

# Machine Transliteration of Names in Arabic Text

Yaser Al-Onaizan and Kevin Knight  
Information Sciences Institute  
University of Southern California  
4676 Admiralty Way, Suite 1001  
Marina del Rey, CA 90292  
{yaser,knight}@isi.edu

**Paper ID:**

**Keywords:** Machine Transliteration

**Contact Author:** Yaser Al-Onaizan

**Under consideration for other conferences (specify)?** None

## Abstract

We present a transliteration algorithm based on sound and spelling mappings using finite state machines. The transliteration models can be trained on relatively small lists of names. We introduce a new spelling-based model that is much more accurate than state-of-the-art phonetic-based models and can be trained on easier-to-obtain training data. We apply our transliteration algorithm to the transliteration of names from Arabic into English. We report on the accuracy of our algorithm based on exact-matching criterion and based on human-subjective evaluation. We also compare the accuracy of our system to the accuracy of human translators.

# Machine Transliteration of Names in Arabic Text

Paper ID:

## Abstract

We present a transliteration algorithm based on sound and spelling mappings using finite state machines. The transliteration models can be trained on relatively small lists of names. We introduce a new spelling-based model that much more accurate than state-of-the-art phonetic-based models and can be trained on easier-to-obtain training data. We apply our transliteration algorithm to the transliteration of names from Arabic into English. We report on the accuracy of our algorithm based on exact-matching criterion and based on human-subjective evaluation. We also compare the accuracy of our system to the accuracy of human translators.

## 1 Introduction

Human translators and machine translation systems are often faced with the task of transliterating phrases like person names and locations. Transliteration is the process of replacing words in the source language with their approximate phonetic or spelling equivalents in the target language. Transliterating names between languages that use similar alphabets and sound systems is often very simple, since the phrase mostly remains the same. However, the transliteration becomes far more difficult when transliterating between languages with very different sound and writing systems.

When transliterating a name from Arabic into English, there are two types of transliterations:

- **Forward Transliteration:** This refers to the transliteration of an Arabic name into English. Typically, many variations of the transliterated name are acceptable. This is especially true when transliterating between two languages with many phonetic incompatibilities, such as English and Arabic. For example, the Arabic name “ياسر *yāsir*” can reasonably be transliterated in any of the following ways: *Yasir*, *Yassir*, *Yaser*, *Yasser*, etc.
- **Backward Transliteration:** This refers to the reverse transliteration process in order to obtain the original of an English name that has

already been transliterated into Arabic. In this case, typically, only one transliteration is acceptable. For example, when encountering the name “روبرت *rubrt*” or “روبيرت *rubyrft*” while translating Arabic text into English, the name should only be transliterated as *Robert*.

Transliterating names from Arabic into English in either direction is a difficult task, mainly due to the differences in their sound and writing systems. For instance, vowels in Arabic come in two varieties, long and short. Short vowels are rarely written in Arabic in newspaper text, which makes pronunciation highly ambiguous. Also, because of the differences in their sound inventory, there is no one-to-one correspondence between Arabic sounds and English sounds. For example, English **P** and **B** are both mapped into Arabic “ب *b*”; Arabic “ح *h*” and “هـ *h*” into English **H**; and so on.

In this paper, we describe Arabic-to-English name transliteration system using probabilistic finite state machines.

## 2 Related Work

Kawtrakul et al. (1998) present a back transliteration system from Thai into English in the context of document retrieval. In their approach, loan words are first segmented into syllables using a combination of rules and statistical techniques. Then, syllables are mapped to phonemes based on some transcription rules. The phoneme sequence of the loan word is compared to the phonetic sequence of a set of English words found in a phonetic dictionary and the word with the most similar phonetic sequence is selected as the transliteration.

The approach described by Kawtrakul et al. (1998) requires a phonetic dictionary of English in order to match phonetic sequences. Only those words with known phonetic sequences in the dictionary can be mapped by the transliteration system. Also, applying such technique to Arabic will most likely fail because without short vowels, the pronunciation is highly ambiguous, and so is its corresponding phonetic sequence.

Arbabi et al. (1994) describes an algorithm for the forward-transliteration of Arabic names into a number of Romance and Germanic languages including English, French, and Spanish. The translit-

eration process starts by vowelizing the given Arabic name by inserting the appropriate short vowels which originally are not written but necessary for the correct pronunciation of the names. Then, the vowelized Arabic name is converted into its phonetic Roman representation using a parser and table lookup. The phonetic representation is then used in a table lookup to produce the spelling in the desired language.

The vowelization rules described by Arbabi et al. (1994) apply only to Arabic names that conform to strict Arabic morphological rules. Any name that does not conform to the morphological rules is ignored and hence no transliteration will be attempted. This restriction limits the applicability of this approach since many person and organization names do not conform to morphological rules, especially loan words and foreign names.

