Bi-directional Conversion Between Graphemes and Phonemes Using a Joint N-gram Model

Lucian Galescu, James F. Allen

Department of Computer Science
University of Rochester, U.S.A.
{galescu,james}@cs.rochester.edu

Abstract

We present in this paper a statistical model for language-independent bi-directional conversion between spelling and pronunciation, based on joint grapheme/phoneme units[1] extracted from automatically aligned data. The model is evaluated on spelling-to-pronunciation and pronunciation-to-spelling conversion on the NetTalk database and the CMU dictionary. We also study the effect of including lexical stress in the pronunciation. Although a direct comparison is difficult to make, our model's performance appears to be as good or better than that of other data-driven approaches that have been applied to the same tasks.

1. Introduction

The problem of spelling-to-pronunciation (S2P) conversion has been studied to a large extent in the context of text-to-speech systems, which usually require an intermediate phonemic representation. Also, a significant amount of research on S2P conversion has stemmed from work on psycho-linguistic models of reading aloud [2]. The reverse problem, of pronunciation-to-spelling (P2S) conversion, has received far less attention. A new impetus to research on automatic P2S and S2P techniques has arisen in connection with the recent advances in large vocabulary continuous speech recognition, and especially in spontaneous speech recognition, where it has become more and more important to find a way of dealing with words outside the recognizer's vocabulary and with pronunciation variants that differ from the known baseforms. Moreover, as the technology is being transferred to new languages, it has become increasingly important to find automatic techniques that would avoid long cycles of development and maintenance of pronunciation dictionaries by expert lexicographers.

Many S2P conversion techniques fall into two major categories: rule-based systems and pronunciation by analogy (PbA) systems. The first derive pronunciations by dictionary lookup, and if this fails, by successively applying carefully crafted pronunciation rules to the input word. PbA systems use some measure of similarity between words to retrieve partial pronunciations for fragments of the input word, which are then concatenated to obtain the final pronunciation. An excellent review (and critique) of these two approaches can be found in [3], which, to our knowledge, gives the only extension of the PbA framework to be applied to P2S conversion. [4] contains an extensive review of different S2P techniques and a comparative evaluation.

There are comparatively fewer studies on the P2S problem. An interesting approach is presented in [5], where a hierarchical framework incorporating several levels of linguistic description is used for bi-directional conversion.

Other bi-directional models are presented in [6,7,8,9], although not all include P2S evaluations. [10] describes an HMM-based approach to the P2S problem. These are all based on statistical finite-state transduction methods, which are data-driven techniques that attempt to infer probabilistic transformation rules from large databases of spelling/pronunciation pairs. Our work continues in this tradition.

Specifically, we developed probabilistic techniques to align spellings with pronunciations automatically. From the resulting alignments a set of grapheme-to-phoneme (GP) *correspondences* (following the terminology in [3]) is induced. An n-gram model trained on data aligned according to this set of correspondences can then be applied for bi-directional conversion between spelling and pronunciation.

In the next section we present the theoretical formulation of our model. We devote most of the paper to describing the model design, the evaluation methodology, and the results of the evaluation. We conclude with a discussion in which we relate our model and its performance to other approaches.

2. The bi-directional model

2.1. Inferring correspondences (alignment)

As a first step to building a pronunciation model, the spelling and pronunciation of each word in the training data need to be aligned. Many of the current approaches to the S2P problem work under the assumption that a one-to-one alignment between letters and phonemes is available. To do the trick, the phoneme set is augmented with a null (pseudo-) phoneme, and with special double phonemes standing for diphthongs and other groups of phonemes that correspond to one letter (e.g., /T_S/, /K_S/, etc.). Besides the arbitrariness of such data manipulations (see, for example, [11], for arguments against one-to-one correspondences), this is also problematic from a technical point of view: data aligned on a strict one-to-one basis is difficult to obtain, and doing it automatically introduces yet another level of imprecision [12]. Moreover, as will be seen in section 3., we found that it is possible to have more than two phonemes corresponding to one letter (at least for English; this is, of course, a language-dependent issue).

