Advances in Profile Assisted Voicemail Management

Konstantinos Koumpis

Vienna Telecommunications Research Center - ftw.
Tech Gate Vienna, 1 Donau City St.
1220 Vienna, Austria
koumpis@ftw.at

Abstract. Spoken audio is an important source of information available to knowledge extraction and management systems. Organization of spoken messages by priority and content can facilitate knowledge capture and decision making based on profiles of recipients as these can be determined by physical and social conditions. This paper revisits the above task and addresses a related data sparseness problem. We propose a methodology according to which the coverage of language models used to categorize message types is augmented with previously unobserved lexical information derived from other corpora. Such lexical information is the result of combining word classes constructed by an agglomerative clustering algorithm which follows a criterion of minimum loss in average mutual information. We subsequently generate more robust category estimators by interpolating class-based and voicemail word-based models.

keywords: automatic categorization, speech recognition, stochastic language models, class-based clustering, voicemail.

1 Introduction

The enormous growth of available spoken audio recordings has led to a comparable growth in the need for methods to assist users in managing the knowledge contained therein [1]. Standard knowledge management approaches typically organize content in portals, use text search and analysis tools, and rely heavily on text as the medium for transferring knowledge. The ongoing migration of computing for information access from the desktop and telephone to mobile devices such as personal digital assistants (PDAs) and smart phones introduces new challenges as these devices offer limited audio playback and display capabilities. Voicemail represents a growing volume of real-world speech data that involves a conversational interaction between a human and a machine with no feedback from the machine, and for which manual organization is a time consuming task, particularly for high-volume users. There are situations in which users would prefer to receive messages of certain content types and leave the remaining ones to be reviewed later at a more convenient location or time. For example, imagine a user attending a business meeting abroad requiring constant information flow from his head office but not messages from his local recreation club. On the

contrary, the same person being on holiday would most likely be interested only in messages related to friends and family rather than those concerning work.

As text categorization utilities are becoming central into advanced email tools, users are likely to wish their migration into voicemail too. Voicemail recipients today rely almost exclusively on caller line identity, the presentation of the caller's telephone number or name, to filter incoming messages. A few alternative solutions have been proposed for efficient voicemail retrieval which include browsing and searching of message transcriptions via a graphical user interface [2], generation and delivery of text summaries on mobile phone displays using wireless protocols [3], extraction of the identity and phone number of the caller from voicemail messages [4], and re-ordering voicemail messages based on criteria of urgency and business relevance [5]. A profile assisted voicemail management system can instead sift through a stream of arriving messages to find those relevant to predefined categories. Each message can be in exactly one, multiple or no category at all. Unlike search queries, user profiles are persistent, yet adaptive, and tend to reflect long term information needs.

Many standard machine learning techniques have been applied to automated text categorization problems, such as decision trees, naive Bayes, neural networks, k-nearest neighbour classifiers and support vector machines [6–8]. Stochastic language models are of particular interest as input features because they can incorporate local dependencies and thus preserve semantics, as a result of modelling word sequences within the framework of standard Markov based approximations. Character level language models have been found to be effective in text classification [9] and author attribution [10] tasks. This paper deals with a corpus containing transcriptions of spoken messages. Spoken language is different from written language as it is often ungrammatical, lacks punctuation and capitalization, and its textual representation almost always contains substitution, deletion and insertion errors. Training stochastic language models using a small corpus, such as voicemail, carries the risk of assigning zero probabilities to a large number of likely events. Because some words are similar to other words in their meaning and syntactic function one can expect getting better estimates with fewer parameters by grouping words into classes.

The rest of the paper is divided into five sections. Section 2 describes the voicemail data and the categorization protocol used, respectively. Section 3 discusses the methodology employed to perform message categorization, and experimental results are given in section 4. Finally, we summarize our conclusions and discuss future work in section 5.

