Syllable-based Phonetic transcription by Maximum Likelihood Methods

R.A.Sharman

MP167, IBM(UK) Labs Ltd, Hursley Park, Winchester SO21 2JN, UK

Introduction

The transcription of orthographic words into phonetic symbols is one the principal steps of a text-to-speech system[1]. In such a system a suitable phonetic pronunciation must be supplied, without human intervention, for every word in the text. No dictionary, however large, will contain all words, let alone proper names, technical terms and other textual items commonly found in unrestricted texts. Consequently, an automatic transcription components is usually considered essential.

Hand-written rule sets, defining the transcription of a letter in its context to some sound, view the process as that of parsing with a context-sensitive grammar. This approach to transcription has been challenged more recently by a variety of methods such as Neural nets[2], Perceptrons[3], Markov Models[4], and Decision Trees[5]. Some approaches have used additional information such as prefixes and suffixes[8], syllable boundaries[3], sometimes combined with the use of parts-of-speech to assist in the disambiguation of multiple pronunciations. In the phonetic transcription of proper names special techniques can be employed to improve accuracy[9] such as detecting the language of origin of the name and using different spelling to sound rules. Each method has its own advantages and disadvantages in terms of computational speed, complexity and cost. However, none of these methods by itself is completely adequate.

The present method uses the two-step conversion process described elsewhere[1,3] in which the structure of the word plays a central role. First the orthographic word is divided into its syllables, and secondly the syllable sequence is converted to a phonetic string. This not only accords with linguistic intuition, but it also allows the two processes to be handled by different techniques, choosing the technique most suited to each step. The question of whether a *morphological* or *syllabic* decomposition of the word might produce better results is not further analysed here. (For the present study data was available for syllables and not for morphs, so in the sense the comparison could not be carried out by the techniques proposed). The effects of other factors, such as part-of-speech tagging, domain-dependent information, and other information sources, were ignored, although these could be useful in practical systems.

The technique proposed for syllabification is based on the principle of Hidden Markov Modelling, well known in speech recognition[7]. This presupposes the existence of some training material containing words in both their orthographic and syllabic form. Using this data a model of syllable sequences can be designed and trained to identify syllable boundaries. Once the most likely syllable division of the word has been found the phonetic transcription can be produced by a variety of direct transcription methods, such as the one used here based on Decision Trees[5]. The training of such a method presupposes the existence of some training data containing words in both their syllabic and phonetic forms. Using the latter data a Decision Tree can be trained to transcribe syllables in context into phone sequences. The advantage of using decision trees is that they not only learn general rules, but also capture idiosyncratic special cases automatically. The resulting process should perform transcription with high accuracy.

Such a two-stage approach has been shown to yield improvements[3] but only where perfect syllabification information is available, consequently a reliable syllabification technique is required. The remainder of this paper

discusses only the syllabification process in detail, since the decision tree methodology is well described elsewhere[5], whereas the syllabification algorithm proposed is novel. An experiment using a very large set of word-syllable-pronunciation strings was used to train the two models, and then tests performed to determine the accuracy of the resulting transcription.

A Maximum Likelihood Model of syllabification

The purpose of this step is to make explicit the hidden syllable boundaries in the observed words. These often, but not always coincide with the *morphological* boundaries of the constituent parts of each word. However, so as not to confuse the question of the derivation of a word from its roots, prefixes and suffixes, with the question of the pronunciation of the word in small discrete sections of vowels and consonants, the term *morphology* is not used here. Strictly speaking the term *syllable* might be more accurately applied only after transcription to phonemes. However, we shall use it here to apply to such pronunciation units described orthographically. The purpose of such analysis is to obtain information which will be used by the phonetic transcription stage to make better judgements on the pronunciation of consonant and vowel clusters in particular.