We decided to avoid these problems by automatically aligning the graphemic and phonemic strings presented in the

[1] Here, the term *grapheme* is used with the sense of "functional spelling unit" corresponding to a single phoneme (cf. [1]). In principle, we allow the possibility that a grapheme may correspond to multiple phonemes in order to handle diphthongs, insertions, and pronunciations of single letters.

input. The resulting grapheme-to-phoneme correspondences are constrained to contain at least one letter and one phoneme. The set of correspondences, with an associated probability distribution, is inferred using a version of the EM algorithm [13]. Due to space limitations, we refer the reader to [14] for a detailed exposition of the learning process. Examples of correspondences induced from the two data sets are given in section 3.

Because our alignment procedure is fully automatic and purely statistical (in that it does not use linguistic information like phonetic categories), and does not require seeding with manually-designed correspondences, it can be readily applied to other languages. For the time being, however, we have applied it only to English.

2.2. The joint n-gram model

N-gram models are used successfully in speech recognition and other applications for statistical modelling of sequences of data. They are flexible and compact, and fast decoding algorithms exist for them. In [15] an n-gram model is described that predicts words and syntactic/semantic tags simultaneously. Analogously, we can define a pronunciation model that is based on joint grapheme/phoneme pairs. This idea was also adopted in the stochastic grammar approach of [6] (the particular instantiation that they use is, in fact, a bigram model), and in the multigram model of [9].

Formally, if the spelling of a word w is $\gamma(w) = l_1 l_2 ... l_n$, and its pronunciation is $\pi(w) = p_1 p_2 ... p_m$, where l_i are letters, and p_i are phonemes, we associate with it the joint probability $P(\gamma(w), \pi(w))$.

Then, the S2P task can be formulated as finding, for the input γ, the most likely transcription π^*:

$$\pi^* = \arg\max P(\gamma, \pi) \tag{1}$$

Similarly, the P2S task can be formulated as finding, for the input π, the most likely transcription γ^*:

$$\gamma^* = \arg\max P(\gamma, \pi) \tag{2}$$

In the joint GP correspondence model, for every GP alignment $a = \langle g, f \rangle_{1,N} = \langle g_{1,N}, f_{1,N} \rangle$ of a word, where each unit $\langle g, f \rangle_i = \langle g_i, f_i \rangle$ is a correspondence between the grapheme g_i and the phoneme unit f_i, we define the probability

$$Q'(a) = Q'(\langle g, f \rangle_{1,N}) = \prod_{i=1}^{N} Q(\langle g, f \rangle_i \mid \langle g, f \rangle_{1,i-1}) \tag{3}$$

The conditional probability distributions Q represents the joint grapheme/phoneme probability model.

We can express $P(\gamma, \pi)$ in terms of Q, by summing over all possible alignments a of γ and π allowed by the set of GP correspondences:

$$P(\gamma, \pi) = \sum_{a \in A(\gamma, \pi)} Q'(a) \tag{4}$$

where we denote by $A(\gamma, \pi)$ the set of allowable alignments.

Finally, combining (1) with (4), and (2) with (4), we obtain the equations of the S2P and P2S tasks, respectively, based on the joint grapheme/phoneme probability model:

$$\pi^* = \arg\max \sum_{a \in A(\gamma, \pi)} Q'(a) \tag{5}$$

$$\gamma^* = \arg\max \sum_{a \in A(\gamma, \pi)} Q'(a) \tag{6}$$

A bi-directional joint n-gram model $Q'(a)$ is obtained by restricting to a given size the contexts in the conditional probability Q. The joint n-gram probabilities are estimated from aligned data using a standard maximum likelihood approach.

In decoding, the space of all possible alignments that match the input is searched; this can be done efficiently for n-gram models, but the summation in equations (5) and (6) may still be too expensive. A common approach is to replace the summation with maximization. The Viterbi algorithm [16] can accomplish the search efficiently.

