Error analysis of a public domain pronunciation dictionary

Olga Martirosian and Marelie Davel

Human Language Technologies Research Group
CSIR Meraka Institute / North-West University
omartirosian@csir.co.za, mdavel@csir.co.za

Abstract

We explore pattern recognition techniques for verifying the correctness of a pronunciation lexicon, focusing on techniques that require limited human interaction. We evaluate the British English Example Pronunciation (BEEP) dictionary [1], a popular public domain resource that is widely used in English speech processing systems. The techniques being investigated are applied to the lexicon and the results of each step are illustrated using sample entries. We find that as many as 5553 words in the BEEP dictionary are incorrect. We demonstrate the effect of correction techniques on a lexicon and implement the lexicon in an automatic speech recognition (ASR) system.

1. Introduction

Strik and Cucchiarini [2] warn that when constructing an automatic speech recognition (ASR) system to be used as a baseline when researching improvement techniques, one must keep in mind that the data used to build the system may contain errors. If these errors are not corrected in the baseline system but are found and corrected in the process of using the system for research, the results from the improvement technique may be overestimated. It is important to validate the baseline system prior to further experimentation, in order to be confident that the method that has been developed for the purpose of improving an ASR system is causing, at the very least, the majority of the improvement observed.

Pronunciation lexica are used to train speech technology systems by describing the pronunciation of words according to manageable units, typically phonemes. These lexica provide the mapping from the orthographic (written) form of a word to its pronunciation, which is useful in both text to speech (TTS) and automatic speech recognition (ASR) systems. A pronunciation lexicon is also used to generate generalised grapheme to phoneme rules, for the purposes of providing pronunciations for words that do not appear in the lexicon.

Because the pronunciation lexica are so fundamental to speech technology systems, much care must be taken to select a lexicon that is as free of errors as possible. For ASR systems, incorrect pronunciations in the lexicon may lead to the incorrect training of the system and consequently to a system that does not function to its full potential. For rule extraction algorithms the correctness of the lexicon is equally important, as each erroneous entry can cause an incorrect grapheme to phoneme rule to be generated, thereby compromising the pronunciation prediction accuracy of the set of rules.

The development of a pronunciation lexicon tends to focus on word coverage. The inclusion of entries with erroneous spelling allows a speech recognition system to learn the phonemes of a word without the need for correction of spelling errors in speech transcriptions. Therefore, entries with erroneous spelling are often included in the lexica to assist with the convenience of building speech technology systems. However, the decision of how erroneous the spelling of a word needs to be in order to be included in the lexicon is not left up to each individual user, but rather the larger group of users. Due to this, a single researcher working on a specific task in speech technology may need to filter the lexicon that is designed to be practical for a larger group of researchers in order to make it suitable for their field of research.

Because pronunciation lexica are often compiled from many sources and because automatic means of lexicon extension are sometimes used, the entries in the lexicon can become flawed. In large lexica, although a high percentage of the entries are correct, the incorrect entries can detrimentally influence a speech technology system that is developed using the lexicon. If one would like to implement the lexicon to its full potential, the removal of the erroneous entries is required.

This study focuses on the implementation of mechanisms to identify incorrect entries in a lexicon that require limited human intervention. However, the automated correction of these entries is not yet investigated and erroneous entries are simply removed from the lexicon. Section 2 provides the general background of the pattern recognition techniques that were implemented to gain more information about the lexicon. Section 3 describes the techniques used in the context of analysis of a dictionary. Section 4 provides a description of the lexicon selected for this study as well as an outline of the process followed in the experiments. Section 5 provides the results of each technique that was implemented and provides samples of entries that are filtered out using that technique. Section 5 also describes the ASR system that was developed for the purposes of gauging the improvement that the filtering provides. Section 6 concludes with a description of potential further work.

2. Background

Our dictionary analysis approach builds on published techniques related to (1) grapheme to phoneme (G2P) alignment, (2) grapheme to phoneme rule extraction and (3) variant modelling.

