Name Searching with Artificial Neural Networks

Mark Lewellen

Lernout & Hauspie-AppTek
1420 Beverly Road, Suite 350
McLean VA, 22101 USA
mlewellen@apptek.com

Abstract

This paper reports on research into the influence of word representation methods on the ability of artificial neural networks (ANNs) to recognize spelling variants of natural language words. We use backpropagation to train a multilayer perceptron with many-to-one mappings of romanized Chinese surnames to Chinese characters. The network is trained with five different methods of word representation and each is tested against five types of spelling variation. The results support the hypothesis that word representations significantly affect the ability of an ANN to recognize spelling variants of natural language words. Furthermore, different word representations have identifiable strengths and weaknesses for specific types of spelling variation. Relatively holistic representations such as a bigram+unigram coarse-coded symbol memory and split fixed-length letter buffer improve ANN performance by reducing its dependence on letter position.

1 Background and Motivation

Artificial neural networks (ANNs) have become increasingly popular in natural language processing (NLP) in recent years, due to their ability to learn from on-line language resources, yet generalize in the event of sparse training data. Among the NLP applications for which ANNs have been employed are speech recognition (Bourland and Morgan 1994), speech generation (Sejnowski and Rosenberg 1986), part-of-speech tagging (Ma and Isahara 1998, Schmid 1994), sentence boundary disambiguation (Palmer and Hearst 1997), spelling correction (Lewellen 1998b) and name searching (Lewellen 1998a). It is this last topic—name searching—which is the area of research reported on in this paper.

A motivation for applying ANNs to name searching is the sizeable demand for high-quality name searching in global information systems. Applications as varied as hotel and airline reservations systems, world-wide shipping, law enforcement and visa or passport issuance all have a great need of quickly identifying personal information through use of a person's name. However, names may be spelled with variant methods, contain nicknames or initials, be in transposed order, or contain typographical errors. Years of research and practical experience have shown that monolithic approaches to name searching can only achieve a certain level of accuracy, and that a multi-faceted approach incorporating a variety of language or culture-specific modules is preferable for higher performance (Lewellen 1998a, Borgman and Siegfried 1992, Hermansen 1985). Thus, for example, specialized techniques have been developed for searching romanized names in Chinese (Lewellen 1998a) and Arabic (Roochnik 1993). Although a Chinese or Arabic speaker might correctly point out that name searching could be performed more accurately using Chinese and Arabic scripts instead of Roman letters, such an approach is typically not feasible in global information systems, due to the requirements of staffing users around the world who are competent in reading and writing those scripts. Thus, the research reported here focuses on romanized names of a specific type—Chinese. Romanized Chinese names are particularly difficult for name search systems due to their short length, wide spelling variation (caused by multiple dialects and romanizations) and homography (many different names may be romanized in the same way).

ANNs are viewed as a promising technology for name searching, since they are able to learn from existing on-line listings of names, yet identify these names when novel or erroneous (i.e. never-before-seen) spellings are encountered. In order to utilize ANNs to recognize erroneous and variant name spellings, methods for converting natural language words to forms appropriate for ANN input must be developed. The method of ANN word representation was shown by (Lewellen 1998a,b) to be a key to ANN performance in functions such as name searching and spelling correction, when using a type of Fuzzy ARTMAP ANN. The research presented here intends to test whether these same conclusions hold up when tested with the most common type of

ANN—a multilayer perceptron trained with backpropagation.

2 ANN Word Representations

2.1 General Representation Issues

ANNs are prototypically composed of multiple layers: an input layer, an output layer, and a variable number of hidden layers. Each layer is composed of a fixed number of nodes, each of which has an activation typically represented by a value between 0 and 1. To input words to an ANN, a word must thus be transformed into a fixed-length vector of numeric values.

Generally, representations are regarded as *local* or *distributed*. A local representation represents a single entity with a single node; for example a letter of a word could be represented by one of 26 binary-valued nodes, where each node corresponds to a letter of the alphabet. A local representation of the letter *n* is displayed in the top row of Figure 1.

