

A Contribution of Intrinsic Speech Variabilities to Errors Done by Speech Recognition

Miloš Cernák

Institute of Informatics, Slovak Academy of Sciences
Dúbravská cesta 9, 845 07 Bratislava, Slovakia
`milos.cernak@savba.sk`

Abstract. A usual way of ASR accuracy evaluation is calculation of Word Error Rate (WER) and Sentence Error Rate (SER). The misrecognitions that contribute to WER are classified into three categories: deletions, insertions and substitutions. The paper presents a study about a contribution of intrinsic speech variabilities to the each of the error category. Decision tree (DT) analysis is used. Five DT styles are examined: CART, C4.5, and then Minimum Message Length (MML), strict MML and Bayesian styles decision trees. We apply these techniques to data of the computer speech recognition fed by intrinsically variable speech.

1 Introduction

A usual way of ASR accuracy evaluation is calculation of Word Error Rate (WER) and Sentence Error Rate (SER). The misrecognitions that contribute to WER are classified into three categories: deletions, insertions and substitutions. The paper presents a study about a contribution of intrinsic speech variabilities to the each of the error category. Decision tree (DT) analysis is used. Five DT styles are examined: CART, C4.5, and then Minimum Message Length (MML), strict MML and Bayesian styles decision trees. We apply these techniques to data of the computer speech recognition fed by intrinsically variable speech.

This paper contributes to the comparison of decision tree classifiers for speech recognition. Several decision tree methods are examined. The diagnoses is based on ASR experiments that use intrinsically variable speech, witch include fast, slow, loud, soft (low, not whispered) speech, plus questioning and normal style speech.

The paper is structured as follows: Next section 2 introduces decision tree classifiers in test. Section 3 describes used data and ASR experiments, on which decision tree (DT) diagnosis was applied. Experiments and comparisons are described in section 4. Finally, sec. 5 concludes the paper.

2 Decision Tree Classifiers

Decision trees are classifiers that represent their classification knowledge in tree form (usually in binary tree form). Each interior node of a decision tree is a

test on an attribute. Satisfying that test causes the instance being classified to take one branch out of that node, failing the test causes the instance to take the other branch. A decision tree is used to classify an instance by starting at the root node of the decision tree and following the path the attribute tests dictate until a leaf node is encountered. Each leaf node in a decision tree is a decision, i.e., represents a classification. An instance that ends up at some particular leaf node is classified with the class assigned to that leaf node. A second kind of tree is a class probability tree. This has a vector of class probabilities at each leaf instead of a decision [1].

The basic algorithm builds a tree top down using the standard greedy search principle, based on recursive partitioning. The partitioning algorithm includes stopping, splitting and pruning rules.

2.1 Classification and Regression Trees

Most of the automatic failure diagnosis use either CART or the C4.5 method (see e.g. [2],[3]). We can classify three most popular CART styles, which comes from the splitting rule that is applied:

- CART style using Gini index of diversity. Gini looks for the largest class in the training list and strives to isolate it from all other classes. It produces good results for a large variety of classification problems and is thus the default rule used for CART. Gini index of diversity minimizes the risk involves when making predictions once having made the test, using the following equation [1]:

$$\begin{aligned}
 G(class|test) &= \sum_{i=1}^T Pr(outcome\ i)G(class|outcome\ i) = \\
 &= \sum_{i=1}^T \frac{n_{i,\cdot}}{n_{\cdot,\cdot}} \sum_{j=1}^C \frac{n_{i,j}}{n_{i,\cdot}} \left(1 - \frac{n_{i,j}}{n_{i,\cdot}}\right)
 \end{aligned}
 \tag{1}$$

Here $n_{i,j}$ corresponds to the number of examples at the node being evaluated that fall in test outcome i and have class j , $n_{\cdot,j}$ is the number that has class j regardless of test outcome, and $n_{i,\cdot}$ is the number that has test outcome i regardless of class. T is total number of tests, and C is total number of classes.

