Mining Transliterations from Wikipedia using Dynamic Bayesian Networks

Peter Nabende
Alfa-Informatica, University of Groningen
p.nabende@rug.nl

Abstract

Transliteration mining is aimed at building high quality multi-lingual named entity (NE) lexicons for improving performance in various Natural Language Processing (NLP) tasks including Machine Translation (MT) and Cross Language Information Retrieval (CLIR). In this paper, we apply two Dynamic Bayesian network (DBN)-based edit distance (ED) approaches in mining transliteration pairs from Wikipedia. Transliteration identification results on standard corpora for seven language pairs suggest that the DBN-based edit distance approaches are suitable for modeling transliteration similarity. An evaluation on mining transliteration pairs from English-Hindi and English-Tamil Wikipedia topic pairs shows that they improve transliteration mining quality over state-of-the-art approaches.

1 Introduction

Transliteration mining is aimed at addressing the problem of *unknown* words in NLP applications of which named entities (NEs) constitute the highest percentage. Currently, there is growing interest in using automated methods to harness large amounts of correct multi-lingual named entities (NEs) from various ever accumulating Web-based data resources such as newspaper websites and online encyclopedia (most notably Wikipedia) with the aim of improving word coverage and the effectiveness of NLP systems such as MT and CLIR.

In this paper, we present the application of two DBN-based edit distance approaches in mining transliterations from Wikipedia. Our motivation to apply the DBN-based approaches for modeling transliteration is founded on our observation of their successful application in NLP tasks (such as cognate identification (2005) and pronunciation classification (2005)) which have requirements similar to transliteration mining. While transliteration mining currently demands new approaches to complement or improve performance over existing methods, there was not yet any investigation about the use of the DBN-based edit distance approaches for mining transliteration pairs from 'noisy' data. The first approach is based on the classic HMM framework but models two observation sequences (hence the name Pair HMM) instead of one observation sequence. The second approach is based on the representation and implementation of a *memoryless stochastic transducer* (initially proposed by Ristad and Yianilos (1998) as a DBN model for learning string edit distance. We propose to evaluate the use of the two approaches in mining transliterations with respect to two subtasks. In the first subtask, we follow the same evaluation setup as that for a recent shared task on transliteration mining (Kumaran et al., 2010b) while using the same standard corpora. This first subtask ensures a comparison of the DBN model results against those for state-of-the-art systems that participated in the shared task since the DBN models are applied to the same standard transliteration corpora. In the second subtask, we investigate the possibility of applying proposed DBN models in mining transliterations from Wikipedia's article content in addition to the Wikipedia paired topics.

2 Wikipedia

Wikipedia is a free Web-based multi-lingual encyclopedia with an ever increasing number of articles in over 270 languages. In some articles for each language Wikipedia, access is provided to pages about the same topic in other language Wikipedias. Figure 1 shows two Wikipedia articles about the same topic, one in English titled as "Arab spring" while the other is in Arabic and is accessed using the Arabic Wikipedia interlanguage link (WIL) which exists on the English page as shown.

Proceedings of Recent Advances in Natural Language Processing, pages 385–391,
Hissar, Bulgaria, 12-14 September 2011.

Figure 1: Two Wikipedia articles about the same topic but written using different writing systems.

Many studies have recently found it inexpensive to automatically construct multi-lingual lexicons by using only Wikipedia inter-language links. In the first subtask, we use Wikipedia topic pairs that have been identified from inter-language links to constitute the collection of raw data to which the DBN models are applied and evaluated for mining transliteration pairs. In the second subtask, we propose to extend the application of the DBN models beyond mining from only linked text to also mining from the unlinked text in comparable articles. We postulate that the possibility to apply DBN models in mining from noisy unlinked comparable Wikipedia text would imply an extended use in mining transliterations from a variety of other similar sources where we expect to get many named entities such as from many emerging bilingual newspaper websites. In the following section, we introduce the concepts underlying the framework of DBNs and the models we have proposed to evaluate in mining transliteration pairs.