2 Voicemail Data

We have used the LDC Voicemail Corpus-Part I [11]. This corpus contains 1801 messages (14.6 hours, averaging about 90 words per message). As a training set for the categorization tasks we used 1789 out of 1801 available messages (151K words) of the corpus. The reason for which 12 messages had to be excluded from the training set was that they did not contain enough information to determine their category. For evaluation purposes we used the test set of the corpus comprising 42 messages (2K words) as well as the test set of the Voicemail Corpus-Part II comprising 50 messages (4K words). Apart from the human

transcriptions (denoted SR-HT), which contained some noise in the form of repetitions and broken words, we also used transcriptions with a word error rate (WER) of 42.5% produced by a hybrid multi-layer perceptron / hidden Markov model speech recognizer (denoted SR-SPRACH) [12]. Additionally, we obtained another set of transcriptions with a WER of 31% (denoted SR-HTK) produced by the more complex HTK Switchboard system adapted to voicemail [13].

Table 1. Taxonomy for the message priority- and content-based categorization tasks. Further details and examples can be found in [14]

.

Priority-based categorization

Category	Description
high	an immediate action by the recipient is required, expected or implied (often following a request)
medium	some attention by the recipient will be required
low	rather trivial content, no need for immediate attention

Content-based categorization

Category	Description
technical	specific technical issues related to projects
office	daily issues (excl. technical)
business	complementary professional tasks not covered by the above
family	related to family members (spouse, children, parents etc.) or concern family issues
friends	related to friends (incl. colleagues but not concerning work)
private	miscellaneous content concerning the recipients not covered by any of the above

2.1 Voicemail Categorization Protocol

Voicemail messages are typically short, conveying the reason for the call, the information that the caller requires from the voicemail recipient and a return telephone number. Herein we consider two tasks, categorization by *content* and by *priority*. The categories in both tasks are mutually exclusive and exhaustive, that is, every message belongs to one, and only one, of the categories. The data labelling is a result of subjective analysis of the message transcriptions. The attributes that the message recipient will perceive along with the categorization criteria, are determined by individual needs. These needs change over time and with the physical and social environment. As the data is not organized per voicemail subscriber, we assumed a general voicemail recipient profile, which might not be fully compatible with the criteria of each individual voicemail recipient. Finally, during the labelling process for the categorization tasks no attempt was

made to associate the message priority or content with the identity of speakers and thus the task does not share similarities with speaker recognition [15].

Table 1 outlines the taxonomy related to the priority- and content-based categorization tasks. Given the relatively small size and the nature of the corpus, we decided to use 3 and 6 categories, respectively because in a dense category space there would be only few example messages in each category. The distribution of messages in the training and test sets for the priority- and content-based tasks are given subsequently in Table 2.

Table 2. Category distributions across the training and test sets for priority and content, respectively.

Priority-based categorization

Category	Training set	Test set
high	37.4%	29.3%
medium	51.4%	54.3%
low	11.1%	16.3%

Content-based categorization

Category	Training set	Test set
technical	13.1%	5.3%
office	16.9%	23.4%
business	38.7%	35.1%
family	5.9%	12.8%
friends	16.4%	12.8%
private	9.0%	10.6%

3 Categorization using Stochastic Language Models

We approach voicemail categorization in a Bayesian learning framework. We assume that the message transcription was generated by a parametric model (in our current implementation this is limited to a language model), and use training data to calculate Bayes optimal estimates of the model parameters. Then, using these estimates we classify new test messages using Bayes rule to turn the generative model around and calculate the probability that a category would have generated the test message in question. Categorization then becomes the task of selecting the most probable category. Details of the above approach are given bellow.

