**OLAKUNLE ANIFOWOSE**

**Department of Electrical and Computer Engineering, Temple University**

**A short Literature Review on**

**NLP on Spoken Documents Without ASR**

***Tua31270@temple.edu***

1. **Summary**

This report is a review of paper NLP on Spoken Documents without ASR by Mark Dredze, Glen Coppersmith and Ken Church. This paper investigates the possible fusion of several disciplines such as automatic speech recognition (ASR), machine Learning, natural language processing, text classification and information retrieval. The ultimate goal is to simplify the processing of spoken documents and optimize resources. Simply fusion of the disciplines above tends to increase errors. In this paper an alternative approach is proposed which involves applying text processing directly to the speech without depending on ASR. The paper claims this method finds long repetitions in speech and clusters them into pseudo terms and that document clustering and classification is very effective on pseudo terms.

1. **Introduction**

**Natural Language processing** (**NLP**) is a field of computer science and linguistics concerned with the interactions between computers and human (natural) languages. Natural-language understanding is can be referred to as an Artificial intelligence problem because natural-language recognition requires extensive knowledge about the outside world and the ability to manipulate it. NLP has significant overlap with the field of computational linguistics, and is often considered a sub-field of artificial intelligence. Research into modern statistical NLP algorithms requires an understanding of a number of disparate fields, including linguistics, computer science and statistics.

1. **NLP on Spoken Documents Without ASR**

The goal of this approach to NLP is to reduce the annotation requirement for constructing a competent Large Vocabulary Continuous Speech Recognition (LVCSR) System. Traditionally applying NLP to speech corpora requires a high resource ASR system to provide automatic word or phonetic transcripts Natural language processing tools can play a key role in understanding text document collections. In a large collection of text, NLP tools can classify documents by category and organize documents into similar groups for a high level view of the collection (clustering). These tools can be applied so that the user can quickly see the topics covered in the news articles, and organize the collection to ﬁnd all articles on a given topic. The ultimate goal in this paper is to apply NLP tools to a large collection of speech thereby allowing users to understand the contents of a speech collection while listening to only small portions of the audio. Previous works have investigated the problem of topic identiﬁcation from audio documents using features extracted from a speech recognizer. Considerable interest was shown in difﬁcult cases where the training material is minimally annotated with only topic labels. In cases like this, the lexical knowledge that is useful for topic identiﬁcation are usually not available, therefore automatic methods for extracting linguistic knowledge useful for distinguishing between topics are relied upon. The research also investigates the problem of topic identiﬁcation on conversational telephone speech from the Fisher corpus under a variety of increasingly difﬁcult constraints. However, unlike text, which requires little or no preprocessing, audio ﬁles are typically ﬁrst transcribed into text before applying standard NLP tools. Automatic speech recognition (ASR) solutions, such as large vocabulary continuous speech recognition (LVCSR) systems, can produce an automatic transcript from speech, but they require signiﬁcant development efforts and training resources. Terms that may be most distinctive in particular spoken documents often lie outside the predeﬁned vocabulary of an off-the-shelf LVCSR system. This means that unlike with text, where many tools can be applied to new languages and domains with minimal effort, the equivalent tools for speech corpora often require a signiﬁcant investment. This paper builds upon a simple method for finding repetitions in speech described by Jansen (2010). The approach described by Jansen identifies long repeated patterns in acoustic signal. This acoustic repetition often corresponds to terms useful for information retrieval tasks. Applying this discovery procedure to a large untranscribed corpus of speech creates a vast number of repeated regions that are subsequently grouped using a simple graph-based clustering method. The resulting groups are called pseudo-terms since they typically represent a single word or phrase spoken at multiple points throughout the corpus. Each pseudo-term takes the place of a word or phrase in bag of terms vector space model of a text document making it easy to apply standard NLP algorithms. Jansens method for identifying repeated speech patterns uses dotplots.

1. **Dotplots Algorithm**

Dotplots are a graphical method of comparing sequences, initially developed for identifying recurring protein sequences in the bioinformatics community and later extended to large scale text processing more generally. Given a character string X = x1x2 ...xn, where each xi is an element of some alphabet, the traditional dotplot on X is an n×n Boolean matrix deﬁned as Mij = (xi == yj ). Substrings repeated at different points in X manifest themselves as diagonal line segments in the visualization of M; substrings consisting of a single repeated token that manifest themselves as large blocks. Recent work extends these techniques to acoustic processing in order to identify repeated intervals of speech or audio. In this section, a detailed explanation deﬁning how this extension is accomplished is given and how using standard image processing techniques, motivate a new efﬁcient search algorithm to discover repeated intervals.