A distributed representation represents a single concept with not one, but a combination of nodes. The values of the individual nodes may be arbitrary, as in the bits of a byte representing an ASCII code, or they may represent specific characteristics, as in a representation of +bilabial, +voiced, +stop for the phoneme /b/. A *coarse-coded* representation is a distributed representation in which many nodes participate in representing an entity, and additionally each node participates in the representation of many entities.

Another way of characterizing a representation scheme is on a scale from maximally *sparse* to maximally *dense*. For example, to represent an individual letter, a maximally sparse representation would be a local representation that activates only one out of 26 nodes at a given time, while a maximally dense representation would utilize the smallest number of nodes possible to represent a letter (five, as $2^5 > 26 > 2^4$) and may have all of the nodes activated at a particular time. A letter representation situated more toward the center of the sparse-dense continuum is a coarse-coded eleven-node buffer that has four nodes (36%) activated at any time (see Figure 1).

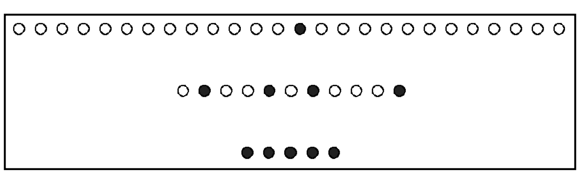

Figure 1: Sparse and dense representations for the letter *N*. A maximally sparse representation (top) is a local representation with 1 of 26 nodes activated, while a distributed representation toward the middle of the sparse-dense continuum (center) has 4 of 11 nodes activated, and a maximally dense letter representation (bottom) could have all of its nodes activated.

2.2 Word Representation Issues

The representation of words for ANN input involves several issues that, taken together, are unusual and problematic for ANNs.

- **Symbols**. Letters are symbols, while ANN values are numeric. The difficulty in using symbols in a numeric system is that numbers have properties that exist between them that do not exist in a symbol set. For example, 2 is twice the size of 1, while 26 is 26 times the size of 1; however, B is not twice A, nor is B more closely related than Z to A. To reflect this difference, non-continuous values are used to represent symbols in ANNs, generally using binary values 0 and 1[1]. Individual input or output nodes for symbol-processing ANNs may thus be regarded as either “on” (“activated”) or “off”. The use of binary values instead of continuous values, however, results in a loss of information content that must be compensated for by increasing the size of input representations.
- **Variable Length**. Words vary substantially in length, from one letter to an indefinite number in double digits. It is undesirable to create large fixed-length representations of perhaps 20 letters when much of that representational capability will seldom be used. Furthermore, a large representation might still not be large enough for occasional extremely long words. While the data tested here—romanized Chinese surnames—does not contain long words, variable length remains a concern. A second problem with variable length is exemplified in the spellings *Ng*, *Ung*, *Wong*, *Zhang* and *Chuang*. In these names, the word-final letter combination *ng* would in some representation methods be represented across different input nodes and consequently different trained weights.
- **Position Shifts**. In the previous example, specific letters appear in different positions (as defined by distance from the beginning). Groups of letters may also be shifted into different positions by affixes and spelling variations or errors. The “position-altering” edits—insertions and deletions—may cause large portions of words to be shifted into different positions. For example, if an ANN is trained on the name *Chao*—with letters represented sequentially from the left—but encounters the spelling *Tchao* in

[1] Bipolar values (1 and –1) may also be used. The target values may be softened as well, for example to 0.1 and 0.9 for binary representations and –0.8 and 0.8 for bipolar representations.

testing, all letters of the original spelling will have been shifted to different input nodes.

In order to identify variant spellings of words, we desire an ANN word representation that can overcome these difficulties and is:

- able to represent all the letters of a word;
- in a form that enables an ANN to identify spelling similarities and differences;
- in a holistic (non-brittle) form that can account for position shifts of groups of letters;
- concise enough to enable processing of a large number of words in a reasonable time.

