# EMPIRICAL PROPERTIES OF MULTILINGUAL PHONE-TO-WORD TRANSDUCTION

*Geoffrey Zweig*
Microsoft Research
gzweig@microsoft.com

*Jon Nedel*
U.S. Department of Defense
jnedel@gmail.com

## ABSTRACT

This paper explores the error-robustness of phone-to-word transduction across a variety of languages. We implement a noisy channel model in which a phonetic input stream is corrupted by an error model, and then transduced back to words using the inverse error model and linguistic constraints. By controlling the error level, we are able to measure the sensitivity of different languages to degradation in the phonetic input stream. This analysis is carried further to measure the importance of each phone in each language individually. We study Arabic, Chinese, English, German and Spanish, and find that they behave similarly in this paradigm: in each case, a phone error produces about 1.4 word errors, and frequently incorrect phones matter slightly less than others. In the absence of phone errors, transduced word errors are still present, and we use the conditional entropy of words given phones to explain the observed behavior.

***Index Terms***— Speech recognition, phonetic decoding, transduction, multilingual, ASR

## 1. INTRODUCTION

State-of-the-art speech recognition systems currently apply all the information sources at their disposal simultaneously in the decoding process. These sources consist of the pronunciation dictionary, the context model or decision tree, the language model, and the actual acoustic model or gaussians. This consolidation is most complete in decoders based of the Finite State Transducer paradigm [1, 2] where the dictionary, language model, and decision tree can be fully combined in advance of any decoding, but it is present in other decoding architectures as well, for example in the form of language model lookahead [3]. While this strategy is highly effective, from the research point-of-view it may be easier to implement and test new modeling techniques in a more decoupled framework.

Therefore, there has been a significant amount of work in recent years to support modularized recognizers for research purposes. In the FLaVoR architecture developed at Leuven University [4, 5], decoding is broken into a two step process, the first generating phone lattices and the second applying morpho-phonological and morpho-syntactic constraints to produce words. Similarly, in the *Automatic Speech Attribute Transcription* paradigm [6], it is proposed that the recognition process should proceed bottom up through multiple stages.

In an effort to better understand the properties of a modularized system, this paper studies the intrinsic difficulty of converting from phones to words. The first stage uses the phone set of [7] and associated acoustic models to recover a one-best phone sequence. The second stage uses a finite state transducer scheme to recover words from phones. In contrast with previous work on multi-stage decoding, our work relies solely on an error model in the transduction phase to formally model the mistakes that are made at the phone recognition level. The error model is an unconstrained model of IID insertions, substitutions and deletions, and more general than the *single error* model of [5]. The advantage of using the error model approach is that it allows us to directly implement a noisy channel model of speech communication, and to pose and answer a number of interesting questions. Specifically, we conduct a class of experiments that involves corrupting a reference phone sequence with a known error model, and then measuring our ability to recover words. This allows us to answer several questions that have not been well studied before:

1. How easy is it to recover words from a correct but unsegmented phone string, and how does this vary across languages?
2. As the phonetic input stream is corrupted with errors, how quickly is our ability to recover words degraded? Are there threshold effects where a small number of phonetic errors can always be detected and recovered from? How does this vary across languages?
3. Are errors in some phones more important that errors in others, and how does this vary across languages?
4. How do the computational requirements of the phone-to-word transduction process vary as the phonetic input is progressively degraded?

The remainder of this paper is organized as follows: in Section 2 we present the formulation of our method. Section 3 describes the CallHome dataset, and the phone recognizer that was used for the different languages. Section 4 examines the robustness of the transduction process to phonetic errors, and presents an explanation for the observed behavior. Section 5 addresses the question of whether some phones are more important than others. Section 6 offers concluding remarks.

