KNOWLEDGE UTILIZATION IN HANDWRITTEN ZIP CODE RECOGNITION

Jonathan J. Hull and Sargur N. Srihari

Department of Computer Science
State University of New York at Buffalo
Buffalo, New York 14260

ABSTRACT

The process of recognizing the postal codes or zip codes in a handwritten address can be aided by many sources of external knowledge. City and state names are obvious examples that can be used in conjunction with a city-state-zip directory to provide evidence about digits in a zip code. This paper describes an extension of this methodology that uses knowledge about legal *street names* and *sujfixes* to constrain the digits in a zip code. The technique does not require complete recognition of all characters in words. Rather, a feature description of words is used to index a set of possible zip codes. Some preliminary experiments with the ZIP+4 database are discussed. It *is* shown that even a relatively simple description of two words in the street line of an address can significantly reduce the number of zip codes that could appear on a piece of mail.

1. Introduction

Reading the zip code within the destination address area of a mailpiece is of central importance in automated mail sorting [6} The problem is not always amenable to straightforward alphanumeric character recognition — particularly when the address is handwritten. Fortunately, the numeric zip code in a destination address does not usually occur in isolation. There is additional or redundant information in the form of a city name, a state name, a street address, the name of a destination, and perhaps an attention line Mao. there is usually a return address and sometimes advertising material. All this knowledge has some potential to contribute to recognition accuracy. Previous Artificial Intelligence approaches to the general reading problem have focused on the use ol intra-word knowledge [2,5]. A robust methodology is needed to focus the various sources of inter-word knowledge on the reading problem [3,4).

The most obvious use of inter-word or *external* knowledge in handwritten zip code recognition is for the city and state to confirm a five digit zip code. State information constrains the first digit whereas city information constrains the second and possibly the third digits. Thus if the city and state information is known, the number of alternatives is reduced for zip code recognition. There exist multi-line reading equipment today that are capable of using city and state information as well as street information in determining a nine-digit zip code.

Besides the knowledge in the city/state/zip line (or lines), there is a wealth of knowledge in the street line that could be utilized. Several sources of external knowledge that could be used to constrain the digits in a zip code are illustrated in Figure 1. For example, if mail for a particular city is being sorted, recognizing the pre-directional code as *N* might limit the zip code tp one that occurs north of some known boundary. A better use of external knowledge is to recognize the post-directional code (e.g., NW, NE, etc.) when sorting mail for cities that use such codes (e.g„ Washington, D.C.). This could significantly reduce the number of zip codes that could match a mail piece. Recognizing the street name or the organization would have a similar effect. If combined with a suitably arranged dictionary, this would in many instances specify the zip code.

The focus of this research is on a technique of using exter nal knowledge to constrain the digits in a five-digit zip code. Usually, accurate recognition of all the characters in a word, such as the city name, is required to use external knowledge. There is usually little tolerance for broken, touching, or smeared characters. Such a recognition procedure would be appropriate for clean well-printed text, in which case there may be no need to look beyond the zip code. What is desired instead is a technique that is robust in the presence of noise and can extract some useful information from textual portions of a destination address. Such a technique should be able to constrain the digits of a postal code even if it cannot fully recognize the text.

The technique proposed here computes a noise-tolerant feature description of a specific word or words in an address. This feature description is then used to access a dictionary and return a number of zip codes that correspond to words with that feature description. This cither produces a unique identification of the zip code. or. by constraining the digits of the zip code, provides information that could be used to advantage in recognition. This methodology does not require exact recognition of all the characters in the words it examines. Only some features have to be calculated and these features may provide only a gross description of the word. However, such a feature description may provide useful knowledge even though it is tolerant to noise and easy to compute.

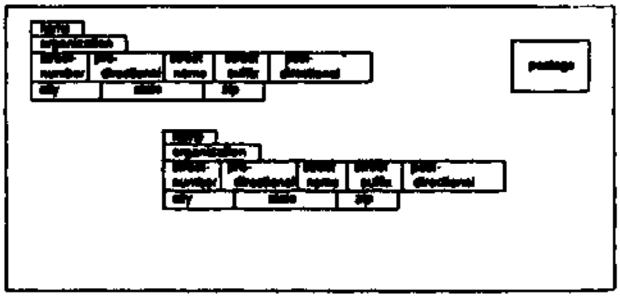


Figure 1. Template for the face of a mailpiece.

2. Method for Constraint Generation

A method is presented for constraining the digits in a zip code. The digital images of isolated words in an address are input. It is assumed that the words come from known categories such as the first word- in a street name. A feature description of an input word serves as an index into a dictionary that returns one or more zip codes associated with the input word. This set of zip codes is referred to as the *neighborhood* of the words with that feature description. The objective of this procedure is to find as small a neighborhood as possible. Ideally, a simple feature extraction routine could be used to find a neighborhood that contains a single zip code.