2.3 Previous Research

Early examples of ANN word representations include left-aligned fixed-length letter buffers (FLLBs) (discussed below), representations based on letter shape (Lacouture 1988), sliding window representations (Sejnowski and Rosenberg 1986, McCulloch et. al. 1987), and "receptive field table" representations, a type of "coarse-coded symbol memory" (CCSM) (Touretzky and Hinton 1988, Touretzky 1990). These representations were developed with different goals in mind, however: letter shape representations are useful for character recognition, sliding window representations are appropriate for speech generation, and the receptive field table representation was designed with the goal of illuminating cognitive processes involved in reading.

(Lewellen 1998a,b) developed several word representations designed specifically for recognition of variant spellings of natural language words. In this research, different word representation methods had substantial effects on overall ANN performance, and had identifiable strengths and weaknesses for specific types of variation as well. For example, in a left-aligned FLLB—in which a fixed number of letter representations are used from left to right—the ANN had substantial difficulty with "position-altering" variation (insertions and deletions) that causes subsequent letters to the right to be shifted into different positions. This problem was mitigated through use of "bi-directional" and better yet, "split" FLLBs. In a bi-directional FLLB, letters are placed from the left to the middle of the buffer, and from the right to the middle of the buffer, filling as many letter positions as possible. In a split FLLB, letters are also positioned from the left and the right, but any given letter may only be utilized once. For example, three FLLB representations for the surname *Chang* are shown in Figure 2.

FLLB	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*
Left-aligned	**C**	**H**	**A**	**N**	**G**			
Bi-directional	**C**	**H**	**A**	**N**	**H**	**A**	**N**	**G**
Split	**C**	**H**	**A**				**N**	**G**

Figure 2. Fixed-length letter buffer (FLLB) representations of the surname *Chang*.

Of the three types of FLLB, the split FLLB performed the best both on datasets of romanized Chinese surnames and English homonyms.

(Lewellen 1998a) also developed two CCSM representations, building on research by (Touretzky 1990, Touretzky and Hinton 1988, Rosenfeld and Touretzky 1988). In a CCSM, the ANN interface is not divided into individual letter buffers (as in an FLLB); rather, many coarse-coded symbol representations are simultaneously activated, as shown in the simplified CCSM representation in Figure 3. (Lewellen 1998a) developed two types of CCSMs: one that employs bigrams as the symbol set (as in Figure 3), and a second that uses both bigrams and unigrams. The two CCSMs outperformed the FLLB representations, with the bi/unigram CCSM achieving the best overall performance.

3 Test Design

The test design presented here is motivated by several questions:

- Do different ANN word representations cause differences in the capability of backpropagation ANNs to perform name searching?
- Which representations are most effective?

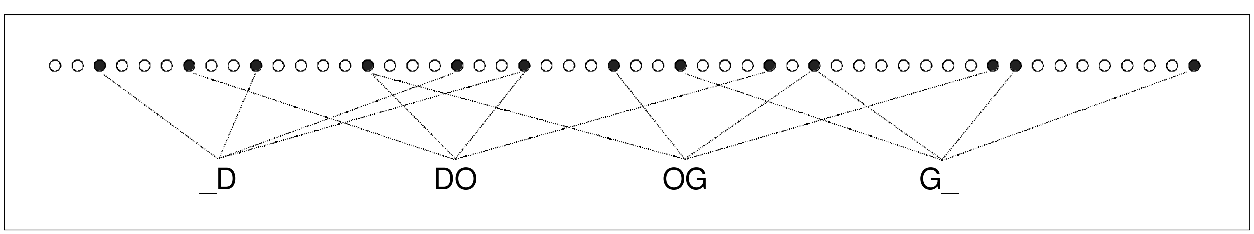

Figure 3. A simplified CCSM, representing the word *dog* with a bigram symbol set. The bigrams simultaneously activate distributed codes; the union of these codes constitutes a CCSM word representation.

- Do the word representations have identifiable strengths or weaknesses on specific types of spelling variation?
- Do the test results conform with the results reported in (Lewellen 1998a,b)? Can the results of all three studies be synthesized into useful generalizations about the influence of word representations on ANN performance?

