Spelling as a Complementary Strategy for Speech Recognition

Keith Vertanen

Department of Computer Science
Montana Tech of the University of Montana
Butte, Montana, USA
kvertanen@mtech.edu

Per Ola Kristensson

School of Computer Science
University of St Andrews
St Andrews, Fife, UK
pok@st-andrews.ac.uk

Abstract

We compare a variety of strategies for incorporating spelling to create more robust voice-only speech interfaces. These strategies use different combinations of speaking the word, spelling the word, and spelling the word using a phonetic alphabet. For correcting a single recognition error, spelling the word or speaking and spelling the word reduced error rates substantially. Phonetic-spelling was very accurate with error rates on a 5K task approaching zero. Most importantly, multiple input strategies can be used simultaneously with only a modest degradation in performance compared to allowing only a single input strategy. Thus our work shows that spelling-based input strategies offer the potential of a simple, natural and effective way for users to both avoid and correct recognition errors.

Index Terms: speech recognition, error correction

1. Introduction

Recovery from recognition errors in a speech interface can be a time-consuming and frustrating process. While the process can be improved via multimodal correction, in some situations voice-only input is preferred, or sometimes required (e.g. in-car appliances, telephone dialog systems, or for users who cannot type due to a disability). The use of spelling in speech interfaces has been investigated for recognition of proper names [1, 2, 3], city names [4], spontaneous speech [5], and to help correct dictation errors [6, 7].

However past work has not systematically compared the various ways users may perform a spelling. For example, a user may spell a word, speak-and-spell a word, or speak-and-spell-and-speak a word. There is also no prior investigation of the accuracy tradeoffs associated with including one or more spelling-based strategies when users choose not to provide a spelling.

In this paper, we provide a thorough within-subject accuracy comparison of seven input strategies involving speaking, spelling, and phonetically-spelling a word. This systematic comparison enables us to understand the potential accuracy offered by individual input strategies and whether multiple strategies can be effectively combined into a single recognition system.

2. Data Collection

2.1. Input Strategies

Our experimental task was to correct a single isolated substitution error that had been made during recognition of spoken WSJ sentences. For each substitution error, we collected seven different utterances using the different ways of combining speaking the word, spelling the word, or phonetically spelling the word. The input strategies we tested were:

- **Word (W)** – The word pronounced normally: "cat".
- **Spelling (S)** – The spelling of a word: "C A T".
- **Word + spelling (WS)** – The word followed by its spelling: "cat C A T".
- **Word + spelling + word (WSW)** – The word before and after the spelling: "cat C A T cat".
- **Phonetic (P)** – The military phonetic-spelling of a word: "charlie alpha tango".
- **Word + phonetic (WP)** – The word followed by its phonetic-spelling: "cat charlie alpha tango".
- **Word + phonetic + word (WPW)** – The word before and after its phonetic-spelling: "cat charlie alpha tango cat".

2.2. Audio Collection and Recognition

We collected our audio data using a Flash-enabled web page fielded on the popular crowdsourcing market Amazon Mechanical Turk. Our human intelligence task (HIT) specified that we required native speakers of North American English who could record in a quiet environment using a headset microphone. We provided both text and audio examples for each of the input strategies. We created a HIT for each of the unique words in our set of substitution errors. Four unique workers recorded each word using all seven input strategies. We paid $0.25 per HIT and required US workers with a 95% HIT acceptance rate.

We used the HTK speech recognizer with a speaker-independent US-English acoustic model trained on 211 hours of wideband (16 kHz) WSJ audio data. We trained cross-word triphones with a 3-state left-to-right HMM topology. We used a 39-dimensional feature vector with 13 Mel-frequency cepstral coefficients, deltas and delta deltas, and utterance-wide cepstral mean normalization. Our model had 10000 tied-states, 32 continuous Gaussians per state (64 for silence states), and diagonal covariance matrices. We used the CMU pronunciation dictionary (39 phones plus silence).

Our trigram language model was trained on newswire text from the CSR-III and English Gigaword corpora (1.5B words). We used the WSJ 5K open vocabulary set. We used interpolated modified Kneser-Ney smoothing with no count-cutoffs. We entropy-pruned the model to reduce its size.