The most common problem associated with n-gram models is that the number of probabilities to be estimated can be large, and thus a very large amount of data would be needed for training. When there isn't sufficient data for training, the model can be smoothed using lower-order models or class-based models [17,18].

Another deficiency of the n-gram models is that the context span is fixed and quite short. Many linguistic phenomena which have been modelled with n-grams may involve long-range dependencies. A typical solution to this problem is to augment the model's vocabulary with multi-unit tokens which may be designed manually or automatically [19].

3. Evaluation

Although the technique just described is language-independent, so far we only conducted an evaluation on English. We used two sets of data, the NetTalk manually aligned dictionary [20], and the CMU pronunciation dictionary [21]. The first was chosen because many S2P techniques have been tested on this data, and so, even if evaluation conditions may differ from one approach to another, it may serve as an indication to where we stand compared to the state of the art. The second data set is much larger, and contains many abbreviations, proper names, and loan words. Also, as this dictionary was primarily designed for continuous speech recognition, it incorporates many deviations from standard baseforms (allophonic variations, deletions, insertions). Only part of this dictionary was manually proofed. Thus, this dictionary provides an opportunity for more comprehensive and more realistic testing, especially for the P2S conversion.

We are also interested in predicting accented phonemes. Both dictionaries include stress markers in the pronunciation, but in the NetTalk transcriptions some of the stress markers are assigned to null phonemes. It would not have been difficult to correct this deficiency, but the results would not have been comparable to those in the literature. Therefore, we only tested the conversion of graphemes to accented phonemes on the CMU dictionary, which doesn't have this problem.

The ultimate measure of performance for both conversion tasks is the *word accuracy* (*WACC*), i.e., a transcription is counted as correct if and only it coincides with the one given in the dictionary. A decision has to be made when there may be possible correct outputs, as is the case with homographs and homophones. Our model does not encode syntactic and/or

semantic information about words, which would be required for making such a decision. We decided against the common *ad hoc* approach of removing homophones and homographs from the data. A statistical model can output several hypotheses if they have close scores, or an n-best list, and it could be left to other modules in a natural language system to decide which one best fits whatever linguistic constraints there might be (e.g., part of speech).

In our scoring strategy, whenever a word had several pronunciations in the test set, the result of the S2P conversion was matched against the one yielding the minimal *phoneme error rate* (*PER*). A similar strategy was applied for P2S conversion, when a pronunciation would correspond to different spellings, by choosing the word with the minimal *letter error rate* (*LER*). The symbol error rate is defined as the number of total errors (deletions, insertions, and substitutions) over the number of all symbols.

Because some authors prefer to report symbol accuracy, i.e., proportion of symbols correct to the number of all symbols, we also provide *phoneme accuracy* (*PACC*) and *letter accuracy* (*LACC*) results.

3.1. Experiments on the NetTalk dictionary

The NetTalk dictionary is publicly available and was originally developed and manually aligned by Sejnowski and Rosenberg for training the NetTalk neural network [20]. It contains of 19,802 words, some with multiple pronunciations, so that the total number of entries is 20,008. The phone set uses 52 phonemes, including the null phoneme and five double phonemes, to insure a one-to-one correspondence between the letter string and the phoneme string. Three stress markers are used, and two other markers provide syllabic information. A detailed description of all the symbols can be found in [22].

For the evaluation, we split the whole vocabulary in 10 randomly selected disjoint test sets of 1,980 words, in which we then included, for each word, all the pronunciations in the NetTalk dictionary. Corresponding to each test set, a training set was formed by including all unique entries in the dictionary for the rest of 17,822 words. The null phoneme was eliminated from all the pronunciations.

