

# A POS-BASED FUZZY WORD CLUSTERING ALGORITHM FOR CONTINUOUS SPEECH RECOGNITION SYSTEMS

S. Momtazi, H. Sameti, M. Bahrani, N. Hafezi  
Speech Processing Lab, Computer Engineering Department,  
Sharif University of Technology, Tehran, Iran  
momtazi,bahrani,hafezi@ce.sharif.edu, sameti@sharif.edu

## ABSTRACT

*Using word base n-gram language models in continuous speech recognition systems is so prevalent. For using this type of language models, we should extract them from large corpora. Since Persian corpora are not rich, therefore the extracted language models are not credible. For this reason, most researchers extract class n-grams instead of finding word n-grams. In this research a new idea for fuzzy word clustering is represented that each word can be assigned to more than one class. The Fuzzy c-mean algorithm is used for our clustering method and we have examined its various parameters of it. Finally, this algorithm was applied on 20000 most frequent Persian words extracted from "Persian Text Corpus". The extracted language models are evaluated by perplexity criterion and the results show that a considerable reduction in perplexity has been achieved. Also, the results of this language model were evaluated on speaker independent continuous speech recognition system and improved the system accuracy.*

## 1. INTRODUCTION

For improving the accuracy of continuous speech recognition, it is usual to use n-gram language models. [1] Although the bi-gram language model is one of the simplest types of n-gram models, but there are some problems in computing word bi-gram models. One of these problems is the lack of sufficient training data. Also there is an idea by which we can use the bi-gram statistics of similar words instead of their own bi-gram statistics. For example, consider this sentence: "He woke up Sunday morning at 7 a.m.". For computing the bi-gram statistics between the words of the sentence, we should use the statistics between "Sunday" and "morning" but we see that there is no difference between "Sunday morning" and "Monday morning" or "Sunday night", etc. because the functions of these words are similar in language. So, it seems that one method of reducing the number of word history to be modeled in the n-gram case is to consider some words as equivalent. This can be implemented by mapping a set of words to a word class by using a clustering function. By clustering words, we will have comprehensive and complete language models without the need to have rich corpora.

For achieving this goal, we should use word clustering techniques and then, instead of referring to bi-gram statistics between two words, we can refer to bi-gram statistics between the classes of those words.

Suppose  $w_1, \dots, w_T$  are the consecutive words of a sentence  $S$  and  $C_1, \dots, C_T$  are their related classes. The probability of this sentence can be computed as follows:

$$P(S) \cong \prod_{i=1}^T P(w_i | w_{i-1}) \cong \prod_{i=1}^T P(C_i | C_{i-1}) P(w_i | C_i) \quad (1)$$

In this research, we use fuzzy methods for clustering the words. The reasons of using fuzzy word clustering are discussed in the next section. Section 3 deals with feature vector description of words which is the important part in our new algorithm. Computing the class bi-gram statistics and class n-gram language models are discussed in section 4 and 5. The experimental results and comparison with similar methods go on in sixth section. Finally, in section 7, concluding remarks are made.

## 2. WHY FUZZY CLUSTERING?

As we said, the goal of clustering words is assigning similar words to a unique class based on their functions in the language. Since the n-gram statistics of words are the most applied models and many words in different contexts are represented beside different categories of words, we can not assign them only to one class. In the other hand, there are some words that can not be assigned in a special class and we should permit them to be assigned in more than one class.

For example consider the word "مهـر" –mehr- which have two different meanings. Its first meaning is kindness and the second one is the name of seventh month of Persian calendar. With these two different meanings, it will be assigned to two different classes. We have the same problem with some words that have more than one part of speech (POS) like "زيبـا" –zib-/ which appears as NOUN (a first name for girls in Persian language), ADJECTIVE (beautiful) and ADVERB (beautifully). As another example, consider the word "مرد" –mard- or –mord-. As in Persian language we use Arabic script, short vowels are not written and capitalization is not used. So this word has various meanings and POS tags, based on its pronunciation and meaning. In some situations it is pronounced as "mard" (man) and in some other situations it is pronounced as "mord" (he/she died). It is necessary

to mention that because separating the statistics of Persian homograph words based on their POS tags or meanings is very difficult, it was not the focus of in this research.

For the above reasons, we have reached to this idea that if one word can be assigned to more than one cluster, we may get better results after using this word clustering method. That is why we have thought about some fuzzy clustering algorithms. One of the important algorithms is fuzzy c-mean clustering technique.