Stalls and Knight (1998) present an Arabic-to-English back-transliteration system based on the source-channel framework. The transliteration process is based on a generative model of how an English name is transliterated into Arabic. It consists of several steps, each defined as a probabilistic model represented as a finite state machine. First, an English word  $w$  is generated according to its unigram probabilities  $P(w)$ . Then, the English word  $w$  is pronounced (i.e., converted to sound sequence  $e$ ) with probability  $P(e|w)$ , which is collected directly from an English pronunciation dictionary. Finally, the English phoneme sequence is converted into Arabic writing with probability  $P(a|e)$ , which we discuss in details in Section 4.

The pronunciation model  $P(e|w)$  converts English letter sequences into English sound sequences. The model proposed by Stalls and Knight (1998) uses a pronunciation dictionary to do this conversion. Therefore, only words with known pronunciations in the dictionary can be transliterated.

One way to overcome this limitation is to train a model that can map any given English letter sequence into its corresponding English sound sequence. This mapping is a complex task because of the mismatch between English spelling and English pronunciation. This difficulty, coupled with the difficulty of mapping Arabic letter sequences to English sound sequences, renders this choice unattractive. Instead we propose a spelling-based model that maps directly into Arabic letter sequences, which can be trained on an English/Arabic name list as we describe in Section 5. But before we present any further details, we describe our evaluation data next.

### 3 Evaluation Data

Our evaluation corpora consists of two data sets, a development and a blind test set. The two sets consist of a list of person names extracted from Arabic newspaper articles. The development test set contains 854 names (377 unique names) and the blind test set contains 218 (85 unique names). The person names are then transliterated into English. The transliterations are then thoroughly reviewed and any obvious mistakes corrected.<sup>1</sup> The corrected transliterations form the gold-standard we will compare our results with.

We would like to investigate the suitability of the models proposed here for back- and forward-transliteration. Therefore, each name in the list is classified in one of three categories **ARABIC**, for names of Arabic origin; **ENGLISH**, for names of English origin; and **OTHER**, for names of other origins including Chinese, Russian, Indian, etc. The names were classified by a bilingual speaker (a native speaker of Arabic). The classification is not always clear cut. In some cases, the first name of a person might be of one category and the last name of another (e.g., “علي رودريغيز” *Ali Rodriguez*). In such cases, the category is chosen based on the identity of the person if it is known, otherwise the category of the last name is chosen. The distribution of person names in the two data sets into the three categories is shown in Table 1.

The evaluation criterion used for evaluating the accuracy of the different models presented throughout this paper is based mostly on exact matching (i.e., a transliteration is considered correct only if it exactly matches the one in the gold-standard). We also present a human-subjective evaluation for the final transliteration model in Section 8.

Our transliteration algorithm provides top  $n$  candidates for any given name. We will present the evaluation results of our transliteration models in terms of **Top-1** (the correct transliteration is the top candidate) and **Top-20** (the correct transliteration is among the top 20 candidates). The top-20 list is often useful in providing phrasal translation candidates to statistical machine translation systems, where contextual information from the document can be exploited to help rank the correct candidate highest. For example, phrase frequency counts can

<sup>1</sup>Possible transliterations for a name were researched using the context in which the person name was mentioned in the original Arabic document using Web resources to determine the correct transliteration. For example, if the person is a US Senator, the official US Senate Web pages are used to determine the correct spelling of the Senator’s name.

be used to rank *Bill Clinton* higher than *Bell Clinton*.

#### 4 Phonetic-Based Model

The transliteration model proposed by Stalls and Knight (1998) has three components  $P(w)$ ,  $P(e|w)$ , and  $P(a|e)$ .  $P(w)$  is a typical unigram model that generates English word sequences according to their unigram probability. A given English word sequence  $w$  is converted to its corresponding phoneme sequence  $e$  with probability  $P(e|w)$ . The probability model  $P(e|w)$  is directly estimated from The CMU Pronouncing Dictionary of English. For words with only one possible pronunciation in the dictionary, the  $P(e|w)$  probability is 1.

An English phoneme sequence  $e$  is converted to Arabic letter sequence  $a$  according to the  $P(a|e)$  model. Stalls and Knight (1998) trained this model on a small 150-word dictionary of names given their Arabic writing and English pronunciation. In the initial model used by Stalls and Knight (1998), each English phoneme can produce zero or more Arabic letters. They have reported several problems pertaining to mapping English vowels to their equivalent Arabic ones. To address these problems, each English vowel was replaced in the model (and subsequently in the training data) with one of three symbols, depending on its position in the word (e.g., **AH** in word-initial position was replaced with **AH-S**; word-final was replaced with **AH-F**; word-medial was **AH**).