Using HDecode we performed recognition on the WSJ0 `si_dt_05` and WSJ1 `si_dt_05` test sets (923 total utterances). The word error rate (WER) was 9.3%. From these results, we selected words that were a) substitution errors, b) in the 5K vocabulary, and c) two or more letters long. We excluded one-letter words to avoid confusion between the read version of a word and its spelled version. We selected only words where the two words to the left and right were correctly recognized, or where the error occurred near the boundary of the sentence. This resulted in 229 error locations. For each location we added the reference word at that error location to our set of words. This gave us 99 unique words that we had four different workers record in each of the seven forms.

2.3. Collection Results

Workers took on average 1.5 minutes to complete each HIT. A total of ten unique workers completed 2793 utterances in 21 hours. We listened to every utterance we received. We rejected the work of one non-native English speaker. In six utterances, the worker misspoke. In these cases, we eliminated that utterance along with the other six recorded by the worker in that particular HIT. Overall we were pleasantly surprised with the audio quality.

We measured utterance length as the time between the worker clicking MIC ON and MIC OFF. Workers could playback their last utterance and re-record as necessary. If a worker had multiple microphone cycles for an utterance, we used the length of the last one recorded. We also measured how often workers re-recorded utterances.

The mean utterance length (in seconds) for each strategy were: W 2.3, S 3.9, WS 5.0, WSW 6.1, P 4.9, WP 5.5, WPW 6.3. As expected, speaking just the word was the quickest. Spelling was slower than reading the word and phonetic-spelling was slower still. However, note that even the most tedious WPW strategy only required an additional four seconds. For situations where the user may anticipate recognition errors (e.g. when correcting a previous misrecognition or speaking a strange name), a few extra seconds spent spelling a word might be time well spent compared to risking a time-consuming correction episode. We saw little difference in the number of microphone cycles between the input strategies (mean 1.1 cycles). This suggests, at least when users are given the text to read (including any spelling), the various strategies are at a similar level of difficultly.

3. Recognition Experiments

3.1. Recognition Grammar and Dictionary

Recognition used HVite and a simple word network that connected the start-word to every word in the 5K vocabulary. Each word was then connected to the end-word. Thus the recognizer was forced to recognize exactly one word from the 5K vocabulary. We used the same word network for recognizing each type of input utterance. Initially, we were interested in the relative accuracy of the input strategies and not their absolute error rate. Thus we used a word network with no language model information (i.e. it was a simple grammar of 5K words with no *a priori* preference between the words).

We changed the pronunciation dictionary to reflect whether a word was spoken, spelled, or a combination thereof. For normal spelling, we used the pronunciation in the CMU dictionary for the words "A.", "B.", etc. For phonetic-spelling, we used the pronunciation for "alpha", "bravo", etc. The words "golf", "lima" and "whiskey" had two pronunciations in the CMU dictionary. We used only the first pronunciation. When words contained an apostrophe, the spelling strategies used the pronunciation for the word "apostrophe".

Our acoustic model has a long silence model `sil` and a short pause model `sp`. The `sil` phone has three emitting states. The `sp` model has a single emitting state and can transition without emission. The `sp` model is skipped over when determining cross-word triphone contexts while the `sil` model is used in triphone contexts. We tested three silence separation strategies:

- **short separation** – The `sp` model was used between the pronunciation of every letter in normal and phonetic spelling. `sp` was also used between any initial word pronunciation and a spelling, and between any spelling and a final word pronunciation. For example, the WSW entry for "cat" is: `k ae t` *`sp`* `s iy` *`sp`* `ey` *`sp`* `t iy` *`sp`* `k ae t`.
- **long separation** – The same as short separation but using the `sil` phone in place of `sp`.
- **long/short separation** – The `sil` model was used to separate normal word pronunciations from any spelled version. The `sp` was used between every letter in normal and phonetic spelling. For example, the WSW entry for "cat" is: `k ae t` *`sil`* `s iy` *`sp`* `ey` *`sp`* `t iy` *`sil`* `k ae t`.

Input strategy	Silence separation short	long	long/short
Word	49.6*	49.6*	49.6*
Spelling	30.9*	11.3	30.9*
Word + spelling	19.0	3.6	10.5
Word + spelling + word	20.3	4.4	9.5
Phonetic	1.3*	1.3	1.3*
Word + phonetic	0.3	0.5	0.5
Word + phonetic + word	1.3	0.5	0.3
Mean	17.5	10.2	14.7

Table 1: Word error rate for different input strategies and silence separation methods. In some rows certain silence separation methods are equivalent (indicated by *).