2.1. Grapheme to phoneme alignment

Many grapheme to phoneme rule extraction algorithms first require that grapheme to phoneme alignment be performed. Each word in the training dictionary is aligned with its pronunciation on a per-grapheme basis, as illustrated in Table 1 where ϕ indicates a null (or empty) grapheme or phoneme. The alignment process involves the insertion of graphemic and phonemic nulls into the lexical entries of words. A graphemic null is inserted

Table 1: *Grapheme to phoneme alignment example*

R O S E	→	/ R OW Z ϕ /
R O W S	→	/ R OW ϕ Z /
R O O T	→	/ R UH ϕ T /
M A X ϕ	→	/ M AE K S /

when more than a single phoneme is required to pronounce a single grapheme. A phonemic null is inserted when a single phoneme is required to realise more than one grapheme.

Viterbi alignment [3] is typically used to obtain these mappings, where the alignment algorithm makes use of the probability of each grapheme being mapped to a particular phoneme. We use the alignment technique described in more detail in [4]: Initial probabilities are calculated by selecting the entries in a dictionary that have the same phonemic and orthographic lengths. Once these probabilities are calculated, iterative forced Viterbi alignment is performed on the lexicon. Graphemic null generator pairs are extracted in order to be able to insert graphemic nulls while predicting unknown words.

2.2. Grapheme to phoneme rule extraction

Various automatic rule extraction techniques exist, including decision trees ([5]), pronunciation-by-analogy models ([6]), Dynamically Expanding Context (DEC) ([7]) and IB1-IG, a *k*-nearest neighbour classifier ([8]).

In our analysis we utilise the *Default&Refine* algorithm for the extraction of grapheme to phoneme rules [9]. This algorithm makes use of two observations: Graphemes are usually realised as one phoneme more often than all others, and that graphemes have different realisations as phonemes based on their context in a word. The algorithm extracts G2P rules for each grapheme independently. The following process is applied: All the realisations of a grapheme are considered and the rule that correctly predicts most of the realisations is selected as the default rule. The rule containing the smallest possible context that correctly predicts most of the left over occurrences of a grapheme is now selected. This process is applied iteratively until all realisations of a grapheme are correctly predicted. During prediction, a grapheme's context is tested against rules, starting from the rule with the largest context, until a match is found. The final rule does not have a context and therefore matches every context in which the grapheme can occur.

2.3. Variant modelling

Most of the G2P rule extraction mechanisms mentioned above can only train on words having single pronunciations (rather than more than one pronunciation for a single word). Pseudo-phonemes and generation restriction rules have been developed as a way to model varying pronunciations of words as a single pronunciation [10]. Pseudo-phonemes are used to represent two or more phonemes which can appear in a certain place in the pronunciation of a word. When two or more pseudo-phonemes appear in a word, generation restriction rules are applied to limit the combinations of phonemes that can be generated from the set of pseudo-phonemes. The rules ensure that if the pseudo-phonemes are removed and the lexicon is expanded, nothing will have been added or removed from its original form.

3. Approach

There are two ways in which a lexicon can be verified: direct observation and indirect analysis. Direct observation of a lexicon is the analysis of a lexicon through direct observation of its behaviour. This process involves measuring the lengths of the orthographic and phonemic representations, looking at different words that have duplicate pronunciations and the examination of the lexicon for distinguishable errors in both the orthographic and the phonemic transcriptions. Indirect analysis requires the implementation of techniques to transform the lexicon into different formats, each of which allows different errors to become more distinguishable. Indirect analysis techniques include the alignment of the lexicon, extraction of grapheme to phoneme rules and the implementation of pseudo-phonemes along with generation restriction rules.

A number of methods have been implemented in an attempt to isolate the incorrect entries in a lexicon. Each general method is explained below along with the ways in which it was applied in order to implement verification on the lexicon.

3.1. Word pronunciation length relationships

The relationship between a word's orthographic and phonemic representation can be an indicator of whether a word's spelling or pronunciation is wrong. The extraction of words whose orthographic and phonemic transcriptions differ above a certain threshold can allow one to obtain a manageable list of possible erroneous entries from a lexicon.