In this algorithm, the centroids of classes and the degree of membership of i-th word in the j-th class is computed iteratively by the following formulas [5]:

$$u_{ij} = \frac{\left| \frac{1}{d^2(X_i, C_j)} \right|^{\frac{1}{q-1}}}{\sum_{k=1}^K \left| \frac{1}{d^2(X_i, C_k)} \right|^{\frac{1}{q-1}}} \quad (2)$$

$$C_j = \frac{\sum_{i=1}^N (u_{ij})^q X_i}{\sum_{i=1}^N (u_{ij})^q} \quad (3)$$

Where  $X_i$  is the feature vector of i-th word,  $C_j$  is the centroid of j-th class,  $K$  is the number of classes,  $d^2(X_i, C_j)$  is the distance between the i-th word and j-th class and  $q$  is the fuzzification factor [11] and controls the fuzziness of the resulting cluster.

The result of this updating process is vectors that represent virtual words as the centroids of classes. Based on the updates of new centroids of classes in each step, the memberships of each word to classes will be updating until the difference between two centroids of a specific class in two consecutive steps becomes less than a specified threshold. In this case, the algorithm will be terminated.

The parameters in this algorithm which we should decide about them are feature vectors, centroid initialization and distance measure. For centroid initialization and distance measure, we have used the previous applied methods and their combinations. For feature vectors, a new idea is introduced that has a good effect on decreasing the perplexity of language.

### 2.1. Centroid Initialization

In most practical applications of fuzzy clustering algorithms, not considering the type of elements (word, text, image, etc.), the procedure of initializing the centroids of classes is a random process. [8] In these methods, the elements membership matrixes in the classes are filled randomly at first. Then the loop for computing the degree of membership and the centroids of classes starts. In other applied methods, at the beginning of process some elements are selected as the centroids of classes and then the loop starts. According to [17], the second initialization method works better than the first one; but because of catching to local minima each of these methods is not recommended.

The other method for initialization is a new idea that we present in this paper and it leads to much better results. This method uses the results of crisp clustering methods for initializing the centroids at the beginning of algorithm. For example we can use the results of Martin [15] or Brown [7] algorithms which are the two most important algorithms in crisp word clustering.

### 2.2. Distance Measure

The different distance measures which we can use for computing the distances between words are similar to the usual distance measures that are used for vectors. One of them is Euclidian distance which its formula is:

$$d_e(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (4)$$

Another method for computing the distance between vectors is Cosine distance as follows:

$$d_c(x, y) = 1 - \frac{\sum_{1 \leq i \leq n} x_i y_i}{\sqrt{\sum_{1 \leq i \leq n} x_i^2 \sum_{1 \leq i \leq n} y_i^2}} \quad (5)$$

In 1998, Korkmaz [14] used the combination of these two measures by multiplying them to each other which had a good affection on the results.

$$d(x, y) = d_e(x, y) * d_c(x, y) \quad (6)$$

## 3. FEATURE VECTOR

Similar to fuzzy clustering for various types of data, in fuzzy word clustering, each word is represented by a feature vector. The ordinary vector for each word is the bi-gram statistics between that word and other words. Also, this ordinary vector can be defined as the bi-gram statistics between other words and that word. The other way for defining ordinary vectors is the combination of two above statistics. [13]

A new idea for defining the feature vectors is presented in this paper. It uses the POS statistics of each word as its feature vector. In Persian language, every single word can get different POS tags in different contexts based on its syntactic category and its meaning. In this research, the statistics of POS tags for each word have been extracted from "Persian Text Corpus" through the following steps: [3]

Step1. All texts of "Persian Text Corpus" were saved in a database; every entity in this database has two fields, word and tag.

Step2. The varieties of words and the frequency of occurrence of them were extracted.

Step3. The 20000 most frequent words were defined as the "vocabulary" of system; the other words were considered as "out of vocabulary".

Step4. The number of occurrence of each POS tag for each word of vocabulary was extracted. As we fixed the

number of POS tags to 166, the result of this step is saved in a 20000\*166 matrix which we called it "POS Matrix". For example, the word "zib/" in "Persian Text Corpus" has 3 different POS tags according to the text which it has been used in. It appears as non-quantitative simple adverb (ADV\_NQ\_SIM) 21 times, singular private noun (N\_SING\_PR) 68 times and simple adjective (ADJ\_SIM) 557 times. The statistics of other POS tags for this word are zero. Therefore the row of POS matrix for "zib/" has three non-zero elements (21, 68 and 557) and has zero values in all other elements.

Based on the above discussion, we have used a new type of feature vectors for word clustering. In our implemented method, each row of POS matrix, which is normalized by relative word frequency, is the feature vector of that word. Clearly, two words that have the same POS distribution have similar feature vectors and both of them are assigned to one class.

