A SIGN-TO-SPEECH TRANSLATION SYSTEM USING MATCHER NEURAL NETWORKS

PETER Z. REVESZ RAGHAVA-RAO K. VEERA

Department of Computer Science and Engineering
University of Nebraska–Lincoln

Abstract

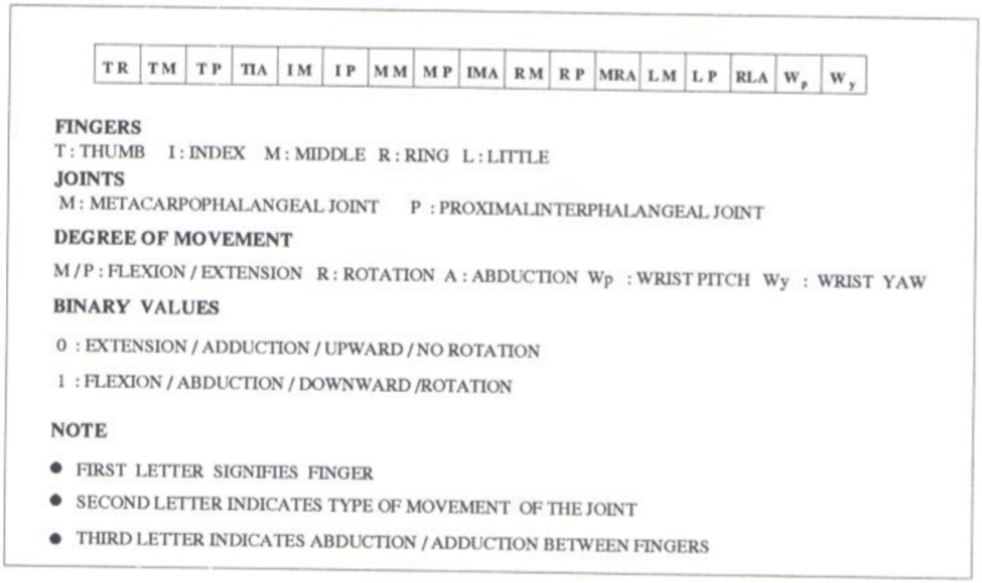
Sign language translation is an interesting but difficult problem for which neural network techniques seem promising because of their ability to adjust to the user's hand movements, which is not possible to do by most other techniques. However, even using neural networks and artificial sign languages, the translation is hard, and the best-known system, that of Fels & Hinton (1993), is capable of translating only 66 root signs and their conjugations. This paper improves their results to 395 root signs while preserving a high accuracy (i.e., over 96 %) in translation. The use of matcher neural networks (Revesz 1989, 1990) and asymmetric Hamming distances are the key sources of improvement.

INTRODUCTION

Sign languages have been used for a long time. Sign language was a common means of communication throughout the Great Plains among the various American Indian tribes who spoke different vocal languages. Today sign languages form a very important group of languages, with over a hundred thousand people worldwide who use them daily as their primary mode of communication. American Sign Language (ASL) is a complete and well-formed language developed naturally within the American deaf community. Its grammar is quite distinct from that of English or any other spoken language (Wilbur 1979).

While there have been many studies on ASL (Poizner et al. 1987, Stungis 1981, Loomis et al. 1983, Kramer & Leifer 1989) including some work towards an automatic translation to English, it turns out to be a very complex language to study directly. Instead of attacking the problem of ASL head on, some authors considered the more manageable problem of translating various artificial sign languages to English. The best results to date are those of Fels and Hinton (1993) who achieved a robust, i.e. over 90% accurate, sign-to-word translation of an artificial sign language to English using neural networks. Their study however leaves much room for improvement. They achieved translation of only 66 root signs plus their conjugations. This falls short of an average English vocabulary, which has over 20,000 words, and even of the basic English vocabulary, which has about 850 root words (Ogden 1968).