In order to answer these questions, a backpropagation network was trained on a corpus of Chinese name spellings, using five different representation methods. The trained networks were then tested with corpora containing specific types of spelling variation.

3.1 Training and Testing Corpora

The training corpus was derived from a large corpus of Chinese name romanization/character mappings. The most frequently-occurring mappings were chosen, using a threshold of five occurrences. The resultant list contains 369 romanized surnames. Of this number, two are involved in many-to-many relations and were removed from the corpus. The 367 remaining surname romanizations map in many-to-one relations to 186 Chinese characters.

The test corpora were derived from the same Chinese name corpus. To help identify name errors and exceptional spellings, all romanization/character mappings occurring two times or less were identified. From this list, names were selected by hand for each of the five types of variation displayed in Figure 4. Differing numbers of variants were found for each of the five types. An insufficient amount of reversal variation was found, so the reversal test set was created by hand, with edits evenly distributed through initial, medial and final portions of the name. The final test set corpora sizes are:

substitutions	212
reversals	91
insertions	132
deletions	80
complex	75

3.2 Word Representations and Architecture

The training corpus is represented with five types of ANN word representation: left-aligned, split and bi-directional FLLBs, and bigram and bi/unigram CCSMs. These representations are used as input to the ANN, while the output is a buffer of 186 nodes, with each node corresponding to a Chinese character.

The FLLBs are all composed of eight letter representations, each a distributed code of 11 nodes. Of the 11 nodes, 4 nodes are activated at any time, and no more than 2 nodes may participate in another letter representation. A total of 88 nodes is used for each type of FLLB input.

Type of variation	*Source spelling*	*Variant spelling*	*Chinese character*
substitution	*Lin*	*Lih*	林
reversal	*Hsiung*	*Shiung*	熊
insertion	*Kwok*	*Kwock*	郭
deletion	*Yang*	*Yan*	杨(楊)
complex	*Chao*	*Chow*	赵(趙)

Figure 4. Examples of spelling variation[2].

The bigram CCSM is a buffer of 195 nodes, six of which are activated by any bigram. No more than one node of the six may participate in any one other bigram representation. The bi/unigram CCSM is composed of 210 nodes. Again, each symbol representation is composed of six nodes, of which no more than one may be part of any one other symbol representation.

The backpropagation ANN consists of three layers. The input layer is composed of 88, 195 or 210 nodes, depending on the word representation, while the output layer is composed of 186 nodes. The hidden layer is composed of 97, 70 or 67 nodes, depending on the word representation as well. The number of hidden-layer nodes was varied in order to create roughly equivalent numbers of connections for the network as a whole. Thus, for an input layer of 88 nodes, a hidden layer of 97 nodes is used, while the 195-node input layer connects to a 70-node hidden layer and the 210-node input layer uses a 67-node hidden layer. Nodes are represented in bipolar format, with target values of –0.9 and 0.9.

The networks were trained until each input/output mapping was learned to within a certain degree of error. Upon meeting this criterion for two successive epochs, training ceased, in order to preserve generalization capabilities. The degree of error used to terminate training was 0.3 for each node. Since each of the 186 output nodes correspond to one Chinese character and the input/output mappings are composed of many-to-one relations, 185 nodes are required to have an activation of –0.6 or less, while the target node must have an activation of 0.6 or greater.

4 Evaluation

4.1 Interpretation of Test Results

4.1.1 Performance measures

Three measures of performance were developed for each of the 25 test sets. The first measure is the

[2] In cells with two characters, a simplified Chinese character precedes the traditional character in parentheses.

number of times that the network correctly identifies the target output (Figure 6), as determined by the maximally activated output node. For this measure, higher scores indicate better performance. The second measure is the average rank score of the target output node (Figure 7). Rank scores are assigned based on the activation of the target node: the most highly-activated node receives a rank of 1, while the 10^{th} most highly-activated node, for example, receives a rank score of 10. Thus, unlike Figure 6, lower scores indicate better performance in Figure 7. The third measure is the median rank score of the target output node (Figure 8). Here again, lower scores indicate better performance.