3.2. Alignment analysis

The alignment of a word to its pronunciation gives one further insight into the length relationship of a word and its pronunciation, and in addition identifies words which do not match their pronunciation. During alignment, graphemic and phonetic nulls are inserted in order to align every grapheme to a phoneme. Potential errors can be flagged at this stage through the analysis of the placement and number of nulls inserted into both the orthographic and phonemic representations of a word.

3.3. Grapheme to phoneme rules

Grapheme to phoneme (G2P) rules are extracted for one grapheme at a time and are sorted such that the number of occurrences that gave rise to any one of the rules is easily obtainable. By inspecting the rules that are generated by the smallest number of occurrences, one can gain insight into potential errors because outlying pronunciations would be flagged. This analysis does assume a certain level of accuracy in the lexicon, as with a high error rate most pronunciations would be erratic. However, it is not dependent on phoneme or grapheme ubiquity as the least likely pronunciation is selected regardless of number of total occurences.

3.4. Duplicate Pronunciations

Words that have the same pronunciation as other words usually have similar orthographic length. For example, the words CAUSE, CAWS, CORES and CORPS have the same pronunciation and their spelling consists of four to five letters. One way to isolate problematic entries is to search for words that have the same pronunciation and to compare their orthographic lengths.

3.5. Variant analysis

The generation restriction rules that accompany words which contain more than one pseudo-phoneme can allow one to flag possibly incorrect entries in the lexicon. When pronunciation variants do occur in a lexicon, they usually differ by one or two phonemes. If restriction rules are being generated for more than three sounds, it can mean one of three things: (1) The entries are correct and the word truly does allow for vastly different pronunciations, (2) the alignment of the word has not aligned graphemes to the correct phonemes, or (3) that some of the variants are incorrect. Once a list of generation restriction rules is obtained, the list of multiple pseudo-phonemes occurring in words is short enough to be evaluated manually.

4. Experimental Setup

4.1. Dictionary

The BEEP dictionary [1] has been selected for the evaluation of this study. It is a freely available online English pronunciation dictionary that is comparable with other available online lexicons with regard to its size and content [11]. It was compiled through the amalgamation of several public domain lexicons and has not undergone a strict quality control process.

4.2. Process

A series of steps is followed for the verification of the BEEP dictionary.

4.2.1. Pre-processing

For pre-processing, unusual punctuation patterns are removed. These are removed temporarily, as the entries are not erroneous, but make dictionary analysis difficult.

4.2.2. Removal of systematic errors

Through inspecting the result of an initial alignment of the dictionary, a list of systematic errors was compiled, specifically with regard to repeated phonemes. It was found that in words where a letter was repeated, the phonemic representation of which was usually repeated as well, even where such repetition does not occur. This phenomenon was found to occur frequently in the lexicon but could not be explained by naturally occurring phenomena in speech.

4.2.3. Spelling verification

In an attempt to verify the spelling used for words in the lexicon, a word list was extracted and the spelling checked automatically. However, the list of incorrect spelling contained over 146 000 words, and after a general manual inspection was found to be invalid and discarded. Checking the spelling of the BEEP dictionary may be beneficial, however, the program that would perform the checking would require a more comprehensive coverage of English words.

4.2.4. Lengthened pronunciations

One of the methods that can be used to isolate errors in the lexicon is checking for which entries the phonemic representation of the word is longer than the orthographic representation. In order to make this method function correctly, the list generated has to be refined by identifying where graphemic nulls should be inserted and taking that into account.

4.2.5. Graphemic null analysis

The graphemic nulls identified in Section 4.2.4 were investigated further. The list consisted of sequences including the letter 'X' always needing a graphemic null and the letter 'U' needing a graphemic null in certain situations (such as the word ACUTE having the pronunciation */ AX K Y UW T /*). However, the graphemic nulls that were in the list were not always applicable, and with manual verification the phonemes */ OW AX /* were found to be invalid in situations where a word didn't contain the letter sequence 'ower' (an example of valid use of the phonemes being the word BESTOWER with the pronunciation */ B IH S T OW AX R /*).

4.2.6. Lengthened spelling

The lexicon was then analysed to isolate the orthographic representations of words that were more than a selected threshold longer than their phonemic representations. These words would require investigation as lengthened spelling may indicate an error.