In this paper we improve previous translation works using artificial sign languages. In particular we design an interface which robustly translates 395 signs to spoken English words and which is capable of adapting to different users with different hand structures. Our test results show that the accuracy of our translation system remains quite high, i.e. 96%. This can be considered a very high accuracy, considering the complexity that the system has to deal with in providing an adaptive interface between



TR	TM	TP	TIA	IM	IP	MM	MP	IMA	RM	RP	MRA	LM	LP	RLA	W_p	W_y

FINGERS

T : THUMB I : INDEX M : MIDDLE R : RING L : LITTLE

JOINTS

M : METACARPOPHALANGEAL JOINT P : PROXIMALINTERPHALANGEAL JOINT

DEGREE OF MOVEMENT

M / P : FLEXION / EXTENSION R : ROTATION A : ABDUCTION Wp : WRIST PITCH Wy : WRIST YAW

BINARY VALUES

0 : EXTENSION / ADDUCTION / UPWARD / NO ROTATION

1 : FLEXION / ABDUCTION / DOWNWARD /ROTATION

NOTE

- FIRST LETTER SIGNIFIES FINGER
- SECOND LETTER INDICATES TYPE OF MOVEMENT OF THE JOINT
- THIRD LETTER INDICATES ABDUCTION / ADDUCTION BETWEEN FINGERS

Figure 1: A 17-joint feature list representation of hand signs

a user and a text-to-speech synthesizer. To provide such an interface we use neural networks as the kernel of the translation system. The particular neural networks that we use are adapted from Revesz (1989, 1990) and are called matcher neural networks.

Our translation system consists of a CyberGlove monitoring the user's hands, a computer that uses neural networks to analyze the CyberGlove inputs and identifies the signs made and associates them with English words, and finally a DECtalk text-to-speech synthesizer, which reads out the input words. In this paper we will describe the novel aspects of this system, concentrating on the sign language itself, the neural networks and their interactions.

BASIC CONCEPTS

Feature Lists

In this paper we will use a special type of feature lists to describe signs. More specifically, we will use joint feature lists. That is, we take the feature list of each sign to be a pattern of 0's and 1's where at each position a 1 signifies the bending or a 0 signifies the straightening of a certain joint.

The human hand is a complex structure with several kinds of bones that are interconnected by various joints having different degrees of freedom. For our sign language we considered 17 different joint movements which are listed in Figure 1 and described below. We assume some basic knowledge of hand anatomy such as that found in April (1990).

Altogether we consider 17 joint movements. Five of these are the flexion or extension of the MCPs (Metacarpophalangeal Joints), five the flexion or extension of PIPs (Proximal Interphalangeal Joints), four the abduction or adduction between adjacent MCPs, one for thumb rotation, and one each for wrist pitch and wrist yaw.

All of these 17 joint movements are sensed by the CyberGlove. The CyberGlove is a glove with very fine sensors that are located over or near the critical joints. The

SIGNS	**FEATURES**				
	TP	**IP**	**MP**	**RP**	**LP**
IS	1	1	1	1	1
A	1	1	1	0	0
THERE	0	0	1	1	1
SHE	0	1	0	1	0
HE	1	0	1	0	1
WOMAN	0	0	1	0	0

Figure 2: A simple example using PIP feature lists

CyberGlove sensors give digitized output values that vary linearly with the angle of the joints over which they are located. Therefore these sensors can easily monitor the movement of the joints described above.

We are not using the angle values directly as provided by the CyberGlove, but we convert each angle value into a "1" if the joint is bent or into a "0" if the joint is straightened. Figure 2 shows a simplified example of joint feature lists considering only the five PIP joints. There the representations of the signs for "is", "a", "there", "she", "he", and "woman" are listed. For example, the word "there" is represented by the pattern 00111 which signifies that while making the sign for this word we have to straighten the thumb and the index PIP joints and bend the other three.

Asymmetric Hamming Distance

Joint feature lists are the primary means of distinguishing between different hand signs. Intuitively we want the feature lists to be as different as possible so that minor errors of one or two bits do not lead to confusion among different signs.