This methodology differs from a conventional approach to contextual postprocessing since it does not require an attempt at complete recognition of all the digits in a zip code before attempting to utilize other information in an address — again by attempting complete recognition of all the isolated characters in a word [1]. Instead, a description of the gross visual characteristics of words are used to access a knowledge-base and return likely candidates for the zip code, even before attempting recognition of the zip code.

An example of this procedure is presented in Figure 2. There are twenty six five-digit zip codes that could appear on a piece of mail with the city and state names of "Wilmington, Delaware" on it. Thus the city and state information constrain the number of possibilities to twenty six from a theoretical maxImum of KX),(XX) or the practical maximum of about 41,000 (about this many /.ip codes are in use). The map in Figure 3 shows the geographic distribution of ten of the twenty six codes. The remaining sixteen are assigned to unique businesses or to Post Office boxes.

Under the assumption that a mail stream is being sorted for the citv of Wilmington, it is not passible to route a piece of mail to its correct five digit zip code without reading information in some area of the address besides the city/state/zip line. One possibilitv is to use the number of characters as a feature description for the second word in the line above the city/state/zip line. Under the assumption that this word is either a pre-directional code (N,S,F,W,NW,NF,SW,SL) or the first word in a street name, a suitably organized dictionary could be used to determine the zip codes associated with all the words having that feature description. Such a dictionary structure is shown in Figure 2(c). The dictionary is indexed by the number of characters in a word. The neighborhoods of zip codes are shown and the size of each neighborhood (ns) is indicated. Using this dictionary, the example address in Figure 2(b), and the assumption that the number of characters in the first word of the street address, *Van,* is correctly determined, the number of zip codes that could match the address is reduced from twenty six (Figure 2(a)) to nineteen.

19801 - 19810, 19850, 19885 - 19899
(a)

Mr. John Q. Public
376 Van Buren Street
Wilmington, Delaware 19802
(b)

feature	zip codes	ns
1	19801-10, 19893, 19895	12
2	19807-10, 19850, 19887	6
3	19801-10, 19850, 19885-9, 19896-7, 19899	19
4	19801-10	10
5	19801-10, 19888, 19892, 19893	13
6	19801-10, 19890, 19898	12
7	19801-10, 19889, 19896, 19897	13
8	19801-10, 19891, 19892, 19894	13
9	19801-10	10
10	19801-10, 19890, 19891	12
11	19804, 19807, 19808, 19810	4
12	19804, 19806-10	6
13	19805, 19807	2
15	19850	1

(c)

Figure 2. Example of constraining zip codes using feature descriptions: (a) the 26 five-digit zip codes for Wilmington, Delaware, (b) an example address, (c) the dictionary for one feature: length of the first word in the street name.

3. Statistical Simulation

A statistical simulation of the technique proposed, in this paper for constraining the digits in zip codes was constructed to test the effect of several alternative feature sets. Each feature set has different characteristics in terms of computability and tolerance to noise. The feature set used in practice would depend on these considerations as well as how tightly constrained the zip code should be. The computability and noise tolerance factors are very implementation-dependent. However, the constraint factors can be estimated by the statistics defined below.

The database for the simulation was a subset of the ZIP+4 National Directory file for the State of Delaware. The ZIP+4 file contains all the information needed to assign a 9-digit zip code to any address in the United States [7]. This is done by storing, among other information, the ranges of numbers on every street that correspond to each 9-digit zip code. The complete street name can contain up to four fields (explained in section 2): predirectional, street name, street suffix, and post-directional.

It was desired to test the proposed methodology on a model of a mail stream where the usual strategy of using external knowledge to constrain the digits in a zip code by examining the city and state names would not work. This is a subset of the database that has the same city and state names. Therefore, the largest such subset in the given database was chosen. This yielded a file of all the street names and five-digit zip codes in the City of Wilmington.

The Wilmington database was pre-processed so that it included the full spelling for the street suffixes as well as its usual USPS abbreviation. This was done by substitutions such as

Figure 3. Geographic distribution of zip codes in Wilmington, Delaware.

BOULEVARD for BLVD. LANE for LN. CIRCLE for CIR, STREET for STR, etc. In many cases this yielded two records for every one input: the original and a copy with the substitution for the suffix. There were 2316 records in the original database and 4622 in the preprocessed version. The preprocessed database should represent a large percentage of the ways the names of streets would be written for destinations in Wilmington.

Several experiments were conducted for different feature descriptions. Two constraints were used in each expenment: either the first word or the first and last words from the complete street name. The first word was the pre-directional, if it existed, otherwise it was the first word in the street name. The second word was either the post-directional, the street suffix, or the last word in the street name, chosen in that order. It was assumed that these words could be located in an address by finding the second word and the last word in the line immediately above the city/state/zip line. Presumably the first word in the line is the number of the street address. In the example address of Figure 2(b), the first and last words in the complete street name are *Van* and *Street.*

Dictionaries similar to that shown in Figure 2(c) were constructed for several different feature sets. The ability of these feature sets to determine which of the twenty six possible five-digit zip codes matched addresses in the database was simulated by computing the three statistics defined below.

