INCORPORATING LANGUAGE CONSTRAINTS IN SUB-WORD BASED SPEECH RECOGNITION

Hakan Erdoğan, Osman Büyük, and Kemal Oflazer

Faculty of Engineering and Natural Sciences
Sabanci University, Orhanlı Tuzla, 34956, İstanbul
{haerdogan,oflazer}@sabanciuniv.edu.tr,
buyuk@su.sabanciuniv.edu.tr.

ABSTRACT

In large vocabulary continuous speech recognition (LVCSR) for agglutinative and inflectional languages, we encounter problems due to theoretically infinite full-word lexicon size. Sub-word lexicon units may be utilized to dramatically reduce the out-of-vocabulary rate in test data. One can develop language models based on sub-word units to perform LVCSR. However, it has not always been beneficial to use sub-word lexicon units, since shorter units have higher acoustic confusability among them and language model history is effectively shorter as compared to the history in full-word language models. To reduce the aforementioned problems, we propose using the longest possible sub-word units in our lexicon, namely half-words and full-words only. We also incorporate linguistic rules of half-word combination into our statistical language model. The language constraints are represented with a rule-based WFSM which can be combined with an N-gram language model to yield a better and smaller language model. We study the performance of the proposed system for Turkish LVCSR, when the language constraint takes the form of enforcing vowel harmony between stems and endings. We also introduce novel error-rate metrics that are more appropriate than word-error-rate for agglutinative languages. Using half-words with a bi-gram model yields a significant reduction in word-error-rate as compared to a bi-gram full-word model. In addition, combining a tri-gram half-word language model with the vowel-harmony WFSM improves the accuracy further when rescoring the bi-gram lattices.

1. INTRODUCTION

Hidden Markov Models are used in modern speech recognition systems for acoustic modelling. Statistical or rule-based language models can be used in order to represent the recognized language. If the speech recognition application permits the use of a limited-vocabulary hand-crafted grammar, then high recognition accuracies may be achieved. In such applications, when the speaker utters a sentence which is not covered by the grammar, the system cannot recognize the sentence. In a system-initiative spoken dialog system, if the user can be directed with well-prepared questions, a dynamic limited grammar system may achieve great performance. However, for mixed-initiative systems and for other applications such as dictation and broadcast news transcription, a large vocabulary system with a statistical grammar (language model) is required. Such systems are called large vocabulary continuous speech recognition (LVCSR) systems.

N-gram language models for LVCSR achieve acceptable performance for English. In English dictation systems, an accuracy of 90-95% may be achieved. On the other hand, in agglutinative and inflectional languages like Turkish, Finnish, Czech, etc., LVCSR is problematic when only the words in the language are used as lexicon units. Total number of words is very high and any vocabulary size becomes inadequate [1-5]. A lexicon that contains sub-word units may be utilized as a solution to the coverage problem [1-7]. These sub-word units are typically morphological components of words or syllables in the language [6-7].

Despite solving the coverage problem, using sub-word units in LVCSR for agglutinative languages does not always achieve acceptable performance [1,4,8]. A reason for the bad performance is the shortness of sub-word units as compared to full-words. As expected, shorter units become more acoustically confusable with each other. Also, LVCSR decoders have a tendency to insert short units into wherever they can since matching short units to acoustic data is easier. Another reason for the performance drop in sub-word systems is the smaller effective language model history size while using N-gram language models. In a sub-word system, the language model effectively uses a shorter history as compared to a full-word based system. The shorter the sub-words, the shorter the effective history one uses in an N-gram language model. As a result, appropriately choosing sub-word units is crucial for LVCSR systems. The main criteria in this choice should be to cover the words in the language as much as possible while maintaining small acoustic confusability between units not to decrease the recognition rates. In other words, unnecessary and short units should not be inserted into the lexicon of the speech recognizer while words in the language are covered with an acceptable rate. In this paper, we propose some new ideas to construct a lexicon of half-word and full-word units for an LVCSR system in Turkish.

Another potential problem while using sub-word units in speech recognition is that, the recognizer may output a sequence of sub-words that may not form a legitimate word sequence in the language. In this paper, we propose to use a rule-based weighted finite state machine (WFSM) to accept only allowable sub-word unit sequences. This WFSM can be composed with an N-gram language model to improve the overall language model and reduce its size. In speech recognition, modelling the language constraints with a WFSM is a new approach and with this approach not only language model size can be decreased but also recognition rates can be improved.