3.2. Single Strategy Experiment

In our first recognition experiment, we assume the system knows which input strategy the speaker is using. For each strategy we had a set of 389 utterances. We tested each set using pronunciation dictionaries built only for that particular input strategy. We tested each set with three different dictionaries, one for each silence separation method.

Table 1 shows the results of the first experiment. Note that since our recognizer can only return a single word, our reported WER is equivalent to the percent of inputs that failed. Across all silence separation methods, speaking just the word was not an effective strategy. Spelling performed better and phonetic-spelling performed much better. Prefixing the spelling or phonetic-spelling with the word improved accuracy further. In fact, saying the word followed by the phonetic spelling had near perfect accuracy. This was despite our recognizer using no information about the probability of the different words. Speaking the word both before and after the spelling performed similarly to speaking the word only before the spelling.

Workers tended to pause longer between the word and the spelling section and between the spelling section and any final word. The pauses during spelling or phonetic-spelling were often much shorter. Despite this, as shown in table 1, on average the best silence separation strategy was to use a `sil` phone everywhere. Using only the `sil` phone, spelling and phonetic-spelling improved markedly. We observed that users adopted a very staccato speaking style during spellings. We conjecture there was little coarticulation between phones in different letters' pronunciations, making the context-blocking nature of the `sil` phone better. For the remainder of this paper, we will use dictionaries with long silence separation.

3.3. Combined Strategy Experiment

An interface based on a single input strategy would force a rigid interaction style where any deviation from the supported strategy would likely result in misrecognition. Enabling multiple input strategies simultaneously would allow users to intelligently select between strategies based on the anticipated difficulty of an input. We therefore compared our single strategy results with two combined strategy systems. The first combination assumed users would not use phonetic-spelling (combining W, S, WS, and WSW). The second combination used all seven strategies. The combined systems concatenated the strategy-specific pronunciation dictionaries. All utterance types were then recognized against this single dictionary.

Input strategy	Single	Word/spell	All
Word	49.6	49.9	49.9
Spelling	11.3	12.6	12.6
Word + spelling	3.6	4.9	4.9
Word + spelling + word	4.4	4.6	4.6
Phonetic	1.3	n/a	1.3
Word + phonetic	0.5	n/a	0.5
Word + phonetic + word	0.5	n/a	0.5

Table 2: Word error rate for systems supporting a single input strategy or multiple strategies.

As shown in table 2, the combined systems had only a small increase in error rate for the W, S, WS and WSW utterances. Both combined systems performed similarly on these utterances. The system combining all strategies was able to recognize phonetic-spelling utterance types with the same accuracy as the strategy-specific baseline. Thus it appears that the phonetic-based utterances are so different from normal word or spelling utterances that they do not confuse the recognizer. Notably, normal word recognition experienced very little accuracy degradation even when six additional input strategies were active.

The main disadvantage of using the combined systems was the additional time required for recognition. The combination without phonetic-spellings was 1.7 times slower than the single-strategy baseline. Using all strategies was 2.3 times slower than the single-strategy system. But given the advantage of allowing a choice of input strategies, we will use a system that combines all seven strategies for the remainder of this paper.

3.4. Larger Vocabulary Size Experiment

Thus far we have used a small vocabulary of 5K words. When users provided spelling information, very accurate input was possible despite not using a language model.

We also investigated spelling-based input using larger vocabularies. We built dictionaries and word networks that included the top 10K, 20K and 64K words from our newswire data. All words in our audio collection were in every vocabulary size. Since larger vocabularies slowed the HVite decoder, we narrowed our search beam to 250 compared to the beam of 500 used in previous experiments. As shown in table 3, recognition became harder

Input strategy	Vocabulary size 5K	10K	20K	64K
Word	49.9	54.2	59.1	67.1
Spelling	12.9	15.4	17.7	22.9
Word + spelling	4.9	6.9	8.7	11.3
Word + spelling + word	4.6	6.2	7.2	9.8
Phonetic	4.6	4.6	4.9	6.7
Word + phonetic	0.8	1.5	1.5	2.1
Word + phonetic + word	2.1	2.3	2.3	3.6
Mean	11.4	13.0	14.5	17.6

Table 3: Word error rate for different vocabulary sizes.