## 2. FORMULATION

In the noisy-channel model we adopt, we assume that the sender begins with a sequence of words he or she intends to communicate, and speaks a phonetic sequence determined by the pronunciations of those words. A phone recognizer then processes the audio and produces an errorful version of the intended phones. The receiver gets this corrupted phone sequence and must decode the likeliest sequences of intended words. This can be more precisely stated if we let $\mathbf{w}_i$ denote the intended words, $\mathbf{p}_i$ denote the intended phone sequence, and $\mathbf{p}_c$ denote the corrupted phone sequence. The job of the decoder is then to determine

$$\begin{aligned}
\arg\max_{\mathbf{w}} P(\mathbf{w}|\mathbf{p}_c) &= \arg\max_{\mathbf{w}} P(\mathbf{w})P(\mathbf{p}_c|\mathbf{w}) \\
&= \arg\max_{\mathbf{w}} P(\mathbf{w}) \sum_{\mathbf{p}_i} P(\mathbf{p}_i, \mathbf{p}_c|\mathbf{w}) \\
&= \arg\max_{\mathbf{w}} P(\mathbf{w}) \sum_{\mathbf{p}_i} P(\mathbf{p}_i|\mathbf{w})P(\mathbf{p_c}|\mathbf{p}_i, \mathbf{w}) \\
&\approx \arg\max_{\mathbf{w},\mathbf{p_i}} P(\mathbf{w})P(\mathbf{p}_i|\mathbf{w})P(\mathbf{p_c}|\mathbf{p}_i)
\end{aligned}$$

The factors involved in the maximization each have simple interpretations: $P(\mathbf{w})$ is given by the language model; $P(\mathbf{p}_i|\mathbf{w})$ is given by the pronunciation model; and $P(\mathbf{p}_c|\mathbf{p}_i)$ is given by the phone-level error model. In all the experiments described subsequently, we use a first-order error model with insertion and deletion probabilities for every phone, and substitution probabilities for all pairs of phones. Table 1 illustrates an example of our noisy channel model.

There is a simple representation of this model in terms of finite state transducer operations. Denote the intended word sequence by $W$, the pronunciation dictionary by $P$, the language model by $L$, the error model by $E$, the process of sampling a random path through a finite state acceptor by $sample$, and the process of finding the likeliest path by $bestpath$. Then the received (corrupted) phone sequence $R$ is given by $R = sample(W \circ P \circ E)$. The operation of decoding can be represented as $bestpath(R \circ E^{-1} \circ P^{-1} \circ L)$.

Given this formulation, it is possible to explore the questions raised in section 1. To find the intrinsic difficulty of recovering words from phones in the error-free case, we implement the noisy channel model with an "identity" error model that never inserts or deletes, and always replaces a phone by itself. To study the sensitivity of the decoding process to phone errors, we construct error models with various error rates, and then compute $bestpath(sample(W \circ P \circ E) \circ (E^{-1} \circ P^{-1} \circ L))$. Finally, it is possible to explore the importance of single phones. Let $E_p$ be the original error model $E$, except that errors involving phone $p$ are adjusted to have zero probability. Then measuring the difference between using $E$ and $E_p$ in the round-trip process gives an indication of the importance of $p$. We have explored the use of this methodology in five of the CallHome languages and using an acoustic model that uses a universal phone set. The database and acoustic model are described next.

| | |
|---|---|
| Intended words | I'm sorry we'll blame him |
| Intended phones | aI m S a r i: w i: l b l ei m H I m |
| Corrupted phones | aI m S a r i: w i: D l ei m H I m |
| Recovered words | I'm sorry we blame him |

**Table 1**. Steps in the noisy channel model

## 3. DATABASE AND ACOUSTIC MODELS

### 3.1. CallHome

In order to work with a data set with roughly equal resources across a variety of languages, we used the CallHome database [8]. This database has speech, transcriptions, and lexica in Egyptian Arabic, Mandarin Chinese, English, German, Japanese, and Spanish. The audio data for each language consists of 120 telephone conversations of up to 30 minutes each (100 conversations for German). Eighty of the conversations are marked as training data and 20 each for development and test, except for German which has development data only. Since the experiments did not involve parameter tuning, and a test set is absent for German, all results are reported on the development set. Due to a high out-of-vocabulary rate for the Japanese lexicon, we did not use the Japanese language data.