Three measures of performance are employed due to concerns that one measure might in some cases be misleading, due to the nature of the data tested. Correct identification of target output (Figure 6) is not always indicative of performance, for there may be several output possibilities that are equidistant from the input. For example, in the insertion test set, the surname spelling *Wou* is supplied as a variant of *Wu* (both as spellings of 吴). However, 11 other names in the training set also differ by one letter from *Wou*: *Sou, Kou, Hou, Lou, Jou, Mou, Dou, Tou, Zou, Ou* and *Woo*. The 12 spellings map to 11 characters; it is somewhat arbitrary which of the corresponding 11 nodes will be maximally activated and all are appropriate. The rank score evaluation (unlike correct identification of target output) is able to capture this information with rank scores of 1 to 11. In the five tests, the target output node obtained rank scores of 1, 9, 10, 107 and 141. Thus, three representations performed well and two (the left-aligned and bi-directional FLLBs) did not. The correct identification of target output measure (Figure 6) only accounts for the good performance of one representation, while the rank-score measures (Figures 7 and 8) account for all three representations that performed well. Given the short length of many romanized Chinese names, such situations are commonplace. Thus, evaluating performance based on the maximally-activated node alone may be arbitrary and misleading.

Consequently, performance measures that capture information about rank scores are desirable as well. While the average rank score (Figure 7) accounts for situations in which many spellings are equidistant from the test input spelling, the figure may be skewed by arbitrarily large rankings. In general, after the 10 to 30 best rankings, the activations cluster at similar values about the lower target value of –0.9. Assigning ranks in this area is somewhat arbitrary, since sizeable differences in rank may correspond to minute variations in activations. To account for the arbitrariness of poor rank scores, median rank scores (Figure 8) are calculated as well. The median rank scores are viewed as the most accurate of the three indicators.

For all three measures, overall indicators of performance are developed for each representation method. In the case of correct identification of target output (Figure 6), the overall figure is the sum total of correct identifications. For the average rank scores (Figure 7), a weighted overall average is used, with weights based on test-set size. Similarly, a weighted overall median score is developed for Figure 8.

The overall indicators of performance are presented together in normalized form in Figure 9. This method of presentation facilitates comparison of overall performance by normalizing all three measures to a baseline of 1. Higher scores correspond to better performance. With this method, different representation scores may be compared, as well as different performance measures.

4.1.2 Many-to-many mappings

In some cases, a variant spelling supplied for testing is the same as a name spelling used in training that maps to a different Chinese character. This situation arose due to the method of data collection: *mappings*, not *spellings*, that occurred five times or more were used for training, and mappings that occurred two times or less were candidates for test sets. Therefore, *spellings* that map to different characters could qualify for both the training and test sets. In testing, these spellings cannot correspond to maximally-activated nodes, as the trained spelling/character mapping always achieves the maximum activation. Additionally, an analysis of preliminary results indicated that the test spellings involved in many-to-many mappings frequently caused large rank scores, and, as pointed out above, large rank scores tend to be somewhat arbitrary. For both these reasons, spellings involved in many-to-many mappings are not included in the test results presented below. The modified test set sizes are shown in Figure 5.

	Sub.	Rev.	Ins.	Del.	Com.
Original	212	91	132	80	75
Modified	104	88	98	36	48

Figure 5. Test set sizes.

4.2 Test Results

Test results for correctly-identified output, average rank scores and average median scores are presented in Tables 6-8, while a normalized overall comparison of performance is presented in Figure 9.

	Sub.	Rev.	Ins.	Del.	Com.	Tot.
Left FLLB	31	4	15	3	3	56
Split FLLB	35	9	34	3	5	86
BiD FLLB	36	6	5	2	2	51
Bigram CCSM	21	3	34	8	0	66
Bi/Unigram CCSM	21	8	28	7	3	67

Figure 6. Number of times target output correctly identified.