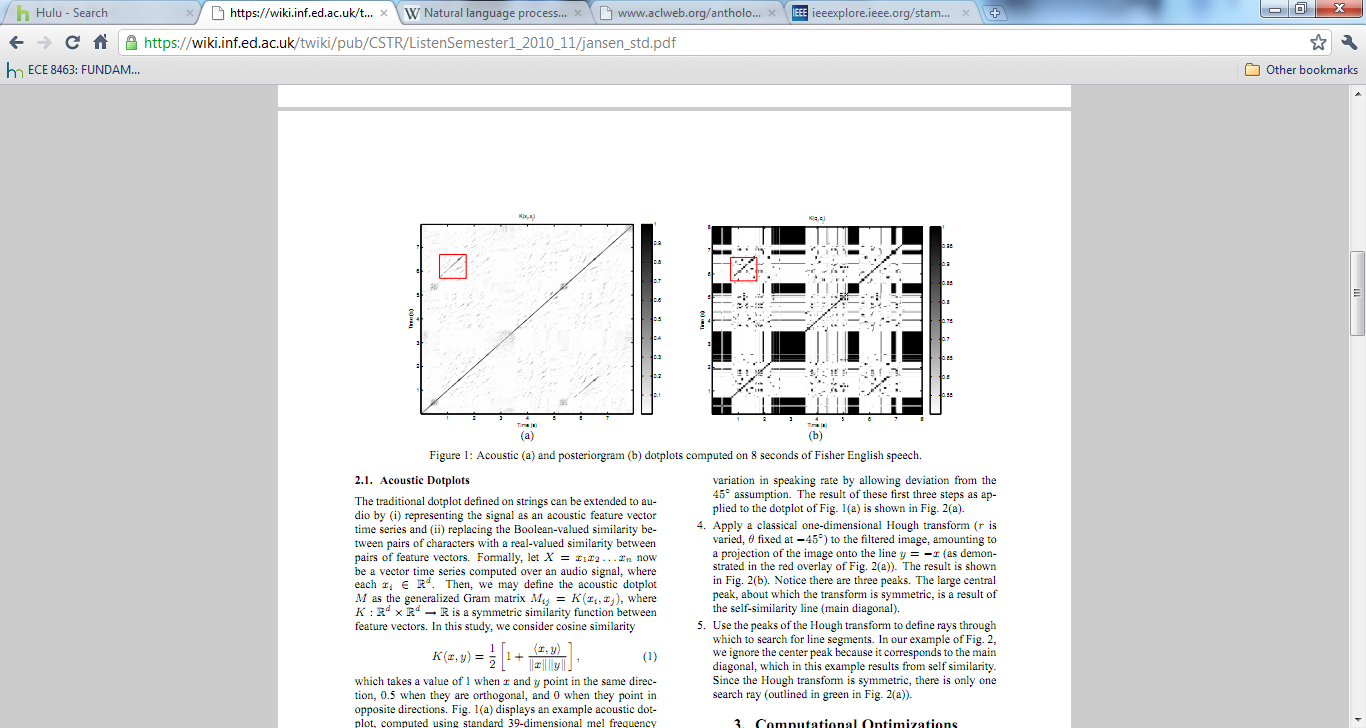


Figure 1: Acoustic (a) and posteriorgram (b) dotplots computed on 8 seconds of Fisher English speech.

**Acoustic Dotplots**

Traditional dot defined on strings can be extended to audio by representing the signal as an acoustic feature vector time series and replacing the Boolean-valued similarity between pairs of character with a real-valued similarity between pair of feature vectors. A vector time series is computed over an audio signal and represented as X=. The acoustic dotplot can then be defined as the generalized Gram matrix where K is a symmetric similarity function between feature vectors. Cosine similarity is used

Which takes a value of 1 when x and y point in the same direction, 0.5 when they are orthogonal and 0 when they point in opposite directions. Figure 1(a) above shows and example acoustic dotplot computed using standard 39-dimensional mel frequency cepstral coefficient features on 8 seconds of speech from the fisher English Corpus. Because K is symmetric in its arguments, the dotplot is symmetric and the main diagonal line arises from self-similarity. Using the acoustic dotplot computed for a pair of utterances the problem of ﬁnding repeated intervals can be reduced to a search for diagonal line segments. This is a common image processing task that can be accomplished efﬁciently using the following steps:

1. Transform the dotplot matrix M into binary matrix Mby applying a similarity threshold δ such that = 1 if > δ and 0 otherwise.

2. Apply to the thresholded image a diagonal μ-percentile ﬁlter of length Δ with orientation parallel to the target line segments (45◦ relative to the x-axis of the image), allowing only diagonal line structures of sufﬁcient length (and thus sufﬁcient repeated interval duration) and density to pass.

3. Apply a Gaussian ﬁlter of width σ with orientation orthogonal to the target line segments (−45◦ relative to the x-axis of the image) to the thresholded image, transversely smearing the remaining line segments. This ﬁlter accommodates variation in speaking rate by allowing deviation from the 45◦assumption. The result of these ﬁrst three steps as applied to the dotplot of Fig. 1(a) is shown in Fig. 2(a).

4. Apply a classical one-dimensional Hough transform (r is varied, θ ﬁxed at −45◦) to the ﬁltered image, amounting toa projection of the image onto the line y = −x (as demonstrated in the red overlay of Fig. 2(a)). The result is shown in Fig. 2(b). Notice there are three peaks. The large central peak, about which the transform is symmetric, is a result of the self-similarity line (main diagonal).

5. Use the peaks of the Hough transform to deﬁne rays through which to search for line segments. In our example of Fig. 2,we ignore the center peak because it corresponds to the main diagonal, which in this example results from self-similarity. Since the Hough transform is symmetric, there is only one search ray (outlined in green in Fig. 2(a)).

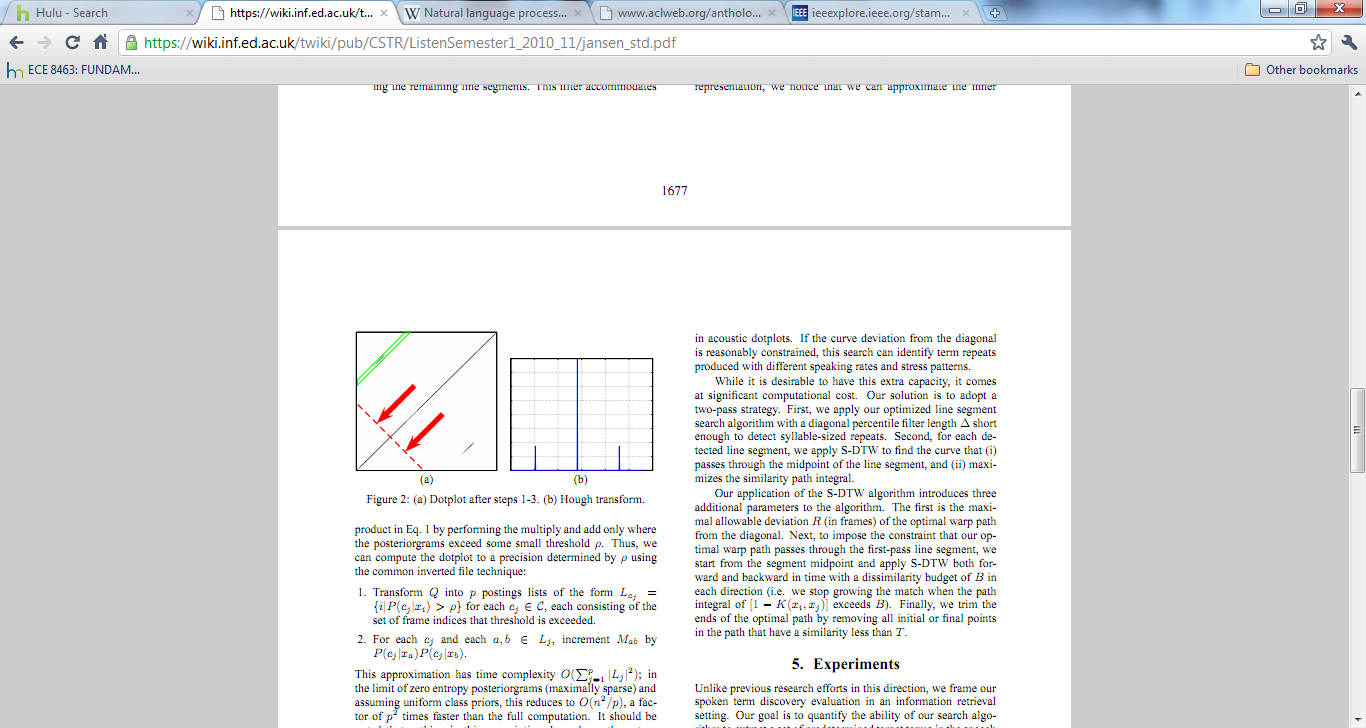


Figure 2: (a) Dotplot after steps 1-3 (b) Hough transform

Also included in this algorithm is the application of a diagonal median filter to a duration k seconds. The choice of k is used to determine an approximate threshold on the duration of the matched regions discovered. Large k values (~ 1 sec) will produce a relatively sparse list of matches that correspond to long words or short phrases while a smaller k values (< 0.5 sec) will admit shorter words and syllables that may be less informative .