### 3.2. The UPR

To conduct our experiments, we need a phone-level error model for each language, reflecting realistic error patterns. To obtain these error models, we decoded the training data with acoustic models based on a universal phone recognizer (UPR) provided by the Department of Defense [7]. This recognizer uses 259 phones based on the International Phonetic Alphabet (IPA), and represents an effort similar to that pioneered with the GlobalPhone project and others [9, 10].

The UPR system was built using the HTK Recognizer, version 3.3 and was trained iteratively, starting with data that was transcribed at the phone level, and later incorporating data that was transcribed at the word level. In the first stage of training the UPR, phonetically transcribed data was taken from the Phonetic Switchboard Corpus [11, 12] in English, and the OGI-MLTS Corpus [13] in English, German, Hindi, Japanese, Mandarin Chinese, and Spanish. Word-level transcriptions were later used to incorporate data from LDC data sets (e.g. CallHome and CallFriend) in a variety of languages. The total amount of acoustic training data used in the five languages studied here varied from about 15 hours in German to 88 hours in English. The overall training process was designed to ensure that sounds represented by a given phone are consistent across languages and that important phonemic distinctions in one language are annotated in all languages.

The UPR acoustic models have diphone acoustic context, with 17 gaussians per state. The acoustic features were 39-dimensional, consisting of cepstra, deltas and double-deltas, and decoding was performed at the speaker-independent level.

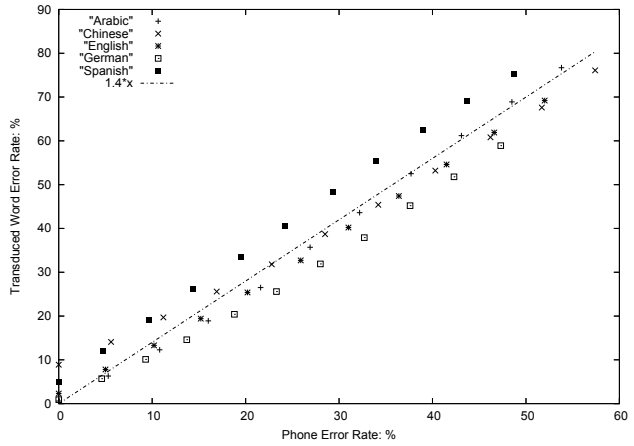


**Fig. 1**. Output word error rate vs. input phone error rate

The UPR makes use of n-gram phonotactic language models trained on transcripts of LDC data as well as data found on the Web. Language-specific phonotactic bigram language models were built for all the languages used in our experiments. Further details of the UPR phone set, acoustic model, and phonotactic language models can be found in [7].

The UPR can be run using either a truly universal model or using language-specific models. We used language-specific models to decode the CallHome training data and create the error models. The phone-error rates on the test data varied from 56.4% in English to 63.0% in German.

## 4. ROBUSTNESS TO PHONETIC ERRORS

This section reports on the sensitivity of the transduction process to the overall error level in the input phone stream. The experiments all use a base error model that is obtained by decoding the CallHome training data with the UPR, aligning it to the reference phoneme strings, and computing the various substitution, insertion and deletion probabilities. This is done separately for each language. To obtain error models at a variety of absolute error levels, we then scale this matrix down by moving probability mass from insertions, deletions and non-identity substitutions to identity substitutions. By corrupting the reference phones with the various error matrices and then measuring our ability to recover the correct words, we determine the sensitivity of the decoding process to input errors.