	Sub.	Rev.	Ins.	Del.	Com.	All
Left FLLB	22.3	54.9	42.2	53.3	74.8	49.5
Split FLLB	21.1	59.2	34.8	40.2	59.6	43.0
BiD FLLB	23.1	59.5	59.7	56.8	79.0	55.6
Bigram CCSM	40.1	77.7	23.3	40.7	83.0	53.0
Bi/Unigram CCSM	40.6	50.1	20.3	28.8	82.2	44.4

Figure 7. Average rank scores of test sets.

	Sub.	Rev.	Ins.	Del.	Com.	All
Left FLLB	4	39.5	17	15.5	51.5	18
Split FLLB	4	29.5	4.5	10.5	37.5	13
BiD FLLB	3	30.5	23.5	36	77.5	25
Bigram CCSM	12.5	61.5	3	25.5	69.5	24
Bi/Unigram CCSM	7	25.5	3	9.5	77.5	13

Figure 8. Median rank scores of test sets.

In the substitution test sets, the FLLB representations all outperform the CCSM representations. The FLLB representations all obtain similar results, while the bi/unigram CCSM outperforms the bigram CCSM, having roughly the same target output and average rank scores, but a much better median rank score.

For the other type of position-maintaining variation—reversals—the FLLBs again are fairly similar, with the split and bi-directional representations slightly out-performing the left-aligned FLLB, based on much better median scores. The bigram CCSM fares worse than the FLLBs, while the bi/unigram CCSM outperforms all other representations.

For position-altering variation (insertions and deletions), the split FLLB performs the best of the FLLBs, followed by the left-aligned and bi-directional FLLBs. The CCSMs outperform the FLLBs (with the exception of the bigram FLLB on deletions). The bi/unigram CCSM fares the best of all representations.

For the complex variation set, the split FLLB far outperforms all other representations. The left-aligned FLLB performs slightly better than the remaining three similarly-performing representations.

Overall, the split FLLB and bi/unigram CCSM achieve the best results, with a slight edge to the split FLLB. At a lower level, the left FLLB performs better than the bigram CCSM, which in turn has an edge over the bi-directional FLLB. The bi/unigram CCSM performs the best for reversal, insertion and deletion variation, while the split FLLB performs best for complex variation, and the FLLBs all perform best for substitution variation.

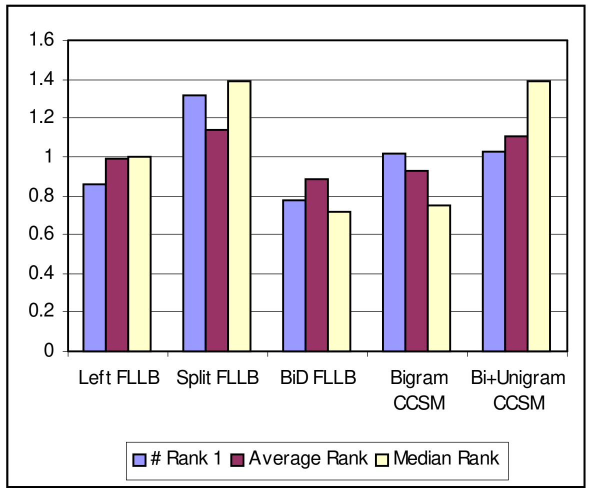

Figure 9. Comparison of word representation overall performance. Three methods of evaluation are represented; each method is normalized to a baseline of 1. Higher values indicate better performance.

5 Discussion and Analysis

A motivation of the current work is to develop generalizations about the effect of ANN word representations by synthesizing the current results with those presented in (Lewellen 1998a,b), which used a Fuzzy ARTMAP type of ANN. (Lewellen 1998a) used romanized Chinese surname data while (Lewellen 1998b) used a dataset of English homonyms, testing only FLLB representations. The results of all three studies demonstrate that different ANN word representations have substantial effects on an ANN's capability to recognize spelling variants and errors. It is not the case that the network is able to learn and generalize regardless of representation type. We posit that this is due to the somewhat unique nature of words: words are composed of a variable number of symbols that may undergo position shifts.