**Posteriorgram Representation**

The acoustic dotplot technique has its short-comings. It can operate on any vector time series representation of the speech signal or a standard spectrogram. It cannot guarantee a high cosine similarity value between spectra of distinct speakers producing the same phoneme. To make discovery possible across multispeaker a speaker-independent representation is needed. Phonetic posteriorgrams are chosen such that each frame is represented as the posterior probability distribution over a set of speech sounds

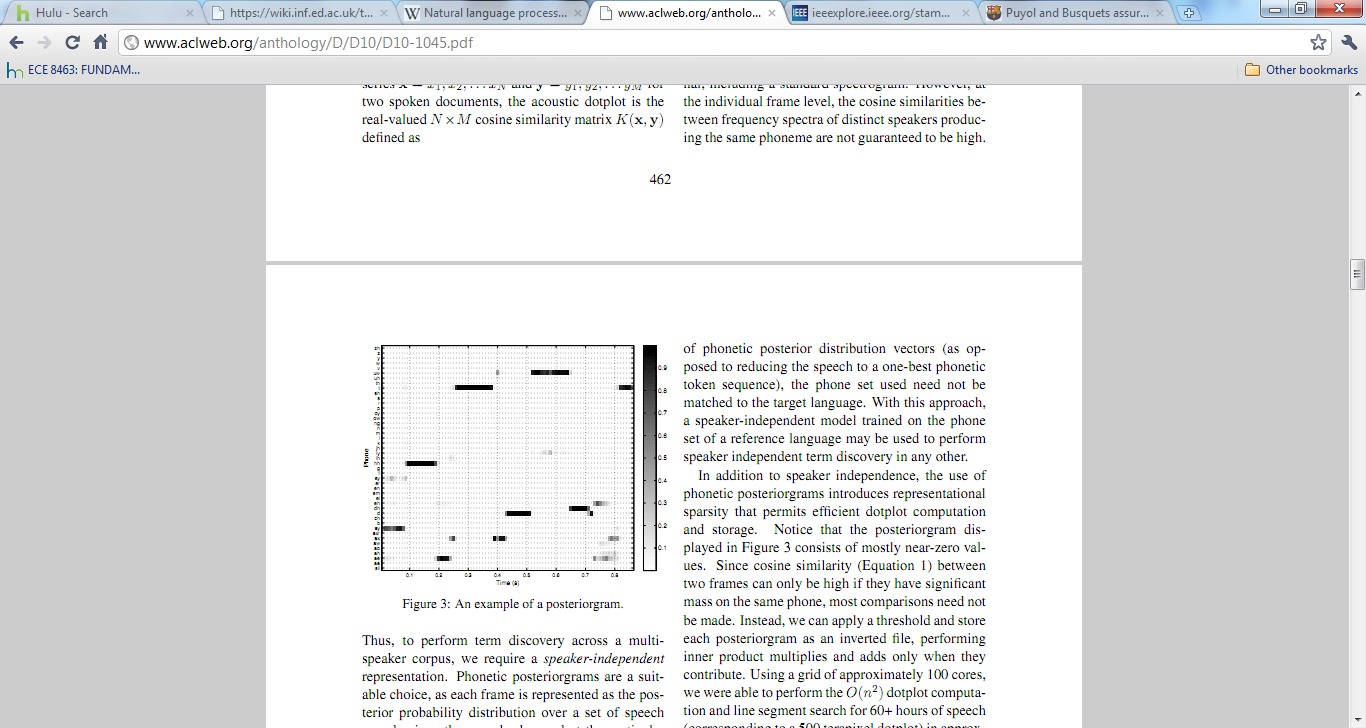


Figure 3: An example of a posteriorgram

The figure above shows an example posteriorgram for the utterance “I had to do that” computed with a multi-layer perceptron (MLP). Each row of the figure above represents the posterior probability of the given phone as a function of time through the utterance and each column represents the posterior distribution over the phone set at that point in time. Typically constructing speaker independent acoustic model typically requires a significant amount of transcribed speech. In this paper a speaker independent acoustic model is trained in a higher resource language or domain to interpret multi-speaker data in the zero resource target setting. This approach doesn’t require knowledge of the language to detect when a word of sufficient length has been repeated in it. Cosine similarities of phonetic posterior distribution vectors are computed instead of reducing the speech to a one-best phonetic sequence. This way a speaker-independent model trained on the phone set of a reference language can be used to perform speaker independent term discovery in another language. Another advantage to using phonetic posteriorgrams is that it introduces representational sparsity that makes computation of dotplots efficient.

**Multi-Layer Perceptron**

This is the tool used to compute posteriorgram for the utterances. It is a feature extraction technique for phoneme recognition that uses short-term spectral envelope and modulation frequency features. These features are derived from sub-band temporal envelopes of speech estimated using Frequency Domain Linear Prediction (FDLP). While spectral envelope features are obtained by the short-term integration of the sub-band envelopes, the modulation frequency components are derived from the long-term evolution of the sub-band envelopes. Time-varying spectrum of speech is usually derived as a sequence of short-term spectral vectors, each vector representing instantaneous values of spectral magnitudes at the individual carrier frequencies. An alternative functionally equivalent representation is a collection of temporal envelopes of spectral energies at the individual carrier frequencies. The Fourier transform of these time-varying temporal envelopes yields a set of modulation spectra of speech, where each modulation spectral value represents the dynamics of the signal at the given carrier frequency.

FREQUENCY DOMAIN LINEAR PREDICTION

FDLP is an efﬁcient technique for auto regressive (AR) modeling of temporal envelopes of a signal. An autoregressive model is a type of random process which is often used to model and predict various types of natural and social phenomena. The FDLP represents a dual technique to the conventional Time Domain Linear Prediction (TDLP). In the case of TDLP, the AR model approximates the power spectrum of the input signal, whereas FDLP ﬁts an all pole model to the Hilbert envelope (squared magnitude of the analytic signal).The FDLP technique is implemented in two parts - ﬁrst, the discrete cosine transform (DCT) is applied on long segments of speech to obtain a real valued spectral representation of the signal. Then, linear prediction is performed on the DCT representation to obtain a parametric model of the temporal envelope.