4.2.7. Duplicate pronunciations

For the purpose of this test, the lexicon was traversed, specifically looking for words that had the same phonemic representation but orthographic representations varying in length.

4.2.8. Alignment

Alignment looks for probabilities of graphemes being realised as certain phonemes, and aligns them accordingly. It can thus be a strong source of information in the search for incorrect entries in the lexicon. For the purpose of flagging incorrect entries, two methods were attempted: listing entries with a high total number of nulls and listing entries with a high number of consecutive nulls.

4.2.9. Pseudo-phonemes

Generation restriction rules of pseudo-phonemes are inspected as described in Section 3.5. Variants that required more than three pseudo-phonemes were investigated with the expectation that one or more of the variant pronunciations would be incorrect.

4.2.10. Grapheme to phoneme rules

The G2P rules are implemented using the Default&Refine algorithm. This algorithm allows one to see how many instances of a grapheme each single rule is extracted from. By selecting rules that are extracted from single instances, entries with anomalous pronunciations can be isolated. Rules were extracted from the BEEP dictionary, and rules that were extracted from single instances of a grapheme were extracted. These rules were used to find the instances which gave rise to them.

5. Experimental Results

5.1. Dictionary analysis

The summary of how many entries were removed by each lexical verification technique can be found in Table 2. The table also indicates whether a verification required is automated (requiring no human intervention) or semi-automated (requiring validation of the list of possible errors by a human). Where validation is required, the size of the list requiring validation is

also reported. All of the steps listed in this section were implemented in sequence. With the exception of pre-processing the sequence of implementation was not considered significant. Thus, the erroneous entries found in one step may have been identified in a later steps, but were removed before its execution.

5.1.1. Pre-processing

Unusual punctuation removal involved the removal of punctuation which does not occur in general English writing. This process also removed many acronyms from the dictionary. Examples of removed words are: VICU ∼ NA and W._R._A._C..

5.1.2. Removal of systematic errors

Entries whose pronunciations contained the same phoneme successively were investigated. 5711 instances were originally identified, but minor inspection revealed that some repeated phonemes were legitimate (such as the transcription for ACCOMPANYING being */ AX K AH M P AX N IH IH NG /*), and those entries were left in the lexicon. In total 4730 entries were removed from the lexicon. Examples of removed entries are: ADMITTER, which was transcribed as */ AX D M IH T T ER /*, and CHIPPIE, which was transcribed as */ CH IH P P AY /*. The separate counts of each of the occurrences removed can be found in Table 3.

Table 3: *Table showing number of lexical entries taken out due to repeated phonemes*

Double Phoneme	Number Removed
AX AX	959
T T	942
N N	586
L L	479
P P	391
D D	275
S S	246
M M	199
K K	182
R R	178
B B	156
G G	56
EY EY	23
F F	14
IY IY	11
SH SH	9
CH CH	8
AA AA	7
OW OW	6
Z Z	3

5.1.3. Spelling verification

No words were removed using spelling verification as a dictionary containing enough English words to allow it to accurately assess the spelling in the BEEP dictionary was not found.

5.1.4. Lengthened pronunciations

For this test, entries whose phonemic representations that were longer than their orthographic representations were identified and investigated for errors. This function yielded a list of 1284 entries. The list was found to contain many proper noun entries, some of whose pronunciations were suspicious but could not be categorised as incorrect. The list was manually filtered down to 253 entries that were removed form the lexicon. Examples of entries removed include the word APRICATION having the pronunciation */ EY P R IH V AE R IH K EY SH N /*, and the word EFFECTIVITY having the pronunciation */ IH F EH K T AX B IH L IH T IY /*.

5.1.5. Graphemic null analysis

Once erroneous phonemic sequences were identified in the graphemic null list, the entries whose pronunciations contained the phonemic sequence were written to a file. This list contained 362 entries, but was manually filtered to 189. Examples of the removed entries are the word DELEGATOR having the pronunciation */ D EH L IH G AA T OW AX /* and the word VENTOR having the pronunciation */ V EH N T OW AX /*.

