, s_1} \prod_{i=1}^{|\tilde{\mathbf{e}}|} t_{s_i, s_{i+1}} P(\tilde{e}_i, \tilde{f}_i | s_i)$$

Again noting that the start state of the HMM $\mathcal{H}$ is 0, we have $Q(\mathbf{e}, \mathbf{f}) = \sum_{s=1}^{S} \beta(\mathbf{e}, \mathbf{f}, s) t_{0,s}$. We denote a subsequence of a string $\mathbf{u}$ as $\mathbf{u}_n^m = (u_n, u_{n+1}, \ldots, u_m)$. Using these definitions, we can define $\alpha(\mathbf{e}_1^i, \mathbf{f}_1^j, s)$ as

$$\begin{cases} 1 & i = j = 0, s = 0 \\ 0 & i = j = 0, s \neq 0 \\ t_{0,s} P(e_1, f_1 | s) & i = j = 1 \\ \sum_{s'=1}^{S} t_{s',s} \alpha(\mathbf{e}_1^i, \mathbf{f}_1^{j-1}, s') P(\epsilon, f_j | s) & i = 1, j > 1 \\ \sum_{s'=1}^{S} t_{s',s} \alpha(\mathbf{e}_1^{i-1}, \mathbf{f}_1^j, s') P(e_i, \epsilon | s) & i > 1, j = 1 \end{cases}$$

Finally for $i > 1$ and $j > 1$,

$$\begin{aligned} \alpha(\mathbf{e}_1^i, \mathbf{f}_1^j, s) = & \\ & \sum_{s' \in \mathcal{S}} t_{s',s} [\alpha(\mathbf{e}_1^i, \mathbf{f}_1^{j-1}, s') P(\epsilon, f_j | s) + \\ & \alpha(\mathbf{e}_1^{i-1}, \mathbf{f}_1^j, s') P(e_i, \epsilon | s) + \\ & \alpha(\mathbf{e}_1^{i-1}, \mathbf{f}_1^{j-1}, s') P(e_i, f_j | s)] \end{aligned}$$

Similarly the recurrence for $\beta(\mathbf{e}_i^N, \mathbf{f}_j^M, s)$

$$\begin{cases} t_{s,S+1} & i = N + 1, \\ & j = M + 1 \\ \sum_{s'=1}^{S} t_{s,s'} \beta(\mathbf{e}_i^N, \mathbf{f}_{j+1}^M, s') P(\epsilon, f_j | s') & i = N, j < M \\ \sum_{s'=1}^{S} t_{s,s'} \beta(\mathbf{e}_{i+1}^N, \mathbf{f}_j^M, s') P(e_i, \epsilon | s') & i < N, j = M \end{cases}$$

For $i < N$ and $j < M$, $\beta(\mathbf{e}_i^N, \mathbf{f}_j^M, s) =$

$$\begin{aligned} & \sum_{s' \in \mathcal{S}} t_{s,s'} [\beta(\mathbf{e}_i^N, \mathbf{f}_{j+1}^M, s') P(\epsilon, f_j | s') + \\ & \beta(\mathbf{e}_{i+1}^N, \mathbf{f}_j^M, s') P(e_i, \epsilon | s') + \\ & \beta(\mathbf{e}_{i+1}^N, \mathbf{f}_{j+1}^M, s') P(e_i, f_j | s')] \end{aligned}$$

In order to proceed with the E.M. estimation of the parameters $\mathbf{T}$ and $\mathbf{P}_s$, we collect the *soft* counts $c(e, f|s)$ for emission probabilities by looping over the training data $\mathcal{D}$ as shown in Figure 1.

Similarly the soft counts $c_t(s', s)$ for the transition probabilities are estimated as shown in Figure 2.

Finally the probabilities $P(e, f|s)$ and t_{ij} are re-estimated as

$$\hat{P}(e, f|s) = \frac{c(e, f|s)}{\sum_{e \in \mathcal{E}, f \in \mathcal{F}} c(e, f|s)} \qquad (2)$$

$$\hat{t}_{s',s} = \frac{c_t(s', s)}{\sum_s c_t(s', s)} \qquad (3)$$

We can also compute the most probable alignment $(\tilde{\mathbf{e}}, \tilde{\mathbf{f}})$ between the two strings $\mathbf{e}$ and $\mathbf{f}$ as