3 Dynamic Bayesian networks

The possibility to have random variables relate to time in a Bayesian network enables DBNs to represent probability distributions over a sequence of random variables comprising of observations that are related to an underlying sequence of hidden states. A DBN model is formally defined as a pair $\langle \mathcal{B}_0, \mathcal{B}_{\rightarrow} \rangle$ where $\mathcal{B}_0$ is a Bayesian network over an initial distribution over states, and $\mathcal{B}_{\rightarrow}$ is a two slice Temporal Bayes Net (2-TBN) (Murphy, 2002). It is the 2-TBN that is *unrolled* a number of times to fit a given observation sequence while providing the semantic definitions of the DBN over the whole observation sequence. The structure of a DBN is a directed acyclic graph (DAG) where each node represents a domain variable of interest, and each directed arc represents the dependency between the two nodes it connects. The DBN framework already generalizes various methods including some of the common and successful methods in NLP such as HMMs. HMMs, being the simplest DBNs, provide a natural starting point for our investigation of the use of DBNs in mining transliteration pairs. The classic HMMs in particular have already been applied in detecting transliteration pairs from bilingual text. However, because of space restrictions, we defer the introduction of HMMs in general and proceed to introduce the edit distance-based Pair HMMs for modeling transliteration similarity.

3.1 Pair HMMs

The Pair HMM approach is based on modifications to a pairwise alignment finite state automaton (Durbin et al., 1998). In this paper, we build upon the Pair HMM structure proposed by Mackay and Kondrak (2005) to compute word similarity. Figure 2 is a finite state representation of a Pair HMM similar to the one proposed by Mackay and Kondrak (2005). The Pair HMM in Figure 2 models word similarity as an edit operation cost for measuring the similarity between two words. The edit operations are encoded in the edit operation states

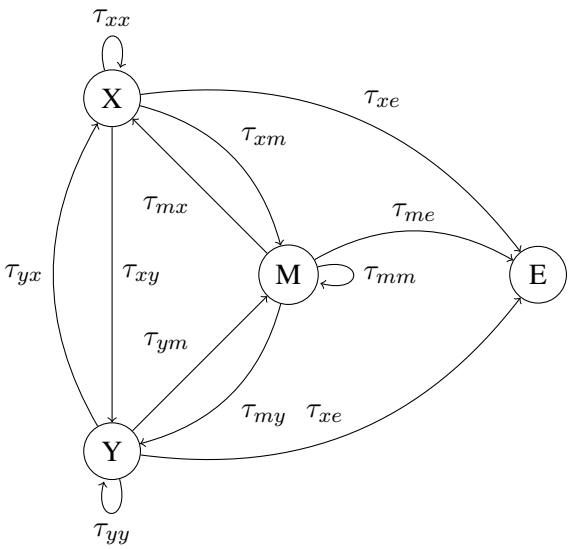

Figure 2: A finite state representation of a Pair HMM for modeling edit operations using three emission states: M (match), X (delete), and Y (insert). The τ's illustrate transition parameters.

which are denoted in Figure 2 by three nodes as follows: M (for emitting an aligned pair of symbols), X (for deleting a symbol from one word), and Y (for inserting a symbol) in the other word. The E node denotes the non-emitting end state. With respect to related work, we do not include a start state and instead assume that the edit operation process starts in any of the edit states where the three starting parameters are defined to be the same as the transition parameters from the M state to one of the edit states including M. We base our application of the Pair HMMs on a number of requirements that were proposed by Nabende et al. (2010) in adapting Mackay and Kondrak's word similarity Pair HMM for computing transliteration similarity. Specifically, we ensure the use of distinct emission parameters in the deletion (X) and insertion (Y) states because of different writing systems. Nabende (2010c) also investigated the effect of transition parameter changes on identifying English-Russian transliteration pairs and the result was that transition parameters are important for computing transliteration similarity. With respect to this paper, we also conducted an investigation on changes in transition parameters for seven language pairs and the conclusion was the same, that transition paramters are important for computing transliteration similarity. Therefore, the Pair HMMs we apply in transliteration mining are based on the finite state representation in Figure 2 but with different settings for transition parameters. In one setting we used only three transition parameters where the transition parameter takes the same value for outward transitions from a given edit state. In the second setting, we used five transition parameters where we assume some symmetries in the source and target language. We used similar parameters for transitions to and from the deletion and insertion states as in Mackay and Kondrak (2005). That is, we used the same parameter for staying in the X or Y state, another same parameter for moving from either X or Y to the end state, and the same parameter from the substitution state to the X or Y state. In the third setting, we used nine distinct parameters for transitions between the Pair HMM states. Apart from evaluating the effect of transition parameters, we also evaluated the use of different Pair HMM scoring algorithms including the following: the forward and Viterbi algorithms, and their combination with a random Pair HMM to compute log-odds ratios which we use as transliteration similarity estimates. The results for our preliminary investigation showed a stable and better transliteration identification performance for the three different Pair HMMs when we compute the transliteration similarity estimates as log-odds ratios of the Pair HMM forward score and the random model score compared to the other scoring algorithms.