#### 4. BUILDING LANGUAGE MODEL

After clustering the words completely, it is necessary to extract bi-gram statistics between classes in order to build the bi-gram model based on clustering method.

In all crisp clustering algorithms, the sum of bi-gram statistics between words are used for extracting the bi-gram statistics between two classes,  $C_i$  and  $C_j$  by the following formula:

$$N(C_i, C_j) = \sum_{\substack{w_k \in C_i \\ w_l \in C_j}} N(w_k, w_l) \quad (7)$$

Which  $N(w_k, w_l)$  is the bi-gram statistics between two words,  $w_k$  and  $w_l$ .

With considering the fuzzy membership of words to classes, we should use a new approach to compute the bi-gram statistics between classes because the degree of membership of a word to a class is an important factor here. In the other words, instead of computing  $P(w_k, w_l)$ , we should use the formulas in which the degree of membership are considered. One of these formulas is the production multiplication of both membership functions in bi-gram statistics as follows:

$$N(C_i, C_j) = \sum_{w_k, w_l} u_k^i u_l^j N(w_k, w_l) \quad (8)$$

As there is another operation which is min function instead of multiplication, the equation 8 will change as:

$$N(C_i, C_j) = \sum_{w_k, w_l} \text{Min}(u_k^i, u_l^j) N(w_k, w_l) \quad (9)$$

Unexpectedly, the primary results show that multiplication produces better results than min function. For building language models based on clustering, we should compute the probabilities of the sentences. As we discussed in the previous section, it can be computed by multiplication of two consecutive words in a sentence. The equation 10 is the bi-gram probability for two

consecutive words  $w_R$  and  $w_L$ . [9]

$$P(w_R | w_L) = \sum_{C_j} [P(w_R | C_j) \sum_{C_i} P(C_j | C_i) P(C_i | w_L)] \quad (10)$$

Here,  $P(C_j | w_R)$  is the degree of membership of  $w_R$  to j-th class. With the following equations, the parameters of equation 10 can be computed:

$$P(C_j | C_i) = \frac{N(C_i, C_j)}{N(C_i)} \quad (11)$$

$$P(w_R | C_j) = \frac{P(C_j | w_R) P(w_R)}{P(C_j)} \quad (12)$$

After computing the above probabilities, we can compute the perplexity of context. The equation 13 is the formula for computing the entropy of context. By using the amount of entropy, we can compute the perplexity with equation 14 as follows:

$$H = -\frac{1}{N} \sum_{i=1}^N \lg(P(w_i | w_{i-1})) \quad (13)$$

$$\text{Perplexity} = 2^H \quad (14)$$

As we know, the perplexity of language is a good criterion for comparing different language models.

#### 5. EXPERIMENTAL RESULTS

The class language models extracted from our word clustering method are tested for evaluating two different criteria: perplexity and word error rate of a CSR system. We computed the perplexity of language with class bi-gram models on "Persian Text Corpus" which contains 8500000 words (463800 sentences). The test set is the last 43800 sentences of this corpus not included in training data. Table 1 shows the perplexity of Persian corpus based on fuzzy class bi-gram models with different number of classes.

Table 1- the perplexity of Persian Text Corpus based on fuzzy class bi-gram models with different number of classes

	Fuzzy Korkmaz method	Fuzzy c-mean (bi-gram feature vector)	Fuzzy c-mean (POS feature vector)
100 classes	1159.863	956.915	863.055
200 classes	1041.372	811.140	704.342
500 classes	1076.834	850.694	732.553

For comparing the new clustering method with Korkmaz algorithm [4, 12], which use a greedy algorithm for clustering words in definite number of classes, and fuzzy c-mean clustering with bi-gram feature vector, the two columns of table 1 shows the perplexity using these methods. According to this table, there is an obvious reduction in perplexity using our suggested clustering method.

To evaluate our statistical language models in CSR system, we used SHARIF speech recognition system [2], which is a Persian speaker independent continuous speech recognition system. This system performs modeling of mono-phones using Hidden Markov Model (HMM) and utilizes the word search algorithm described in [16] for word recognition. In this algorithm, while recognizing the phonemes, the lexicon tree is also searched in order to find the best word sequence according to the phoneme sequence.

To run experiments, the HMMs were trained for each 30 phonemes of Persian language using 5940 sentences (about 5 hours of read speech) of FARSDAT speech database [6]. We performed the experiments on 140 sentences of FARSDAT database which don't overlap with the training data.

In general, in speech recognition systems, the language model score can be combined with acoustic model score through two methods: "during search" and "at the end of search" [10]. In this paper we have used "during search" method. In "during search" method when search process recognizes a new word while expanding the different hypothesis, the new hypothesis score is computed via multiplication of following three terms: the n-gram score of new word, the acoustic model score of new word and current hypothesis score.