[2] $|\tilde{\mathbf{u}}| = |\tilde{\mathbf{v}}| > |\mathbf{u}| + |\mathbf{v}|$ would imply $\exists i$ s.t. $\tilde{u}_i = \tilde{v}_i = \epsilon$ which contradicts the definition of joint ϵ-extension.

$$
\begin{aligned}
c(e,f|s) = & \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{s'} \alpha(\mathbf{e}_1^{i-1},\mathbf{f}_1^{j-1},s')t_{s',s}P(e_i,f_j|s)\beta(\mathbf{e}_i^{N},\mathbf{f}_j^{M},s)\mathbf{1}(e_i=e,f_j=f) \\
& + \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{s'} \alpha(\mathbf{e}_1^{i-1},\mathbf{f}_1^{j},s')t_{s',s}P(e_i,\epsilon|s)\beta(\mathbf{e}_i^{N},\mathbf{f}_j^{M},s)\mathbf{1}(e_i=e,f_j=f) \\
& + \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \sum_{i=1}^{N}\sum_{j=1}^{M}\sum_{s'} \alpha(\mathbf{e}_1^{i},\mathbf{f}_1^{j-1},s')t_{s',s}P(\epsilon,f_j|s)\beta(\mathbf{e}_i^{N},\mathbf{f}_j^{M},s)\mathbf{1}(e_i=e,f_j=f)
\end{aligned}
$$

Figure 1: EM soft count $c(e, f|s)$ estimation.

$$
\begin{aligned}
c_t(s',s) = & \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \sum_{i=1}^{N}\sum_{j=1}^{M} \alpha(\mathbf{e}_1^{i-1},\mathbf{f}_1^{j-1},s')t_{s',s}P(e_i,f_j|s)\beta(\mathbf{e}_i^{N},\mathbf{f}_j^{M},s) \\
& + \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \sum_{i=1}^{N}\sum_{j=1}^{M} \alpha(\mathbf{e}_1^{i-1},\mathbf{f}_1^{j},s')t_{s',s}P(e_i,\epsilon|s)\beta(\mathbf{e}_i^{N},\mathbf{f}_j^{M},s) \\
& + \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \sum_{i=1}^{N}\sum_{j=1}^{M} \alpha(\mathbf{e}_1^{i},\mathbf{f}_1^{j-1},s')t_{s',s}P(\epsilon,f_j|s)\beta(\mathbf{e}_i^{N},\mathbf{f}_j^{M},s) \\
& + \sum_{(\mathbf{e},\mathbf{f})\in\mathcal{D}} \frac{1}{Q(\mathbf{e},\mathbf{f})} \alpha(\mathbf{e}_1^{N},\mathbf{f}_1^{M},s')t_{s',S+1}\mathbf{1}(s=S+1)
\end{aligned}
$$

Figure 2: EM soft count $c_t(s', s)$ estimation.

$$\arg\max_{(\tilde{\mathbf{e}},\tilde{\mathbf{f}})\in J(\mathbf{e},\mathbf{f})} \sum_{0=s_0,\ldots,s_{|\tilde{\mathbf{e}}|+1}=S+1} t_{0,s_1}\prod_{i=1}^{|\tilde{\mathbf{e}}|} t_{s_i,s_{i+1}} P(\tilde{e}_i,\tilde{f}_i|s_i)$$

The pair $(\tilde{\mathbf{e}},\tilde{\mathbf{f}})$ is considered as an alignment between the training pair $(\mathbf{e},\mathbf{f})$.

3 Transduction of the Transliterated Output

Given an alignment $(\tilde{\mathbf{e}},\tilde{\mathbf{f}})$, we consider all possible *sub-alignments* $(\tilde{\mathbf{e}}_i^j,\tilde{\mathbf{f}}_i^j)$ as pairs of substrings obtained from $(\tilde{\mathbf{e}},\tilde{\mathbf{f}})$ such that $\tilde{e}_i \neq \epsilon$, $\tilde{f}_i \neq \epsilon$, $\tilde{e}_{j+1} \neq \epsilon$ and $\tilde{f}_{j+1} \neq \epsilon$. We extract all possible sub-alignments of all the alignments from the training data. Let $\mathcal{A}$ be the bag of all sub-alignments obtained from the training data. We build a weighted finite state transducer that transduces any string in $\mathcal{E}^+$ to $\mathcal{F}^+$ using these sub-alignments.