A problem occurs when carrying out the actual transliterations. A given Arabic letter sequence might generate an ill-formed English phoneme sequence, since there is no restriction in the model to force, for instance, word-final phonemes to appear only at the end of the sequence. Such ill-formed sequences need to be filtered out, which can be done using an additional finite-state machine.

Instead of extending the phoneme set with position markers, we decided to move such a dependence into the model itself without changing the phoneme set. In our adaptation, we use three separate states for mapping English phonemes to Arabic letter sequences. The mapping is conditioned on the position of the phoneme (e.g., word-initial, word-medial, and word-final). As a result, no ill-formed phoneme sequences will be generated. The two different models are shown in Figure 1. The model we propose has more parameters to estimate and hence more training data is needed, but is otherwise as accurate as the original model when trained on the same training data. The advantage of this model is that com-

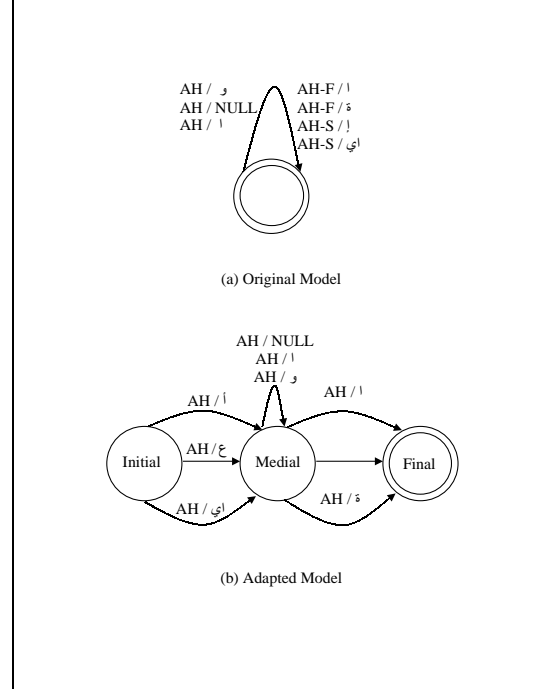


Figure 1: The shape of the original phonetic-based model proposed by Stalls and Knight (1998) and our adaptation.

posing other finite state machines with it is much faster. The model was trained on a list of 1426 Arabic/English name pairs.

Here are the most likely mappings of the English phoneme **AH** in the modified model after training<sup>2</sup>:

a	e	Position	$P(a e, \text{Position})$
ا	AH	Initial	0.43
ع	AH	Initial	0.24
اي	AH	Initial	0.05
NULL	AH	Medial	0.37
ا	AH	Medial	0.36
و	AH	Medial	0.27
ا	AH	Final	0.93
ت	AH	Final	0.06

According to this model, the probability of transliterating Arabic word  $a$  into English word  $w$  is given by the following equation:

$$P_p(w|a) \simeq \sum_{\forall e} P(w)P(e|w)P(a|e) \quad (1)$$

The actual transliteration process is a graph-search problem through millions of possible map-

<sup>2</sup>NULL is the empty string.

Data Set	Distribution (%)		
	ARABIC	ENGLISH	OTHER
Dev. Test Set	21.78	30.09	48.13
Blind Test Set	48.62	30.28	21.10

Table 1: The distribution in the two data sets of the names into the three categories **ARABIC**, **ENGLISH**, and **OTHER**. **ARABIC** refers to names of Arabic origin, **ENGLISH** is of English origin, and **OTHER** is for names of other origins including Russian, Chinese, and Indian.

pings to find the best path with English word sequence  $w$  that maximizes  $P_p(w|a)$  for a given Arabic word sequence  $a$ , as described by Knight and Graehl (1997).

## 5 Spelling-Based Model

One serious limitation of the phonetic-based model described above is that only English words with known pronunciations can be produced. For back-transliterating person names of English origin, this is not a big problem because many of such names are typically found in the dictionary. However, applying this technique to transliterate names of origins other than English is not going to work, because many such names are not likely to be in the dictionary despite the fact that the dictionary has more than 100,100 entries in it, as shown in Table 2. Moreover, if we want to apply this technique to transliterate a name into a language other than English, a large pronunciation dictionary is needed for that language, which is not easily obtainable.

Also, human translators often transliterate words based on how they are spelled in the source language. For example, *Graham* is typically transliterated by humans into Arabic as “غراهام” *grāhām* and not as “غرام” *grām*. Also, both “هوجز” *hwǧz* and “هيوز” *hywz* occur in our corpus as possible transliterations for *Hughes* (both occurred as a transliteration for *Karen Hughes*). To back-transliterate such instances, one would need to consider spelling-based mappings not just sound mappings.