Table 2 presents the word error rates (WER) obtained when incorporating different class n-gram models in the CSR system. The two first rows show the results using two other methods. The results show that a considerable reduction in word error rates has been achieved by using class n-gram models.

Table 2- The percentage of accuracy and correctness obtained with incorporating different class n-gram models in CSR system

Language Model Type	Class Number	Accuracy	Correctness
Fuzzy Korkmaz method	100	69.5%	70.7%
	200	72.4%	73.8%
	500	71.6%	76.2%
Fuzzy c-mean (bi-gram feature vector)	100	71.9%	73.6%
	200	74.2%	76.9%
	500	73.4%	78.0%
Fuzzy c-mean (POS feature vector)	100	73.2%	77.9%
	200	75.6%	81.7%
	500	74.8%	79.9%

## 6. CONCLUDING REMARKS

In this paper, we presented a new fuzzy algorithm for word clustering. The different criteria parameters in fuzzy word clustering method were considered such as feature vectors of words, centroid initialization and distance measure. To reach an optimal clustering method, these parameters were evaluated several times. The criterion for choosing appropriate parameters was the perplexity of

context and the less perplexity is better. Finally the extracted class-based models of this research were incorporated in Persian Continuous Speech Recognition System and decreased the word error rate by about 12 percent.

## REFERENCES

- [1] Allen J., "Natural Language Understanding", The Benjamin-Cummings Publishing Comp-any, 1995.
- [2] Babaali B., Sameti H., "The Sharif Speaker-Independent Large Vocabulary Speech Recognition System", The 2nd Workshop on Information Technology & Its Disciplines, Kish Island, Iran, Feb. 24-26, 2004.
- [3] Bahrani M., Sameti H., Hafezi N., Momtazi S., "A New Word Clustering Method for Building n-gram Language Models in Continuous Speech Recognition Systems", submitted in ICASSP07.
- [4] Bazargani N. "Word Clustering for Building Persian Language Models", Master of Science Thesis, Amirkabir University, Tehran, Iran, 2004.
- [5] Bezdek J.C., "Pattern Recognition with Fuzzy Objective Function Algorithms", Plenum Press, New York 1981.
- [6] Bijankhan M. et al., "FARSDAT-The Speech Database of Farsi Spoken Language", In Proc. The 5<sup>th</sup> Australian Int. Conf. on Speech Science and Tech., Vol. 2, Perth, 1994.
- [7] Brown P.F., Della Pietra V.J., deSouza P.V., Lai J.C., Mercer R. L., " Class-Based n-gram Models of Natural Language", Computational Linguistics, 18(4):467-479, 1992.
- [8] Fung G., "A Comprehensive Overview of Basic Clustering Algorithms", 2001.
- [9] Jardino M., Adda G., "A Class Bi-gram Model for Very Large Corpus", Proceeding 3<sup>rd</sup> European Conference on Spoken Language Processing, Yokohama, Japan, 1994.
- [10] Harper M. P., Jamieson L. H., Mitchell C. D., Ying G., "Integrating Language Models with Speech Recognition", AAAI-94 Workshop on the Integration of Natural Language and Speech Processing, pp. 139-146, Aug. 1994.
- [11] Klawonn F., Höppner F., "What Is Fuzzy about Fuzzy Clustering? Understanding and Improving the Concept of the Fuzzifier", IDA 2003, pp. 254-264.
- [12] Korkmaz E.E., "A Method for Improving Automatic Word Categorization", M.Sc thesis, Department of Computer Engineering, Middle East Technical University, September 1997.
- [13] Korkmaz E.E., Ucoluk G., "A Method for Improving Automatic Word Categorization", Computational Natural Language Learning, CONLL97.
- [14] Korkmaz E.E., Ucoluk G., " Choosing a Distance Metric for Automatic Word Categorization", In Proceeding of NeMLaP3/CONLL98, Workshop on New Methods in Language Processing and Computational Natural Language Learning, Sydney, 1998.
- [15] Martin S., Liemann J., Ney H., "Algorithms for Bi-gram and Tri-gram Word Clustering", Speech communication 24, 1998.
- [16] Ney H., Haeb-Umbach R., Tran B. H., Oerder M., "Improvements in Beam Search for 10000-Word Continuous Speech Recognition", IEEE Int. Conf. on Acoustics, Speech and Signal Processing, pp. 13-16, March 1992.
- [17] Zhang L., "Comparison of Fuzzy c-means Algorithm and New Fuzzy Clustering and Fuzzy Merging Algorithm", <http://www.cse.unr.edu/~lzhang/>, 2001.