3.2 Transduction-based DBN models

The second edit distance-based DBN approach that we propose for mining transliterations from Wikipedia finds its origins as an alternative DBN representation of a *memoryless stochastic transducer* using the general probabilistic graphical modeling (PGM) framework. The DBN template-based representation simplifies the investigation of a variety of edit operation-specific dependencies for computing word similarity and has led to successful applications in pronunciation classification (Filali and Bilmes, 2005) and cognate identification (Kondrak and Sherif, 2006). In this approach, random variables are defined to correspond to the objects that contribute to the computation of edit distance for a pair of words. The main objects of interest include: an edit operation variable (denoted by Z_i); source and target character variables; variables that capture the position of the characters in the source and target words; and consistency variables that check the yield of the edit operation variable against the actual pair of characters at a given position. The dependencies between the random variables follow naturally. For example, the following include some of the dependencies that were defined by Filali and Bilmes (2005) to model the memoryless stochastic transducer. Dependencies between string position variables (sp_i and tp_i) and character variables (s_i and respectively t_i) (where the idea is that knowledge about a position in a string leads to knowledge about the character at that position. For consistency checking, consistency variables are defined to depend on the character variables and the edit operation variable. Filali and Bilmes (2005) use an ASR-based graphical modeling approach where a *frame* is used to represent a set of random variables and their attributes at a given time. In the ASR-based graphical modeling approach, the term *prologue frame(s)* is used to refer to the initial Bayesian network $\mathcal{B}_0$ and the term *chunk frame* is used to refer to the Bayesian network that is *un-*

rolled a number of times to fit a given observation sequence as defined for a 2-TBN. Using the ASR-based notation, the initial, chunk and epilogue networks that model the memoryless stochastic transducer are as shown in Figure 3. The DBN template defined by the three networks in Figure 3 is also referred to as *memoryless and context-independent* (MCI) since it models the memoryless stochastic transducer which is also context-independent in the sence that the edit operation variable has no local dependencies on the source and target characters, but has a global context dependency that allows it to generate the source and target strings.

In Figure 3, the sp_i and tp_i nodes refer to variables used to track the current position in the source and target strings respectively. The sp_i and tp_i combined with z_{i-1} capture the transition semantics in the network. The s_i and t_i variables represent the current character in the source and target strings respectively. The sc_i and tc_i nodes enforce consistency constraints by having a fixed observed value 1 where the only configuration of

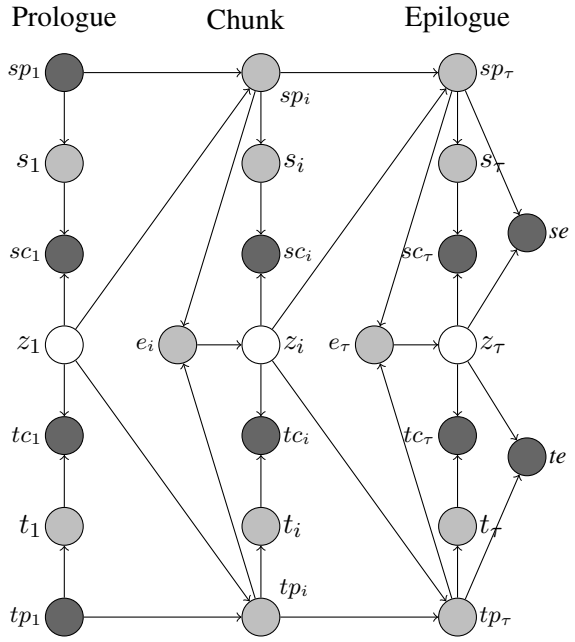

Figure 3: Graphical representation for the MCI DBN template. Following the common convention for representing graphical models, dark nodes represent observed variables which can be either deterministic or stochastic, gray nodes represent deterministic hidden variables, and unshaded nodes represent hidden variables. Adapted from Filali and Bilmes (2005).

their parents is such that the source component of the edit operation variable z_i is s_i or an empty symbol and the target component of z_i is t_i or an empty symbol ϵ, and that z_i does not generate (ϵ, ϵ). e_i denotes the 'end' variable which is a 'switching' parent of z_i and it is used to indicate when we are past the end of both the source and target strings; that is, when $sp_i > m$ and $tp_i > n$ where *m* and *n* are the lengths of the source and target strings respectively. The *se* and *te* nodes represent variables that ensure that we are past the end of the source and target strings respectively. Most of the edges in Figure 3 represent deterministic relationships between variables, more specifically, edges that are associated with position variables, consistency variables, character variables and end variables. For these, we use deterministic conditional probability tables. The emission probabilities that are used to generate the source and target strings are encoded in the edit operation variable z_i by way of dense conditional probability tables.