- CART style using information gain (entropy) splitting rule. The Entropy rule, which is very similar to Twoing in practice, strives for similar splits. The split is motivated by minimization of entropy between a parent node and a sum of entropies of two child nodes. Information gain maximizes the information gained about the class making the test. The following formula

is used [1]:

$$\begin{aligned}
 I(class|test) &= \sum_{i=1}^T Pr(outcome\ i)I(class|outcome\ i) = \\
 &= - \sum_{i=1}^T \frac{n_{i..}}{n_{...}} \sum_{j=1}^C \frac{n_{i,j}}{n_{i..}} \log_2 \frac{n_{i,j}}{n_{i..}}
 \end{aligned} \tag{2}$$

- CART style using "twoing". The philosophy of twoing is far different from that of Gini. Rather than initially pulling out a single class, twoing first segments the classes into two groups, attempting to find groups that together add up to 50 percent of the data. Twoing then searches for a split to separate the two subgroups. The twoing rule strikes a balance between purity and creating roughly equal-sized nodes.

2.2 C4 Method

C4.5 style uses Quinlan's gain ratio splitting rule [4]. Information Gain measure in terms of Eq. 2 can be defined as

$$\begin{aligned}
 Gain(class|test) &= I(test) - E(class|test) = \\
 &= - \sum_{j=1}^C \frac{n_{i,j}}{n_{i..}} \log_2 \frac{n_{i,j}}{n_{i..}} + \sum_{i=1}^T \frac{n_{i..}}{n_{...}} \sum_{j=1}^C \frac{n_{i,j}}{n_{i..}} \log_2 \frac{n_{i,j}}{n_{i..}}
 \end{aligned} \tag{3}$$

where $I(test)$ measures randomness of the distribution of examples under test over C possible classes, and $E(class|test)$ is expected information for the tree with $class$ as root.

The Quinlan's modification consists of dividing $Gain(class|test)$ by the following expression

$$IV(class) = \sum_{j=1}^C \frac{n_{i,j}}{n_{i..}} \log_2 \frac{n_{i,j}}{n_{i..}} \tag{4}$$

obtaining the Gain Ratio

$$Gain_R(class|test) = \frac{I(test) - E(class|test)}{IV(class)} \tag{5}$$

According to Quinlan

the rationale behind this is that as much as possible of the information provided by determining the value of an attribute should be useful for classification purpose.

2.3 Minimum encoding styles

Minimum encoding approaches were developed for fitting models to data problems. The problem of finding a good model is converted to a problem of finding minimum encoding of the data, using concepts from Shannons theory of information. Minimum encoding styles can also be considered as extensions of decision trees, which may result in decision graphs. They are based on minimum description length principle and minimum message length principle (MDL/MML). These principles use "encoding length" to measure the quality of hypotheses. We examined three styles as defined and implemented by the IND program: strict MML (**SMML**) that is closely related to the theory, a modified approach **MML** which does not penalize large tree as strongly, and **Bayesian trees**. The theoretical background is described in [5],[6], and their application in ASR diagnosis task can be found in [7].

3 ASR experiments

3.1 OLLO database

The speech database used for ASR experiments is the Oldenburg Logatome Corpus (OLLO) [8]. It contains 150 different non-sense utterances (logatomes) spoken by 40 German and 10 French speakers. Each logatome consists of a combination of consonant-vowel-consonant (CVC) or vowel-consonant-vowel (VCV) with the outer phonemes being identical.

A large drop in recognition accuracy in ASR is not only encountered in noisy environments, but even when clean speech in known conditions is to be recognized. This drop is often caused by speech intrinsic variabilities (as for example speaking rate, style or effort, speaker's age, gender or health condition, regional dialect or accent).

To provide an insight into the influence of speech intrinsic variabilities on speech recognition, OLLO covers several variabilities such as speaking rate and effort, dialect, accent and speaking style (statement and question). The OLLO database is freely available at <http://sirius.physik.uni-oldenburg.de>.

