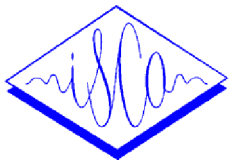
ISCA Archive
http://www.isca-speech.org/archive



Speech and Language Technology in Education (SLaTE2009)
Wroxall Abbey Estate, Warwickshire, UK
September 3-5, 2009

A Rule-Based Language Model for Reading Recognition

Jian Cheng, Brent Townshend

Knowledge Technologies, Pearson
299 S. California Ave, Palo Alto, California 94306, USA
jian.cheng@pearson.com

Abstract

Systems for assessing and tutoring reading skills place unique requirements on underlying ASR technologies. Most responses to a "read out loud" task can be handled with a low perplexity language model, but the educational setting of the task calls for diagnostic measures beyond plain accuracy. Pearson developed an automatic assessment of oral reading fluency that was administered in the field to a large, diverse sample of American adults. Traditional N-gram methods for language modeling are not optimal for the special domain of reading tests because N-grams need too much data and do not produce as accurate recognition. An efficient rule-based language model implemented a set of linguistic rules learned from an archival body of transcriptions, using only the text of the new passage and no passage-specific training data. Results from operational data indicate that this rule-based language model can improve the accuracy of test results and produce useful diagnostic information.

1. Introduction

The U.S. Government's National Center for Education Statistics commissioned a reading fluency assessment as part of the 2003 NAAL (National Assessment of Adult Literacy) called the Fluency Addition to NAAL, in which each respondent read aloud from lists and passages of text. We focus here on the passages. All oral reading responses were digitally recorded and subsequently analyzed for measures of accuracy and fluency. For the 2003 NAAL project, 181,420 response recordings were collected. To expedite the scoring of these responses and to extract additional information from the responses that human raters cannot provide, technology from Ordinate Corporation (now the Knowledge Technologies group of Pearson) was used to automate and augment the analysis of the oral readings.

The evidence that Pearson's automatic scoring of reading accuracy is reliable and valid was presented in [1]. This paper presents the methods for building suitable language models for automatic speech recognition (ASR) used in scoring a person's skill in reading passage aloud. A rule-based language model (RBLM) was proposed to improve recognition accuracy on read-aloud performances, using a set of linguistic rules learned from a collection of transcriptions of other passages being read aloud. Contrary to traditional methods, language models for new passages can be built without transcriptions of readings of the same passage. Furthermore, the rule-based recognition process can yield extra diagnostic linguistic information about the test takers' reading habits that can be reported and analysed.

2. Task analysis

NAAL reading passages were relatively simple expository or narrative texts of about 150-200 words in length. Each of the 18,142 adult respondents read aloud 2 out of 8 passages. There has been previous work [2, 3] using ASR in reading passage. In general a large number of transcriptions are required to build suitable language models to ensure recognition accuracy. A dedicated language model may be built for each passage to capture the stochastic language structure accurately. The requirement for a large number of transcriptions is a disadvantage because each time a set of new passages are introduced, many spoken responses for each passage needs to be transcribed before language models can even be built. Therefore, one goal of this research is to find a method to build suitable language models for a new reading passage based only on the text of the new passage without any passage-specific training data. Instead, information in existing transcriptions of other material can be extracted and applied to new passages. It turns out that these new rule-based language model improve recognition performance, and consequently improve the accuracy of test results. Beyond guiding the recognition process, another goal of the rule-based model development is to identify and aggregate information about the fine structure of the reader's performance. For example, knowing the recognition path taken through one or more rule-based language models should indicate which reading skills have been mastered and which still need work.

For our specified passage-reading task, we know the exact sentences test takers are expected to say. One of the scores, the number of words read correctly, is measured based on the difference between what the test taker says and the expected text. The smaller the difference, the better. This is the big difference between the ASR task in passage reading and ASR in other applications, in which we usually only know that incoming speech relates to a special knowledge domain (a constrained dictionary and grammar). The traditional ASR methods for building and applying language models are not optimal for the special domain of reading testing. In addition, significant numbers of test takers are non-native speakers with low oral proficiency. Their responses may not follow the English grammar and it is important to have the ability to detect such errors.

3. The language models

In a bigram language model, only one previous word w_{i-1} will be used to estimate the likelihoods of the current word. A trigram language model considers only previous two words w_{i-1}, w_{i-2}. We introduce feasible methods that model much longer sequential dependencies.