To address these limitations, we propose a new spelling-based model that can be used alone or in conjunction with the phonetic-based model. The new model outperforms the phonetic-based model, even when evaluated on names found in the phonetic dictionary as we will discuss in more detail in Section 8.

The spelling-based model we propose directly maps English letter sequences into Arabic letter sequences with probability  $P(a|w)$ , which is trained on an English/Arabic name list without the need for English pronunciations. Since no pronunciations are needed, this list is easily obtainable for many lan-

guage pairs. We also extend the model  $P(w)$  to include a letter trigram model in addition to the word unigram model. This makes it possible to generate words that are not already defined in the word unigram model but obey English patterns. The word unigram model can be trained on any list of words. When trained on a list of person names, the transliterations will be most accurate for person names. For the experiments reported in this paper, the unigram model was trained on the list of names (without their pronunciations) from the CMU dictionary. The letter trigram is also trained on the same list.

The transliteration score according to this model is given by:

$$P_s(w|a) \simeq P(w)P(a|w) \quad (2)$$

For a given Arabic name  $a$ , the actual transliteration process is carried out by searching for the English word sequence that maximizes  $P_s(w|a)$ .

In our spelling-based model, a sequence of one or more English letters is mapped to a sequence of zero or more Arabic letters.<sup>3</sup> English letter sequences are typically longer than their Arabic equivalents for many reasons. First, because Arabic short vowels are not written and need to be “guessed” by the model. Second, English names often have silent letters that mostly are not reflected in the Arabic equivalent (e.g., **Knight** is transliterated as “نايت” *nāyṭ*). This phenomenon was also reflected in the learned model. Here is an example of some of the parameters learned during training:

<sup>3</sup>To reduce the parameters to be estimated and prevent data sparseness without loss of any practical modeling power, English letter sequences were restricted to a maximum of 3 letters, while Arabic ones were restricted to a maximum of 2 letters.

Data Set	Dictionary Coverage (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Dev. Test Set	29.03	89.10	59.12	61.59
Blind Test Set	55.66	100	47.82	67.43

Table 2: The percentages of names in development and blind test sets that are found in the CMU Pronouncing Dictionary presented by the category of each name. **OVERALL** is a weighted average of the three categories.

a	w	Position	$P(a w, \text{Position})$
إ i	I	Initial	0.87
إي iy	I	Initial	0.10
أي ay	I	Initial	0.03
ي y	I	Medial	0.56
أ ā	I	Medial	0.38
NULL	I	Medial	0.06
ي y	I	Final	0.99

Here are some examples of the letter sequence alignments for pairs of Arabic name/top transliteration as provided by our system.

**Example I:** Given the name “صدام *ṣdām*,” its top transliteration was **SADDAM**, and the letter sequence alignment was:

SA	ص s
DD	د d
A	أ ā
M	م m

**Example II:** Given the name “إيران *iyrān*,” its top transliteration was **IRAN**, and the letter sequence alignment was:

I	إي iy
R	ر r
A	أ ā
N	ن n

**Example III:** Given the name “أوبنهايمر *wbnhā-ymr*,” its top transliteration was **OPPENHEIMER**, and the letter sequence alignment was:

O	أو w
PP	ب b
E	
N	ن n
H	ه h-
E	
I	أي ay
M	م m
R	ر r

## 6 Combining The Phonetic-Based and Spelling-Based Models

The phonetic-based and spelling-based models can be linearly combined into a single transliteration model. The transliteration score for an English word  $w$  given an Arabic word  $a$  is a linear combination of the phonetic-based and the spelling-based transliteration scores as follows:<sup>4</sup>

$$P(w|a) = \lambda P_s(w|a) + (1 - \lambda) P_p(w|a) \quad (3)$$

## 7 Improving Transliterations

In this section we discuss two different techniques that were used to improve the transliteration accuracy. In the first technique, the given word to be transliterated is pre-processed to correct any typos and spelling errors. The spelling correction model described in Section 7.1 is also implemented using a finite state machine which can be easily added to the transliteration composition pipeline. In the second technique to improve transliterations, transliterations are post-processed to filter any unlikely transliterations as described in Section 7.2.

### 7.1 Spelling Corrections

Typos and misspellings are very common in Arabic newspapers, especially in on-line editions. Typical typos stem from replacing a letter with another that has a similar shape, especially when they are mapped to adjacent keys on the keyboard layout (e.g., “ع” and “غ”; “ف” and “ق”; and so on). These letters have very different sounds and without being corrected, names with those typos will most likely be transliterated incorrectly. For example, the name “عوزاليس *wnzālys*” is a misspelled version of the name “غوزاليس *gwnzālys*” (*Gonzalez*).