There are several ways to measure the difference between two feature lists. The Hamming distance is widely used when the probabilities of 1 getting corrupted to 0 and 0 getting corrupted to 1 are equal. These two types of errors are often referred to as omission and addition errors respectively. However, in the CyberGlove environment the omissions errors are much more frequent. Hence instead of the Hamming distance, we will use an asymmetric Hamming distance measure. The asymmetric Hamming distance between two feature lists A and B is the maximum of the number of omissions from A or the number of omissions from B that must occur for the two patterns to become indistinguishable. For example, the asymmetric Hamming distance between the signs "there" and "she" in Figure 2 is two. This is because 00010 can be obtained from 00111 by two omissions and from 01010 by one omission.

Let $\omega(P)$ be the number of 1's in the pattern P. Then we can define the asymmetric Hamming distance more formally as follows: *If P_1 and P_2 are two patterns, their asymmetric Hamming distance is $A(P_1, P_2) = \max(\omega(P_1 \wedge \bar{P}_2), \omega(P_2 \wedge \bar{P}_1))$.* (In this definition the complement and the conjunction are bit-wise operations.)

Reconstructibility

Note that if at most one omission occurs in the patterns 00111 and 01010, then they remain distinguishable. This is because, if any of the patterns 00110, 00101 or 00011 is received via the CyberGlove, then we can guess that the user is signing "there", and will not confuse it with "she" which would be registered as either 01010, 01000, or 00010. Thus the asymmetric Hamming distance can be used as a measure of reconstructibility (after omission errors) of patterns. A pattern is k-reconstructible if it is still distinguishable after k number of omission errors. The following is an

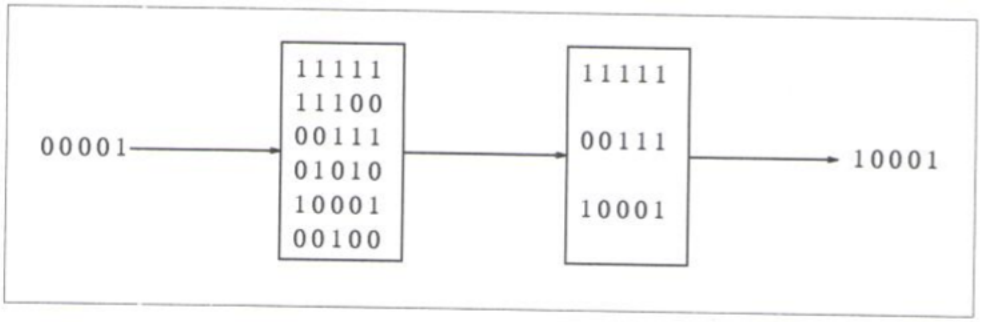


Figure 3: Matcher neural network operation

important fact whose proof we will present in the full paper.

Considering only omission errors, a set of patterns S is k-reconstructible if and only if for each pair of patterns P_i and P_j in S the Asymmetric Hamming Distance $A(P_i, P_j) = k + 1$.

Matcher Neural Networks

Matcher neural networks (MNNs) are biologically motivated neural networks that were introduced in Revesz (1989,1990). We will describe here only the algorithmic part of these networks and refer to the original references for their biological context.

Matcher neural networks are especially suitable for the reconstruction of patterns with a large number of omission errors. Like many neural network algorithms, the operation of MNNs can also be divided into two phases: learning and associative recall. The learning part of MNNs is instantaneous, i.e., all the patterns to be learned are memorized instantly and stored in separate memory locations. For each input pattern, the associative recall phase is carried out in four steps. The first step consists of finding those memorized patterns that match the 1's in the input. The second step consists of selecting from the matching patterns those with the smallest asymmetric Hamming distance from the input. The third step involves finding the percentage of 1's within each group of corresponding bits of the selected patterns. Finally, the fourth step yields as output a binary pattern in which the i^{th} bit is 0 if and only if the percentage of 1's within the i^{th} group is below 50.