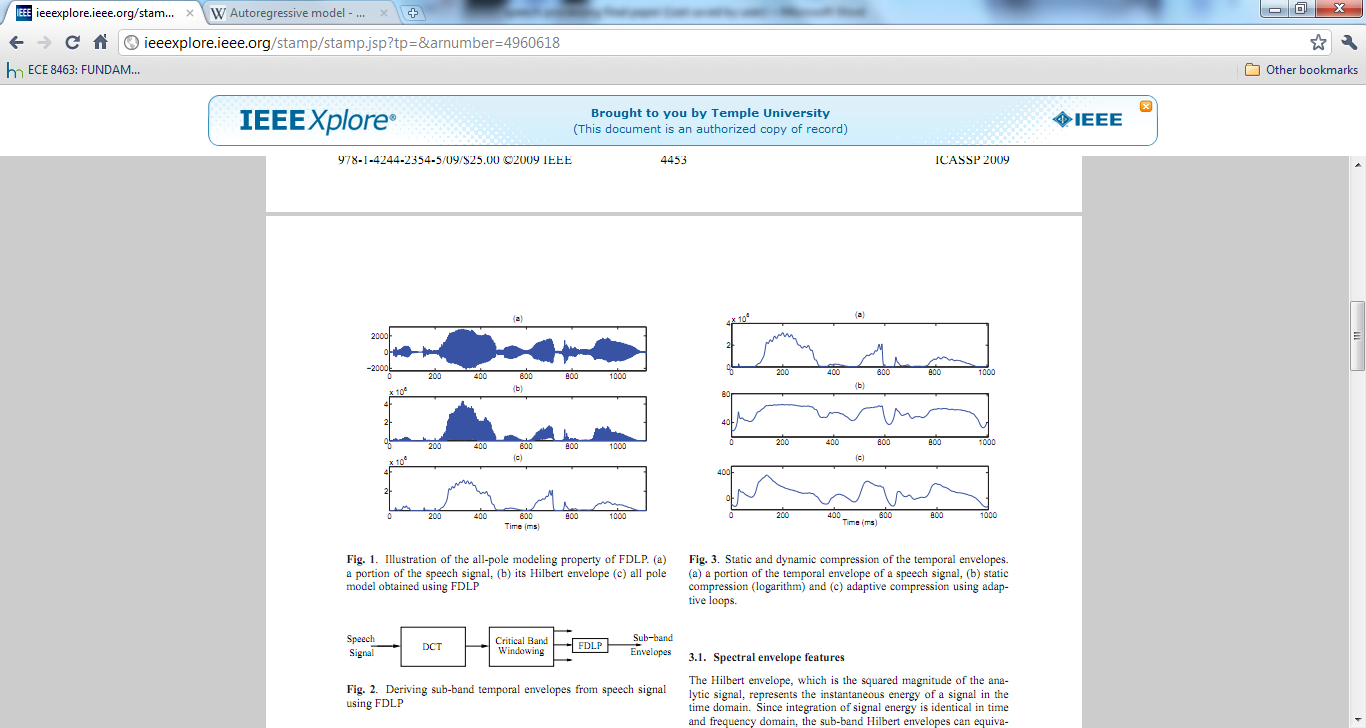


Figure 4Illustration of the all-pole modeling property of FDLP. (a) a portion of the speech signal, (b) its Hilbert envelope (c) all pole

model obtained using FDLP

Fig. 4 above illustrates the AR modeling of FDLP. It shows (a) a portion of speech signal, (b) its Hilbert envelope computed using the Fourier transform technique and (c) an all pole approximation to the Hilbert Envelope using FDLP.

1. **Creating Pseudo terms**

Pseudo terms are computed using acoustic repetitions described in the previous section. A set M of matched region containing pair of speech interval are obtained where m =[ indicating that speech from to is an acoustic match to speech from to .The size of the set is dependent on the frequency of occurrence of a particular term. If a term occurs k times, the set has distinct elements that corresponds to that term. These distinct terms are grouped into clusters. The resulting clusters are called pseudo terms. Once the pseudo terms are obtained, it is easy to represents spoken documents as a collection of pseudo-terms. Matched regions are represented as vertices in a graph with the edges representing similarities between those regions. A graph-clustering algorithm that extracts connected components is used where G = (V,E) is an unweighted, undirected graph with vertex set V and edge set E. Each vi ∈ V corresponds to a single speech interval ([) present in M (each m ∈M has a pair of such intervals, so |V | = 2|M|) and each eij ∈ E is an edge between vertex vi and vj . The set E consists of two types of edges. The ﬁrst represents repeated speech at distinct points in the speech database as determined by the match list M. The second represents near-identical intervals in the same utterance .When multiple intervals are contained in the same utterance, an edge is introduced if fractional overlap and

Weights that reflect acoustic similarity between match intervals are introduced to improve clustering. The weights allow shorter pseudo-terms to be considered without greatly increasing false alarms.

Table 1 Pseudo-terms resulting from a graph clustering of matched regions(k=0.75,T=0.95)

|  |  |
| --- | --- |
| Counts | Terms |
| 5 | Keep track of |
| 5 | Once a month |
| 2 | Life insurance |
| 2 | Capital punishment |
| 9 | Paper; newspaper |
| 3 | Talking to you |

The table above contains several examples of pseudo-terms and the matched regions included in each group obtained using the algorithm described in this paper.

1. **Test Database**

The experiments in this paper used the Switchboard Telephone Speech Corpus. Switchboard is a collection of roughly 2,400 two sided telephone conversations with a single participant per side. Over 500 participants were randomly paired and prompted with a topic for discussion. Each conversation belongs to one of 70 pre-selected topics with the two sides restricted to separate channels of the audio. To develop and evaluate the methods described in this paper, three data sets were created from the Switchboard corpus: a development data set, a held out tuning data set and an evaluation data set. The development data set was created by selecting the six most commonly prompted topics (recycling, capital punishment, drug testing, family ﬁnance, job beneﬁts, car buying) and randomly selecting 60 sides of conversations evenly across the topics (total 360 conversation sides.) which corresponds to 35.7 hours of audio.. All algorithm development and experimentation were conducted exclusively on the development data. For the tuning data set, an additional 60 sides of conversations was selected evenly across the same six topics used for development, for a total of 360 conversations and 37.5 hours of audio. This data was used to validate the experiments on the development data. Once the parameter had been selected for evaluating the system, an evaluation dataset was created. The evaluation dataset consisted of 6000 conversation sides containing 61.6 hours of audio related to six conversation topics (family, life, news, media, public education, exercise/fitness, pets, taxes). For the experiments the duration of speech was varied between 0.6s and 1.0s while the overlap threshold ranged between 0.75 and 1.0. The statistics on the number of features of Pseudo term generated for different settings of duration and threshold can be seen below.