This paper is organised in the following way. In section 2, we review N-gram language models. In section 3, we discuss ways to split words and reveal our choices for splitting. We provide details of our new language model which is a composition of an N-gram WFSM and a rule-based WFSM in section 4. Section 5 discusses new evaluation metrics for LVCSR of agglutinative languages. In section 6, the experimental setup and results are provided. The conclusions and future plans are summarized in the final section.

2. STATISTICAL LANGUAGE MODELS

A statistical language model enables to compute the probability of a sentence in a language. If we assume that a sentence is constructed from words, the sentence probability can be written as follows [9]:

$$P(W) = P(w_1, w_2, ..., w_n) = \prod_{i=1}^{n} P(w_i \mid w_1, ..., w_{i-1}).$$

In this equation W represents the sentence and each w_i represents a word in the sentence. N-gram language models approximate each term in the right hand side of the equation by using only N-1 words in the history. The equation becomes:

$$P(W) \approx \prod_{i=1}^{n} P(w_i \mid w_{i-N+1}, ..., w_{i-1})$$

Usually, a large text corpus is used to calculate each N-gram probability. Smoothing techniques are used to avoid assigning zero probabilities for unseen N-grams [9].

As indicated in the introduction section, since the number of words in Turkish is very large, the use of words as the lexicon entries in an N-gram language model is not reasonable. In this case, utilization of sub-word units will be necessary and beneficial. When sub-word units are used, we replace the sequence of words in a sentence with the corresponding sequence of sub-words and perform N-gram language modelling using sub-words as basic units instead of full-words.

3. SPLITTING WORDS INTO SUB-WORDS

There is more than one way to split a word into its parts, especially in Turkish and other agglutinative languages. Some splitting options can be listed as follows:

1. Full-word (no split),
2. Stem + ending,
3. Stem + morph1 + morph2 + …,
4. Syllabifying.

In this paper, we will progressively use first, second and fourth techniques in order to split a word into its parts. In forming a hybrid lexicon, a word will be included in the vocabulary as a full word if it can be found among most frequent stems. If this is not possible, we will try to split the word into two half-words as stem + ending. If this split is also not possible, then the word will be syllabified. We will not use the third choice above, since this splitting procedure which can be seen as a morphological analysis of the word will increase acoustic confusability by unnecessarily choosing small units.

After this splitting procedure, speech recognizer's lexicon will include the following units:

1. Stems (used as a full-word or a half-word). Examples: ev/house, sokak/street, his/feeling.
2. Endings (used as final half-words only). Examples: -ler, -lar, -lerde, -imizin, -imizdekiler.
3. Syllables. Examples: <a>, <e>, <de>, <bak>, <kır>, <trak>,<ler#>.

Some syllables can also act as single-syllable stems, like "bak/look". In order to distinguish the stem "bak" from syllable "bak", an extra symbol is needed. Syllable units are contained within angular brackets to avoid confusion. We also append a "#" symbol to indicate word-final syllables to enable conversion of syllable sequences to word sequences at the output. Similarly "-" symbol is added in front of endings in order to prevent confusion with other units and to enable word reconstruction at the output

We split words into stem + ending parts by using a two-level morphological analyzer [10]. When there is more than one morphological analysis possible, we choose the one with the longest stem. We also require that the ending part is at least 2 phonemes long to avoid using acoustically confusable short endings. Our final lexicon contains all frequent stems, endings and syllables in the training data.

4. A NOVEL SUB-WORD LANGUAGE MODEL

N-gram language models can be represented by a weighted finite state machine (WFSM) [11]. Software packages are available to train N-gram language models and convert them into WFSMs [12]. The resultant N-gram WFSM is usually large and complex. The weights in this WFSM correspond to N-gram probabilities mentioned in section 2 and they are learned from text data. In order to obtain reliable probabilities, a text corpus containing millions even billions of words is required. This text corpus should be cleaned, tokenized and checked for typos before it can be used in N-gram language model training. Since this cleaning procedure is usually not perfect, some noise is always left in the training data and this noise causes imperfect weights in the WFSM. Also, to avoid zero probabilities and to enable detection of word sequences that were never seen in training data, linguistically impossible word sequences receive a small but non-zero probability in the language model. If acoustic evidence strongly prefers a word sequence that was never seen in training data, that sequence can be emitted by the decoder. When only an N-gram statistical language model is used in a sub-word recognizer, the decoder can output linguistically impossible sentences in the language of interest. For example when the hybrid lexicon introduced in the previous section is used; decoder could output such a sentence in Turkish LVCSR:

- <a> <lis#> ev -inde -ler <a> <vi> <za#> bul –ındırmayı sev -mez.