As a simple example of the operation of MNNs, consider Figure 3. There the middle box contains the patterns memorized by the MNN. Note that it is the same set of patterns as shown in Figure 1. The pattern to the left of the box is the input pattern. In this case, the first step of the associative recall phase will find that only the patterns 11111 and 00111 match all the 1's in the input. The second step will select the latter of these two patterns because its asymmetric Hamming distance from the input is only one while the distance of the other is three. The next two steps of the algorithm in this case are trivial and will result in simply the pattern 00111 to be returned as desired.

Matcher neural networks are similar in many respects to the sparse distributed memories of Kanerva (1988). However, there are several important differences in both the biological context and the algorithmic parts. Four important differences in the algorithmic parts between SDMs and MNNs are that (1) SDMs use distributed storage while MNNs use localized storage for each memorized pattern (2) SDMs use Hamming distances while MNNs use asymmetric Hamming distances (3) SDMs select all patterns whose *addresses* are within a fixed Hamming distance while MNNs select the closest patterns to the input according to the AHD measure and (4) SDMs use counters and sum counter values of the selected patterns while MNNs sum corresponding bit values of the selected patterns. Some of these differences may seem minor, but as we will see in the next section they can significantly effect the relative performance of the two algorithms.

TESTING METHODS AND RESULTS

Since in this paper we use an artificial sign language, we can choose the set of signs used in our language from the set of all possible signs. In theory considering all 17 joint movements and classifying each as either "0" or "1", we could have 2^{17} different hand signs. However, in practice the total number of possible hand signs is much smaller because certain combinations of joint movements are physically impossible.

In addition to the physical constraints the requirement of high reconstructibility also considerably restricts the number of signs that we can have in our language. One should keep in mind that the CyberGlove sensors for the 17 joint movements may often lead to one or two bit errors in the input feature lists. As noted before, to guarantee even 1-reconstructibility we have to have between each pair of feature lists of signs in our language an asymmetric Hamming distance of two or more.

The problem of finding maximal sets of binary patterns in which each pair has a Hamming distance of k or more for various fixed values of k has been studied by several authors in coding theory (e.g., MacWilliams & Sloane 1981, Brouwer et al. 1990). Surprisingly this is a very difficult combinatorical problem with even the best solutions at present giving only some lower and upper bounds on the maximal set sizes for most values of k. Naturally our problems are compounded by considering the asymmetric Hamming distance and the physical restrictions mentioned above. In choosing our set of symbols we cannot guarantee maximality. Nevertheless we have chosen the set of signs in our language such that between the feature lists of each pair of those signs there is an asymmetric Hamming distance of at least two and each sign is easily made. In total we chose 395 one-handed signs.

With 395 signs we can represent the 395 most frequent words in the English language as measured by Jones & Wepman (1966). This represents about half of the vocabulary of basic English (Ogden 1968). In associating the signs with English words, we chose the signs that are the easiest to make and have the largest mean asymmetric Hamming distances to other patterns to represent the most frequent English words.

For the testing of the sign to speech translation system we did the following. We randomly generated hundred signs to be tested. We allowed repetition in the signs to be tested following the expected frequency of the signs in ordinary signing. The matcher neural network had in its memory the feature list of each sign to be tested.

Each of the hundred CyberGlove generated feature lists was given in sequence as input to the matcher neural network. We noted the number and types of errors in the CyberGlove generated feature lists and in the feature lists reconstructed by the matcher neural network. For comparison we copied the CyberGlove-generated feature lists into a file and entered them in the same order into an SDM. The SDM was autoassociative, had hundred random addresses and 8 bit counters. We tried various values for the Hamming distance.