In Nabende (2010a), three DBN model generalizations based on the MCI DBN model were adapted to compute transliteration similarity and identify transliterations between English and Russian NEs. In the preliminary experiments in this paper, we evaluate the three DBN model generalizations that were adapted in Nabende (2010a) on the same seven language pairs that we used to evaluate several Pair HMMs as described in section 3.1 above. The three DBN model generalizations represent different dependencies on the edit operation random variables including: edit operation *memory* dependencies that capture memory from previous edit states of a DBN model; *contextual* dependencies of the edit operation variable on either source and / or target string elements; and dependencies that account for the *length* of the edit steps needed to represent an observation sequence.

3.3 Preliminary transliteration identification experiments

We conducted a preliminary transliteration identification (TI) experiment to help choose DBN models for use in mining transliteration pairs from Wikipedia. Several Pair HMMs and transduction-based DBN models introduced in the previous section were evaluated on standard transliteration corpora (Li et al., 2009; Li et al., 2010) for seven language pairs including: English-Bangla (e-b), English-Chinese (e-c), English-Hindi (e-h),

Models	e-b	e-c	e-h	e-k	e-r	e-t	eth
	Top-1 accuracy						
Phm	93	68	**89**	**86**	89	83	62
Mci	87	30	75	72	98	65	35
Mem	89	49	72	57	89	72	49
Cs1	96	70	86	84	98	83	74
Cs2	96	80	86	85	97	85	79
Ct1	95	68	84	84	98	84	76
Ct2	96	**82**	86	86	98	**86**	**85**
Ls1	96	71	83	77	98	84	73
Ls2	95	70	85	81	98	80	70

Table 1: DBN model transliteration identification results involving seven language pairs. The row with Phm represents the best Pair HMM result per language pair. The remaining results are for the transduction-based models with Mci referring to the MCI DBN template in Figure 3. cs1 and ct1 refer to context-dependent DBN models where Z_i depends on the current source (cs1) or target (ct1) character. In Cs2 and ct2, Z_i depends on the current and previous characters.

English-Kannadda (e-k), English-Russian (e-r), English-Tamil (e-t), and English-Thai (eth). Table 1 shows the TI Top-1 accuracy results (out of 100%) for the transduction-based DBN models against the best Pair HMMs (represented by Phm) involving the seven language pairs. As Table 1 shows, the context-dependent DBN models generally achieve a better performance compared to other DBN models. Table 1 also shows that Pair HMMs outperform the transduction-based DBN models in identifying transliterations between English and Hindi, and between English and Kannada. Based on the TI Top-1 accuracy results in Table 1, we chose to evaluate the Pair HMMs and context-dependent DBN models in mining transliterations from Wikipedia.

4 Transliteration mining experiments

4.1 Experiments using NEWS 2010 shared task Wikipedia data

We applied the best Pair HMMs and context-dependent DBN models from the preliminary TI experiments above to transliteration data provided for the NEWS 2010 shared task (Kumaran et al., 2010a) on mining single word transliteration pairs for three language pairs: English-Hindi, English-Russian, and English-Tamil. Two sets of data were provided per language pair including: 1000 hand picked pairs of single NEs as seed data for training transliteration mining systems; and many noisy Wikipedia topic pairs obtained using Wikipedia inter-language links (WILs) as raw data. Our data pre-processing on the noisy Wikipedia topic pairs involved using simple regular expressions to filter out most of the irrelevant entities including: characters from other writing systems; temporal and numeric expressions; and punctuation symbols. To reduce on data sparseness, we converted all characters in the English and Russian datasets to lowercase. After pre-processsing, the remaining number of Wikipedia topic pairs per language pair were as follows: 14620 (86.2%) for English-Hindi, 296053 (85.6%) for English-Russian, and 13249 (95.4%) for English-Tamil.