Additionally, the studies concur that FLLB and CCSM representations have identifiable strengths and weaknesses for processing different types of variation. FLLB representations perform similarly and very well on the simple type of position-maintaining variation—substitutions—since a substitution edit involves only the altering of one letter at one position. For this reason also, the bi/unigram CCSM outperforms the bigram CCSM, since unigrams are more tolerant of a one-letter edit.

In both studies involving CCSMs, the bi/unigram CCSM achieves the best performance for reversals and position-altering variation—insertions and deletions. In the current study, the bi/unigram CCSM is slightly superior for reversals, while in (Lewellen 1998a) it is far superior. The three FLLBs perform similarly to each other for reversals, while the bigram CCSM performs the least well. In the case of reversals, we attribute the good performance of the bi/unigram CCSM again to the use of unigrams. In the case of position-altering variation, we posit that the strength of an *n*-gram CCSM over an FLLB for position-altering variation is that position is *relative* for bigrams and *independent* for unigrams, but *absolute* for FLLBs. For position-altering variation, both CCSMs outperform all the FLLBs. Among the FLLBs, the holistic characteristics of the split FLLB enable it to outperform the other FLLBs.

Despite these broad consistencies, there remain some inconsistencies. The bigram CCSM performed much better in (Lewellen 1998a) than in the current study. There, the bigram CCSM performed as well as the bi/unigram CCSM for position-altering variation, at a level so much higher than the FLLBs that the overall bigram CCSM scores exceeded those for all of the FLLBs. Here, the bigram CCSM underperformed the split and left CCSMs. We have not been able to identify a reason for this difference, but a potential cause is the smaller sample size in the current study, as well as the improved results of the left-aligned FLLB.

While the left-aligned FLLB achieved better figures than the bi-directional FLLB and bigram CCSM in this study, it performed worse in the previous studies. This altered behavior may be attributed to the distribution of data in the current study. Since the spelling variation is naturally-occurring, spellings were not selected with an eye toward an even distribution among initial, medial and final position of the spelling variation (with the exception of reversals). For position-altering variation, such differences are major factors for left-aligned FLLBs, but not the other representations. For example, an insertion edit in the beginning of a name—such as *Tchao* for *Chao*—shifts all the original letters to different positions. However, position-final insertions—such as *Ger* for *Ge*—affect only the final inserted letter of a left-aligned FLLB. An analysis of the insertion data shows the following distribution, and consequent results:

Position	*# Occurrences*	*Left Average*	*Left Median*	*Bi/unigram average*	*Bi/unigram median*
Initial	9	108.2	146	27.3	5
Medial	48	38.4	18.5	19.8	3
Final	27	24.0	6	27.8	7
Ambig.	14	48.3	21	3.4	1

Figure 10. Position of insertion edits.

Only 9 of 98 insertion edits occurred unambiguously[3] in initial position. The severe effect of initial position on left-aligned FLLBs is demonstrated by much higher average and median scores. However, bi/unigram CCSMs, for example, are affected relatively little by such variation. Since the left-aligned FLLB would have achieved much worse scores had the edit positions been more evenly distributed, the current results may be viewed to overstate the performance of left-aligned FLLBs. The poor performance of the left-aligned FLLB on evenly-distributed data would then be consistent with previous research.

Another inconsistency in results is that the split FLLB attains marginally better results than the bi/unigram CCSM in the current evaluation, while the bi/unigram CCSM was far superior to all other representations in the previous evaluation. One cause is that the bi/unigram CCSM performed better on complex variation previously, while the split FLLB performs much better on complex variation in the current study. Given that the current complex variation sample is somewhat small, and the great variability of the complex data, it may be viewed as less reliable and discounted, particularly in view of the fact that the bi/unigram CCSM achieves the best performance for three of the four basic variation types (insertions, deletions and reversals). Given this adjustment, the bi/unigram CCSM slightly outperforms the split FLLB in the current study, albeit not by such a large margin as in the previous study.