### 4.1. Accuracy and Speed

Figure 1 plots transduced word error rate (WER) as a function of the input phone error rate (PER). To a first approximation, the two are related by $WER = 1.4PER + \epsilon_{lang}$. The slope in all cases is approximately 1.4, and there is a language dependent y-intercept. These results show no evidence of redundancy – if redundancy were present, one would expect a threshold effect in which very low phone-error rates would have little or no impact on word error rate. In terms of runtime, we have found that whereas the word error rate scales linearly with phone error rate, the runtime increases exponentially from less than one-two thousandth realtime in the absence of error to one-tenth realtime with about 50% phone error rate. Again, this is similar across the languages studied. All experiments were run with a fixed beam such that there was little accuracy loss at high phone error rates.

| | Phone-to-word WER | Entropy: bits |
|---|---|---|
| Egyptian | 0.6% | 0.0020 |
| German | 0.9 | 0.029 |
| English | 2.3 | 0.080 |
| Spanish | 5.0 | 0.18 |
| Mandarin | 8.9 (CER) | 0.44 |

**Table 2**. Conditional entropy of words given phones

### 4.2. Conditional Entropy: Explaining the y-intercept

The transduced word error rate achieved in the absence of any phone errors is not zero, and differs by over a factor of ten from 0.6% to 8.8% across the different languages. To understand the observed differences in the y-intercept, we examine the conditional entropy of words given phones, which can be computed as the entropy of the words less the mutual information between phones and words. To define the mutual information between phones and words, let $\mathbf{r_s}$ be the phone sequence for utterance $s$ in the database. Let $\mathbf{l_s}$ be the word sequence. Note that sums over $s$ are thus over the observed data segments. Let $R$ and $L$ be phone-sequence and word-sequence variables respectively that take specific values such as $\mathbf{r_s}$ and $\mathbf{l_s}$. Then

$$\begin{aligned} M(L;R) &= \sum_{L,R} P(L,R) \log \frac{P(L,R)}{P(L)P(R)} \\ &\approx \sum_{s} \log \frac{P(\mathbf{r}_s, \mathbf{l}_s)}{P(\mathbf{r}_s)P(\mathbf{l}_s)} \\ &= \sum_{s} \log \frac{P(\mathbf{r}_s|\mathbf{l}_s)}{\sum_{\mathbf{w}} P(\mathbf{r}_s|\mathbf{w})P(\mathbf{w})} \end{aligned}$$

$P(\mathbf{w})$ is given by the language model. $P(\mathbf{r_s}|\mathbf{w})$ is the probability of an observed phone string given a word string. It is given by the sum over all the alignments of $\mathbf{r_s}$ to the phones in $\mathbf{w}$ of the probability of the substitutions, insertions and deletions in the alignment, and can be computed using dynamic programming.

The quantity $M(L;R)$ is a measure of how much information the phones provide about the words. If we let $H(L)$ be the entropy of the language, then $M(L;R) - H(L)$ provides a measure of the excess information that is available when inferring words from phones, and its negative is in fact the entropy of the language conditioned on knowledge of the phone strings. In general, $M(L;R)$ is difficult to compute since it involves summing over all possible word sequences in the de-

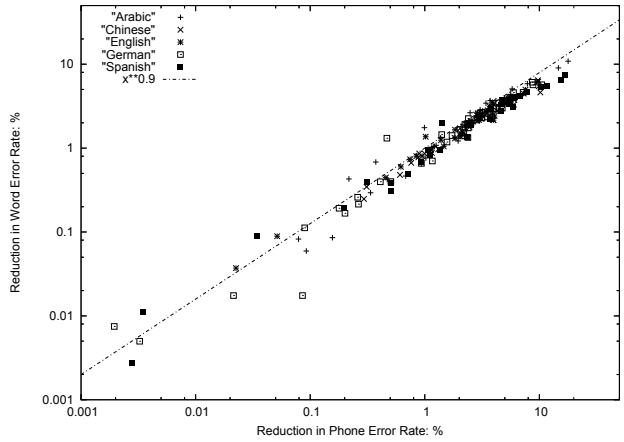


**Fig. 2**. Sensitivity to individual phones

nominator. To simplify the computation, we have approximated the sum over all data segments by a sum over the words in the lexicon weighted by their unigram frequency. Essentially this uses a notional data set consisting of the words in the lexicon. Table 2 shows the conditional entropy along with the round trip word word error rates. It can be seen that there is a good correlation between this entropy and the observed word error rate.

