DIRECTED GRAPHICAL MODELS OF CLASSIFIER COMBINATION: APPLICATION TO PHONE RECOGNITION

Jeff A. Bilmes <bilmes@ee.washington.edu>
Katrin Kirchhoff <katrin@ee.washington.edu>
SSLI-Laboratory, Dept. of Electrical Engineering, University of Washington
EE/CS Bldg, Box 352500, Seattle, WA 98195, USA

ABSTRACT

Classifier combination is a technique that often provides appreciable accuracy gains. In this paper, we argue that the underlying statistical model of classifier combination should be made explicit. Using directed graphical models (DGMs), we provide representations of two common combination schemes, the mean and product rules. We also introduce new DGMs that yield novel combination rules. We find that these new DGM-inspired rules can achieve significant accuracy gains on the TIMIT phone-classification task relative to existing combination schemes.

1. INTRODUCTION

When multiple independently trained pattern classifiers are combined, the resulting accuracy is often better than any of the individual classifiers. This has been demonstrated for automatic speech recognition (ASR) [7, 10, 18] and for pattern classification [12, 13, 20, 29].

Classifier combination can fuse together different information sources to utilize their complementary information. The sources can be multi-modal, such as speech and vision, but can also be transformations [18] or (e.g., spectral) partitions [5, 25, 24] of the same signal. In each case, combination can produce appreciable gains, even when individual classifiers exhibit widely varying accuracies.

Combination rules often operate directly on classifier probabilities. One method (the mean rule) computes a weighted average of classifier outputs. Another method (the product rule) multiplies and then renormalizes these probabilities. Other techniques compute the maximum, minimum, or median of the classifier outputs [20]. Other methodologies [27, 19] jointly train separate classifiers which are combined in various ways. In ASR, classifier combination can occur at different levels including the feature stream [2, 16, 10, 17], the HMM state [18], or at higher levels such as at the syllable [31] or sentence [7].

It is often said that classifiers should be combined if they are different. Classification is related to regression, where several theoretical studies[13, 4] have shown that mean-rule combination is successful (a lower mean-squared error) when the errors of each system are independent. In this case, error reduction is related to ensemble bias (the degree to which the averaged ensemble output diverges from the true target function) and variance (the degree to which the ensemble members disagree) [21, 29, 4]. Generally, a low error requires both a low bias and variance, but since variance is reduced by averaging, it is sufficient to combine classifiers with low bias.

When working with probabilistic decision-making systems, it is usually advantageous to explicitly state the assumed underlying statistical model. For example, a hidden Markov model, easily defined by its conditional independent properties [3], is often used to represent speech for ASR. In a mixture model, it is assumed that a hidden and unknown cause selects each mixture component.

A model may also be used to represent classifier combination. Explicating the models leading to a given combination rule could provide insight about when that rule best applies. If a model is found that matches the data well, a combination rule derivable from the model, also matching the data, can be selected. Alternatively, a given model can be improved by relaxing the most glaring simplifying assumptions, thereby leading to new combination schemes.

This paper investigates directed graphical models (DGMs) for the classifier combination problem. A useful way to understand a given model, and to measure how well it matches data, is to make explicit all of its conditional independence properties. DGMs are a type of graphical model [22] where these properties may be visualized. This makes it easy to experiment with different models and produce novel combination schemes. The paper provides DGMs for two popular schemes (the sum and product rules) and evaluates new combination rules resulting from novel models of combination.

Section 2 reviews conditional independence and DGMs. Section 3 provides DGMs for the mean and product rules. Section 4 considers new DGMs and combination rules for classifier combination. Section 5 evaluates these rules on the TIMIT phone-recognition task. Finally, Section 6 concludes and discusses future work.

Notation: In this paper, capital letters (X) represent random variables and lower case letters (x) refer to their possible values. $p_X(X = x) = p(X = x) = p(x)$ is the probability of the event $X = x$. Note that X can be a vector random variable. We will use matlab-like notation to refer to ranges, so $X_{1:t}$ refers to the set $\{X_1, X_2, \ldots, X_t\}$. If there are T random variables, and $A \subseteq 1{:}T$, then $X_A \subseteq X_{1:T}$ is the subset of the random variables $X_{1:T}$ indexed by elements within A.