The GP correspondences were obtained on the whole database. 494 units were inferred, but in each of the training sets around 480 units were typically used. Some examples of correspondences are given in Table 1. The upper part of the table contains some less straightforward, but obviously correct correspondences, while the lower part contains some obviously wrong correspondences, resulting from incorrect alignments. It should be noted, however, that the notion of "correct" correspondence is ill-defined, and we only rely on the performance of the correspondence model in the conversion tasks as a quality measure.

The correspondence model was then used for aligning the data using an adapted version of the Viterbi algorithm [16]. A back-off 4-gram model (the GP model) with Witten-Bell discounting was then trained on the aligned data, using the CMU-Cambridge Toolkit [23].

Cross-evaluation results for the S2P and P2S conversion tasks are shown in Table 2. Closed test performance is included, in which a model was estimated from the whole data set. The high word accuracy shows that 4-grams are able to describe quite well the information encoded in the dictionary, thus having good potential for lexicon compression.

It has been found in other studies that vowel quality confusion is the leading source of errors in S2P conversion. Indeed, we found that substitution between a schwa and another vowel is responsible for a full 13% (absolute) of word errors.

NetTalk		CMU	
eigh	/e/	'S	/AH0 Z/
che	/S/	U	/Y AH0 W/
xi	/K/	Z	/T S/
ngue	/G/	EAULT	/OW1/
oub	/W/	IST	/AH0/
pbo	/b/	EB	/EH1/
r	/Y/	M	/IY1 EH1 M/
ic	/A/	C	/IY1/

Table 1: Examples of GP correspondences inferred from the NetTalk and the CMU dictionaries.

S2P	***WACC*** [%]	***PACC*** [%]	***PER*** [%]
closed	93.75	98.70	1.31
GP	63.93	91.74	9.00
P2S	***WACC*** [%]	***LACC*** [%]	***LER*** [%]
closed	91.01	98.51	1.89
GP	58.13	92.22	10.03

Table 2: S2P and P2S closed and open test results for the NetTalk task.

3.1.1. Class-based smoothing

Classes of correspondences were obtained using the greedy algorithm for automatic classification described in [18]. We used the implementation found in the SRILM toolkit [24]. A 3-gram class-based model was built, and then interpolated with the baseline 4-gram. We assigned a weight of 0.1 to the class-based model. The training and testing were done on the first of the 10 pairs of evaluation data.

Table 3 shows the results on both P2S and S2P for smoothing with 400 and 300 classes (there were 476 correspondences in the training set). The GP1 model is the baseline (the slightly lower word accuracy, compared to the cross-evaluation average reported in Table 2 indicates that this particular test set has a larger proportion of difficult words). The class-based interpolated models are indicated by the number of classes in each.

S2P	***WACC*** [%]	***PACC*** [%]	***PER*** [%]
GP1	63.3	91.8	8.9
400	63.7	92.0	8.7
300	63.5	92.0	8.8
P2S	***WACC*** [%]	***LACC*** [%]	***LER*** [%]
GP1	57.8	92.2	9.8
400	57.9	92.1	9.8
300	57.9	92.1	9.8

Table 3: S2P and P2S results for the class-based smoothed models.

A small improvement can be seen on all measures on the S2P task; there is no significant improvement on the P2S task.

It appears that it would be most beneficial to use classes for only a small number of units, as otherwise over-smoothing may occur.

3.1.2. Chunk-based models

In speech recognition, including a number of phrases (multi-word units) into the vocabulary of an n-gram model has been shown to lead to some improvement. Useful phrases can be found automatically by iteratively "gluing" together the most promising sequences of two units. Various techniques have been used to measure how promising a sequence is (sometimes called "stickiness" measures): bigram frequency, mutual information, change in bigram log-likelihood, etc. [25].

In two pilot experiments, we tested the usefulness of augmenting the vocabulary of our basic model some multi-correspondence units (chunks). In the first experiment, we used bigram frequency as a stickiness measure, and in the second was based on mutual information (MI). In each case, the training data was processed to replace all the appropriate correspondence sequences with chunks. We then estimated chunk-based 4-gram models and interpolated them, with equal weights, with the baseline model. We note in passing that the interpolated chunk-based model is theoretically a multigram model [9], although it is estimated differently.