Suppose that the word string a speaker said is $\mathbf{W} = w_1, w_2, ..., w_n$, $w_i \in V$. V is used to denote a vocabulary. Then a priori probability for this word string is

$$P(\mathbf{W}) = \prod_{i=1}^{n} P(w_i | w_1, ..., w_{i-1}).$$

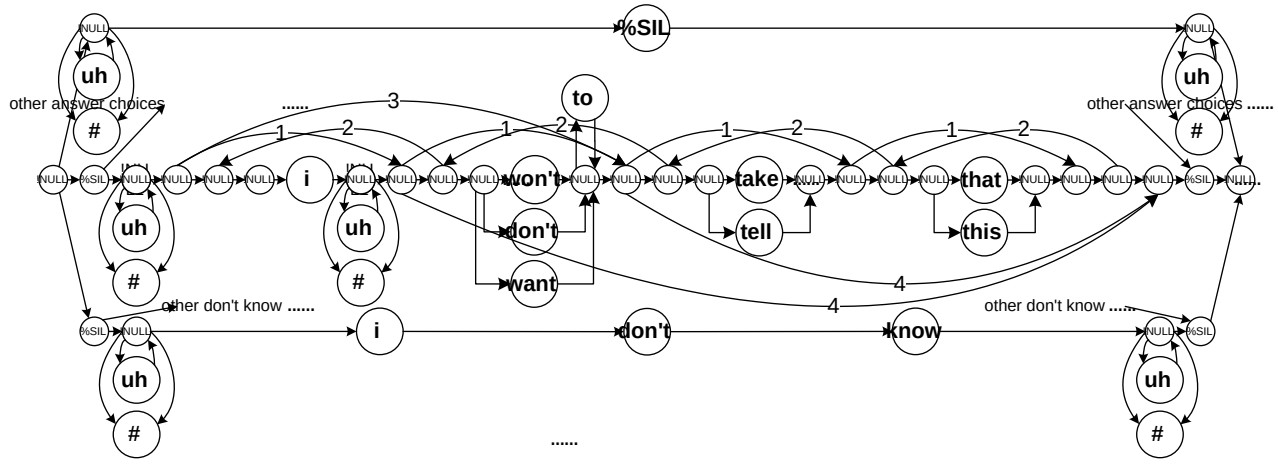


Figure 1: A simplified example of the rule-based language model. The passage text is *i won't take that*.

We hope that we know the value of $P(w_i|w_1, ..., w_{i-1})$ after a speaker produces $w_1, w_2, ..., w_{i-1}$. This value can help us to recognize the next word. But there are too many arguments in $P(w_i|w_1, ..., w_{i-1})$ even for moderate values of i and reasonable vocabulary size. It is impossible to estimate every $P(w_i|w_1, ..., w_{i-1})$ based on previous transcriptions. We need to build a language model to estimate them. Suppose that the language model we use is $\mathbf{L}$, then the output likelihood of this language model can be calculated as

$$P(\mathbf{W}|\mathbf{L}) = \prod_{i=1}^{n} P(w_i|w_1, ..., w_{i-1}, \mathbf{L}).$$

The advantage of using the language model is that the probability that a speaker will choose his i^{th} word does not explicitly depend on the entire history of all his previous words, but depends on the equivalence classes $\Phi_{\mathbf{L}}(w_1, ..., w_{i-1})$. A different language model determines how to choose the appropriate equivalence classification and gives a method to estimate $P(w_i|\Phi_{\mathbf{L}}(w_1, ..., w_{i-1}))$. Thus

$$P(\mathbf{W}|\mathbf{L}) = \prod_{i=1}^{n} P(w_i|\Phi_{\mathbf{L}}(w_1, ..., w_{i-1})).$$

When the word sequence is sufficiently long, the cross-entropy $H(\mathbf{W})$ of a language model on $\mathbf{W}$ can be simply approximated as $H(\mathbf{W}) = -1/n \cdot log_2 P(\mathbf{W})$, and then the preplexity of a language model is the reciprocal of the geometric average probability $P(\mathbf{W}|\mathbf{L})^{1/n}$. The perplexity is a popular measure of a text's complexity within a language model and can be treated as the effective branching factor. In passage reading, the perplexity within a reasonable language model is usually very low.