2. CONDITIONAL INDEPENDENCE (CI) AND DIRECTED GRAPHICAL MODELS (DGMS)

A random variable X is conditionally independent (CI) of a different random variable Y given a third random variable Z under a given probability distribution $p(\cdot)$, if the following relation holds:

$$p(x, y|z) = p(x|z)p(y|z)$$

for all x, y, and z. This is written $X \perp\!\!\!\perp Y | Z$. Many properties of CI are given in [22, 26]. CI is a powerful property — when CI assumptions are made, a model might undergo enormous simplifications.

Directed graphical models (DGMs) [14, 26] are one type of

graphical model (GM) [22] where the graph is directed and acyclic. A GM specifies a family of statistical models and a set of computationally efficient algorithms for decision making. A particular graphical model is associated with a collection of random variables and a set of probability distributions over that collection. A GM's edges in one way or another specifies a set of conditional independence properties that are true under all the members of the associated family.

Nodes in a graphical model can be either *hidden*, which means they have an unknown value, or they can be *observed*, which means that the values are known. Certain types of GMs allow the dependencies to switch [9], and are often called multinets. In this case, the edges in a GM can change as a function of other variables in the network.

There are several equivalent schemas that can formally define the CI relationships implied by a DGM. One simple method (equivalent to d-separation [14, 22] and described in Figure 1) is called the Bayes-ball procedure [28].

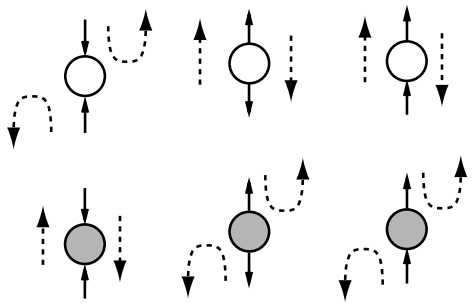

Figure 1: The Bayes-ball procedure makes it easy to answer questions about a DGM such as "is $X_A \perp\!\!\!\perp X_B | X_C$?", where A, B, and C are disjoint sets of node indices. First, shade all nodes having indices in C and imagine a ball bouncing from node to node along the edges in a graph. The answer to the above query is TRUE if and only if a ball starting at some node in A can reach a node in B, when the ball bounces according to the rules depicted in the figure. The dashed arrows depict whether a ball, when attempting to bounce through a given node, may bounce through that node or if it must bounce back.

When using a DGM to represent a physical process, it is important for the DGM to represent those properties needed to solve a given task (such as prediction or classification). A mismatch can occur in a variety of ways. For example, the model's CI properties might not be reflected by the data. Alternatively, the CI properties might be correct, but the implementations could be wrong (e.g., representing a non-linear dependence using only linear regression).

A DGM can also represent classifier combination. Given the right model, one need not assume that errors are independent since error interdependency can be modeled. Once a model is specified, a combination rule, correct with respect to the model, can be derived using the associated CI properties. Since it is possible to check model accuracy with respect to the data, it is possible also to check a corresponding combination scheme. While we do not measure model accuracy in this paper, we investigate several new models and their associated combination strategies.

3. MODELS FOR SUM AND PRODUCT RULES

In this section, we examine DGMs that can lead to the mean and the product rule. Consider the left DGM in Figure 2, where C is a class variable, $X_{1:N}$ is a feature vector, and H is a hidden discrete random variable. Under this model, the following is a valid expansion:

$$p(c|x) = \sum_h p(c, h|x) = \sum_h p(c|x, h)p(h|x)$$

As is well known, this is a mixture of experts [15] but several additional simplifications can be made. If the edge from $X_{1:N}$ to H is removed, this yields $p(c|x) = \sum_h p(c|x, h)p(h)$ which is the weighted mean rule. If H is uniformly distributed, then $p(c|x) = \frac{1}{N}\sum_h p(c|x, h)$ which is just the the average of each classifier. If it is further assumed that $C \perp\!\!\!\perp X_{A_h} | \{H = h\}$, then for appropriate A_h, the rule becomes $p(c|x) = \frac{1}{N}\sum_h p(c|x_{A_h^c}, h)$ where $x_{A_h^c}$ is a subset of features.[1] This last assumption is that certain features are conditionally independent of C for a particular assignment to the hidden variable H. The result is a rule that can combine heterogeneous feature vectors.