Assume that the correct utterance is "Alice evinde avize bulundurmayı sevmez. / Alice does not like to have a chandelier in her house." There are linguistically impossible sub-word sequences in the decoder output. Errors in this recognition result can be listed as follows:

1. Two endings consequently recognized. Example: "–inde –ler" is meaningless.
2. The ending following a stem does not obey the Turkish vowel harmony rule. Example: "bul –ındırmayı", should be "bul –undurmayı".
3. Although the word "avize/chandelier" is included in the lexicon of the speech recognizer, a sequence of syllables similar to the word with a better acoustic match to the acoustic data may be preferred, such as "<a> <vi> <za#>".

In fact when the statistical language model is well trained, these types of problems will be reduced, but never avoided totally. In order to remove the problems listed above we are proposing to use a new rule-based WFSM that accepts only linguistically acceptable sub-word sequences. The main role of the WFSM is to enforce vowel harmony between stems and endings. It also enforces the correct ordering of stems and endings to form a valid word sequence.

The rule based WFSM is depicted in Figure 1. The WFSM contains parallel alternative branches to form a single allowable word that obeys vowel harmony rule in Turkish and accepts a sequence of such words.

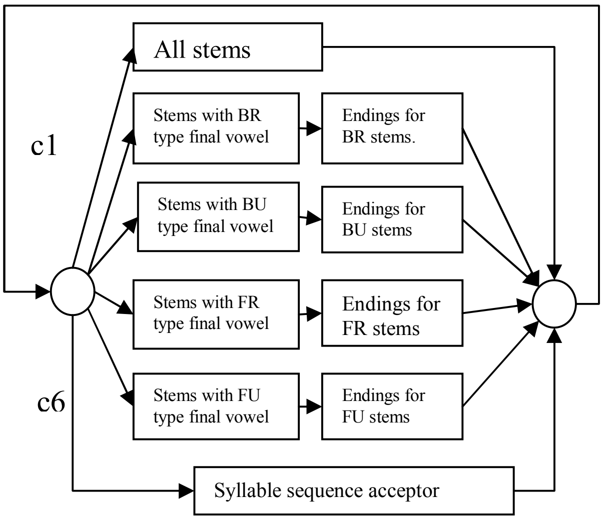
All stems
c1
Stems with BR type final vowel
Endings for BR stems.
Stems with BU type final vowel
Endings for BU stems
Stems with FR type final vowel
Endings for FR stems
c6
Stems with FU type final vowel
Endings for FU stems
Syllable sequence acceptor

Figure 1 Rule-based WFSM that accepts sub-word sequences that obey vowel harmony in Turkish.

To realize the WFSM, we form four distinct classes of stems depending on their final vowel type. The vowel type depends on the position of the tongue (front (F) or back (B)) and lips (rounded(R) or un-rounded (U)). In Turkish, there are eight vowels "a, ı, e, i, o, u, ö, ü". The first two are back and un-rounded (BU), the next two are front and un-rounded (FU), followed by two back and rounded (BR) and two front and rounded (FR) vowels. In Turkish language, when attaching suffixes to stems, vowels of suffixes may change according to the last vowel of the stem they are attached to. This property is called vowel harmony. For example suffixes –lar and –ler both make plural nouns, but we have to choose the one that has vowel harmony with the preceding stem. There are some rare exceptions to this rule when for example a palatal "l" is the last consonant in the stem (examples: alkol, ampul, tuval). In that case, even if the last vowel is a back vowel, it acts as a front vowel. We incorporate such exceptions into our rules easily by including such stems into the front vowel classes instead of the back vowel classes. The allowable endings that attach to the class of stems can be found from the training data. We group together all endings that were attached to a class of stems in the training data. Even though the stem classes are distinct, there is overlap between ending classes since some suffixes do not change form according to vowel harmony (-ki, -ken) and some suffixes only change form according to being front or back regardless of the roundness attribute (-lar or –ler, -da or -de). The ending classes are corrected by expert intervention when necessary to avoid errors due to noise in training data.

In Figure 1, the weights in the branches are shown as c1,...,c6. By using a small weight for the syllable branch, the probability of choosing this branch can be decreased. Thus, the third problem listed above will not be observed too frequently.