Let $(\mathbf{u},\mathbf{v})$ be an element of $\mathcal{A}$. From the training data $\mathcal{D}$, observe that $\mathcal{A}$ can have multiple realizations of $(\mathbf{u},\mathbf{v})$. Let $N(\mathbf{u},\mathbf{v})$ be the number of times $(\mathbf{u},\mathbf{v})$ is observed in $\mathcal{A}$. The empirical probability of transducing string $\mathbf{u}$ to $\mathbf{v}$ is simply

$$P(\mathbf{v}|\mathbf{u}) = \frac{N(\mathbf{u},\mathbf{v})}{\sum_{\mathbf{v}:(\mathbf{u},\mathbf{v}')\in\mathcal{A}} N(\mathbf{u},\mathbf{v}')}$$

For every pair $(\mathbf{u},\mathbf{v}) \in \mathcal{A}$, we also compute the probability of transliteration from the HMM $\mathcal{H}$ as $Q(\mathbf{u},\mathbf{v})$ from Equation 1.

We construct a finite state transducer $\mathbf{F}_{\mathbf{u},\mathbf{v}}$ that accepts *only* $\mathbf{u}$ and emits $\mathbf{v}$ with a weight $w_{\mathbf{u},\mathbf{v}}$ defined as

$$w_{\mathbf{u},\mathbf{v}} = -\log(P(\mathbf{v}|\mathbf{u})) - \lambda\log(Q(\mathbf{u},\mathbf{v})) + \delta \quad (4)$$

Finally we construct a global weighted finite state transducer $\mathbf{F}$ by taking the union of all the $\mathbf{F}_{\mathbf{u},\mathbf{v}}$ and taking its closure.

$$\mathbf{F} = \left[\bigcup_{(\mathbf{u},\mathbf{v})\in\mathcal{A}} \mathbf{F}_{u,v}\right]^+ \qquad (5)$$

The weight δ is typically sufficiently high so that a new english string is favored to be broken into fewest possible sub-strings whose transliterations are available in the training data.

We tune the weights λ and δ by evaluating the accuracy on the held-out data. The n-best paths in the weighted finite state transducer $\mathbf{F}$ represent our n-best transliterations.

4 Results

We evaluated our system on the standard track data provided by the NEWS 2009 shared task organizers on five different languages – Tamil, Hindi, Russian, and Kannada was derived from (Kumaran and Kellner, 2007) and Chinese from (Li et al., 2004). The results of this evaluation on the test data is shown in Table 1. For a detailed description of the evaluation measures used we refer the readers to NEWS 2009 shared task whitepaper (Li et al., 2009).

Language	**Top-1 Accuracy**	**mean F_1 score**	**MRR**
Tamil	0.327	0.870	0.458
Hindi	0.398	0.855	0.515
Russian	0.506	0.901	0.609
Chinese	0.450	0.755	0.514
Kannada	0.235	0.817	0.353

Table 1: Results on NEWS 2009 test data.

5 Conclusion

We described a system for automatic transliteration of pairs of strings from one language to another using ϵ-extension hidden markov models and weighted finite state transducers. We evaluated our system on all the languages for the NEWS 2009 standard track. The system presented is language agnostic and can be trained for any language pair within a few minutes on a single core desktop computer.

References

Nasreen Abdul Jaleel and Leah Larkey. 2003. Statistical transliteration for english-arabic cross language information retrieval. In *Proceedings of the twelfth international conference on Information and knowledge management*, pages 139–146.

Kevin Knight and Jonathan Graehl. 1997. Machine transliteration. In *Computational Linguistics*, pages 128–135.

A. Kumaran and Tobias Kellner. 2007. A generic framework for machine transliteration. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 721–722, New York, NY, USA. ACM.

Haizhou Li, Min Zhang, and Jian Su. 2004. A joint source-channel model for machine transliteration. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 159, Morristown, NJ, USA. Association for Computational Linguistics.

Haizhou Li, A Kumaran, Min Zhang, and Vladimir Pervouchine. 2009. Whitepaper of news 2009 machine transliteration shared task. In *Proceedings of ACL-IJCNLP 2009 Named Entities Workshop (NEWS 2009)*.

Lawrence Rabiner. 1989. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286.

Yilu Zhou, Feng Huang, and Hsinchun Chen. 2008. Combining probability models and web mining models: a framework for jproper name transliteration. *Information Technology and Management*, 9(2):91–103.