Mean rules are useful for combining uni-modal distributions into a single multi-modal distribution. Since mixing increases entropy [6], such a procedure is poor for representing low-entropy distributions where probability is concentrated in narrow input-space regions. In such cases, the product rule is useful, where each classifier must supply probability to the correct class, but may also supply probability to incorrect classes as long as one or more of the other classifiers do not supply probability to those incorrect classes. These are therefore called "AND" style combination schemes [18] since only the logical AND of each classifier's probabilistic decision will survive combination. It is also the case that such a combination scheme is useful when the underlying distributions factorize over the probabilistic space of C [11].

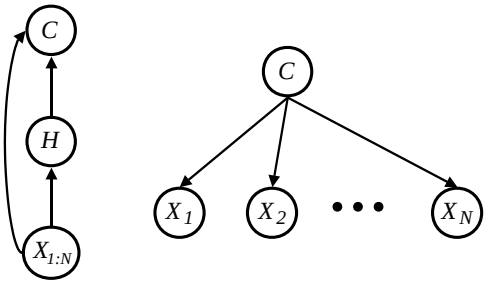

Figure 2: Models for Sum and Product Rules.

The product rule can be derived using the right DGM in Figure 2. This is the graph for a naive Bayes classifier [8], which states that features are independent given the class ($X_{i:j} \perp\!\!\!\perp X_{l:m} | C$ for all i, j, l, m such that $i{:}j \cap l{:}m = \emptyset$). Given this this graph, the product rule may be derived as follows:

$$\begin{aligned} p(c|x_{1:N}) &= \frac{p(x_{1:N}|c)p(c)}{\sum_{c'} p(x_{1:N}|c')p(c')} \\ &= \frac{\prod_i p(x_i|c)p(c)\left(\frac{p(c)^{N-1}/\prod_i p(x_i)}{p(c)^{N-1}/\prod_i p(x_i)}\right)}{\sum_{c'}\left(\prod_j p(x_j|c')p(c')\right)\left(\frac{p(c')}{p(c')}\right)^{N-1}} \\ &= \frac{\prod_i p(c|x_i)}{\sum_{c'}\left(\frac{p(c)}{p(c')}\right)^{N-1}\prod_j p(c'|x_j)} \end{aligned}$$

This rule has been successful for HMM state combination in ASR [16, 10, 18]. This appears surprising since the corresponding assumptions are certainly not true — e.g., neither different feature

[1]The notation A_h^c means the complement of the set A_h and can be defined as $A_h^c \triangleq 1{:}N \setminus A_h$.

representations derived from nor different spectral sub-bands of the same signal are CI given the class [1]. On the other hand, producing low-entropy distributions over HMM states from a product of sometimes incorrect classifiers might outweigh this inaccuracy. Alternatively, as argued in [3], an assumption that is incorrect for predictive accuracy does not ensure discriminative inaccuracy.

4. NEW DGM COMBINATION MODELS AND THEIR RULES

It is possible to model classifier combination using distinct hidden random variables for each classifier output and the target. A specific form of this has been called stacked generalization [30], where a random relationship is learned between classifier outputs and the true target. In general, different DGMs over these random variables lead to distinct and potentially novel combination schemes. Interestingly, we find that certain relatively simple schemes do not require some of the seemingly egregious independence assumptions such as feature independence.

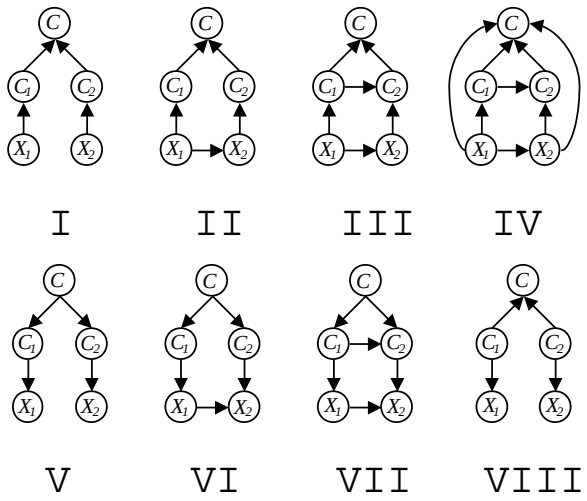

Figure 3: Eight (out of many) possible models for combination.