5.1.6. Lengthened spelling

For this test, words whose orthographic length differed from their phonemic length by more than a threshold value were investigated. The threshold value was tested iteratively. A threshold of four yielded a list of 1366 words, which was judged to contain too many correct entries. A threshold of six yielded a list that contained less than 50 entries. Therefore, orthographic representations that were a threshold of five characters longer than their pronunciation were flagged as possibly erroneous. A list of 209 entries was extracted , which was analysed manually and filtered down to a list of 69 entries that were removed from the lexicon. Examples of the words removed form the lexicon are the word PRESENTIMENTAL having the pronunciation */ P R IH Z E N T L /* and the word SEMITRANSPARENT having the pronunciation */ S EH M IH T R AX N T/*.

5.1.7. Duplicate pronunciations

For this test, words whose pronunciations were identical were analysed by comparing their orthographic length and extracting ones that differed by more than a set threshold. The algorithm that was implemented for this experiment calculated the mean length of all the orthographic representations and worked out by how much the length of each of the orthographic representations differed from the mean. The threshold for this value was iteratively tested, and the most applicable value was found to be 1.5. A value of two yielded less than 50 entries, and a value of one yielded too many entries to be manually verified. The list of duplicated pronunciations contained 305 sets of words. A set of words would contain between two and four words with identical pronunciations. This list was manually analysed and a list of 95 erroneous entries was extracted that were removed from the dictionary. Examples of words that contained pronunciations for other words include the word NONRESPONDENT having the pronunciation */ N OH N R EH Z IH D AX N T /* and the word DISTINGUISED having the pronunciation */ D IH S G AY Z D /*.

5.1.8. Alignment

A list of entries with a high number of total nulls inserted by alignment was extracted using different thresholds of how many nulls an entry needed to contain in order to be added to the list. Setting the threshold to four nulls yielded a list of over 10 000 entries, a sample of which was verified as mostly correct con-

Table 2: *Table illustrating verification process*

Verification applied	Verification type	# listed possible errors	# removed	% possible errors verified	# entries remaining
None	N/A	0	0	0%	257 059
Punctuation Removed	Automated	576	576	100%	256 483
Repeated Phonemes	Automated	4730	4730	100%	251 753
Lengthened Pronunciations	Semi-automated	1284	253	19.7%	251 500
Incorrect Graphemic Nulls	Semi-automated	362	189	52.2%	251 311
Lengthened Spelling	Semi-automated	209	69	33%	251 242
Duplicate Pronunciations	Semi-automated	$\simeq 305$	80	$\simeq 26.22\%$	251 162
Alignment Errors	Semi-automated	9	9	100%	251 153
Consecutive Phonemic Nulls	Semi-automated	204	84	41.18%	251 069
Singular G2P Rules	Semi-automated	1450	89	$\simeq 6\%$	250 980
Generation Restriction	Semi-automated	$\simeq 90$	50	$\simeq 55.56\%$	250 930
Punctuation Replaced	Automated	-576	-576	100%	251 506
Total		9219	5553	60.23%	251 506

tent. The threshold then was steadily increased to 7 nulls. This threshold yielded a list of 82 entries, however, after verification this list was discarded because all incorrect entries listed in it would be removed by checking entries for successive nulls as was done in the following experiment.

Listing the entries with a high number of successive nulls inserted by alignment gives one insight into where the alignment algorithm experienced difficulty in aligning a grapheme to the correct phoneme. The best threshold for the number of successive nulls was investigated to yield a list of possibly erroneous entries that was short enough for manual verification. Initially set to three nulls, a list of over 3000 entries was generated. Through verification this list was found to contain too many correct entries and thus discarded. The threshold was then set to four successive nulls, and a list of 204 entries was generated, and filtered down to 71 entries through verification. The list of incorrect entries was then removed from the lexicon. Examples of the removed entries are the word ANTISEPTICISM being aligned to the pronunciation */ AE N T IH S 0 0 0 0 IH Z AZ M /* and the word SUBMERSIBILITY being aligned to the pronunciation */ S AX 0 0 0 0 0 0 B IH L IH T IY /*.