Evaluation setup and results

We used the seed datasets to train each Pair HMM and context-dependent DBN models. We trained each Pair HMM using the Baum-Welch expectation maximization algorithm and each context-dependent DBN model using a generalized expectation maximization algorithm. We then applied the trained models to compute transliteration similarity between candidate NEs from each Wikipedia topic pair. After the application of each model, we checked the similarity estimates assigned by the model to candidate transliteration pairs which enabled us to specify different threshold scores ranging from a low threshold for which the system suggests many candidate pairs as transliteration pairs to a high threshold where the system suggests very few candidate pairs as transliteration pairs. The transliteration mining results from each model are evaluated using three related metrics: Precision (P), Recall (R), and F-score.

$$\mathrm{P} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \quad \mathrm{R} = \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$$

$$\mathrm{F} = \frac{2 \times \mathrm{P} \times \mathrm{R}}{\mathrm{P}+\mathrm{R}}$$

where TP, FP, and FN refer to true positives, false positives, and false negatives respectively.

In order to compare the DBN model results against those reported for the NEWS 2010 shared task on transliteration mining, we checked a subset of the transliteration mining result per language pair to set a subjective threshold score that we thought would result in an optimal dis-

Model	P	R	F
PHMM3_FLO	0.930	0.976	0.952
PHMM3_IterT_FLO	0.959	0.949	**0.954**
PHMM9_FL0	0.934	0.975	0.954
PHMM9_IterT_FLO	0.936	0.976	**0.955**
CONs2	0.911	0.891	0.901
NEWS 2010 best result	0.954	0.895	0.924

Table 2: DBN model results against NEWS 2010 shared task results for English-Hindi. P refers to Precision, R to recall, and F to F-score.

crimination between true transliteration and non-transliteration pairs. Table 2 shows the results for the DBN models against the best shared task result on the English-Hindi dataset. In Table 2, the Pair HMMs have in an F-score value that is better than that for the best shared task result while the CONs2 model also posts a promising result.

Table 3 shows the results for Pair HMMs and a context-dependent DBN model against the best shared task result on the English-Tamil dataset. The Pair HMMs again achieve a better F-score value compared to the best shared task result. However, the CONs2 model has a relatively poor performance than it did for English-Hindi above.

For the English-Russian dataset, we applied a context-dependent DBN model (Cont1) that models the dependency of the edit operation variable Z_i on the current target character. Table 4 shows the results of Cont1 against the best shared task result and against those of a Pair HMM with nine distinct transition parameters that was also evaluated during the shared task (Nabende, 2010b). For the English-Russian dataset, none of the DBN models achieved an F-score better than that of the shared task result. Table 4 shows that the context-dependent DBN models results in a better F-score over the Pair HMM using only the forward algorithm to compute transliteration similarity.

Model	P	R	F
PHMM3_FLO	0.913	0.966	0.936
PHMM5_FLO	0.923	0.955	0.939
CONs2	0.790	0.852	0.820
NEWS2010 best result	0.923	0.906	0.914

Table 3: DBN model results against NEWS 2010 shared task results for English-Tamil.

Model	P	R	F
Cont1	0.835	0.815	0.825
PHMM9_F_NEWS2010	0.780	0.834	0.806
NEWS2010 best result	0.880	0.869	0.875

Table 4: DBN model results against NEWS 2010 shared task results for English-Tamil.

4.2 Experiments using Wikipedia's article content

For this set of experiments, we used Wikipedia inter-language links to automatically acquire seed data by restricting our search to only person names following the structured nature of information in Wikipedia infoboxes. For test data, we identified some ten of the most visited pages during the month of August 2009 to serve as our source for mining transliterations. The data pre-processing steps here are similar to those described in the previous section. Our evaluation set comprised of 4811 English NEs and 9334 Russian NEs after pre-processing. From the candidate NEs, we hand-picked 264 transliteration pairs to form the gold set.

We applied three Pair HMMs (PHMM3, PHMM5, and PHMM9) and a context-dependent DBN model to mine transliterations from English-Russian Wikipedia article text in a manner similar to how we applied them in mining transliterations from Wikipedia topic pairs. We trained the DBN models on the automatically acquired seed data and then applied the trained models to compute transliteration similarity between candidate NEs. All the Pair HMMs use the log-odds ratio involving the forward algorithm to compute transliteration similarity. We evaluate the models at different cut-offs of the number of the top-ranked suggestions of transliteration pairs for each model. Table 5 shows theresults for the top ranked 200 suggestions per model. Table 5 shows that the

Model	P	R	F
PHMM3_FLO	0.530	0.402	0.457
PHMM5_FLO	0.755	0.572	0.651
PHMM9_FLO	0.630	0.477	0.543
CONs1	0.760	0. 576	0.655

Table 5: Transliteration mining results for a cut-off of 200 top-ranked suggestions of English-Russian candidate pairs as true transliteration pairs.

context-dependent DBN model achieves a slightly better F-score than the Pair HMMs for the first 200 suggestions of transliteration pairs using the models. However, we found out that an increase in recall resulted in a faster drop of precision for the context-dependent DBN model compared to the drop for the Pair HMMs.