Figure 3 provides eight out of many possible models for two-classifier combination. The differences between them are the edges, and therefore the CI assumptions. Each model uses five random variables, C, the target class, C_1, C_2, hidden variables indicating the classifier outputs, and X_1, and X_2, different input features. These models can be generalized to N classifiers, but the number of possible rules considerably grows in that case.

Under all models, a valid expansion is:

$$p(c|x_1, x_2) = \sum_{c_1,c_2} p(c|c_1, c_2, x_1, x_2)p(c_1, c_2|x_1, x_2).$$

Other than the obvious chain-rule expansions, each model allows for further simplifications to be made about the quantities $p(c|c_1, c_2, x_1, x_2)$ and $p(c_1, c_2|x_1, x_2)$, as described below.

Under Model I, implications include $X_1 \perp\!\!\!\perp X_2$ and $C_1 \perp\!\!\!\perp C_2$ but it is *not* implied that $X_1 \perp\!\!\!\perp X_2|C$. Also, since $C \perp\!\!\!\perp \{X_1, X_2\}|\{C_1, C_2\}$, $C_1 \perp\!\!\!\perp \{C_2, X_2\}|X_1$, and $C_2 \perp\!\!\!\perp X_1$, the combination rule becomes

$$p(c|x_1, x_2) = \sum_{c_1,c_2} p(c|c_1, c_2)p(c_1|x_1)p(c_2|x_2) \quad (1)$$

Note that $p(c|c_1, c_2)$ could be represented either using a neural network, or using a 3-dimensional table "trained" by counting from the list of classifier outputs and targets for each feature.

For model II, it is neither the case that $X_1 \perp\!\!\!\perp X_2$ nor that $C_1 \perp\!\!\!\perp C_2$ (use Bayes-ball to see this). It is still true that $C \perp\!\!\!\perp \{X_1, X_2\}|\{C_1, C_2\}$, and $C_1 \perp\!\!\!\perp \{C_2, X_2\}|X_1$, and now $C_2 \perp\!\!\!\perp X_1|X_2$. Interestingly, this model results in the same combination scheme as case I even though in case II the features are not assumed to be independent. Also, this rule still requires a 3D table for $p(c|c_1, c_2)$

Under model III, it is no longer the case that $C_1 \perp\!\!\!\perp C_2|X_1$ but since $C_2 \perp\!\!\!\perp X_1|\{C_1, X_2\}$ and $C_1 \perp\!\!\!\perp X_2|X_1$,

$$p(c|x_1, x_2) = \sum_{c_1,c_2} p(c|c_1, c_2)p(c_2|c_1, x_2)p(c_1|x_1). \quad (2)$$

The output of the first classifier must be given as input to the second classifier. An potential benefit of this scheme is that the second classifier might detect and then correct mistakes made by the first classifier. Again, $p(c|c_1, c_2)$ requires a 3D table.

In model IV, it is no longer true that $C \perp\!\!\!\perp \{X_1, X_2\}|\{C_1, C_2\}$ leading to

$$p(c|x_1, x_2) = \sum_{c_1,c_2} p(c|c_1, c_2, x_1, x_2)p(c_2|c_1, x_2)p(c_1|x_1).$$

Model V reverses arrow directions and can be seen as generative model of $X_{1:2}$ — one first generates C according to $p(c)$, then produces noisy versions C_1 and C_2 of the class variable, and each of those produce X_1 and X_2 respectively. In this model, $\{C_1, X_1\} \perp\!\!\!\perp \{C_2, X_2\}|C$ and $C_1 \perp\!\!\!\perp X_2|C_2$ but it is not the case that $C_2 \perp\!\!\!\perp X_1|X_2$. This leads to the rule

$$p(c|x_1, x_2) = \sum_{c_1,c_2} p(c|c_1, c_2)p(c_1|c_2, x_1)p(c_2|x_1, x_2). \quad (3)$$

The first classifier uses the first feature stream and the output of the second classifier which uses both features. Under this rule, one need not use a 3D table for $p(c|c_1, c_2)$ since $C_1 \perp\!\!\!\perp C_2|C$. Bayes rule yields $p(c|c_1, c_2) = p(c_1, c_2|c)p(c)/p(c_1, c_2)$ which also equals $p(c_1|c)p(c_2|c)p(c)/p(c_1, c_2)$. It is therefore possible to represent $p(c|c_1, c_2)$ using three 2-dimensional and one 1-dimensional tables, significantly reducing parameters.