Table 2 Statistics on the number of features ( pseudo-terms) generated for different settings of the match duration k and the overlap threshold T.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| k | T | Features | Feat Frequency | Feat. /doc |
| 0.6  0.6  0.6  0.6 | 0.75  0.85  0.95  1.0 | 5809  23267  117788  333816 | 2.15  2.22  2.38  2.32 | 34.7  143.4  779.8  2153.4 |
| 0.75  0.75  0.75  0.75 | 0.75  0.85  0.95  1.0 | 8236  18593  48547  90224 | 2.31  2.36  2.36  2.18 | 52.8  121.7  318.2  546.9 |
| 0.85  0.85  0.85  0.85 | 0.75  0.85  0.95  1.0 | 5645  8832  15805  24480 | 2.52  2.44  2.24  2.10 | 39.5  59.8  98.3  142.4 |
| 1.0  1.0  1.0  1.0 | 0.75  0.85  0.95  1.0 | 1844  2303  3239  4205 | 2.39  2.24  2.06  1.93 | 12.3  14.4  18.6  22.7 |

1. **Document Clustering**

The objective here is to sort spoken documents into groups where each group contains documents that are similar. This makes it easier for a user exploring the speech database to gain a good overview of the contents being discussed in the database. The clustering technique considered in this paper differs from traditional clustering technique because traditional techniques don't generate clusters with highly readable names. When a given a query and the ranked list of documents, the method first extracts and ranks salient phrases as candidate cluster names, based on a regression model learned from human labeled training data. The documents are assigned to relevant salient phrases to form candidate clusters, and the final clusters are generated by merging these candidate clusters. There are several methods for evaluating the effectiveness of clustering algorithm. This paper considers three methods, Purity, Entropy and B-Cubed.

**Purity** this measures the precision of each cluster like how many examples in each cluster belong to the same true topic. Purity ranges between zero and one with one being optimal. The purity of a cluster is deﬁned as the largest percentage of examples in a cluster that have the same topic label. Purity of the entire clustering is the average purity of each cluster:

Where C is the clustering, L is the reference labeling and N is the number of examples. is a specific cluster and is a specific true label.

**Entropy** this measures how the members of a cluster are distributed amongst the true labels. The global metric is computed by taking the weighted average of the entropy of the members of each cluster. Specifically, entropy(C,L) is given by

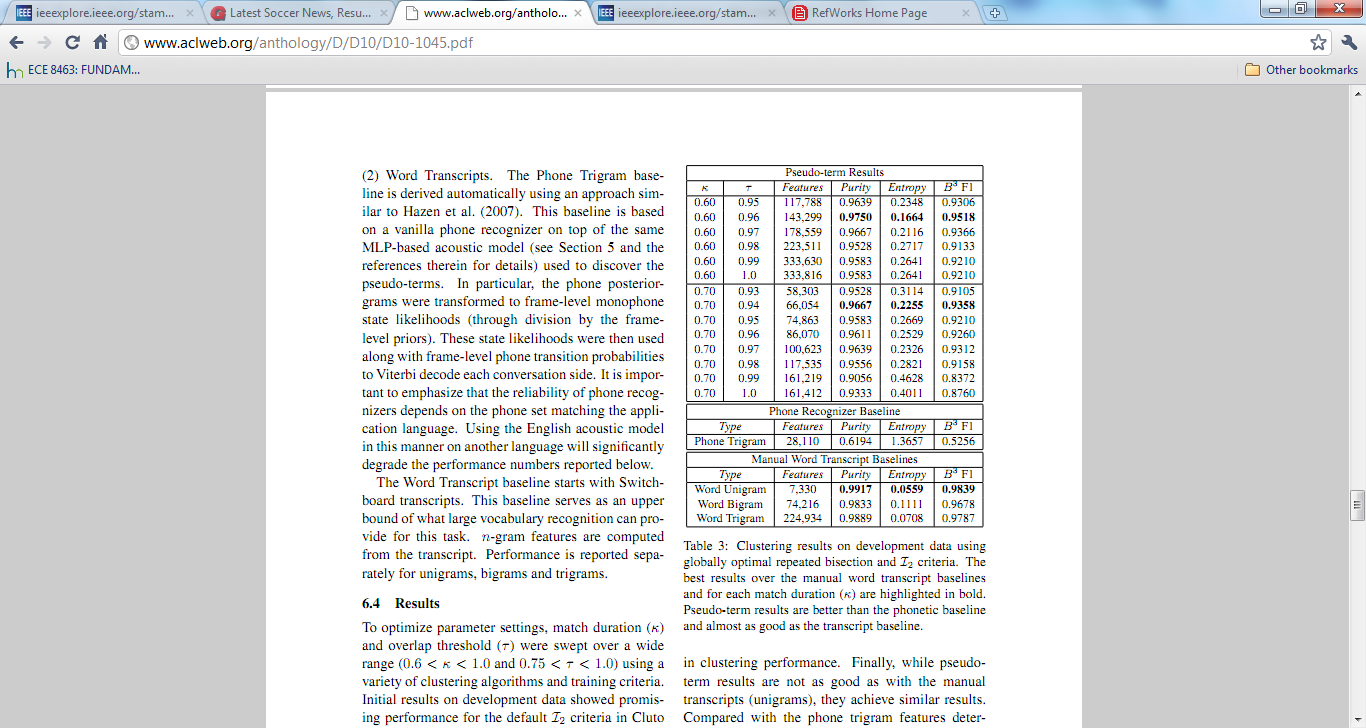
Where is the number of instances in cluster I, is the probability of seeing label in cluster .

**B-Cubed** this measures the clustering effectiveness from the perspective of a user’s that is inspecting the clustering results. It measures what percentage of all examples that share the same label as the selected example will appear in the selected cluster. B-Cubed averages its evaluation over each document and not each cluster making it less sensitive to small errors in large clusters as opposed to many small errors in small clusters.