## 5. SENSITIVITY TO INDIVIDUAL PHONES

By using our noisy channel model, we have been able to study the sensitivity of word error rate to individual phones in two ways. The first uses the corruption process described in section 2. Insertions, deletions and substitutions are made according to the empirically derived error model, with one exception: all errors involving a particular phone are excluded. The corruption process is run separately for each phone, and the resulting strings are transduced to words. The transduced word error rate is then computed, and we compute the decrease in error rate over the baseline where no errors are excluded. To normalize against frequency effects, we also count the number of phone errors that have been excluded from the input. This allows us to create a scatterplot of the number of word errors corrected after transduction against the number of phone errors corrected on the input side. This is shown in Figure 2 for each phone in each of the languages studied.

The second method of computing sensitivity to individual phones avoids the artificial corruption process. This is done by aligning the phone-level UPR output to the reference phone string. Then, for a particular phone, we fix all the errors involving the phone. The remaining steps are identical to the first method, and we obtain another scatterplot. This plot is similar to that of Figure 2 with somewhat greater dispersion. The fact that Figure 2 is on a log-log scale with a slope of about $0.9$, indicates that there is a slight tendency such that phones which are frequently involved in errors are relatively downweighted.

## 6. DISCUSSION

This paper has examined the robustness of phone-to-word transduction in a variety of languages and over a range of error rates. We find that the introduction of a phone error on average creates about $1.4$ word errors, and this is seen to be constant across the five languages studied, and across a wide range of absolute error levels. At the level of individual phones, the sensitivity to errors is almost linear as well, but seems to be optimized in the sense that frequently misleading phones have slightly less impact per error than their more reliable counterparts.

### Acknowledgements

We thank Pat Schone for assistance with the UPR and Hynek Hermansky, Alex Acero, Li Deng and Patrick Nguyen for many helpful comments and references.

## 7. REFERENCES

[1] M. Mohri, F. Pereira, and M. Riley, "Weighted finite state transducers in speech recognition," *Computer Speech and Language*, vol. 16, no. 1, pp. 69–88, 2002.

[2] G. Saon, G. Zweig, and D. Povey, "Anatomy of an extremely fast LVCSR decoder," in *Interspeech*, 2005.

[3] S. Ortmanns, H. Ney, and A. Eiden, "Language-model look-ahead for large vocabulary speech recognition," in *ICSLP*, 1996.

[4] K. Demuynck, T. Laureys, D. Van Compernolle, and H. Van Hamme, "FLaVoR: a flexible architecture for LVCSR," in *Eurospeech*, 2003.

[5] K. Demuynck, D. Van Compernolle, and H. Van Hamme, "Robust phone lattice decoding," in *Interspeech*, 2006.

[6] C.H. Lee, M. Clements, S. Dusan, Eric Fosler-Lussier, K. Johnson, B.H. Juang, and L. Rabiner, "An overview on automatic speech attribute transcription," in *Interspeech*, 2007.

[7] B. Walker, B. Lackey, J. Muller, and P. Schone, "Language-reconfigurable universal phone recognition," in *Eurospeech*, 2003.

[8] "Linguistic data consortium," http://www.ldc.upenn.edu/.

[9] T. Schultz and A. Waibel, "Language independent and language adaptive acoustic modeling for speech recognition," *Speech Communication*, 2001.

[10] C. Corredor-Ardoy, L. Lamel, M. Adda-Decker, and J.L. Gauvain, "Multilingual phone recognition of spontaneous telephone speech," in *ICASSP*, 1998.

[11] J. Godfrey, E. Holliman, and J. McDaniel, "SWITCHBOARD: Telephone speech corpus for research and development," in *ICASSP*, 1992, pp. 517–520.

[12] S. Greenberg, "The switchboard transcription project," Tech. Rep., Johns Hopkins Workshop on Innovative Techniques for LVCSR, 1996.

[13] A. Cole, Y. Muthusamy, and B. Oshikal, "The OGI multilanguage telephone speech corpus," in *ICSLP*, 1992.