When WFSM shown in Figure 1 is composed with an N-gram WFSM, a new WFSM is obtained which not only uses the statistical information obtained from a large text corpus but also enforces vowel harmony and a linguistically correct half-word order. In the combined WFSM, decoder will never produce a recognition result which is not accepted by the rule based WFSM in Figure 1, effectively setting their language model probabilities to zero.

The WFSM displayed in Figure 1 may also be used in a syllable-based speech recognition experiment when the lexicon only contains syllables. When an N-gram syllable language model is composed with the rule based WFSM, a more powerful language model can be obtained since this language model will block meaningless syllable sequences. When such a rule based WFSM is not used, syllable-based experiments usually generate meaningless syllable sequences. As a result, a big improvement can be expected when such a WFSM is used in the language model.

In some of our experiments, we set the weight for the syllable branch to 0 (c6=0 in Figure 1). This effectively disables syllable units and we use only half-words and full-words in recognition. In that case, we call our system a “half-word system”, since we only use full or half-words as lexicon units. When we set all c1,c2,…,c6 to be equal, we call this system “hybrid system” since syllable sequences are also possible in addition to half-word sequences. In the test-set that we worked with, half-word system yielded a better recognition result as compared to the hybrid system. This is due to the fact that we did not have a coverage problem with the half-word system, so we did not need the syllables to get increased coverage. We also did not optimize the weights c1 through c6 in our hybrid experiment and used equal values for them. We expect better results with the hybrid system when we optimize the weights and when we test our system on different test-sets which have reduced coverages while using only half-words. We present our detailed results in section 6.

5. EVALUATION METRICS

Although word-error-rate (WER) is a well-accepted method for evaluating speech recognition performance, for agglutinative languages like Turkish, we feel the need to define new metrics that depend on recognizing sub-words. Occasionally a recognizer may recognize the stem of the word correctly but fail in recognizing the ending part. When we use WER as a metric, we count this as a single substitution error. However since the stem of the word is recognized correctly, we may wish to count it as one correct recognition and one substitution error due to misrecognizing the ending. Another alternative is to remove the ending half-words and calculate error-rate using only the stems. This also gives an idea about the accuracy of the speech recognition system in recognizing the main part (stem) of the words which is the most semantically informative part.

So, we use three different metrics in evaluating our speech recognition systems.

1. **WER**: word-error-rate. We form word sequences from our recognized sub-word sequences (this is not usually ambiguous since a stem + ending is a word and a syllable sequence that ends in a syllable with # symbol forms a word). For statistical-only language models, we may have un-allowed sequences of sub-words. In those cases, we combine parts that we can combine to form words and the parts that cannot be combined are left as is.
2. **HWER**: half-word-error-rate. We re-split the words into two (stem + ending) in both reference and recognized (hypothesis) word sequences consistently. We use the same splitting algorithm as we used during language model training with one exception that the ending part is allowed to be of length one character (one phoneme) in this case. We calculate the error-rate among these recognized half-words.
3. **STER**: stem-error-rate. After dividing the words into two parts, we delete the second part (if the second part exists) for both the reference and the hypothesis texts and calculate the error-rate among stems only.

In our experiments, we provide error-rates using the three metrics introduced above.

6. EXPERIMENTS AND RESULTS

To obtain meaningful LVCSR results, we collected ample amount of acoustic and text data in Turkish. Although minimal as compared to English LVCSR systems, our acoustic and text data are the largest amount of data (as we know of) used in Turkish LVCSR to date.

Our acoustic data consists of read speech (16 kHz, 16 bits) collected using a headset microphone (Plantronics Audio 50) connected to a laptop. 34 of 37 hours of speech data is used for training Hidden Markov Models and 3 hours of the data is set aside for testing. 367 different speakers with equal gender distribution were recorded. All speakers read around 120 sentences among about 1000 phonetically balanced sentences. HTK 3.2 is used to train tri-phone HMM models with tied states [13]. Twelve Gaussian mixtures were used for modelling each state emission probabilities. The test data we used in this paper contains 88 different sentences with 717 words in the sports news domain read by 16 different speakers. The topics of the sentences were chosen from sports news domain since our language modelling text data contains sports news data as well.