Unfortunately, the results for the chunk-based models were rather disappointing. Frequency-based models had a lower performance than the MI-based ones. On the S2P task, the MI-based chunk models were slightly worse than the baseline. On the P2S task, adding up to 200 MI-based chunks actually showed some increase in performance, but a very small one.

Although these experiments were not compelling, we believe that, with more careful (but, admittedly, more time-consuming) joint optimization of the number of chunks and the interpolation parameters, the technique will eventually prove useful. We were encouraged by the fact that the chunks that were found included a number of morphologically meaningful units, like affixes (e.g., "-ize", "-ing", "-ment", "inter-", "-ism", etc.), words that appear often in compounds (e.g., "man", "hand", "form", "port), and also some frequently co-occurring letter units with special pronunciation (e.g., "qu" - /k w/, "ng" - /G k/, etc). However, frequent sequences are also predicted well by the baseline model; from our preliminary analysis we conclude that augmenting the vocabulary with chunks based only on low-frequency sequences with high mutual information will in fact be beneficial.

3.1.3. Models based on one-to-one correspondences

The above results situate our approach among the best that have been applied to this task. This could be due to the fact that the model is based on more meaningful GP correspondences, or to the fact that the model represents jointly the graphemic and the phonemic dependencies.

In a final set of experiments, we compared our model to a joint 4-gram model based on one-to-one correspondences (model 1-1). The estimation procedure was exactly the same as for model GP1. As these correspondences have been thoroughly checked by hand, this gives us an opportunity to check how good the automatically learned GP correspondences really are.

Also, in order to asses the benefits of using some extra information, we built two other one-to-one 4-gram models in which we included in the training data the stress markers for the first one (model 1-1/S), and for the second both the stress markers and the syllable markers (model 1-1/SS).

We trained and tested these three models for the S2P task on the first of the 10 pairs of evaluation data. In order to compare the performance of these models to that of the GP1 model, all the markers and the null phonemes were removed in the output of the 1-1/S and 1-1/SS models before scoring.

Results are shown in Table 4. Compared to the GP1 model, the one-to-one models had a slightly (but consistently) lower word accuracy, but the phoneme accuracy and error rate were practically the same. These results allow us to conclude that the GP correspondences inferred are very reliable, and that the occasional errors probably don't have a big impact on the results (quite likely, the erroneous correspondences occur only once in the data). However, the one-to-one models performed quite well, given that they use less context, which gives us confidence that much of the power of our approach resides in modeling joint dependencies.

S2P	***WACC*** [%]	***PACC*** [%]	***PER*** [%]
GP1	63.3	91.8	8.9
1-1	62.7	91.8	8.9
1-1/S	62.5	91.7	8.9
1-1/SS	62.9	91.8	8.8

Table 4: Comparison between the model based on inferred correspondences and models based on manually designed one-to-one correspondences

3.2. Experiments on the CMU dictionary

The CMU pronunciation dictionary consists of more than 119,000 words, with a total of over 127,000 pronunciations. The phone set is composed of 39 phones. Lexical stress is indicated with one of three stress markers (0 = no stress, 1 = primary stress, 2 = secondary stress).

We reserved 12,000 words for testing (12784 different pronunciations, if stress is included), and the rest of the dictionary (about 90%) was used for training.

This dictionary contains many critical words with uncommon pronunciations, which is the reason most other approaches evaluated on it showed poor performance relative to the results on other English dictionaries. We chose, however, not to remove any words from the corpus.