A language model is essentially an information source which emits a sequence of symbols w_i from a finite alphabet, i.e., the vocabulary. The probability of any word sequence $w_1, w_2, ..., w_i$ is given by:

$$p(w_1, w_2, ..., w_N) = \prod_{i=1,...,N} p(w_i|w_1, ..., w_{i-1}) \quad (1)$$

A simple yet effective approach to approximate the above is the n-gram model [16] according to which the occurrence probability of any test word sequence is conditioned upon the prior occurrence of $n-1$ other words:

$$p(w_i|w_1, ..., w_{i-1}) \approx p(w_i|w_{i-n+1}, ..., w_{i-1}) \tag{2}$$

n-gram language models have the advantage of being able to cover a much larger variation than would normally be derived directly from a corpus in the form of explicit linguistic rules, such as a formal grammar. Open vocabularies can also be easily supported by n-gram language models and are used in all experiments reported in this paper.

The task of classifying a message transcription $\mathcal{M}$ into a category $c \in C = \{c_1, c_2, ..., c_C\}$ can be expressed as the selection of the category which has the largest posterior probability given the message transcription:

$$c^+ = \arg\max_{c \in C}\{p(c|\mathcal{M})\} \tag{3}$$

$$= \arg\max_{c \in C}\{p(\mathcal{M}|c)p(c)\} \tag{4}$$

In the above expression the language model is used to estimate the likelihood $p(\mathcal{M}|c)$ whilst the prior $p(c)$ is assumed to be the same with that of the training set. For computational reasons, products of probabilities in Equation 4 are replaced by sums of negative log probabilities. Categorizing a message involves calculating a sum of negative logs for each category, where the length of the sum equals to the number of n-grams contained in the test message. Each term in the sum is proportional to the frequency with which the corresponding n-gram sequence has occurred in the training data. Note that if one assumes equal priors the above criterion becomes equivalent to perplexity [17], a measure expressing the average number of possible branches after a word in a sequence. Comparing the above measure across different categories for each test message allows the highest ranked category along with a rank value to be returned. The number of returned categories can be specified by the user so that the categorization results may be given in the form of an n-best list for further processing. Such 'soft' decisions allow a message to appear in more than one relevant category giving greater flexibility during retrieval. Finally, adding new messages to a trained n-gram model only requires the recording of word occurrence statistics for those messages.

3.1 Class-based n-gram Models

Training n-grams for categorization using a small corpus carries the risk of assigning zero probabilities to a large number of likely events not present in the available data. The perplexity of test word sequences containing such unseen events will increase significantly. Based on the hypothesis that some words are similar to other words in their meaning and syntactic function, we can derive likely, yet unobserved, word sequences to reduce the effects of data sparseness. Further, this approach can update the probabilities of rarely observed word sequences. For example, if the word "speech" is completely missing from the

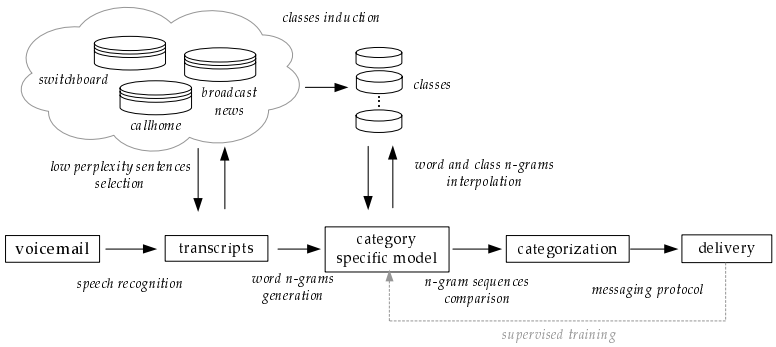

Fig. 1. Overview of the methodology to augment language models with word classes for categorizing spoken messages.