Spaces are reliably used in Arabic to separate words, with very few exceptions. Arabic employs a cursive writing system, so typically letters in the same word are connected to each other. Most letters can be connected from both sides, but some (such as “د”, “ذ”, “ر”, and “ز”) can be connected only from

<sup>4</sup>For the experiments reported in this paper, we used  $\lambda = 0.5$ .

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Without Spell Corr. (Top1)	15.59	59.53	23.44	32.52
With Spell Corr. (Top1)	15.59	59.53	28.95	35.20
Without Spell Corr. (Top20)	18.28	86.00	49.52	53.66
With Spell Corr. (Top20)	19.89	87.16	50.48	54.82

(a) Accuracy on the Development Test Set.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Without Spell Corr. (Top1)	40.57	48.48	6.52	35.78
With Spell Corr. (Top1)	43.40	48.49	6.52	37.16
Without Spell Corr. (Top20)	42.46	100	47.82	61.00
With Spell Corr. (Top20)	46.23	100	47.82	62.85

(b) Accuracy on the Blind Test Set.

Table 3: A comparison of the transliteration accuracy on the development and blind test sets by category, with and without spelling correction. The results shown here are for the phonetic-based model. The **Top1** results considers whether the correct answer is the top candidate or not, while the **Top20** results considers whether the correct answer is among the top-20 candidates.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Without Web Filtering (Top1)	18.28	64.98	40.91	43.21
With Web Filtering (Top1)	19.89	63.42	42.34	43.79
Without Web Filtering (Top20)	26.88	84.82	53.83	57.26
With Web Filtering (Top20)	31.72	85.21	56.69	59.82

(a) Accuracy on the Development Test Set.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Without Web Filtering (Top1)	45.28	59.10	54.35	50.92
With Web Filtering (Top1)	48.11	59.10	54.35	52.75
Without Web Filtering (Top20)	60.38	98.50	56.52	70.19
With Web Filtering (Top20)	70.75	98.50	56.52	76.15

(b) Accuracy on the Blind Test Set.

Table 4: A comparison of the transliteration accuracy on the development and blind test sets by category, with and without Web filtering. The results shown here are for the spelling-based model.

Data Set	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Dev. Test Set	55.91	77.04	42.58	55.85
Blind Test Set	60.38	92.42	50.00	67.89

Table 5: The transliteration accuracy for a human translator.

the right side. After any of these letters, a space might be incorrectly deleted (e.g., “بيترمندلسون” instead of “بيتر مندلسون” *bytr mndlsun*) or inserted (e.g., “عبدالله” *bd āllh* instead of “عبدالله” *bdāllh*).

Additionally, there are common misspellings that can be found even in the most respected Arabic newspapers, e.g., interchanging one form of an alif (“ا”, “آ”, “إ”, or “أ”) with another, especially at the beginning of a word; or interchanging “ذ” and “ظ” at the end of a word; etc.

These kinds of typos and misspellings are more common than we expected. For example, 5% of the names in our development test set were misspelled. Human translators seem to be able to recover from name misspellings when transliterating a name they are familiar with. Our human subject was able to transliterate the name “بوريس” *bwrys* (Boris) correctly, even though it was misspelled as “برويس” *br-wys*. Therefore, we believe that we need to model misspellings explicitly rather than hope that they will not cause wrong transliterations.

We model misspellings by using an additional finite-state machine at the end of the cascade of finite state machines. We would like to estimate the parameters in this model empirically. But since we do not have enough misspellings/correct spelling pairs to train this model, the weights were set manually.

The use of this spelling correction model slightly improves our transliterations, as shown in Table 3.

## 7.2 Web-Based Filtering

As we have described in Section 5, the  $P(w)$  model is a combination of a word unigram model and a letter trigram model. The latter is needed in order to be able to generate words that are not in the word unigram model. However, despite being trained on a long list of names, the letter trigram model occasionally produce unlikely candidates. Unlikely candidates can be eliminated by filtering out candidates with zero Web counts.

The Web-based filtering is useful only for our spelling-based model since all candidates generated by the phonetic-based model are in the pronunciation dictionary and all have non-zero Web counts. A

comparison of the transliteration accuracy with and without the Web-based filtering is shown in Table 4.

## 8 Evaluation and Experimental Results

In this section, we present a comparison of the accuracy of the phonetic-based model, the spelling-based model, and the linear combination in transliterating names from Arabic into English on the development and test sets. We also, present the transliteration accuracy of human translators on the same task. The results presented in Section 8.1 and Section 8.2 are based on the exact-matching criterion (i.e., a transliteration is considered correct only if it exactly matches the one in the gold-standard). We also show the accuracy based on human-subjective evaluation in Section 8.3.