5 Conclusions and future work

In this paper, we evaluated several DBN models in mining transliterations from Wikipedia. Pair HMMs achieved fair improvements in transliteration mining quality over state-of-the-art methods for mining transliterations from English-Hindi and English-Tamil Wikipedia topic pairs. The results also showed the possibility of applying the DBN approaches in mining transliteration pairs from comparable Wikipedia article content with context-dependent models performing better than the Pair HMMs on an English-Russian dataset.

As future work, we would like to evaluate more transduction-based DBN models in mining transliteration pairs from comparable Wikipedia content for other language pairs and on larger datasets than the ones used in this paper.

Acknowledgments

Research in this paper was funded through a second NPT Uganda project from 2007–2011.

References

Richard Durbin, Sean R. Eddy, Anders Krogh, and Graeme Mitchison. 1998. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, UK.

Karim Filali and Jeff Bilmes. 2005. A Dynamic Bayesian Framework to model context and memory in edit distance learning: An application to pronunciation classification. *Proceedings of the Association for Computational Linguistics (ACL)*, Ann-Arbor, Michigan.

Grzegorz Kondrak and Tarek Sherif. 2006. Evaluation of several phonetic similarity algorithms on the task of cognate identification. *Proceedings of the COLING-ACL Workshop on Linguistic distances*, pp. 43–50, Sydney, Australia.

A Kumaran, Mitesh M. Khapra, and Haizhou Li. 2010. Whitepaper of NEWS 2010 Shared Task on Transliteration Mining, *Proceedings of the 2010 Named Entities Workshop*, pp. 29–38, Uppsala, Sweden.

A Kumaran, Mitesh M. Khapra, and Haizhou Li. 2010. Report of NEWS 2010 Transliteration Mining shared task. *Proceedings of the 2010 Named Entities Workshop*, pp. 21–28, Uppsala, Sweden.

Haizhou Li, A Kumaran, Vladmir Pervouchine, and Min Zhang. 2009. Report of NEWS 2009 Machine transliteration shared task. *ACL/IJCNLP 2009 Named Entities Workshop (NEWS 2009)*, Suntec, Singapore.

Haizhou Li, A Kumaran, Min Zhang, and Vladmir Pervouchine. 2010. Whitepaper of NEWS 2010 transliteration generation shared task. *Proceedings of the 2010 Named Entities Workshop*, pp. 12–20, Uppsala, Sweden.

Wesley Mackay and Grzegorz Kondrak. 2010. Computing Word and identifying cognates with Pair Hidden Markov models. *Proceedings of the ninth Conference on Computational Natural Language Learning (CONLL 2005)*, pp. 40–47 Ann Arbor, Michigan.

Kevin P. Marphy. 2002. *Dynamic Bayesian Networks: Representation, Inference and Learning.* PhD Thesis, UC Berkeley, Computer Science Division.

Peter Nabende, Jörg Tiedeman, and John Nerbonne. 2010. Pair Hidden Markov model for named entity matching. *Innovations and advances in computer sciences and engineering*, pp. 497–502, Springer Heidelberg.

Peter Nabende. 2010. Applying a Dynamic Bayesian network framework to transliteration identification. Proceedings of the seventh International language resources and evaluation conference (LREC 2010), pp.244–251, Valletta, Malta.

Peter Nabende. 2010. Mining transliterations from Wikipedia using Pair HMMs. *Proceedings of the 2010 Named Entities Workshop, ACL 2010*, pp. 76–80, Uppsala, Sweden.

Peter Nabende. 2010. Comparison of applying Pair HMMs and DBN models in transliteration identification. *Proceedings of the 20th Computational Linguistics in Netherlands meeting*, pp. 107–122, Amsterdam, The Netherlands.

Eric S. Ristad and Peter N. Yianilos. 1998. Learning string edit distance. *Trans. on Pattern Recognition and Machine Intelligence*, 20(5):522–532.

Martijn Wieling, Therese Leinonen, and John Nerbonne. 2007. Inducing sound segment differences using Pair Hidden Markov Models. In J. Nerbonne, M. Ellison, and G. Kondrak (eds.), *Computing and Historical Phonology: 9th Meeting of ACL Special Interest Group for Computational Morphology and Phonology Workshop*, pp. 48–56, Prague, Czech Republic.