training data while the words "voice" and "spoken" are included, there is still a good chance to be able to model it using a class which contains semantically similar words to the above. An overview of the methodology followed is depicted on Figure 1. Word clustering can provide useful semantically related groups, similar to an automatically generated thesaurus. Suppose that we partition a vocabulary of V words into G classes using a function f_g, which maps a word, w_i, into its class g_i. The resulting model

$$p(w_i|w_{i-2}, w_{i-1}) = p(g_i|g_{i-2}, g_{i-1})p(w_i|g_i) \tag{5}$$

produces text by first generating a string of classes g_1,g_2,...,g_n and then converting it into words w_i with probability $p(w_i|g_i)$, for $i = 1, 2,...,n$. Word classes can be defined either manually or automatically. Manual definition makes use of part-of-speech tags and stem information while automatic definition clusters words as part of an optimization method. For the purposes of this study we adopted an agglomerative clustering algorithm [18]. This algorithm performs a bottom-up clustering, starting with a separate cluster for each of the G most frequent words and at each step merge that pair for which the loss in average mutual information is minimum. Different words in the same class are only distinguished according to their relative frequencies in the training set as a whole and therefore large and relevant sets should be used to generate accurate word classes.

In order to reduce the amount of computation along with the risk of generating irrelevant classes, we selected subsets of various American English transcriptions from the publically available Broadcast News, Switchboard and CallHome corpora. The criterion employed was low sentence perplexity (in practice, <200) over a trigram language model trained on each of the priority and content voicemail categories described in section 2.1. We also required that sentences used to induce word classes contained at least ten words. The corresponding vocabularies for the sentences selected were divided into 1000 classes. Prior to the interpolation with the word-based voicemail language models we retained in the

classes those words that occurred at least ten times in the selected data and we included no more than the ten most frequent words of any class.

4 Experimental Results

Categorization performance in all subsequent experiments is measured in terms of *overall accuracy*, which is defined as:

$$Acc = \frac{\#correctly\ categorized\ messages}{\#messages\ considered} \tag{6}$$

We examined the effects of the following factors in relation to the above best-category only performance measure:

WER quantifies the mismatches between the reference category language models and those of the test messages due to transcription errors.

***n*-gram order** introduces a trade-off between capturing enough context and having poor model estimates due to data sparsity.

smoothing replaces the original counts with modified counts so as to redistribute the probability mass from the more commonly observed events to the less frequent and unseen events. Various smoothing techniques were compared, namely linear, absolute, Good Turing and Witten Bell [19].

The results for the priority- and content-based tasks are given in Figures 2 and 3, respectively. The training set is the same, whether we test on manually transcribed or automatically recognized data. We expect that the performance when testing with automatically transcribed data can be improved by using training data that is automatically generated too. We used neither a stop-list nor stemming as in our previous experiments [14] they were found to hurt accuracy. The larger training data set employed (approx. 80% more messages) offered consistent but relatively small improvements over the baseline word-based language models we had previously reported. This suggests that the stochastic language models employed for the categorization tasks are not too sensitive to training set size variations, as far as statistics for a sufficient number of n-grams have been calculated.

The accuracy in priority-based categorization task was significantly higher than the content-based one due to the smaller degree of confusability (3 vs. 6 target categories). The interpolated word- and class-based n-gram models (right column figures) offered improved accuracy than the word-based n-gram models (left column figures). Class-based clustering from multiple corpora allowed the models to hypothesize out-of-vocabulary words, which often hold the most significance when testing with unseen data. The average absolute improvement due to class-based clustering in categorization accuracy was 2-5%. Despite that the clustering algorithm employed generates pairs of words, roughly equal gains were observed between the bigram and trigram models since all models back-off to lower order n-grams. As it had been observed in previous experiments, transcription errors had a significant impact on categorization accuracy. Moving from human transcriptions to automatic transcriptions with WERs of either 31.0% or 42.5% reduces the accuracy by about 20% absolute.