Each of the 150 logatomes was recorded in six variabilities (speaking rate: fast and slow; speaking effort: loud and soft; speaking style: spoken as question and normal) with three repetitions. This results in 2,700 logatomes per speaker. Influences caused by dialect may be investigated, as speakers without dialect and from four different dialect/accents were recorded. Utterances of ten speakers with no accented speech (five speakers for training and five speakers for testing) were selected for ASR tests.

3.2 Automatic speech recognition test setup

Hidden Markov Models (HMM) and Gaussian Mixture Models (GMM) based speech recognition system is trained using public domain machine-learning library TORCH on the training set that consists of 13446 logatome utterances.

Three states left-right HMM models were trained for each of the 26 phonemes in the OLLO database including silence as well. Gaussian mixture models with 17 Gaussians per state and diagonal covariance matrices were used to model the emission probability densities of the 39 dimensional feature vectors - 13 cepstral coefficients and their derivatives (Δs) and double derivatives ($\Delta\Delta s$). The phoneme HMMs are connected with no skip. We extended the TORCH library in a package of calculation and storage of feature data, necessary for further statistic processing. The decoder collects the feature data by running on the testing set that consists of 13466 logatome utterances. We trained and tested the ASR system with MFCC feature set. All the features were calculated using HTK hcopy tool. We calculated MFCC vectors every 10 msec using windows of size 25 msec. Average phoneme recognition performance of the ASR systems on this task was 76.49 % (see Tab. 1). Similar performance can be achieved also by the HTK tool, but we preferred using TORCH as it is much easier to adapt speech recognition process.

Accuracy	Deletions	Insertions	Substitutions
76.49 %	3.60 %	3.69 %	16.23 %

Table 1. Accuracy of ASR system on logatom recognition task with no grammar.

4 Experiments and comparisons

More than 13K logatomemes were used for AM training, and the same amount of different data from a test set was used for ASR testing. The test set was further split 90% for development (DT training), and 10% for evaluation.

We implemented new measurer within the TORCH library (called at the end of each recognized utterance/logatom), witch dumped all the data used later in DT analysis. Specifically, following vector along with a single recognition was created (the first item was a classified item, others were predictors' values):

- Error category (the classified item)
 - Deletion (values 0,1; 1 for deletion done)
 - Insertion (values 0,1; 1 for insertion done)
 - Substitution (values 0,1; 1 for substitution done)
- Speech variability:
 - V1: Fast speech
 - V2: Slow speech
 - V3: Loud speech
 - V4: Soft, low (not whispered) speech
 - V5: Questioning style speech
 - V6: Normal speech

- Logatom type (VCV or CVC)
- Speaker ID (from S06 to S10)
- Speaker gender (M,F)
- Speaker age (ranges 21-31 and 32-42 years old)

CART style trees were generated using several splitting rules (see Sec. 2.1), subsetting on multivalued discrete variables, cost-complexity pruning and 10-fold cross validation [9]. **C4.5** style trees were generated using Quinlan’s gain ratio splitting rule (see Sec. 2.2), and Quinlan’s pruning rule [4]. The depth of the inducted trees was not limited. For each of the error category, the following decision trees were trained and examined:

1. C4 style using Quinlan’s gain ratio splitting rule
2. CART style using Gini index of diversity
3. CART style using information gain (entropy) splitting rule
4. CART style using ”twoing”
5. a Bayesian tree
6. a MML tree
7. and a strict MML tree

The IND program [1] was used for DT training and testing. For each DT style we trained three DTs, for deletions, insertions and substitutions, 7 x 3 trees in total. For each of the tree misclassification matrix was calculated. Tab 2 shows an example of a strict MML matrix. Our criterion for best tree selection was minimal misclassification rate of classified error made by ASR ([2,2] elements of the matrices). Having the best tree, we examined a path leading to the most probable terminal node with an error made event classified.