1. **Results**

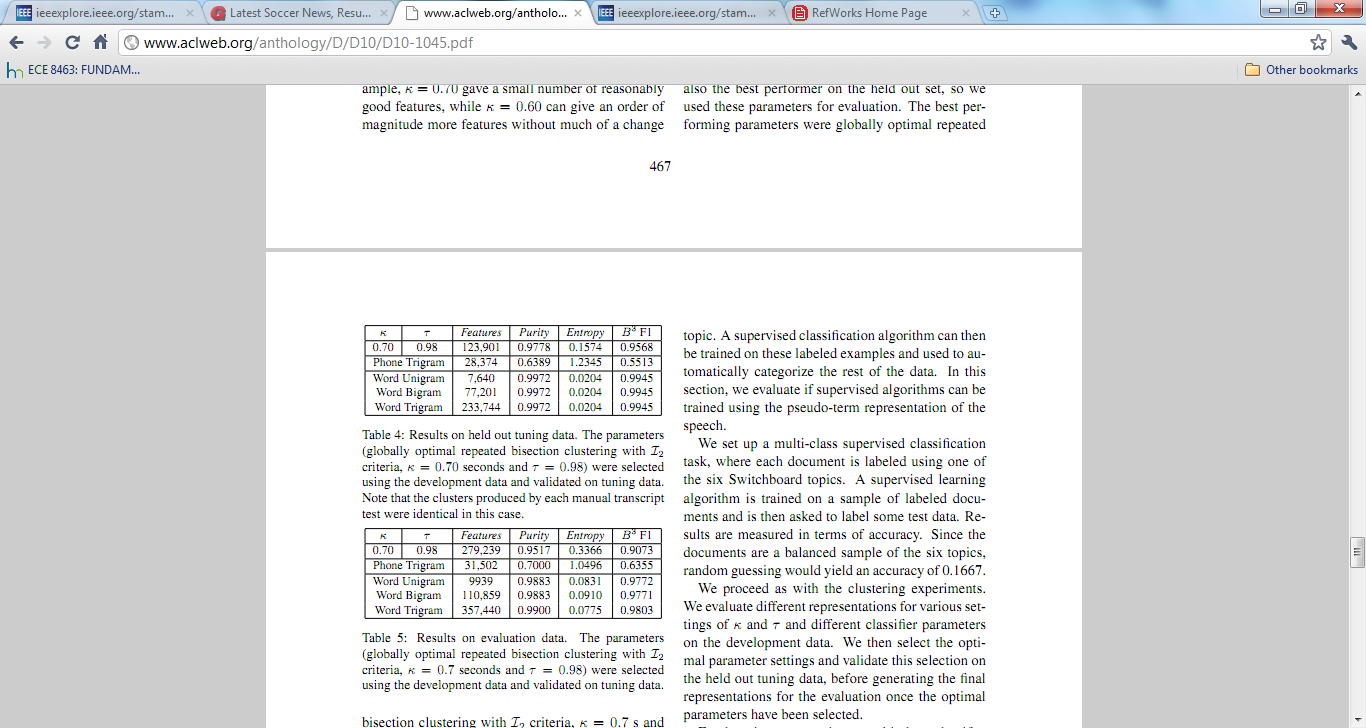
The performance of the pseudo-term feature set is compared to the performance of two baselines namely Phone Trigrams and word Transcripts. The results can be seen in the table below

Table 3: Clustering results. The best results over the manual word transcript baseline and for each match duration (k) are highlighted in bold.

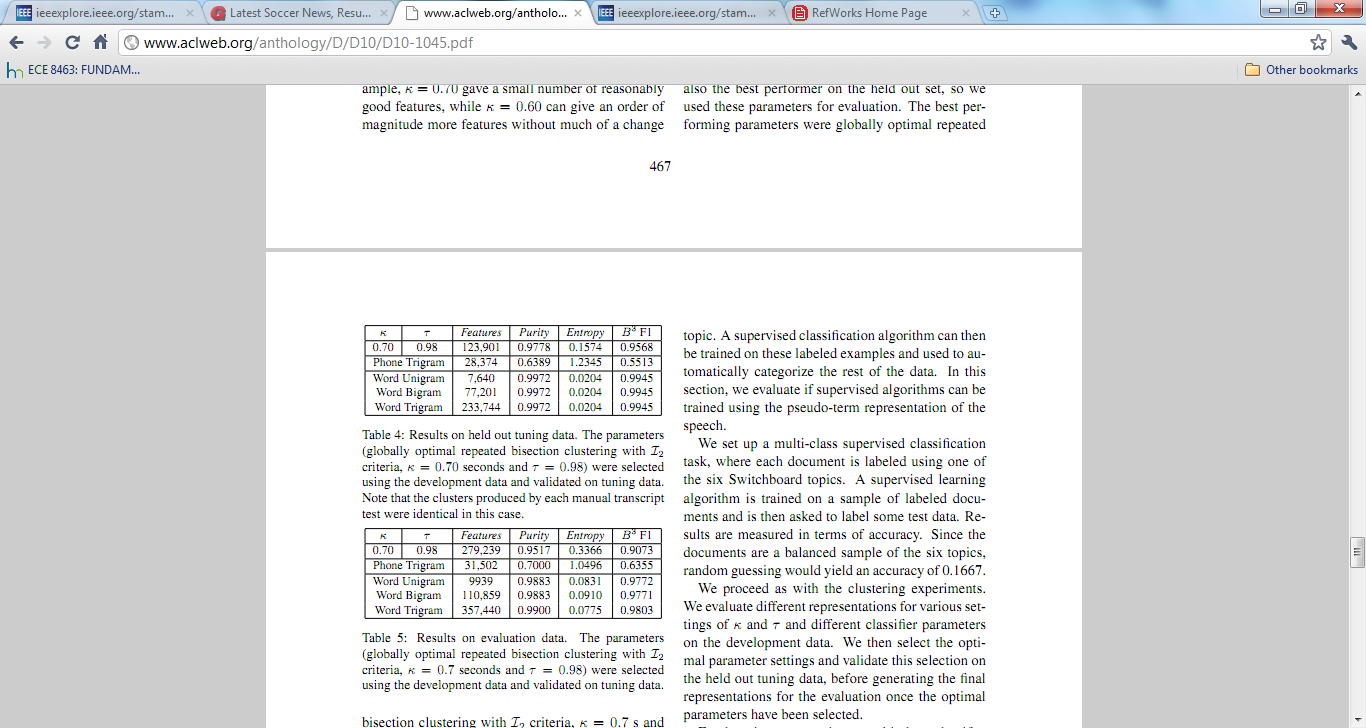


From the table above we can see that the Pseudo-term results are better than the phonetic baseline, the results are not as good as that of the manual transcripts (unigrams). It performs significantly better than phone trigram features determined by a phone recognizer output. It is important to note that all approaches were built using an identical MLP-based phonetic acoustic model. The optimal parameter settings for testing the evaluation data was selected based on results obtained using the development data and the held out tuning data. The parameter settings are k duration and threshold T. An optimal threshold of k =0.7 and T= 0.98 is selected. The results generated on the evaluation data and held out tuning data can be seen in the tables below.