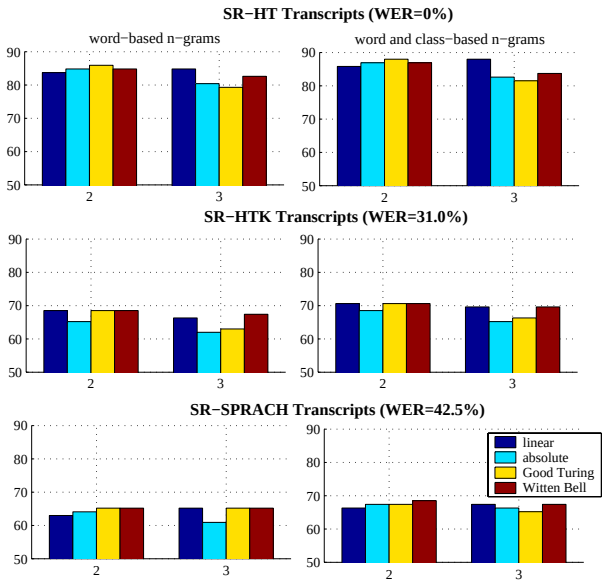

Fig. 2. Accuracy (%) in the priority-based categorization task using different smoothing techniques. The rows of subfigures correspond to transcripts of different WERs, while the n-gram order is shown on the horizontal axis.

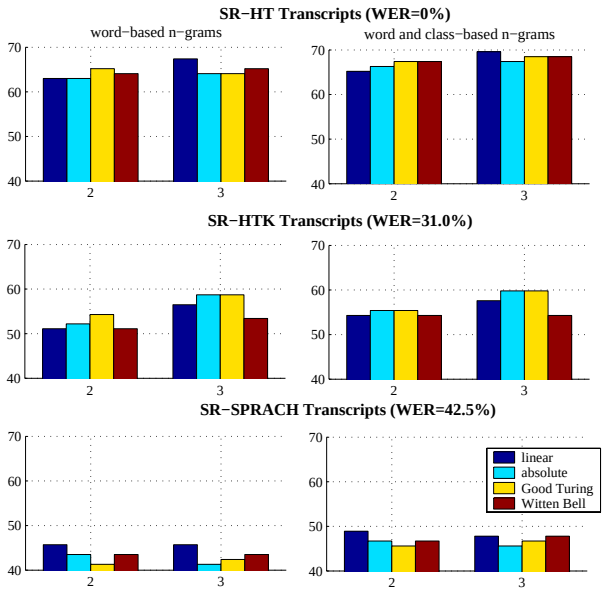

Fig. 3. Accuracy (%) in the content-based categorization task. The subfigure layout follows that of Fig. 2.

The differences across the four smoothing techniques were small. Among them, linear and Witten Bell performed slightly better on average, followed by Good Turing. It is though possible to combine different language models to improve accuracy. Methods as simple as majority voting can be employed to help reduce errors that individual smoothing techniques introduce.

The slight differences across the results made apparent the limitations in system evaluation introduced by a small test set. Although the combined test sets of Voicemail Corpora I and II demonstrated the effects of transcription errors, they were not adequate to demonstrate clear patterns related to other factors. Another issue to be investigated is how to treat messages for which the level of agreement among annotators is low. The use of Kappa statistic [20] could help indicate in a definitive way the correct category, if any, that a message belongs to. In future experiments we also plan to use the training set of Voicemail Corpus II as a validation and as a test set. Finally, the parametric model we employed for voicemail categorization was based only on textual information. It remains to be investigated if any prosodic features can be effectively associated with particular categories.

5 Conclusion

Voicemail data introduces several challenges to information management systems, including ambiguities related to short messages, uncertainties associated with speech recognizers and the need for scalability and portability in new domains and users. We have approached voicemail management in a Bayesian learning framework using stochastic language models to represent individual messages and groups of messages reflecting user profiles. Although still limited by the challenging speech recognition environment and the lack of any deep semantic analysis, we have reported improvements by training on a larger data set and by augmenting the language models with class-based models derived automatically from other corpora.

Acknowledgments

This work is supported by a Marie Curie fellowship.