From the training data we inferred two sets of GP correspondences, one with non-accented phonemes, and one with accented phonemes. The correspondence sets included 1115 and 1554 units, respectively. Some examples are given in table 1, above. As can be seen, there are many more units than were found on the NetTalk dataset. This is due to the large number of proper names, many of foreign origin, abbreviations, and non-standard pronunciations that include various phonological deviations from baseforms. We give below examples of a few uncommon correspondences and dictionary entries in which they occur:

<J, /HH/> and <ILL, /IY1/>	in "TRUJILLO"
<R, /AA1 K T ER0/>	in "DR"
<U, /UW0 W/ >	in "GRADUATE"

Table 1 contains an example of correspondence containing the apostrophe. This is the only non-alphabetic character

allowed; all other were removed. We constrained the correspondences not to allow it as the sole component of the graphemic part, but to attach it to one of the neighboring graphemes.

Again, we built back-off 4-gram models with Witten-Bell discounting based on the Viterbi-aligned training data. Open test results for these models are given in Tables 5 to 7. The GP-S models are based on units with non-accented phonemes, and the GP+S models on units with accented phonemes.

A class-based smoothed GP-S model, based on 900 classes, showed almost no improvement over the basic model on the S2P task and 0.4% (absolute) *WACC* improvement on the P2S task.

Although the CMU task would seem more difficult than the NetTalk task, the availability of a large amount of data for training makes for significantly higher results for S2P conversion. The same is not true, however, for P2S conversion. The fact that each phoneme appears in so many correspondences makes the task a very difficult one, indeed.

It is remarkable that including stress information in the training data improved the word accuracy in S2P conversion (Table 6). Also, the availability of stress information in the input for P2S conversion seemed to help a little.

We found that the apostrophe caused a large number of insertion errors in the P2S conversion, it being responsible for close to 4% (absolute) word errors.

S2P	***WACC*** [%]	***PACC*** [%]	***PER*** [%]
GP-S	71.5	93.6	7.0
P2S	***WACC*** [%]	***LACC*** [%]	***LER*** [%]
GP-S	50.3	91.2	11.5

Table 5: S2P and P2S open test results for the CMU task using non-accented phonemes.

S2P	***WACC*** [%]	***PACC*** [%]	***PER*** [%]
GP+S	72.3	93.8	10.9

Table 6: S2P open test results for the CMU task using accented phonemes in the training data and non-accented phonemes for testing.

S2P	***WACC*** [%]	***PACC*** [%]	***PER*** [%]
GP+S	62.6	91.0	9.6
P2S	***WACC*** [%]	***LACC*** [%]	***LER*** [%]
GP+S	50.6	91.6	11.2

Table 7: S2P and P2S open test results for the CMU task using accented phonemes.

4. Discussion

Because of the lack of standardized data sets and evaluation methodology, it is difficult to compare the performance of the joint n-gram model to that of other approaches. We will make an attempt, though, to relate our model to the state of the art. This will provide us with an opportunity to make some comments on the differences in evaluation methodology between various approaches, and to indicate some of the strengths of our model.

The 63.4% word accuracy in S2P conversion obtained on NetTalk is comparable to the best results obtained on this database. In [22], a 64.8% word accuracy is reported, with a model based on decision trees (ID3). However, the authors acknowledge using the test set for development, which may have lead to "overfitting" the test set. Also, they used only 1,000 word for testing and the rest for training. In [26], a PbA approach based on overlapping chunks achieves 63.96% *WACC* in a 10-fold cross evaluation, but 0.5% of all words were not pronounceable. In our model we use backing-off, which guarantees that every word will receive a pronunciation. A similar PbA model is described in [3], where a 65.5% *WACC* is obtained using the combined scores given by five different scoring strategies. The optimal combination was chosen *post hoc*. The best single strategy fared at 63%. In this study all homographs and the two one-letter words were removed for the purpose of S2P evaluation. Finally, a 65.8% *WACC* is reported in [27], with another decision tree model (CART); this model benefited from several smoothing techniques and from rescoring with a trigram phonotactic model. We noticed in the S2P output several errors stemming from inappropriate vowel and/or stress patterns; a phonotactic model might help correct such errors as well as some of the schwa substitution errors. Similarly, for the reverse conversion, it would be interesting to see if a graphotactic model may be of use.