Under Model VI, it is no longer the case that $X_1 \perp\!\!\!\perp X_2|C$ but it is still true that $C_1 \perp\!\!\!\perp C_2|C$, so 2-dimensional tables can be used to represent $p(c|c_1, c_2)$. This can therefore lead to the same rule as in case V. Model VII can produce the same rule as does model VI, but requires a 3-dimensional table for $p(c|c_1, c_2)$. Finally, model VIII results in the came rule as does models I and II.

In general, different models can lead to exactly the same combination rule — the rule is valid with respect to multiple statistical models. It is therefore not always the case that removing CI assumptions leads to more powerful rules or that adding CI assumptions leads to simplifications. Since an edge in a DGM does not necessarily imply a dependency (only a lack of an edge implies a conditional independency), it can be seen that a single model might lead to more than one combination rule.

5. EXPERIMENTAL RESULTS

In this section, we empirically evaluate some of the rules given in the preceding sections. We use the TIMIT speech corpus (the

MFCC	LPC	Sum	Prod	Min	Max
69.56%	66.92%	70.43%	70.49%	70.43%	70.13%

Table 1: Results for MFCC and LPC base systems, product, sum, min, and max rule combination.

I (tbl)	I (MLP)	III(tbl)	III(MLP)
69.90%	70.25%	72.93%	72.46%

Table 2: Results for DGM combination schemes.

standard training and core test set); the accuracy rates we report are for frame-level phone classification (as opposed to phone recognition). The two base classifiers which we combine using a combination rule are three-layer Multi-Layer-Perceptrons (MLP) trained using MFCC or LPC input feature representations. Both feature streams consist of 12 basic coefficients, energy and first derivatives, resulting in 26 input features. The MLPs use a context window of nine frames. In each case, the number of hidden units in the classifiers were adjusted to equalize the number of parameters between the different cases. Combination using $p(c|c_1, c_2)$ is implemented either using discrete probability tables (tbl), or when appropriate by another MLP.

Table 1 shows the baseline performance and the results of sum and product rule combination and for comparison also provides results using the min and max rules [20]. The accuracy rates obtained by the DGM schemes are shown in Table 2.

Not unexpectedly, we find that performance improves as independence assumptions are relaxed (from I to III). More interestingly, we find that no improvement is found over the sum and product rules when individual classifier outputs are independent (i.e., $C_1 \perp\!\!\!\perp C_2|X_1$) but significant improvements are found when this assumption no longer holds. Moreover, we observe that using an MLP instead of a conditional probability table has a negligible affect on performance.

Overall, the best scheme attempted (model III) achieves a statistically significant improvement (at the $p < 0.0001$ level using a difference of proportions test) both over our baseline systems and over the product and sum schemes.

6. DISCUSSION

This paper argues that 1) since classifier combination is a statistical process, the underlying assumed statistical model should be precisely stated, and that 2) directed graphical models (DGMs) are a rich and flexible language which can be used to reason about different classifier combination schemes. When deciding from among a collection of combination rules, one can consider the corresponding set of underlying statistical models. By choosing the model most accurately reflected by the data, one can select an correspondingly appropriate combination rule. Selecting a combination rule can therefore be seen as a model selection [23] procedure.

It has been shown that multiple different models might lead to exactly the same combination rule, with shared rule being valid with respect to multiple models. Also, a single model can lead to more than one valid combination rule. In the later case, the simplest model could be chosen. This paper listed the models for the product and sum rules, and presented some novel models and rules, some of which have lead to appreciable accuracy gains relative to the product or sum rules for the TIMIT phone classification task.

In the future, we will evaluate the other models for combination, and develop and evaluate ways to automatically check combination model accuracy, and thereby select an appropriate rule. We also intend to evaluate these new rules in large vocabulary speech recognition tasks.

REFERENCES

[1] J. Bilmes. *Natural Statistic Models for Automatic Speech Recognition*. PhD thesis, U.C. Berkeley, Dept. of EECS, CS Division, 1999.