Input strategy	Language model none	uni	bi	tri
Word	56.0	51.2	46.7	34.9
Spelling	17.9	11.5	14.7	9.6
Word+spelling	8.8	3.7	5.9	3.8
Word+spelling+word	8.6	6.1	5.4	4.3
Phonetic	0.8	1.1	1.1	1.1
Word+phonetic	0.3	0.3	0.5	0.3
Word+phonetic+word	0.3	0.5	0.8	0.3
Mean	13.2	10.6	10.7	7.8

Table 4: Word error rate of different 5K language models.

as vocabulary sized increased. The danger of too narrow a search beam is evidenced by the much higher error rate on the P utterances compared to earlier experiments.

3.5. Language Model Context Experiment

Finally, we investigated the input strategies in a voice-only correction scenario in which a user has selected a recognition error and spoken a correction. The error selection may have been done manually by the user (e.g. using the mouse) or automatically using the spoken correction [8]. In such a scenario, the prior words are likely correct and can provide language model context.

We set each word's probability using a 5K unigram, bigram or trigram language model trained on our newswire corpora. The bigram and trigram models used the prior words of context given the location of the error in the WSJ test sets. Recall that we selected 229 substitution errors in the WSJ test sets and these consisted of 99 unique words (the words we recorded from workers). We eliminated one word "point" because it constituted a large number of the error locations (70 out of 229).

The four different recordings for each word and for each input strategy were recognized at each error location. To make the no language model and unigram results comparable to the bigram and trigram results, we duplicated the no language model and unigram results according to the number of times each unique words appeared in the set of error locations. Note this duplication makes the no language model results differ from our previous experiments. We used a search beam of 500. Table 4 shows that more language model context resulted in higher accuracy. The advantage of spelling strategies remained strong and consistent with previous experiments.

4. Conclusions

Our results demonstrate how speech interfaces can be made more robust if users are allowed to supply additional information via spelling and phonetic-spelling. For speech interfaces designers considering incorporating spelling-based strategies, we highlight several important considerations. First, the accuracy of spelling-based strategies depends crucially on using a silence model that blocks phonetic-context between spelled letters. Without such context-blocking, error rates using spelling-based input strategies increased several fold.

Second, interface designers may want to offer a set of possible input strategies since users may incorporate spelling in different ways. We provide the first systematic accuracy comparison of various word input options that involve spelling. We show that multiple spelling-based strategies can be used in tandem with normal input mechanisms without unduly impacting normal recognition. Based on this evidence we recommend speech interface designers consider incorporating spelling-based strategies. They offer the potential for a natural, simple, and effective way to help users avoid and correct errors.

5. References

[1] M. Meyer and H. Hild, "Recognition of spoken and spelled proper names," in *Proceedings of the 5th European Conference on Speech Communication and Technology*. ISCA, 1997, pp. 1579–1582.

[2] G. Chung, S. Seneff, and C. Wang, "Automatic acquisition of names using speak and spell mode in spoken dialogue systems," in *Proceedings of HLT-NAACL*. ACL, 2003, pp. 32–39.

[3] N. Davidson, F. McInnes, and M. A. Jack, "Usability of dialogue design strategies for automated surname capture," *Speech Communication*, vol. 43, no. 1-2, pp. 55–70, 2004.

[4] E. Filisko and S. Seneff, "Developing city name acquisition strategies in spoken dialogue systems via user simulation," in *Proceedings of the 6th SIGDial Workshop on Discourse and Dialogue*, 2005.

[5] H. Hild and A. Waibel, "Integrating spelling into spoken dialogue recognition," in *Proceedings of the 4th European Conference on Speech Communication and Technology*. ISCA, 1995, pp. 1977–1980.

[6] A. E. McNair and A. Waibel, "Improving recognizer acceptance through robust, natural speech repair," in *Proceedings of the 3rd International Conference on Spoken Language Processing*. ISCA, 1994.

[7] B. Suhm, B. Myers, and A. Waibel, "Multimodal error correction for speech user interfaces," *ACM Transactions on Computer-Human Interaction*, vol. 8, no. 1, pp. 60–98, 2001.

[8] K. Vertanen and P. O. Kristensson, "Automatic selection of recognition errors by respeaking the intended text," in *Proceedings of the 11th Biannual IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE Press, 2009, pp. 130–135.