On the CMU dictionary, [12] reports a 62.79% *WACC* in a S2P task including accented phonemes, using a decision tree (ID3) model. This is about the same as our model's performance on the same task (62.6%). However, the authors used a hand-designed set of correspondences which failed to provide alignments for a relatively large part of the dictionary entries; these were removed from dictionary prior to evaluation. The already mentioned CART-based model of [27] achieved a 73.1% *WACC* (using non-accented phonemes) on the top 60,000 most-frequent words from the dictionary, as found in the NAB News corpus. The only other approach that we know of that was tested on the whole data (an earlier, smaller release, though), without removing any of the critical words is the statistical model described in [28]. This model is factored into a phonotactic model and a "matching" model, each of these being a mixture of models encoding different context dependencies. The 57.2% *WACC* obtained with this model is very low compared to the performance of the joint n-gram, even considering the fact that their model was trained on less data. This may be another indication that the joint model is a better alternative .

To our knowledge, no other technique has been applied on either the NetTalk data or the CMU dictionary for P2S conversion. [3] includes experiments in this direction on NetTalk, but the technique described there makes the unrealistic assumption that the null phonemes are present in the input [29].

There are many ways this work can be continued, some of which are under way. One avenue is to explore the use of word frequency in the training of the joint n-grams. In a pilot study on the NetTalk data, we found that using some word frequency information may lead to an up to 1% (absolute) increase in word accuracy. We would like to test this approach on other languages, in order to better asses its language-independence feature and to compare it to systems that have not been tested on the NetTalk and CMU dictionaries. A few other possibilities have already been mentioned above.

We also intend to apply this work in the context of a speech recognition system for recognizing out-of-vocabulary words and for generating pronunciations for specialized domains.

There is ample room for optimizing the structure and the parameters of our models. It appears, though, that the joint n-gram model already has reached the level of performance of the best data-driven approaches reported in the literature.

5. Conclusion

We described the joint grapheme/phoneme n-gram model, a statistical model for bi-directional spelling-to-pronunciation conversion, based on automatically inferred minimal correspondences between graphemes and phonemes. Our method is fully automatic, language-independent, and builds on the successes achieved by n-gram models in other areas of language processing. An initial evaluation showed that this approach compares favorably to other data-driven techniques. We expect to see some improvements over the results presented here by looking for better models in the joint n-gram framework.

6. Acknowledgements

The work reported here was supported by ONR research grant no. N00014-95-1-1088 and DARPA research grant no. F30602-98-2-0133. We acknowledge the active role of Jeff Adams and Paul Vozilla in a preliminary phase leading to this work, while the first author worked as a summer intern for Lernout & Hauspie.

7. References

[1] Coltheart, M., "Writing systems and reading disorders". In L. Henderson (Ed.*), Orthographies and Reading*, pp. 67-79. London, UK: Lawrence Erlbaum, 1984.

[2] Glushko, R. J., "The organization and activation of orthographic knowledge in reading aloud", *Journal of Experimental Psychology: Human Perception and Performance*, 5(4):674-691, 1979.

[3] Marchand, Y., and R. I. Damper, "A multi-strategy approach to improving pronunciation by analogy". *Computational Linguistics*, 26(2):195-219, 2000.

[4] Damper, R.I., Y. Marchand, M.J. Adamson, and K. Gustafson, "Evaluating the pronunciation component of text-to-speech systems for English: A performance comparison of different approaches*". Computer Speech and Language* 13(2):155-176, 1999.

[5] Meng, H., "A hierarchical lexical representation for bi-directional spelling-to-pronunciation/pronunciation-to-spelling generation", *Speech Communication*, 33:213-239, 2001.

[6] Luk R.W.P., and R.I. Damper, "Stochastic phonographic transduction for English*". Computer Speech and Language*, 10:133-153, 1996.

[7] Parfitt, S., and R. Sharman, "A Bi-directional Model for English Pronucnciation". In *Proc. EUROSPEECH'91*, pp. 801-804, 1991.