### 8.1 Evaluating Human Transliterations

We wanted to know how well human translators do in this task. So, we asked a bilingual speaker (a native speaker of Arabic) to transliterate the names in both data sets given the context they appear within in the Arabic document. Then, the transliterations provided by the human subject are compared with those in the gold-standard. The accuracy of the transliterations provided by the human translator is shown in Table 5.

Examples of the transliteration errors made by the human subject are shown in Table 6.

### 8.2 Evaluating Our Transliteration System

In this section, we present a comparison of the different transliteration models described in this paper. We first show in Section 8.2.1 the overall accuracy of the phonetic-based model, the spelling-based model, and the linear combination of them. Then, in Section 8.2.2 we show how the presence of names in the pronunciation dictionary affects the transliterations obtained using our models. We also present some transliteration errors made by our algorithm in Section 8.2.3.

#### 8.2.1 Transliteration Accuracy of Our Algorithm

Table 7 shows the transliteration accuracy of the spelling-based model, the phonetic-based model,



Correct	Human Subject	Arabic
Jon Kyl	John Keele	جون كيل
Barak	Baraq	باراك
Karen Hughes	Karen Meuz	كارين هيوز
Karen Hughes	Karen Hoggs	كارين هوجز
Condoleezza Rice	Condolisa Rice	كوندوليزا رايس
Ariel Sharon	Arial Sharon	اريل شارون
Yehuda Wilk	Yuhuda Welleck	يهودا ويلك
Gilead Sher	Jelaad Shir	جلعاد شير
Hubert Vedrine	Ubair Federene	أوبير فدرين
Richard Boucher	Richard Bautcher	ريتشارد باوتشر
Ronald Reagan	Ronald Regan	رونالد ريغان
Ovadia Yosef	Ufadia Joseph	عوفاديا يوسف
Stephen Miles	Stephan Miles	ستيفن مايلز
John Bolton	George Bolton	جون بولتون
Jean Chretien	Jan Kretaine	جان كريتيان
Vladimir Voronin	Vladimir Vlaronin	فلاديمير فلرونين *
Rudolf Scharping	Rudolph Sharping	رودولف شارينج
Emile Lahoud	Amil Lahud	أميل لحود
Eduard Drumont	Edward Dremon	ادوارد دريمون
Fyodor Dostoevsky	Vieord Dyestoyavsky	فيورد ديستويفسكي *
Dostoevsky	Dyestoyavsky	دستويفسكي
Boris Pasternak	Boris Bastranak	برويس باسترناك *
Ivan Aksakov	Ivan Aksasov	ايفان اكساكوف
Jacques Chirac	Jack Sherak	جاك شيراك
Walter Shipley	Walter Shibly	والتر شيبلي
Dick Cheney	Dick Chaney	ديك شيني
Lawrence Lindsey	Lawrance Linzi	لورانس لينزي
Slade Gorton	Slid Gorotn	سليد جورتون
Pete Laney	Bet Lenny	بيت لينني
Kenneth Lay	Kenneth Lai	كينيث لاي
Tony Garza	Tony Jaroza	توني جارزا
Bulent Ecevit	Boland Ajaweed	بولنت أجاويد
Francis Okello	Francis Akopolo	فرنسيس اكوپولو

Table 6: Examples of the transliteration errors made by the human subject. Names marked with “\*” were misspelled in the Arabic document.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Phonetic-Based (Top1)	15.59	59.53	28.95	35.20
Spelling-Based (Top1)	19.89	63.42	42.34	43.79
Combination (Top1)	17.20	61.87	42.11	42.62
Phonetic-Based (Top20)	19.89	87.16	50.48	54.82
Spelling-Based (Top20)	31.72	85.21	56.69	59.82
Combination (Top20)	28.49	87.94	57.90	60.51

(a) Accuracy on the Development Test Set.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Phonetic-Based (Top1)	43.40	48.49	6.52	37.16
Spelling-Based (Top1)	48.11	59.10	54.35	52.75
Combination (Top1)	47.17	48.48	54.35	49.08
Phonetic-Based (Top20)	46.23	100	47.82	62.85
Spelling-Based (Top20)	70.75	98.50	56.52	76.15
Combination (Top20)	70.75	100	58.70	77.06

(b) Accuracy on the Blind Test Set.