Table 4 Results on Held out Tuning Data with the parameter k=0.7 and T=0.98



**Table 5 Results on Evaluation Data with the parameter k=0.7 and T=0.98**



The results generated on the held out tuning and evaluation data for the optimal setting is compared to the manual word transcripts in the tables above. The result obtained on the tuning data and evaluation data is very similar to those of the development data. The results of the manual transcript are better than the pseudo-term representation presented in this paper. However the pseudo-term representation is quite competitive, it requires fewer resources and performs significantly better than a phone recognizer.

1. **Supervised Document Classification**

The algorithm discussed is in this paper is for performing unsupervised clustering of the speech corpora where no human annotations are necessary. The algorithm can be used to train speech corpora with few labels to automatically categorize the rest of the data. This portion of the research focused on the feasibility of training a supervised algorithm using the pseudo-term representations of the speech. For this experiment each spoken document is labeled using one of the six switchboard corpora topics The supervised learning algorithm is trained on a sample of labeled documents and then expected to label some test data. The evaluation criterion for this experiment is accuracy. For learning purpose a multi-class classifier training algorithm is required. For this experiment four of such algorithm were considered namely MIRA ( a large margin online Learning algorithm), Confidence Weighted(CW) learning (a probabilistic large margin online learning algorithm), Maximum Entropy (a log linear discriminative classifier) and Support Vector Machines(SVM)- a large margin discriminator. Below are tabular result obtained using the multi-class classifier algorithm on the development, held out tuning and evaluation data.

**Table 6: The top 15 results for pseudo-terms on development data. The best pseudo-terms and manual transcript results for each algorithm are bolded. Pseudo term results are better than the phonetic baseline and almost as good as the transcript baseline**

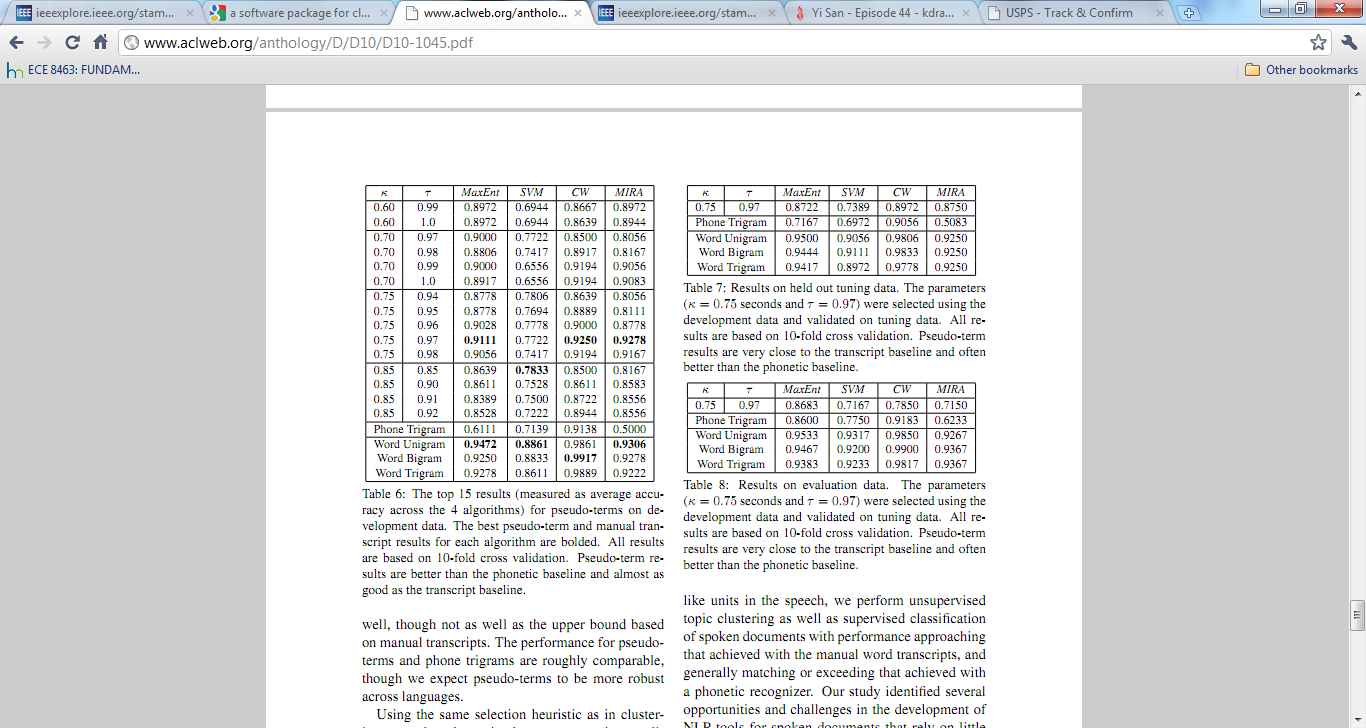


Table 7 Results on held out tuning data. The parameters (k=0.75 seconds and T= 0.97) were selected using the development data and validated on tuning data.