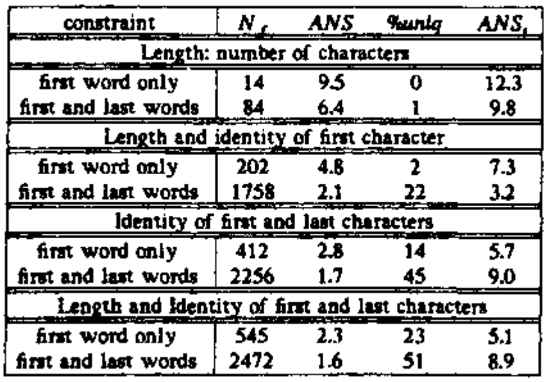
$$ANS = \frac{1}{N_f}\sum_{i=1}^{N_f} ns_i , \quad ANS_t = \frac{\sum_{i=1}^{N_f} ns_i * n_i}{N_t} ;$$

$$\%uniq = 100 * \frac{N_{uniq}}{N_t}$$

where N_f is the number of different feature descriptions present in the dictionary, ns_i is the size of the /' neighborhood, n_t is the frequency of the i' feature description in the mail stream being m o d $N_t = \sum_{j=1} n_j$, and N_{uniq} the number of words that uniquely specify a zip code.

ANS is a static measure of performance that gives the expected number of zip codes in a partition of the dictionary. *ANS,* provides a dynamic measure of performance that indicates the expected number of zip codes associated with a piece of mail in a modeled mail stream. *%uniq* provides the percentage of addresses in the mail stream with a feature description that uniquely specifies a zip code.

These statistics were computed using four different feature descriptions. The results of these expenments are shown below.



constraint	N_f	ANS	$\%uniq$	ANS_t
Length: number of characters				
first word only	14	9.5	0	12.3
first and last words	84	6.4	1	9.8
Length and identity of first character				
first word only	202	4.8	2	7.3
first and last words	1758	2.1	22	3.2
Identity of first and last characters				
first word only	412	2.8	14	5.7
first and last words	2256	1.7	45	9.0
Length and Identity of first and last characters				
first word only	545	2.3	23	5.1
first and last words	2472	1.6	51	8.9

4. Discussion and Conclusions

The results of these experiments illustrate several aspects of this approach. A very simple feature of just the number of characters in two words produced an ANS_t value of about 10. This is much better than the value of twenty six that would be encountered if no constraints were used. If only one letter and the number of characters in one word can be recognized, $ANS_($ is reduced to 7.3. If the first character and the lengths of two words can be computed, a value of 3.2 is obtained for ANS_r When the first and last characters in two words can be recognized, the *ANS* value is only 1.7. Also, the zip code for up to 45% of the mail stream is determined with no other feature tests. This is encouraging but comes at the cost of an increase in *ANS'* to 9.0. A similar trend occurred with the fourth feature set where *ANS,* was about the same at 8.9, but *%ounlq* increased to 51%. If only the first word in the street address is considered, ANS_t is only 5.1, but a zip code is uniquely determined for only 23% of the pieces.

Future work in this area should include an improved simulation of the mail handling environment. The population of addresses used for the experiments discussed here contained at most two entries for each street in each five-digit zip code zone. The simulation thus assumes an equal amount of mail is destined for each street. An improvement would incorporate data that more accurately reflects the amount of mail going to each street.

Acknowledgements

We are grateful to Glen Davis of Arthur D. Little, Inc. for his assistance in acquiring several of the databases used for exper Iments discussed in this paper. We acknowledge the support of the office of Advanced Technology of the United States Postal Service under contract BOA 104230-84-0962.

References

1. W. Doster, "Contextual postprocessing system for cooperation with a multiple-choice character-recognition system," *IEEE Transactions on Computers C26,* 11 (November, 1977).
2. J. J. Hull, "Interword constraints in visual word recognition," *Proceedings of the Conference of the Canadian Society for Computational Studies of Intelligence,* Montreal, Canada, May 21-23, 1986. 134 138.
3. J. J. Hull and S. N. Srihari, "Use of external information in ZIP code recognition," *United States Postal Service Advanced Technology Conference,* October 21-23, 1986, 361-370.
4. J. Schurmann, "Reading machines," *Proceedings of the 6th International Conference on Pattern Recognition,* Munich, West Germany, October 19-22, 1982,1031-1044.
5. S. N. Srihari and J. J. Hull, "Knowledge integration in text recognition," *AAAI-82, Proceedings of the National Conference on Artificial Intelligence,* Pittsburgh, Pennsylvania, August, 1982, 148-151.
6. S. N. Srihari, J. J. Hull, P. Palumbo and C. Wang, "Address block location: specialized tools and problem solving architecture," *United States Postal Service Advanced Technology Conference,* October 21-23, 1986, 116-131.
7. *Zlp+4 national directory technical guide,* United States Postal Service Address Information Center. September 1, 1985.