Table 7: A comparison between the spelling-based model, phonetic-based model, and their linear combination on the development and blind test sets by category.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Phonetic-Based (Top1)	53.70	66.81	48.40	56.85
Spelling-Based (Top1)	59.26	71.18	63.60	66.42
Phonetic-Based (Top20)	68.51	97.81	84.40	88.56
Spelling-Based (Top20)	77.78	95.20	80.80	86.68

(a) Accuracy on the Development Test Set.

Experiment	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Phonetic-Based (Top1)	77.97	48.49	13.64	55.10
Spelling-Based (Top1)	86.44	59.10	90.90	74.83
Phonetic-Based (Top20)	83.07	100	100	93.20
Spelling-Based (Top20)	93.22	98.50	90.90	95.24

(b) Accuracy on the Blind Test Set.

Table 8: A comparison between the spelling-based and phonetic-based models on names in the pronunciation dictionary for the development and blind test sets by category.

and the linear combination on the development and blind test set. The spelling-based model was by far more accurate than the phonetic-based model in all three categories and on both data sets. Because it combines the transliterations of the two models, we expected the linear combination to be the most accurate. However, this was not the case. The linear combination was slightly worse than the spelling-based model when considering only the top candidate, and slightly better when considering the top-20 candidates. The reason we believe is that equal weights were given to the phonetic-based and spelling-based models in the combination. Weighting the spelling-based model higher will most likely give more accurate transliterations.

### 8.2.2 Phonetic-Based vs. Spelling Based on Names in the Dictionary

As we have described in Section 4, the phonetic-based model uses a pronunciation dictionary to convert an English phoneme sequence to an English word sequence. Consequently, only words with known pronunciations (from the dictionary) can be generated using this model. Therefore, the spelling-based model generally has a higher transliteration accuracy. But, does the spelling-based model generate more accurate transliterations for words with known pronunciations? We expected the answer to this question to be no. But much to our surprise, the spelling-based model produced more accurate transliterations on all categories, as shown in Table 8. When top-20 transliterations were considered, the spelling-based model was slightly less accurate.

As expected, the transliterations for names in the pronunciation dictionary are much more accurate than those that are not in it. This is because the word unigram model  $P(w)$  was trained on names in the dictionary.

### 8.2.3 Examples of Transliteration Errors

Table 9 shows some examples of the transliteration errors made by our transliteration algorithm. Some of the errors occurred were in fact not errors but rather acceptable alternative transliterations. However, many were true errors. The human-subjective evaluation described in Section 8.3 helps distinguish between these two cases.

## 8.3 Human-Subjective Evaluation

The evaluation results presented so far consider a transliteration correct only if it matches the gold-standard. In some cases where more than one possible transliteration is acceptable, this criterion is too rigid. To address such cases, we must ask a human

subject to determine the correctness of transliterations. We asked a native speaker of English with good knowledge of Arabic to decide whether any given transliteration is correct or not. This human-based evaluation is done for both the transliterations provided by the human translators and by our transliteration system.

The human subject was presented with the names in the Arabic script, their gold-standard transliterations, and the transliteration that we are evaluating. For our transliteration algorithm, the human subject was provided with the top 20 transliteration candidates as well.

The accuracy of the human translator based on the human-subjective evaluation is shown in Table 10. The accuracy of our transliteration models based on the human-subjective evaluation is shown in Table 11.

The human translator’s accuracy based on the human-subjective evaluation was higher than the exact-matching accuracy by about 11%. Most of the increase came from the forward-transliteration of Arabic names. This was expected because for Arabic names, typically many variant transliterations are acceptable. This was also reflected on the human-subjective evaluation of our spelling-based model. However, the accuracy of our phonetic-based model remains almost the same as in the case of the exact-matching evaluation. This is because names that can be found in the dictionary have only a single spelling that for the most part agrees with our gold-standard. Also, most of the names in the dictionary are English names and with English names the human evaluator was rigid, mostly accepting only the exact-matching spelling.

## 9 Discussion

We have presented and evaluated a transliteration algorithm using phonetic-based and spelling-based models. This algorithm is most accurate on back-transliterating English names. The reason for this is that most names in the dictionary are of English origin. Hence, the language model was mostly trained on English names. One way to improve transliterations of non-English names is to train the language model on a list of non-English names in addition to the dictionary names.