[8] Dermatas, E., and G. Kokkinakis, "A Language-Independent Probabilistic Model for Automatic Conversion Between Graphemic and Phonemic Transcription of Words", In *Proc. EUROSPEECH'99*, pp. 2071-2074, 1999.

[9] Deligne, S., F. Yvon, and F. Bimbot, "Variable-length sequence matching for phonetic transcription using joint multigrams". In Proc. *EUROSPEECH*, 1995.

[10] Rentzepopoulos, P.A., and G.K. Kokkinakis, "Efficient Multilingual Phoneme-to-Grapheme Conversion Based on HMM". *Computational Linguistics*, 22(3):351-376, 1996.

[11] Damper, R.I., and J.F.G. Eastmond, "Pronunciation by Analogy: Impact of Implementational Choices on Performance". *Language and Speech*, 40(1):1-23, 1997.

[12] Pagel, V., K. Lenzo, and A. Black, "Letter to sound rules for accented lexicon compression". In *Proc. ICSLP'98*, Sydney, Australia., 1998.

[13] Dempster, A.P., N.M. Laird, and D.B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm", *J. Roy. Stat. Soc., Ser. B*, vol. 39, pp. 1-38, 1977.

[14] Galescu, L, J. Adams, P. Vozilla, and J. Allen, "Sub-word Language Modeling based on Graphonemes", URCS Technical Report, University of Rochester, 2001 (forthcoming).

[15] Galescu, L., and E.K. Ringger, "Augmenting Words With Linguistic Information For N-Gram Language Models". In *Proc. EUROSPEECH'99*, 1999.

[16] Viterbi, A.J., "Error Bounds for convolutional codes and an asymptotically optimum decoding algorithm*", IEEE Transactions on Information Theory*, vol. IT-13, pp. 260-269, 1967.

[17] Jelinek, F., "Self-organized language modeling for speech recognition". In A. Waibel and K. Lee, editors, *Readings in speech recognition*, pp. 450-506. Morgan Kaufmann Publishers, Inc., 1990.

[18] Brown, P.F., V.J. Della Pietra, P.V. deSouza, J.C. Lai and R.L. Mercer, "Class-Based n-gram Models of Natural Language", *Computational Linguistics* 18(4):467-479, 1992.

[19] Heeman, P.A. and G. Damnati, "Deriving Phrase-based Language Models". In Proc. IEEE Workshop on ASRU, 1997.

[20] Sejnowski, T.J., "The NetTalk Corpus: Phonetic Transcription of 20,008 English Words", 1988.

[21] Weide, R., "The CMU Pronunciation Dictionary, release 0.6". Carnegie Mellon University, 1998.

[22] Bakiri, G., and T. Dietterich. "Achieving High-Accuracy Text-to-Speech with Machine Learning". In B. Damper (Ed.), *Data mining in speech synthesis*, Chapman and Hall, 1999.

[23] Clarkson, P., and R. Rosenfeld, "Statistical Language Modelling using the CMU-Cambridge Toolkit". In *Proc. EUROSPEECH'97*, pp. 2707-2710, 1997.

[24] Stolcke, A., "SRILM - The SRI Language Modeling Toolkit", SRI International, 2000.

[25] Ries, K., F.D. Buo, and A. Waibel, "Class Phrase Models for Language Modeling". In *Proc. ICSLP'96*, 1996.

[26] Yvon, F., "Grapheme-to-phoneme conversion using multiple unbounded overlapping chunks". In Proc. NeMLaP'96, pp 218-228, 1996.

[27] Jiang, L., H.-W. Hon and X.D. Huang, "Improvements on a trainable letter-to-sound converter". In Proc. Eurospeech'97, pp. 605-608, 1997.

[28] Besling, S., "A statistical approach to multilingual phonetic transcription". *Philips Journal of Research*, 49:367-379, 1995.

[29] Damper, R., Personal communication, 2001.