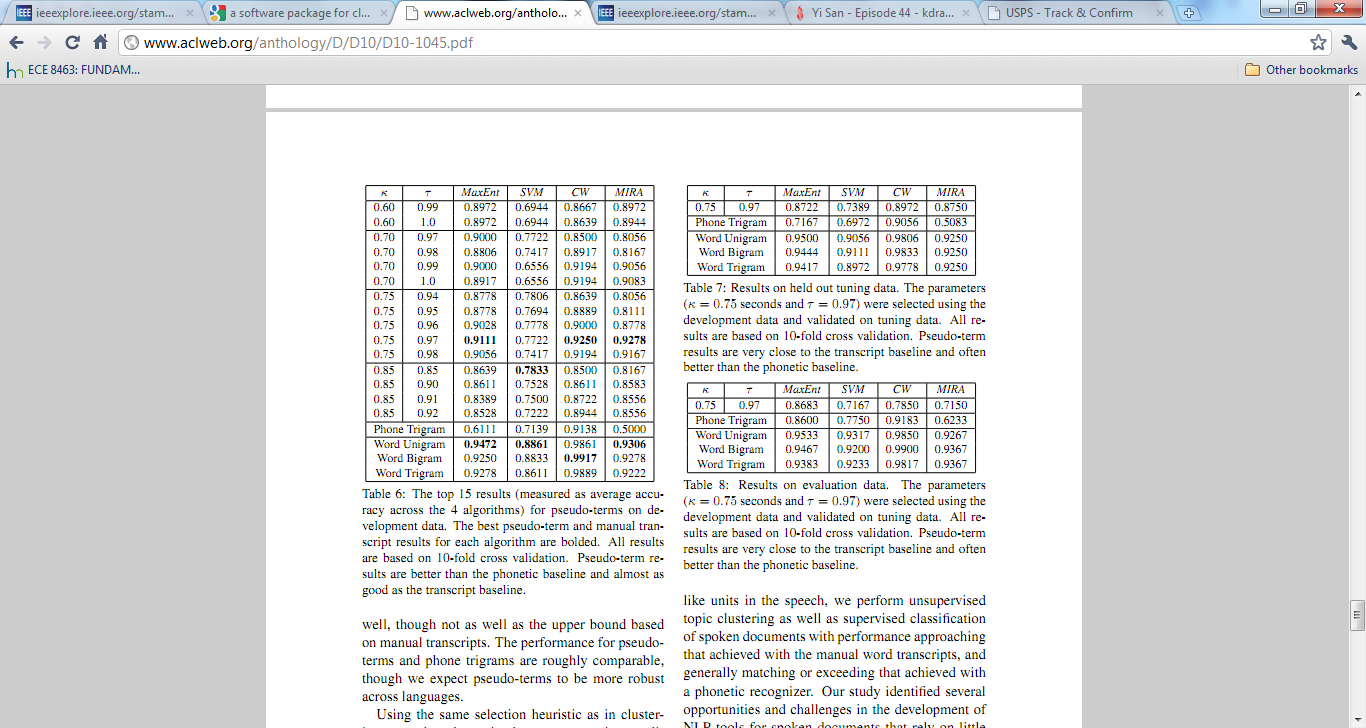
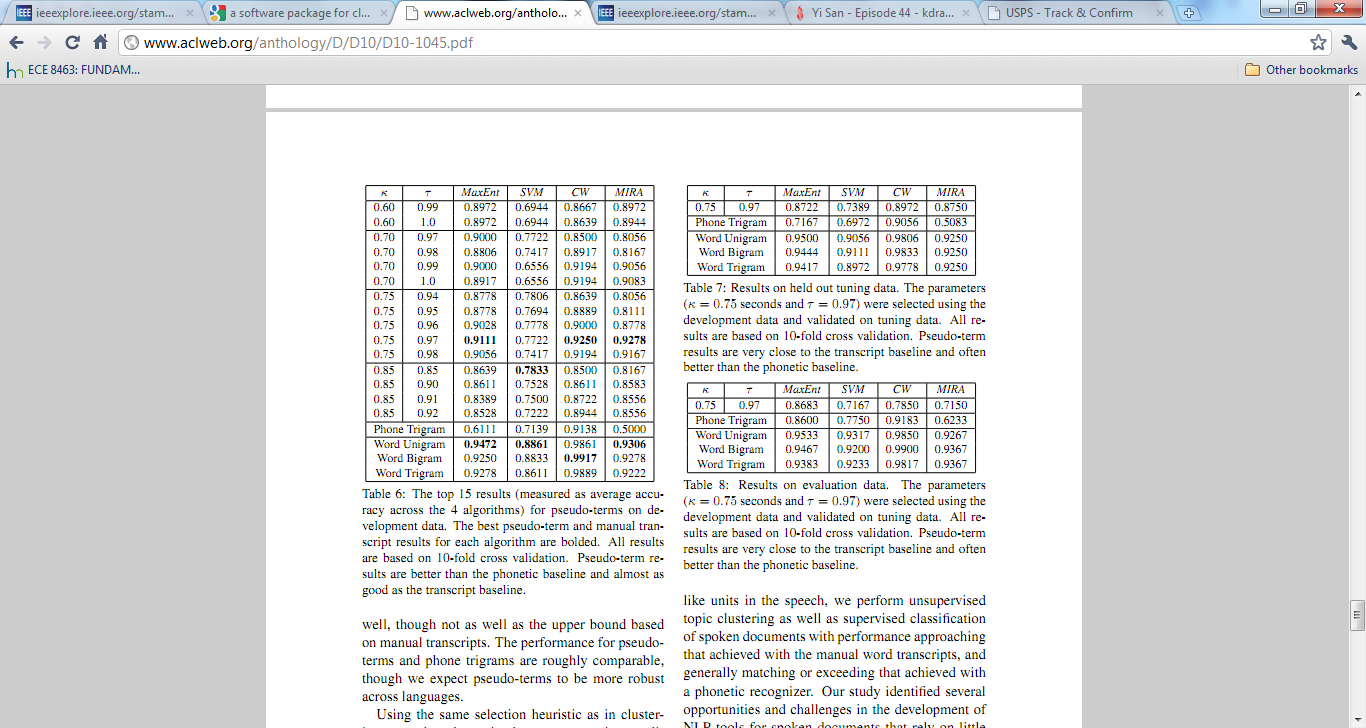


Table 8 Results on evaluation data. The parameters (k=0.75 seconds and T= 0.97) were selected using the development data and validated on tuning data.



1. **Conclusion**

The paper presents an alternative method for applying standard natural language processing to speech utterance without the use of an automatic speech recognizer. The result obtained show that the proposed new approach can classify spoken documents as effectively as baseline techniques such as manual word transcripts and phone recognizer using fewer resources. The new method is favorable because it does not require a lot of linguistic resources. It significantly reduces the amount of annotation required to build an efficient LVCSR system.

**References**

1. H.J. Zeng, Q.C. He, Z. Chen, W.Y. Ma, and J. Ma. 2004 .Learning to cluster web search results. In Conference on Research and development in information retrieval (SIGIR).
2. Aren Jansen, Kenneth Church, and Hynek Hermansky.2010. Towards spoken term discovery at scale with zero resources. In Interspeech.
3. Timothy J. Hazen, Fred Richardson, and Anna Margolis**.** 2007. Topic identiﬁcation from audio recordings using word and phone recognition lattices. In IEEE Workshop on Automatic Speech Recognition and Understanding.
4. Mark Dredze, Aren Jansen, Glen Coppersmith, and Ken Church. 2010. NLP on spoken documents without ASR. In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing (EMNLP '10). Association for Computational Linguistics, Morristown, NJ, USA, 460-470.