4. The structure of the RBLM

To avoid collecting and transcribing many renditions for each new passage, we build language models based on the transcription of a fixed body of similar passages as read by people similar to the target population of readers. The basic proposal is that a simple direct graph needs to be built that has a path from the first word in the reading passage to the last word. Different direct arcs are required to represent the different classes of errors made by the readers, such as skipping, repeating, inserting, and substituting words. For each arc, a probability is assigned to represent the chance that the arc will be chosen. We use a knowledge-based approach, which includes a list of linguistic rules, such as *she* may be substituted by *he*, a single noun may be substituted by a plural noun, to represent some potential arcs. The arc itself can remember which rule it stands for. If such a graph represents many of the renditions encountered, then it should be a useful language model.

We give a simple example to explain the idea. Suppose that the passage text is just the four words *i won't take that*, which is also the expected sequence of words that a test taker will speak. The test taker may keep silent or say *i don't know* if he can't read this sentence. Also suppose that we have the following linguistic rules in the knowledge base (how to elicit these rules will be discussed in Section 5): *won't* may be substituted by *don't* or *want*; *take* may be substituted by *tell*; *that* may be substituted by *this*; *to* may be inserted at the end of *won't*; any word in the answer choice may be skipped or repeated; mouth noise (noted as *#*) or hesitation (noted as *uh*) may be inserted at the end of any word or at the beginning or end of the sentence; ... . With these assumptions, we can get a simplified RBLM shown in Figure 1.

For any passage language model, we also have a leading silence and tail silence (noted as $\%SIL$), which stand for the possible silent pause at the beginning or at the end of the sentence. After applying all the rules, the null nodes can be deleted that have only one parent and one child, and then a direct connection between the parent and child can be given.

From Figure 1, we can see, for every language model, we have a start null node and an end null node. Every arc from the start null node can stand for a category, such as correct, wrong, silence or *i don't know*. The category information can help us figure out which direction the test taker goes. For some categories, it is possible to have multiple choices, such as other answer choices and other *don't know*. The test taker may speak *i'm sorry* instead of *i don't know*. The leading silence follows every category except the silence category itself. If we want to know the relative proportion of the different response choices or *don't know*, we can let the corresponding arcs remember which choice they stand for. In the figure, we omitted some nodes of hesitation and mouth noise to simplify the graph. Type 1 arc stands for skipping any word. Type 2 arc stands for repeating

any word. Type 3 arc stands for skipping a phrase break. Type 4 arc stands for the situation that test taker will jump from any place to the end of the passage (i.e. stop reading). This situation happens quite often in passage-reading tests because of the time constraint. Since we have the rules of skipping or repeating any word, this language model can cover any response whose words are in this model.

One advantage of this language model is that we can record what's really going on. If a test taker speaks *i don't take that*, after matching to this language model to find the maximum likelihood path, we can tell that the rule *won't* is substituted by *don't* and such rule has been fired. Other situations, such as repeating a word, skipping a word, can be caught easily. During the scoring process, we may take advantage of such information to score them differently. For example, the score penalty for repeating a word could be treated as less than that for skipping a word.

The RBLM essentially is a Markov chain. Every state in the language model corresponds to the equivalence class of preceeding words as classified by the language model. The probability that a speaker will choose his i^{th} word depends on a certain state. Between that state and the start null node a path that covers his previous words should exist.

5. Extracting linguistic rules

To extract linguistic rules, many transcriptions of spoken responses to various passages are required. Since using tagged passages can give us the information about the linguistic structures of passages, we tag all the passage texts. It can be done automatically using different tagging tools. The tag set we used is the Penn Treebank Tag set [4]. We also add potential phrase breaks in answers. The linguistic structure can give us flexibility to add rules that are more general. Some general rules that can be figured out easily based on the tagged answer choices are: NN (noun, singular or mass) becomes NNS (noun, plural); VBZ (verb, 3rd person singular present) becomes VBP (verb, non-3rd person singular present); and so on. These rules happen quite frequently in the responses of non-native English speakers.