We collected text data with 5.5 million different sentences and 81 million words, mostly from internet news web sites and e-books in Turkish. The news data contains daily life, sports news and columnists. We obtained 1,170,526 unique words in this text data. The large number of unique words in our text data is due to the agglutinative nature of Turkish language. A bi-gram language model is trained from the text training data. Also, a tri-gram language model is trained from a 2 million sentence subset of the training data due to computational reasons.

We have used four different lexicon types in our speech recognizer during initial experiments:

1. Word lexicon: only most frequent 30000 words in the lexicon
2. Syllable lexicon: only most frequent 2000 syllables in the lexicon
3. Half-word lexicon: most frequent 10000 stems and 3000 endings
4. Hybrid lexicon: most frequent 18000 stems, 3000 endings 2000 and syllables

For all lexicon types, we also manually added, to our acoustic dictionary, stems that are in a large group of test sentences that contain our test data (large test sentences contain literary novel and sports news domains) but are not in the most frequent stems in the training data. This procedure added 132 extra entries to the word lexicon, 73 extra entries in the half-word stem lexicon and 53 extra entries in the hybrid stem lexicon. Assuming that the morphological productivity is the main problem causing high out-of-vocabulary (OOV) rates, we performed such a manual adjustment to remove OOV words due to unknown stems. We confirmed that the OOV rates for syllable, half-word and hybrid lexicons are indeed much lower than that for the word lexicon in our large group of test sentences as shown in Table 1.

Lexicon	Vocabulary size	OOV rate
Word	30132	18.16%
Half-word	13073	1.66%
Syllable	2000	6.04%
Hybrid	23053	0.66%

Table 1 OOV rates for different lexicon types in a larger test-set after adding test stems to the dictionary.

Unit recognition error-rates are obtained for each lexicon type using a bi-gram language model in HTK [13]. The results are presented in Table 2[1]. However, these results do not provide a direct comparison between different lexicon types since units are different in each recognizer. We observe in this table that the hybrid lexicon yields worse results as compared to the half-word lexicon, so for the following experiments, we do not provide results for the hybrid lexicon.

Lexicon Type	Sentence correct %	Unit correct %	Unit accuracy %
Word	11.93	60.57	47.05
Syllable	2.85	54.18	53.11
Half-word	10.70	60.57	57.88
Hybrid	10.01	55.96	53.27

Table 2 Unit recognition rates using bi-gram models

In Table 3, we provide a better comparison among word, syllable and half-word lexicon systems by comparing their comparable WER, HWER and STER performances[2]. These results clearly show that half-word-based language models are superior to word-based language models. Syllable-based language models provide the worst performance since syllables are not constrained to form a valid word when using a statistical language model.

Lexicon Type	Sentence correct %	WER	HWER	STER
Word	11.93	52.95	42.18	43.21
Syllable	2.85	68.80	64.63	62.71
Half-word	11.99	45.11	40.12	36.30

Table 3 Comparable percentage error-rates for different lexicon types using a bi-gram language model.

As it can be observed from the error-rates, best results are obtained using a half-word lexicon with a bi-gram language model. To further improve the results, we perform lattice-rescoring experiments on the half-word system to apply the rule-based vowel harmony WFSM and a tri-gram WFSM to see how much improvement can be obtained. We obtain lattices using the half-word bi-gram system for three randomly chosen speakers from the test-

[1] After the paper was submitted, we observed that we can obtain better results for word and hybrid lexicons by adjusting decoding parameters such as grammar weight. Please see Osman Buyuk's Masters Thesis for new results.

[2] The sentence correct results for half-word lexicon in Tables 2 and 3 are different due to word split mismatches in hypothesis and reference transcripts in the half-word experiment. In Table 3, the mismatches are automatically corrected by consistently re-splitting words.

set. Rule based WFSM is obtained by setting c6=0 (no syllable lexicon) in Figure 1. Lattice rescoring experiment is depicted in Figure 2.

The results of the lattice-rescoring experiment are shown in Table 4. The last row shows the best result than can theoretically be attained by rescoring the lattice. Here, we see that tri-gram language modelling indeed improves the results significantly. Rule-based vowel harmony WFSM still improves upon tri-gram language model performance, although the additional gain is around 3% relative. Applying rule-based WFSM directly to rescore bi-gram lattices results in a bigger improvement than applying it after tri-gram rescoring since bi-gram is suboptimal and returns more half-word sequences that are disallowed according to the linguistic rules.