[2] J. Bilmes, N. Morgan, S.-L. Wu, and H. Bourlard. Stochastic perceptual speech models with durational dependence. *Intl. Conference on Spoken Language Processing*, November 1996.

[3] J.A. Bilmes. Dynamic Bayesian Multinets. In *Proceedings of the 16th conf. on Uncertainty in Artificial Intelligence*. Morgan Kaufmann, 2000.

[4] C. Bishop. *Neural Networks for Pattern Recognition*. Clarendon Press, Oxford, 1995.

[5] Herve Bourlard and Stephane Dupont. A new ASR approach based on independent processing and recombination of partial frequency bands. In *Proceedings ICSLP*, volume I, pages 426–427, 1996.

[6] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. Wiley, 1991.

[7] J.G. Fiscus. A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997.

[8] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29:131–163, 1997.

[9] D. Geiger and D. Heckerman. Knowledge representation and inference in similarity networks and Bayesian multinets. *Artificial Intelligence*, 82:45–74, 1996.

[10] A.K. Halberstadt and J.R. Glass. Heterogeneous measurements and multiple classifiers for speech recognition. *Proceedings ICSLP-98*, pages 995–998, 1998.

[11] G. E. Hinton. Training products of experts by minimizing contrastive divergence. Technical report, Gatsby Computational Neuroscience Unit, 2000.

[12] Tin Kam Ho, Jonathan J. Hull, and Sargur N. Srihari. Decision combination in multiple classifier systems. *PAMI*, 16(1):66–75, January 1994.

[13] R.A. Jacobs. Methods for combining experts' probability assessments. *Neural Computation*, 7:867–888, 1995.

[14] F.V. Jensen. *An Introduction to Bayesian Networks*. Springer, 1996.

[15] M.I. Jordan and R. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

[16] B.E.D. Kingsbury and N. Morgan. Recognizing reverberant speech with RASTA-PLP. *Proceedings ICASSP-97*, 1997.

[17] K. Kirchhoff. Combining acoustic and articulatory information for speech recognition in noisy and reverberant environments. In *Proceedings of the International Conference on Spoken Language Processing*, 1998.

[18] K. Kirchhoff and J. Bilmes. Dynamic classifier combination in hybrid speech recognition systems using utterance-level confidence values. *Proceedings ICASSP-99*, pages 693–696, 1999.

[19] K. Kirchhoff and J. Bilmes. Generalized acoustic classifier combination for speech recognition. In *Proc. of the Automatic Speech Recognition (ASR) Workshop*, Paris, FR, October 2000.

[20] J. Kittler, M. Hataf, R.P.W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[21] A. Krogh and J. Vedelsby. Neural network ensembles, cross validation, and active learning. In *Advances in Neural Information Processing Systems 7*. MIT Press, 1995.

[22] S.L. Lauritzen. *Graphical Models*. Oxford Science Publications, 1996.

[23] H. Linhart and W. Zucchini. *Model Selection*. Wiley, 1986.

[24] P. McMahon, P. Court, and S Vaseghi. Discriminative weighting of multi-resolution sub-band cepstral features for speech recognition. *Proceedings ICSLP-98*, pages 1055–1058, 1998.

[25] N. Mirghafori and N. Morgan. Combining connectionist multi-band and full-band probability streams for speech recognition of natural numbers. In *Proceedings of the International Conference on Spoken Language Processing*, pages 743–746, 1998.

[26] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 2nd printing edition, 1988.

[27] J. R. Quinlan. Bagging, boosting, and C4.5. In *Proc. 14th National Conf. on AI*, 1996.

[28] R.D. Shachter. Bayes-ball: The rational pastime for determining irrelevance and requisite information in belief networks and influence diagrams. In *Uncertainty in Artificial Intelligence*, 1998.

[29] A.J.C. Sharkey. Multi-net systems. In *Combining Artificial Neural Nets: Ensemble and Modular Multi-Net Systems*, pages 3–30. Springer, 1999.

[30] D.H. Wolpert. Stacked generalization. *Neural Networks*, 5:241–259, 1992.

[31] Su-Lin Wu, Michael L. Shire, Steven Greenberg, and Nelson Morgan. Integrating syllable boundary information into speech recognition. In *Proc. IEEE Intl. Conf. on Acoustics, Speech, and Signal Processing*, volume 1, Munich, Germany, April 1997. IEEE.