5.1. Building rules

As the first step, the following four rules were put in our knowledge base: any word can be substituted by any word with a probability 0.0000001; any word can be inserted after any word with a probability 0.0000001; any word can be skipped with a probability 0.001; any word can be repeated with a probability p = 0.001. Then a RBLM was built as discussed in Section 4. For each transcription, we use the Viterbi algorithm [5] to find the maximum likelihood path in the language model. This process is similar to ASR decoding. Note that we assign a very low probability to the garbage models (rules permitting any word to be replaced by any word and any word to be inserted after any word). They are the only rules that allow out-of-vocabulary words to appear, and their probability is fixed to the lowest level no matter how we update other probabilities. They will never be fired unless there is no other choice. The garbage rules will only be used during rules building process. When we build the RBLM, they will be ignored. By collecting garbage model firing patterns, such as context and tags in the language model, we can cluster similar cases and propose rules with the firing frequencies for linguists to review. Starting from the most frequent transcriptions, the most popular rules should be easy to determine since there are usually only one or two rules used in these transcriptions. On the other hand, test takers make the same mistakes quite often. After the most frequent rules are added to the knowledge base, linguists can more easily identify other less frequent ones. To be easy to generalize to new passages, we should try to use the general rules (such as rules based on tags) instead of using the context-dependent rules (such as word substitutions).

By counting the rule-firing situations, word coverage can be understood; i.e., the number of times that garbage models were fired in a transcription divided by the total number of words in the transcription, equal to one minus word coverage. Our goal is to introduce reasonable rules to improve word coverage. After every cycle, a few more rules may be added until no significant improvement in word coverage is observed. After such a state is reached, the rule building process can be finished.

5.2. The probabilities of rules

There is one problem in our procedure: we need to have the probabilities of the rules that we are trying to estimate to figure out the best path. Fortunately, the *expectation-maximization* algorithm [6] can help us to overcome this problem. We can start with a guess at the probabilities of rules (in our case, we assign a small probability to rules), obtain better estimates, and then put them back to run the program again, so that we can obtain an even better estimate.

The probabilities for different rules are estimated using the maximum likelihood method. The maximum likelihood parameter estimate could be obtained by counting. At the individual passage level, for every node, there are several out arcs. If the matched path for a transcription includes that node, we increase the count by one for every out arc of that node. However, we increase the fired count by one only for the out arc included in that path. Others are unchanged. After all the transcriptions are matched, the maximum likelihood probability for a rule is the fired count divided by the visited count.

The above discussion is only based on the individual passage level. The rule can be generalized to the whole domain we are interested in. Its probability is the sum of the fired counts for that rule in any individual passage divided by the sum of the visited counts. It is possible that certain rules have different probabilities when they are applied to the different passages. We can distinguish this situation by using chi-squared test [1] to find out if there is a statistically significant difference in the probabilities between the general and individual levels. If not, we only keep the probabilities for the general level. Otherwise, we also keep the probabilities in the individual level that have significant difference. When we use a rule for a passage, we first check if there is an individual-level probability for that rule. If yes, we use that probability. Otherwise, we use the general level.

The whole process can be iterated several times using the new estimated probabilities combined with some new rules. This procedure essentially is an expectation-maximization algorithm, so the final estimated probabilities should converge to a local minimum.

Currently we treat hesitation and mouth noise as general rules. We only distinguish three different kinds of hesitation and mouth noise in the RBLM. They can be inserted at the beginning of the passage, the end of the passage, or at the end of any word.

[1] When the total number is less than 80, we use Fisher's exact test.

6. Experimental results and analysis

From the 18,142 adult respondents, we randomly selected 2,703 test takers as our training set and 1,301 test takers as our test set. Each test taker read 2 out of 8 potential passages. All the passage responses in the training and test sets were transcribed. Thus, there were about 677 transcribed responses for each potential passage in the training set and about 325 in the test set.

Table 1 lists the percentage of out-of-vocabulary words in the RBLM building process. We can see that it decreased very quickly. In the final model, there are still around 1.5% out-of-vocabulary words. Most of them are caused by wrong passage readings, some unintelligible words, and partial sounds.

Step	0	1	2	3	$Final$
TrainingSet	16.8%	3.4%	1.7%	1.6%	1.47%
TestingSet	16.7%	3.6%	2.0%	1.9%	1.52%

Table 1: The percentage of out-of-vocabulary words in the training and test sets during different iteration steps.

Using the transcriptions in the training set, the final rules and their probabilities were generated. We found that the following rules were used frequently by the test takers: *might* is substituted by *may* with the probability 0.088; contraction format (such as she's) becomes no contraction format with the probability 0.073; *VBZ* (verb, 3rd person singular present) is substituted by *VB* (verb, base form) with the probability 0.048; *NNS* (noun, plural) is substituted by *NN* (noun, singular or mass) with the probability 0.044; *NN* is substituted by *NNS* with the probability 0.010; inserting mouth noise (#) at the beginning and the end of the passage with the probability 0.296 and 0.090 respectively; *the* could be inserted in the middle of a structure like IN JJ NNS with the probability 0.094; mouth noise (#) could happen after every word with the probability 0.047; repeating two or three words happens more than three times more often than skipping two or three words; any word could be replaced by its partial words; and so on.