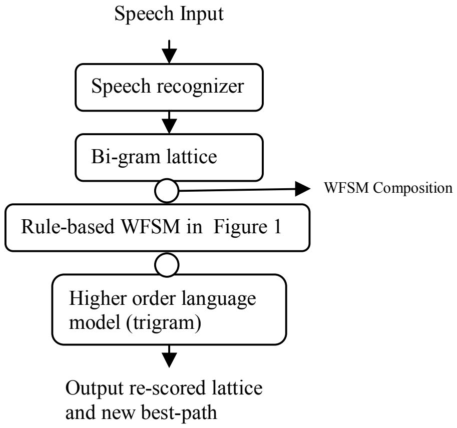
Speech Input
Speech recognizer
Bi-gram lattice
WFSM Composition
Rule-based WFSM in Figure 1
Higher order language model (trigram)
Output re-scored lattice and new best-path

Figure 2 Lattice rescoring paradigm used in testing rule-based WFSM and tri-gram language models.

Language Model used with half-word lexicon	Sentence correct %	WER	HWER	STER
Bi-gram	11.74	40.51	36.51	31.53
Vowel harmony WFSM (Rule based)	11.79	39.66	34.89	30.26
Tri-gram	19.70	33.06	30.19	25.83
Rule-based and Trigram	19.77	32.54	29.10	25.28
N-best bi-gram (oracle)	31.44	21,25	19.20	15.14

Table 4 Percentage error-rates obtained by various language models

7. CONCLUSION AND FUTURE WORK

We have introduced novel methods for determining sub-word lexicons and developing language models for large vocabulary continuous speech recognition for Turkish language. The techniques should be applicable to other agglutinative or inflectional languages as well. The combination of a rule-based WFSM that enforces correct half-word order and vowel harmony between stems and endings and a tri-gram language model achieves encouraging LVCSR performance on a sports news domain speech corpus.

We plan to improve language modeling for Turkish by exploring and improving the hybrid lexicon language models. We also plan to explore class-based language models suitable for the Turkish language in the future.

8. ACKNOWLEDGEMENTS

The authors would like to thank Dilek Hakkani-Tur, Giuseppe Riccardi and Murat Saraclar of AT&T Research for valuable discussions.

9. REFERENCES

[1] Kenan Carki, Petra Geutner, and Tanja Schultz, “Turkish LVCSR: Towards better Speech Recognition for Agglutinative Languages,” ICASSP, 2000.

[2] Kadri Hacioglu, Bryan Pellom, Tolga Ciloglu, Ozlem Ozturk, Mikko Kurimo and Mathias Creutz, "Word splitting for Turkish LVCSR," *SIU,* 2003.

[3] Kadri Hacioglu, Bryan Pellom, Tolga Ciloglu, Ozlem Ozturk, Mikko Kurimo and Mathias Creutz, "On lexicon creation for Turkish LVCSR," *Eurospeech,* 2003.

[4] Ebru Arisoy, *Turkish dictation system for radiology and broadcast news applications,* M.S. Thesis, Bogazici University, 2004.

[5] Helin Dutagaci and Levent M Arslan, "A comparison of four language models for large vocabulary Turkish speech recognition," *ICSLP,* 2002.

[6] Jeff Bilmes and Katrin Kirchhoff, "Factored language models and generalized parallel backoff," *Human Language Technology Conference,* 2003.

[7] Vesa Siivola, Teemu Hirsimaki, Mathias Creutz and Mikko Kurimo, "Unlimited vocabulary speech recognition based on morphs discovered in an unsupervised manner," *Eurospeech,* pp. 2293--2296, 2003.

[8] W. Byrne, J. Hajie, P. Ircing, F. Jelinek, S. Khudanpur, P. Krbec and J. Psutka, "On large vocabulary continuous speech recognition of highly inflectional language - Czech," *Eurospeech,* 2001.

[9] Daniel Jurafsky and James H Martin, *Speech and language processing,* Prentice Hall, New Jersey, 2000.

[10] Kemal Oflazer, “Two-level Description of Turkish Morphology,” *Literary and Linguistic Computing, Vol.9 No.2,* 1994.

[11] M Mohri and M Riley, "Weighted finite-state transducers in speech recognition (tutorial)," *ICSLP,* 2002.

[12] AT&T grm tools library, http://www.research.att.com/projects/mohri/grm.

[13] S. Young, D. Ollason, V. Valtchev and P. Woodland, *The HTK book (for HTK version 3.2)*, Entropic Cambridge Research Laboratory, March 2002.