5.1.9. Pseudo-phonemes

A list of 90 generation restriction rules containing more than three pseudo-phonemes was generated. A list of 49 entries was extracted from these manually and removed from the lexicon. Examples of the removed entries include the word INAPPRECIABLE having the pronunciation */ IH N AX P R OW P R IA T /* and the word UNATTACHED having the pronunciation */ AH N AX T EH N D IH D /*. This method was found to have the most accurate prediction of incorrect entries due to its manual verification percentage being 55.56%.

5.1.10. Grapheme to Phoneme Rules

Grapheme to phoneme rule extraction was implemented and the rules extracted were analysed. The last 50 rules for each grapheme were analysed, where the set of graphemes included three punctuation marks. This process yielded a list of 1450 entries. This list was verified manually and finally, a list of 52 entries was removed from the lexicon. Examples of entries found include the word HYDROPOLITICS having the pronunciation */ H AY D R AX P OH N IH K S /* and the word UNPRECIPITATED having the pronunciation */ AH N P R EH S IH D EH N T IH D /*.

5.2. Effectiveness of error analysis

Error analysis was performed in order to determine the effectiveness of implementing the above techniques on the BEEP dictionary. 200 entries were randomly selected from the final and initial lexica and analysed independently by two researchers. The goal of the exercise was to obtain an estimate of the number of incorrect entries in both, however, more information is required to conclusively categorise entries as either correct or not. It is not a simple task to determine whether a word or pronunciation is correct or not. Some esoteric words were not known to the lexical verifiers and were not included in any word list consulted. Proper nouns included in the lexicon were exceedingly difficult to evaluate because some seemingly incorrect proper nouns may actually be correct. Thus the category of incorrect was expanded to three categories: Conclusive, Proper noun and Questionable.

For the unfiltered BEEP dictionary, 32 entries were selected as being erroneous. The Conclusive category contained 14 entries, including the word BUMPTY with the pronunciation */ B AH M P IY T IY W AY /*. The Proper noun category contained 10 entries, including the word BUZZY'S, with the pronunciation *B AH Z W AY Z*. The Questionable category contained 7 words, including the word CHADLIN, with the pronunciation */ CH AE D L I N /*. In total 16% of the initial lexicon was found to be incorrect.

For the filtered lexicon, 19 entries were selected as being erroneous. The Conclusive category contained 5 entries, including the word TOURNANT'S having the pronunciation */ T AO N AX M AX N T S /*. The Proper noun category contained 7 entries, including the word MESSA'S, having the pronunciation */ M EH S EY Z /*. The Questionable category contained 6 words, including the word RESCURE, having the pronunciation */ R EH S K Y UA /*. In total, 9.5% of the filtered lexicon was found to be incorrect.

5.3. Implications for ASR

An ASR system was implemented to test the functionality of the G2P verification process. The system was implemented using the toolkit HTK [12]. It makes used of 39 normalised Mel Frequency Cepstral Coefficients (MFCCs), which includes delta and acceleration coefficients. It makes use of 3 state Hid-

den Markov Models (HMMs) to model triphones, using a 17 part Gaussian mixture to model observation probabilities. The acoustic data, whose duration is over nine hours long, is telephone speech data from South African call centres. The testing process implements ten fold cross-validation.

For possible improvement of the ASR system, a lexicon built using BEEP but containing only 1511 entries that appear in the data was verified. To illustrate the reduction in human interaction, the G2P rule extraction technique described in Section 5.1.10 was implemented to isolate entries for manual attention. The focus of this lexical verification was the removal of erroneous pronunciation variants and the correction of incorrect pronunciations. 498 entries were flagged (a third of the total lexicon), of which 33 entries were removed and 3 entries were corrected.

Without lexical verification, the accuracy of the ASR with an n-word recognition vocabulary and a flat language model (no statistical language model was used) was calculated to be 51.53% at word level. The accuracy did not increase significantly with verification (the exact accuracy increased 0.02%), even though 2.18% of the lexicon was removed.