For example, the consonant cluster *ph* in the word *loophole* might be pronounced /f/ by analogy with the same cluster in the word *telephone.* However, it might also be pronounced as /ph/ by analogy with the same cluster in the word *tophat.* The deciding factor is where the syllable boundary lies in the word. The most plausible structure for the word *telephone* is *tele+phone,* or possibly, *te+le+phone,* and for *tophat* is *top+hat.* So a possible syllable structure for the word *loophole* might be *loop+hole*, or alternatively *loo+phole,* or maybe *looph+ole.* The syllable model needs to determine what the true, but unobserved, syllable sequence is, given only the observed evidence of the orthographic characters. This can be modelled as a decoding problem in which a hidden sequence of states (syllables) gives rise to an observed sequence of symbols (letters). We need to discover the underlying sequence of states which gave rise to the observations. The complexity arises since the states and observations do not align in a simple way[11]. Syllable models of a similar type have been proposed for prosody[12] but not for transcription, whereas direct models of transcription have been attempted[4].

Let a orthographic word, W, be defined as a sequence of letters, $w_1, w_2, \ldots, w_n$. Let a syllabic word, S, be defined as a sequence of syllables, $s_1, s_2, \ldots, s_m$. The observed letter sequence, W, then arises from some hidden sequence of syllables, S, with conditional probability $P(W|S)$. There are a finite number of such syllable sequences, of which the one given by $\max P(W|S)$ where the maximisation is taken over all possible syllable sequences, is the maximum likelihood solution, and intuitively, the most plausible analysis. By the well-know Bayes theorem, it is possible to rewrite this expression as:

$$\max_S [P(W|S)] = \max_S \left[\frac{P(S|W)P(S)}{P(W)} \right]$$

In this equation it is interesting to interpret the $P(S|W)$ as a probability distribution capturing the facts of syllable division, while the $P(S)$ is a different distribution capturing the facts of syllable sequences. The latter model thus contains information such as which syllables form prefixes and suffixes, while the former captures some of the facts of word construction in the usage of the language. Note that the term $P(W)$, which models the sequence of letters, is not required in the maximisation process, since it is not a function of S. Given the existence of these two distributions there is, in principle, a well-understood method of estimating the parameters, and performing the decoding[7]. The estimation is provably capable of finding a local optimum[13], and is thus dependent on finding good initial conditions to train from. In this application the initial conditions are provided by supervised training data obtained from a dictionary.

A variety of expansions of the terms $P(S|W)$ and $P(S)$ can be derived, depending on the computational cost which is acceptable, and the amount of training data available. There is thus a family of models of increasing complexity which can be used in a methodical way to obtain better modelling, and thus more accurate processing.

The function $P(S|W)$ can be simply modelled as

$$P(S|W) = \prod_{i=1}^{m} \left(s_i | w_j, .. w_k \right)$$

which has the value 0 everywhere, except when $s_i = w_j, \ldots, w_k$ for any j,k, when it has the value 1. This simply says that each syllable is spelled the same way as the letters which compose it. This points the way to a more sophisticated model of syllabification which incorporates spelling changes at syllable boundaries, but this will not be attempted here. Another application of the approach might be in a model of inflexional or derivational morphology where spelling changes are observed at morph boundaries.

The function $P(S)$ can be modelled most simply as a bi-gram distribution, where the approximation is made that:

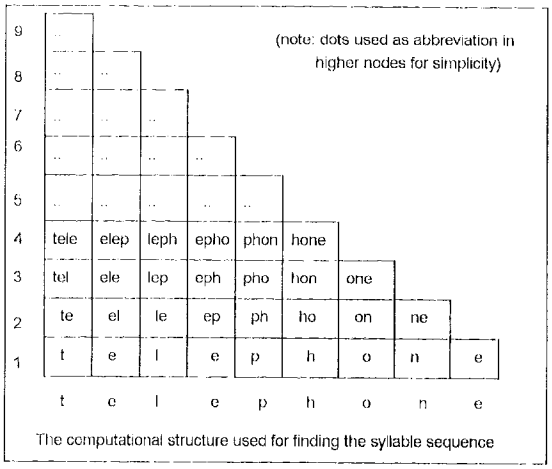
$$P(S) = \prod_{i=1}^{m} P(s_i | s_1, \ldots, s_{i-1}) \approx P(s_1) \prod_{i=2}^{m} P(s_i | s_{i-1})$$

Such a simple model can capture many interesting effects of syllable placements adjacent to other syllables, and adjacent to boundaries. However, it would not be expected that subtle effects of syllabification due to longer range effects, if they exist, could be captured this way.