Our current models do not deal with the issue of metathesis (e.g., metathesis of **v** and **r** between the spelling and the pronunciation of the name *Favre*) in person names across languages. Metathesis in person names into Arabic is often a result of wrong transliterations by the person who transliterated in the original name in Arabic. For example,

Correct	Our Transliterations	Arabic
Abdel Maqsd	Abd Al Maksoud	عبد المقصود
Mahmud Salameh	Mahmoud Slama	محمود سلامة
Amid Khuli	Amid Julie	عميد خولي
Massoud Barzani	Miss Awad Barsanti	مسعود بارزاني
Mikhail Kasyanov	Mikhail Casanova	ميخائيل كاسيانوف
Boris Gryzlov	Boris Grise Love	بوريس غريز洛夫
Ayman Khairy	Amen Kerrey	أمن خيري
Giora Eiland	Guerra Island	غيورا ايلاند
Ann Veneman	Ann Feinman	آن فينمان
Francois Rivasseau	Francois Revco	فرنسوا ريفاسو
Voronin	Vernon	فارونين
Yisrael Hasson	Israel Hassan	اسرائيل حسون
George W. Bush	George Deblois Bush	جورج دبليو بوش
Carl Rove	Carl Rough	كارل روف
Shalom Yerushalmi	Shalom Euro Salmi	شالوم يوروشالمي
Fawzi Abu Samrah	Fauci Abu Samara	فوزي أبو سمرّة
Sovirene	Sovereign	سوفرين
Morarji Desai	Margie Desai	موارجي ديساي *
Getachew Tefera	Jetta Cho Tavera	جيتاتشو تفيرا
Kofi Annan	Coffee Annan	كوفي عنان

Table 9: Examples of the transliteration errors made by our transliteration algorithm. Names marked with “\*” were misspelled in the Arabic document.

Data Set	Accuracy (%)			
	ARABIC	ENGLISH	OTHER	OVERALL
Dev. Test Set	79.03	77.43	54.31	66.85
Blind Test Set	83.96	92.42	52.17	79.82

Table 10: The transliteration accuracy for a human translator based on the human-subjective evaluation.

<b>Experiment</b>	<b>Accuracy (%)</b>			
	<b>ARABIC</b>	<b>ENGLISH</b>	<b>OTHER</b>	<b>OVERALL</b>
Phonetic-Based (Top1)	15.59	59.53	28.95	35.20
Spelling-Based (Top1)	27.96	63.42	44.74	46.69
Combination (Top1)	23.12	61.87	45.22	45.41
Phonetic-Based (Top20)	20.43	87.16	50.48	54.93
Spelling-Based (Top20)	43.55	85.21	59.81	63.88
Combination (Top20)	44.63	87.94	61.25	65.62

(a) Accuracy on the Development Test Set.

<b>Experiment</b>	<b>Accuracy (%)</b>			
	<b>ARABIC</b>	<b>ENGLISH</b>	<b>OTHER</b>	<b>OVERALL</b>
Phonetic-Based (Top1)	43.40	48.49	6.52	37.16
Spelling-Based (Top1)	54.72	59.09	58.70	56.88
Combination (Top1)	56.60	48.48	54.35	53.67
Phonetic-Based (Top20)	46.23	100	47.82	62.85
Spelling-Based (Top20)	75.47	98.48	63.05	79.82
Combination (Top20)	79.24	100	63.05	82.11

(b) Accuracy on the Blind Test Set.

Table 11: A comparison between the spelling-based model, phonetic-based model, and their linear combination on the development and blind test sets by category. The evaluation is based on human-subjective evaluation.

the name *Dostoevsky* was found in our Arabic corpus transliterated as “ديستويفسكي *dystwyfsky*” and “ديستويفسكي *dystwyfksy*” (a metathesis of **k** and **s**); the name *Ordzhonikidze* was found transliterated as “اردجونيكيديزي *ārdġwnykydzy*” and “اردجيکينيدزة *ā-rdġykyndzt*” (a metathesis of **k** and **n**). This causes incorrect transliterations of these names by our system.

The transliteration accuracy on the blind test set for both our system and the human translator is significantly higher than the development test set. This is because the blind set is mostly of highly frequent, prominent politicians; whereas the development set contains also names of writers and less common political figures and hence are less likely to be in our unigram language model (and our pronunciation dictionary in the case of the phonetic-based model).

## References

- Mansur Arbabi, Scott M. Fischthal, Vincent C. Cheng, and Elizabeth Bart. 1994. Algorithms for arabic names transliteration. *IBM Journal of Research and Development*, 38(2).
- Asanee Kawtrakul, Amarin Deemagarn, Chalathip Thumkanon, Navapat Khantonthong, and Paul McFetridge. 1998. Backward transliteration for Thai document retrieval. In *Proceedings of the 1998 IEEE Asia-Pacific Conference on Circuits and Systems (APCCAS)*, pages 563–566.
- Kevin Knight and Jonathan Graehl. 1997. Machine Transliteration. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 128–135. Morgan Kaufmann.
- Bonnie G. Stalls and Kevin Knight. 1998. Translating names and technical terms in Arabic text. In *Proceedings of the COLING/ACL Workshop on Computational Approaches to Semitic Languages*.