6 Conclusion

In sum, the test results presented here from a multilayer perceptron trained with backpropagation bolster the hypothesis of (Lewellen 1998a,b) that word representations significantly affect the ability of an ANN to recognize spelling variants of natural language words; it is not the case that an ANN is able

[3] The position of an edit was considered ambiguous when the insertion edit is adjacent to an identical letter, as in *Renn* for *Ren*.

to learn and generalize regardless of word representation type. Furthermore, different word representations have identifiable strengths and weaknesses for specific types of spelling variation. Relatively holistic representations—the bi/unigram CCSM and split FLLB—improve ANN performance by reducing its dependence on letter position. ANNs employing such holistic representations are able to process the type of spelling variation encountered in name searching of romanized Chinese names.

In future work, we hope to perform further experiments to clarify whether a split FLLB or bi/unigram CCSM is likely to achieve the best performance. Additionally, we would like to compare ANNs employing CCSM and FLLB representations with recurrent ANNs. Since a recurrent ANN usually processes letters from left to right, it is plausible that recurrent ANNs would encounter the same difficulty in processing spelling variation as left-aligned FLLBs. Additionally, we hope to perform experiments with other types of ANNs, in order to further validate the conclusions drawn here about the performance of various word representations, and train on larger numbers of words.

References

Borgman, C. L. and S. L. Siegfried (1992). "Getty's Synoname and its cousins: a survey of applications of personal name-matching algorithms". *Journal of the American Society for Information Science*, 43(7):459-476.

Bourland, H. and N. Morgan (1994). *Connectionist Speech Recognition: A Hybrid Approach.* Kluwer Academic Publishers, Norwell, MA.

Hermansen, J. C. (1985). *Automatic name searching in large data bases of international names.* Washington, D.C.: Georgetown University Dissertation.

Lacouture, Y. (1988). From Mean Square Error To Reaction Time: A connectionist model of word recognition. In D. Touretzky, G. Hinton and T. Sejnowski (Eds.) *Proceedings of the 1988 Connectionist Models Summer School*: 371-378. San Mateo, CA: Morgan Kaufmann Publishers.

Lewellen, M. (1998a). *Neural Network Recognition of Spelling Variation in Chinese Name Searching*. Washington, D.C.: Georgetown University Dissertation.

Lewellen, M. (1998b). Neural Network Recognition of Spelling Errors. *COLING-ACL '98: Proceedings of the Conference, Vol. II*: 1490-1492.

Ma, Q. and H. Isahara (1998). A Multi-Neuro Tagger Using Variable Lengths of Contexts. *COLING-ACL '98: Proceedings of the Conference, Vol. II*: 802-806.

McCulloch, N., M. Bedworth and J. Bridle (1987). NETspeak—A re-implementation of NETtalk. *Computer Speech and Language* 2:289-302.

Palmer, D. D. and M. A. Hearst (1997). Adaptive Multilingual Sentence Boundary Disambiguation. *Computational Linguistics* 23(2):241-267.

Rosenfeld, R. and D. S. Touretzky (1988). A Survey of Coarse-Coded Symbol Memories. In D. Touretzky, G. Hinton and T. Sejnowski (Eds.) *Proceedings of the 1988 Connectionist Models Summer School*: 256-264. San Mateo, CA: Morgan Kaufmann Publishers.

Roochnik, P. (1993). *Computer-based solutions to certain linguistic problems arising from the romanization of Arabic names.* Washington, D.C.: Georgetown University Dissertation.

Schmid, H. (1994). Part-of-speech tagging with neural networks. *Proceedings of COLING-94*: 172-176.

Sejnowski, T. J. and C. R. Rosenberg (1986). "NETtalk: a parallel network that learns to read aloud". The Johns Hopkins University Electrical Engineering and Computer Science Technical Report, JHU/EECS-86/01, Baltimore.

Touretzky, D.S. (1990). BoltzCONS: Dynamic Symbol Structures in a Connectionist Network. *Artificial Intelligence* 42(1-2):5-46.

Touretzky, D.S. and G.E. Hinton (1988). A Distributed Connectionist Production System. *Cognitive Science* 12(3):423-466.