We obtained the following results. We found that out of the hundred input patterns only 45 were correct, while 41 had a 1-bit error and 14 had 2-bit errors. Out of the total of 69 errors 67 were omission and two were deletion errors. Taking this as input the matcher neural network reconstructed 96 patterns correctly. This 96 is the sum of the 45 that were initially correct plus the 40 cases of with single omission errors and 11 cases with double omission errors. In other words 51 out of 55, that is, 93% of the incorrect inputs were corrected by the matcher neural network.

For the SDM working on the same inputs the value of 4 for the Hamming distance selection was the best. However even in that case only 33% of the outputs were correct, and some of the outputs had upto eleven bit errors. (Using asymmetric Hamming distances with the SDM resulted in even worse performance. It was best with a distance value of 5 yielding only 30% correctness.) Note that the SDM had a negative effect on the performance of the sign-to-speech translation system because twelve of the correct inputs were actually destroyed. While this difference in the performance of SDMs and MNNs on our problem is not really surprising, it underscores the importance of the differences between these two types of networks and that they are best for different applications.

CONCLUSIONS

In this paper we improved the work of Fels & Hinton (1993) by increasing from 66 to 395 the number of artificial root signs that can be robustly translated to speech. However we did not considered the conjugation of these signs. The conjugation in Fels & Hinton (1993) involves the movement of fixed hand shapes (the roots) in any of six directions. This movement is detected by an additional device called the Flock of Birds and is analyzed by a separate neural network that works independently of the one that analyses hand shapes. A similar network could be used for the conjugation of our signs as well. We recently purchased a Flock of Birds and are implementing such an algorithm. In the future we also plan to use semantic techniques to improve the accuracy of the reconstructions.

REFERENCES

April, E.W., *Anatomy*, National Medical Series, Harwal Publishing Company, 1990.

Brouwer, A.E, Shearer, J.B, Sloane, N.J. Smith, W.D, A new table of constant weight codes, *IEEE Transactions on Information Theory*, Vol 36 (11), 1990.

Fels, S.S., Hinton, G.E., Glove-Talk: A neural network interface between a Data-Glove and a speech synthesizer, *IEEE Transactions on Neural Networks*, 4 (1), 2-8, 1993.

Jones, L.V., Wepman, J.M. *A Spoken Word Count*, Language Research Associates, 1966.

Kanellakis, P.C., Kuper, G.M., Revesz, P.Z., Constraint query languages, 9th ACM Symposium on Principles of Database Systems, ACM Press, pp. 299–313, Nashville, Tennessee, 1990.

Kanerva, P., *Sparse Distributed Memory*, MIT Press, 1988.

Kramer, J., Leifer, L. 'TalkingGlove': A speaking aid for nonvocal deaf and deaf-blind individuals, *Proc. of 12th Annual RESNA Conference*, 471-472, Louisiana, USA, 1989.

Loomis, J., Poizner, H., Bellugi, U., Blakemore, A., Hollerbach, J., Computer graphic modeling of American Sign Language, *Computer Graphics*, 17 (3), 105-114, 1983.

MacWilliams, F.J., Sloane, N.J, *The Theory of Error Correcting Codes*, 1981.

Ogden, C.K., *Basic English: International Second Language*, World Inc., NY, USA, 1968.

Poizner, H., Klima, E.S., Bellugi, U., *What the hands reveal about the brain*, The MIT Press, 1987.

Revesz, P.Z., Matcher neural networks, *1st International Joint Conference on Neural Networks*, vol. 1, 767–772, Washington D.C., 1989.

Revesz, P.Z., Functional interpretations of neocortical modules, *2nd International Joint Conference on Neural Networks*, vol. 2, 509–514, San Diego, CA, 1990.

Revesz, P.Z., A closed form evaluation for Datalog queries with integer (gap)-order constraints, *Theoretical Computer Science*, vol. 116 (1), 117–149, 1993.

Stungis, J., Identification and discrimination of handshape in American Sign Language, *Perception & Psychophysics*, 29 (3), 261–276, 1981.

Wilbur, R.B., *American Sign Language and Sign Systems*, University Park Press, BA, USA, 1979.