The final rule set was used to build RBLMs. The average perplexity for these models was 1.73. When standard bigram language models were built for comparison, their average perplexity was 2.87. In both cases, we used a non-native triphone acoustic model to do the speech recognition. This acoustic model was trained on a widely representive sample of non-native spoken materials collected by Pearson. None of the data used in these LM experiments had been used for acoustic model training. In the test set (the total number of words is 348,681), we achieved word error rate (WER) 9.0% by using the RBLM, compared to a WER 12.6% when using a passage-specific bigram language model. This suggests that a non-specific rule-based model extracted from 5600 reading performances provides the recognizer with more accurate information than a bigram model based on around 677 reading performances on the specific passage under study.

By taking advantage of the RBLM, for each passage reading, Pearson provided rule-firing details to the National Center for Educational Statistics for further analysis. By analyzing the rule-firing situation in context, test takers' reading errors including real substitutions, omissions, insertions, self-corrections, and reversals, etc. can be accurately obtained using automatic methods.

It is well-known that the children's speech is substantially harder to be recognized than adults' [7]. We tested the model to a children's oral reading dataset. A total of 164 elementary school students (46 first graders, 62 third graders, and 56 fifth graders) were recruited from different parts of the United States, from a range of ethnic and linguistic backgrounds. Roughly half of the students were male and half were female. Each student took the grade-appropriate Benchmark test (3 passages) yielding 492 responses. Each response was 90 seconds in length. All the responses were transcribed. The rules learned from previous adult data were used directly to build RBLMs. All the transcriptions here plus some from similar source were used to build bigram language models. The average number of transcriptions for a bigram language model is 126. The average perplexty is 2.22 for RBLMs and 4.22 for bigram language models. WER was 13.1% by using the RBLM, compared to a WER 26.1% when using a passage-specific bigram language model.

This RBLM has also been successfully applied in a test of oral English proficiency, Versant for English [8] that uses ASR and other automatic techniques to score reading, elicited imitations, and sentence construction tasks, again, without the usual transcription requirement for new items.

7. Conclusions

We proposed an RBLM for reading recognition. This model can be built by applying different linguistic rules to passage texts. Our experimental results showed that the recognition performance of the RBLM was significantly better than that of the bigram language model in passage reading task. After building enough general linguistic rules, the RBLM should be able to be applied to other low perplexity recognition domains without the requirement of transcriptions. Thus, it significantly reduces the time and cost of human transcribers in developing automated tests that score spoken performances. The RBLM opens the possibility of automatically catching spoken grammar mistakes and making diagnostic suggestions.

8. Acknowledgements

The part of work was performed as a subcontract to Westat, Inc., under a contract with the U.S. Department of Education.

9. References

[1] Balogh, J., Bernstein, J., Cheng, J., and Townshend, B., "Automatic evaluation of reading accuracy: assessing machine scores", In SLaTE 2007, 112-115.

[2] Mostow, J., Roth, S.F., Hauptmann, A.G., and Kane, M., "A prototype reading coach that listens", In AAAI 1994, 785-792.

[3] Li, X., Deng, L., Ju, Y., and Acero, A., "Automatic children's reading tutor on hand-held devices", In Interspeech 2008, 1733-1736.

[4] Marcus, M. P., Santorini, B., and Marcinkiewicz, M. A., "Building a large annotated corpus of English: The Penn Treebank", Computational Linguistics 19(2):313-330, 1993.

[5] Viterbi, A., "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm", IEEE Trans. on Information Theory IT-13:260-269, 1967.

[6] Dempster, N. M., Laird, A. P., and Rubin, D. B., "Maximum likelihood from incomplete data via the EM algorithm", J. R. Statist. Soc., B 39:185-197., 1977.

[7] Li, Q. and Russel, M. An analysis of the causes of increased error rates in children's speech recognition, In ICSLP 2002, 2337-2340.

[8] Bernstein, J. and Cheng, J., "Logic and validation of a fully automatic spoken English test", In Holland, V. M. and Fisher, F. P. (Ed.), The Path of Speech Technologies in Computer Assisted Language Learning, 174-194, 2007.