References

1. Moreno, P., Thong, J.M.V., Logan, B., Jones, G.J.F.: From multimedia retrieval to knowledge management. IEEE Computer **35** (2002) 58–66
2. Hirschberg, J., Bacchiani, M., Hindle, D., Isenhour, P., Rosenberg, A., Stark, L., Stead, L., Whittaker, S., Zamchick, G.: SCANMail: Browsing and searching speech data by content. In: Proc. Eurospeech, Aalborg, Denmark (2001)
3. Koumpis, K., Ladas, C., Renals, S.: An advanced integrated architecture for wireless voicemail retrieval. In: Proc. 15th IEEE Intl. Conf. on Information Networking, Beppu, Japan (2001) 403–410
4. Huang, J., Zweig, G., Padmanabhan, M.: Information extraction from voicemail. In: 39th Annual Meeting of Assoc. for Computational Linguistics, Toulouse, France. (2001)

5. Ringel, M., Hirschberg, J.: Automated message prioritization: Making voicemail retrieval more efficient. In: Proc. Conf. on Human Factors in Computing Systems (Ext. Abstracts), Minneapolis, MN, USA (2002) 592–593
6. Yang, Y.: An evaluation of statistical approaches to text categorization. Journal of Information Retrieval **1** (1999) 67–88
7. Lewis, D.D., Schapire, R.E., Callan, J.P., Papka, R.: Algorithms for linear text classifiers. In: Proc. 19th annual Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval. (1996) 298–306
8. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys **34** (2002) 1–47
9. Teahan, W.J., Harper, D.J.: Using compression based language models for text categorization. In: Proc. Workshop on Language Modeling and Information Retrieval, Carnegie Mellon University, USA (2001) 83–88
10. Peng, F., Schuurmans, D., Kaselj, V., Wang, S.: Automated authorship attribution with character level language models. In: Proc. 10th Conf. of European Chapter of Assoc. for Computational Linguistics, Budapest, Hungary (2003) 19–24
11. Padmanabhan, M., Eide, E., Ramabhardan, G., Ramaswany, G., Bahl, L.: Speech recognition performance on a voicemail transcription task. In: Proc. IEEE ICASSP, Seattle, WA, USA (1998) 913–916
12. Koumpis, K., Renals, S.: The role of prosody in a voicemail summarization system. In: Proc. ISCA Workshop on Prosody in Speech Recognition and Understanding, Red Bank, NJ, USA (2001) 87–92
13. Cordoba, R., Woodland, P.C., Gales, M.J.F.: Improving cross task performance using MMI training. In: Proc. IEEE ICASSP. Volume 1., Orlando, FL, USA (2002) 85–88
14. Koumpis, K.: Automatic categorization of voicemail transcripts using stochastic language models. In: Proc. 7th Int. Conf. on Text, Speech and Dialogue, Brno, Czech Republic (2004)
15. Charlet, D.: Speaker indexing for retrieval of voicemail messages. In: Proc. IEEE ICASSP. Volume 1., Orlando, FL, USA (2002) 121–124
16. Gotoh, Y., Renals, S.: Statistical language modelling. In Renals, S., Grefenstette, G., eds.: Text and Speech Triggered Information Access. Springer-Verlag (2003) 78–105
17. Jelinek, F., Mercer, R.L., Bahl, L.R., Baker, J.K.: Perplexity - a measure of difficulty of speech recognition tasks. In: Proc. 94th Meeting Acoustical Society of America, Miami Beach, Florida, USA (1977)
18. Brown, P.F., Pietra, V.J.D., deSouza, P.V., Lai, J.C., Mercer, R.L.: Class-based n-gram models of natural language. Computational Linguistics **18** (1992) 467–479
19. Chen, S., Goodman, J.: An empirical study of smoothing techniques for language modeling. Computer Speech and Language **13** (1999) 359–394
20. Carletta, J.: Assessing agreement on classification tasks: The kappa statistic. Computational Linguistics **22** (1996) 249–254