An efficient computational scheme for syllabification

One complication exists before either the Viterbi decoding algorithm[7] for determining the desired syllable sequence, or the Forward-Backward parameter estimation algorithm[7] can be used. This is due to the combinatorial explosion of state sequences due to the fact that potential syllables may overlap the same letter sequences, as shown in the example above with the word *telephone*. This leads to the decoding and training algorithms becoming $O(n^3)$ rather than $O(n^2)$ in computational complexity, as usual for this type of problem. The difficulty can be overcome by the use of a technique from context-free parsing[14], namely the use of a substring table. The method will be briefly described.

A word of length, n, can contain $n^2/2$ substrings, any of which may potentially be syllables of the word. Using the method of tabular layout familiar from the *Cocke-Kasami-Younger* (CKY) parsing algorithm, these substrings can be conveniently represented as a triangular table, $T_n = \{t_{i,j}\}$ (see diagram below). Where the table contains a non-zero element the index number of a unique syllable can be found. The first step in parsing the word is to generate all the possible substrings and check them against a table of possible syllables. Even for long words with 20 or 30 letters, this is not a prohibitive calculation. If the letter string is identified as a possible syllable then the unique identifying number of the syllable can be entered into the table.



The computational structure used for finding the syllable sequence

The bigram sequence model can now be calculated by the following algorithm, which is an adaptation of the familiar CKY algorithm:

for each letter w[i], i=1,...,n
  for each starting syllable position
    t[i,j],j=1,...,n+1-i
    for each ending syllable position
      t[i+j-1,k],k=1,...,n-i-j
      let x=t[i,j] and y=t[i+j-1]

compute P(s(y)|s(x))

In this way it is possible to calculate all the possible syllable sequences which apply to the given word without being overwhelmed by a search for all possible syllable sequences.

A methodology for constructing a syllabifier

The following methodology can be used to build a practical implementation of the technique outlined above:

1. Collect a list of possible syllables.
2. From the observed data of orthographic-syllabic word pairs, construct an initial estimate of $P(M) = \prod P(m_i|m_{i-1})$. This is the bi-gram model of syllable sequences.
3. Using another list of words, not present in the initial training data, use the *Forward-Backward* algorithm to improve the estimates of the bi-gram model. (This step is optional if the original data is sufficiently large, since the hand annotated text may be superior to the maximum likelihood solution generated by the Forward-Backward algorithm.)

To decode a given orthographic word into its underlying syllable sequence, first construct a table of the possible syllables in the manner given above. Use the variant of the parsing algorithm described above to obtain a value for the most likely syllable sequence which could have given rise to the observed spelling in a way consistent with the *Viterbi* algorithm for strict HMM's.

Training and testing the model

A large collection of words was obtained for which orthography, syllable boundaries and pronunciations were available[11], ultimately from a machine readable dictionary, the Collins English Dictionary. As described[11] the original data was extracted from a type-setting tape in which the words were listed in the usual forms with abbreviations, run-ons, and other typographical devices. These were first regularised by a combination of human and programmed conversioned so that no difficulties were encountered in the current experiment.

The word entries were then divided into training data (220,000 words) and test data (5,000 words) by randomly extracting words. It was observed that the 220,000 words in the training text were composed of sequences of syllables taken from a set of 27,000 unique syllable types. An initial estimate of the syllable bi-gram model can be directly computed by observation. This initial model was able to decode the training data with 96% accuracy and the test data with 89% accuracy. This indicates the requirement for a smoothing technique to generalise the parameters of the bi-gram syllable model. Such smoothing may reduce the accuracy of the model on the training data, but should improve it on the test data.

A further 100,000 words, not previously seen in the dictionary, were obtained from a corpus of 100 million words of Newspaper articles (available on published CD ROM from the *Guardian* and *Independent* newspapers). Numeric items, formatting words, and other textual items not suitable for this test were omitted. Assuming that no new syllable types are required to model this data, the training procedure described above was used to adapt the initial statistics obtained by direct inspection. The performance of the model on the training text was 94% and on the test data 92%. This indicates that some generalisation had occurred which made the model less specific to the initial training text, but more robust on the test text.