	Actual not substitution	Actual substitution	Total
Predicted not substitution	0.537147	0.312036	0.849183
Predicted substitution	0.060178	0.090639	0.150817
	0.597325	0.090639	1.000000

Table 2. Misclassification matrix of the SMML style DT for substitutions.

Table 3 presents the results we got from the trees. The order of found predictors on the path to the most probably error classification is not important.

4.1 Deletions

For deletions, the best tree was CART style using information gain (entropy) splitting rule. The clear predictor for deletions made was specified as fast speech. All the rest of intrinsic speech variabilities were contributing less to this error category. The second major predictor selected was speaker specification (S06), what were male recordings, all speakers in age range of 32-42 years old. The second minor predictor were the rest of speakers, what were mostly (3/4) female recordings, all in age range 21-31 years old.

Error category	Selected DT	Selected predictors
Deletions	CART using entropy	Fast speech and S06 speaker (more probable) Fast speech and rest of spkrs (less probable)
Insertions	Strict MML	VCV logatoms Slow speech (more probable) Loud speech and female spkrs (less probable) Questioning style and female speakers
Substitutions	Strict MML	Male (or age 32-42), CVC type, soft speech Female (or age 21-31), CVC, loud and soft

Table 3. Results for each of the error category.

4.2 Insertions

For insertions and substitutions, strict MML DTs were best trees. Analyzing insertions, primary error predictors were selected as VCV type of logatoms and slow speech. The minor predictors were selected as loud and questioning style speech together with female recordings.

4.3 Substitutions

The hardest specification of predictors was in case of an analysis of substitutions. There were no clear patterns, also overall classification error of the trees were less than for DTs for insertions and deletions. Anyway, CVC type of logatoms and soft speech were in most cases dominant predictors of substitution prediction.

5 Conclusions and future work

We contributed to the topic of ASR evaluation, focusing on classical error categories and intrinsic speech variabilities. We specified predictors for different DT styles which contribute to the each error category, which is a novel approach in ASR diagnosis.

Often the true power of a predictive model comes from insightful predictor creation. Subject-matter expertise is critical. In future we want to look for some data-driven mechanism to decrease our dependency on such the expertise. However, until that time, we will look for new powerful predictors of ASR errors.

6 Acknowledgments

The work has been funded with support from the European Commission, project Euronounce (URL: <http://www.euronounce.net>), and with support from the VEGA grant No. 2/0138/08. This publication reflects the views only of the authors, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

References

1. W. Buntine: Tree classification software. Technology 2002: The Third National Technology Conference and Exposition, Baltimore, USA, December 1992.
2. G. Zhou—M. E. Deisher—S. Sharma: Causal analysis of speech recognition failure in adverse environments. In ICASSP '02, volume 4, pages 3816–3819, 2002.
3. M. Hunt: Speech recognition, syllabification and statistical phonetics. In Proc. of ICSLP, Jeju Island, Korea, October 2004.
4. J. R. Quinlan: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, CA, USA, 1993.
5. C. S. Wallace: Statistical and Inductive Inference by Minimum Message Length. Springer-Verlag, Information Sci. & Stats., May 2005.
6. W. Buntine: A Theory of Learning Classification Rules. PhD thesis, University of Technology, Sydney, 1991.
7. M. Cernak—S. Darjaa: Noisy Speech Recognition Failure Diagnosis Using Minimum Message Length Decision Trees. In Proceedings of the 15th International Conference on Systems, Signals and Image Processing (IWSSIP), June 25-28, Bratislava, Slovak Republic, 2008.
8. T. Wesker—B. Meyer—K. Wagener—J. Anemuller—A. Mertins—B. Kollmeier: Oldenburg logatome speech corpus (OLLO) for speech recognition experiments with humans and machines. In Interspeech 2005, pages 1273–1276, September 2005.
9. L. Breiman—J. Friedman—Ch J. Stone—R. A. Olshen: Classification and Regression Trees. Chapman & Hall/CRC, New York, 1983.