6. Conclusion

This study focused on identifying algorithms to cater for semi-automated lexical verification. Several methods were implemented and their effectiveness analysed. We found that the techniques that identified the most errors were:

- Searching for repeated phonemes and removing entries whose pronunciations contain incorrect repetitions. 4730 entries were removed using this method. However, this method is quite lexicon specific and may not generalise well to other lexica.
- Pronunciations that were longer than their orthographic representation provided a good source of incorrect entries. 253 entries were identified and removed. This method can be applicable to other lexica in English but may, however, be language specific and not perform as well with lexica in other languages.
- Identifying erroneous graphemic nulls found many incorrect entries. 189 entries were identified and removed from the lexicon using this technique. The analysis of graphemic nulls may generalise well to other lexica, however, the specific nulls that were identified may not.

In addition, the most efficient techniques (identifying the largest number of verified incorrect entries as a percentage of the word list requiring manual verification) were found to be:

- The analysis of the generation restriction rules that accompany the implementation of pseudo-phonemes, concentrating on groups of more than three pseudo-phonemes, was very efficient at identifying truly erroneous entries in the lexicon. This method achieved 55.56% accuracy with its list of potential errors. This method is likely to generalise well to other lexica, but is only efficient when one is looking for incorrect pronunciation variants.
- Identifying erroneous graphemic nulls was very efficient as well, achieving a 52.2% accuracy with its predictions.
- Searching for number of consecutive nulls in pronunciations after alignment is performed on a dictionary is also an efficient technique at finding errors in the lexicon. This technique was 41.18% successful in its prediction of incorrect entries.

In total, 5 553 words were removed from the BEEP dictionary. This result was unexpected, as BEEP is a popular dictionary, frequently utilised in a variety of speech technology applications.

Further work includes making use of error analyses of our ASR to determine whether additional consistency checks can be implemented. Once a reliable dictionary has been obtained, we would like to use it as the platform for an analysis of pronunciation variance in South African English.

7. References

[1] BEEP, "The british english example pronunciation (beep) dictionary," Retrieved Jan 2007, from `http://svr-www.eng.cam.ac.uk/comp.speech/Section1/Lexical/beep.html`.

[2] H. Strik and C. Cucchiarini, "Modeling pronunciation variation for asr: A survey of the literature," *Speech Communication*, vol. 29, pp. 225–246, 1999.

[3] A.J. Viterbi, "Error bounds for convolutional codes and a asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, pp. 260–269, 1967.

[4] M. Davel and E. Barnard, "The efficient creation of pronunciation dictionaries: machine learning factors in bootstrapping," in *Proceedings of Interspeech*, Jeju, Korea, 2004, pp. 2781–2784.

[5] A. Black, K. Lenzo, and V. Pagel, "Issues in building general letter to sound rules," in *3rd ESCA Workshop on Speech Synthesis*, Jenolan Caves, Australia, November 1998, pp. 77–80.

[6] Y. Marchard and R.I. Damper, "A multistrategy approach to improving pronunciation by analogy," *Computational Linguistics*, vol. 26, pp. 195–219, 2000.

[7] K. Torkkola, "An efficient way to learn English grapheme-to-phoneme rules automatically," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, Minneapolis, USA, April 1993, vol. 2, pp. 199–202.

[8] W. Daelemans, A. van den Bosch, and J. Zavrel, "Forgetting exceptions is harmful in language learning," *Machine Learning*, vol. 34, no. 1-3, pp. 11–41, 1999.

[9] M. Davel and E. Barnard, "A default-and-refinement approach to pronunciation prediction," in *Proceedings of the Symposium of the Pattern Recognition Association of South Africa*, Grabouw, South Africa, 2004, pp. 119–123.

[10] M. Davel and E. Barnard, "Developing consistent pronunciation models for phonemic variants," in *Proceedings of Interspeech*, Pittsburgh, Pensylvania, 2006.

[11] R.I. Damper, Y. Marchard, J.D.S. Marsters, and A.I. Bazin, "Aligning text and phonemes for speech technology applications using an em-like algorithm," *International Journal of Speech Technology*, vol. 8, pp. 147–160, 2005.

[12] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland, "The htk book. revised for htk version 3.0," July 2000, retrieved from `http://htk.eng.cam.ac.uk/`.