The affect of this syllable model on the overall pronunciation system is as follows: The basic decision tree transcription system when working directly from orthography to phonemes has a word correct accuracy of 86% on training text and 78% on test data. (the result for training data is not 100% as expected because of smoothing and other generalisations in the decision tree construction process). With the use of syllables as marked, and a new decision tree grown on the syllable marked training data, the overall system has a word accuracy rate of 92% on the training data and 89% on the test data.

Conclusions

A method of determining syllable boundaries has been shown. The method can be improved by the use of a tri-syllable model and by the use of more training data. Other extensions could be explored quite easily. The method does not find new syllable types. For this some type of unsupervised clustering method is required. The method leaves unsolved the treatment of unusual or idiosyncratic textual conventions, notations, and numeric information. It seems that rule-based techniques will still be needed.

While the more serious question still to be answered for TTS systems lie elsewhere, for example in prosody[10], the inability of systems to perform transcription with high accuracy makes this still an open question. The problem of transcription is also of interest in Speech Recognition[6] where there is a need to generate phonetic baseforms of words which are included in the recognisers' vocabulary. In this case the work required to generate a pronouncing dictionary for a large vocabulary in a new domain, including many technical terms and new jargon not previously seen, calls for an automatic, rather than manual technique.

In the wider context the method applied here is another example of self-organising methods applied to Natural Language Processing. While these methods have found a fundamental place in speech processing (for example, speech recognition) they have yet to be seriously adopted for language processing. It is a possibility that many more specific tasks in language processing may be amenable to treatment by self-organising methods, with a consequent improvement in the reliability and ease of replication of the NLP systems which incorporate them.

References

1. J.Allen, M.S.Hunnicutt and D.Klatt, *From Text to Speech,* Cambridge University Press,Cambridge, 1987.
2. S.M.Lucas and R.I.Damper, *Syntactic neural networks for bi-directional text-phonetics. pp 127-141* in *Talking Machines,* ed G.Bailly and C.Benoit, North Holland, 1991.
3. W.A.Ainsworth and N.P.Warren, *Application of Multilayer Perceptrons in Text-to-Speech Synthesis Systems,* pp 256-288 of *Neural Networks for Vision, Speech and Natural Language,* ed D.J.Myers and C.Nightingale, Chapman Hall, 1992.
4. S.Parfitt and R.A.Sharman, *A bi-directional model of English Pronunciation,* pp 801-804, Proceedings of EuroSpeech 91, Genoa, 1991.
5. L.R.Bahl, P.V. deSouza, P.S.Gopalakrishnan, D.Nahamoo and M.A.Picheny, *Context-dependent Modelling of Phones in Continuous Speech using Decision Trees,* IEEE ICASSP 1992.
6. F.Jelinek, et al., *The development of a large-vocabulary discrete word Speech Recognition system, IEEE Trans, Speech and Signal Processing, 1985.*
7. L.Rabiner, *Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,* Proc IEEE vol 77, no 2. pp257-286, 1989.
8. S.R.Hertz, J.Kadin, and K.J.Karplus, *The Delta Rule Development System for Speech Synthesis from Text,* Proc IEEE vol 73 no 11. pp 1589-1601, 1985.
9. K.Church, *Pronouncing proper names,* ACL Chicago, 1985.
10. R.Collier, H.C.Van Leeuwen and L.F.Willems, *Speech Synthesis Today and Tomorrow,* Philips Journal of Research and Development, vol 47 no 1, pp 15-34, 1992.
11. S.G.Lawrence and G.Kaye, *Alignment of phonemes with their corresponding orthography,* Computer Speech and Language vol 1, pp 153-165, 1986.
12. M.Giustiniani, A.Falaschi and P.Pierucci, *Automatic inference of a Syllabic Prosodic Model,* Eurospeech pp 197-200, 1991.
13. B.Merialdo, *On the locality of the Forward-Backward Algorithm,* IEEE Transactions on Speech and Audio Processing, pp 255-257 vol 1 no. 2, April 1993.
14. A.V.Aho and J.D.Ullman, *The theory of parsing, translation and compiling,* Prentice-Hall, 1972.