EXPERIMENTS IN TOPIC INDEXING OF BROADCAST NEWS USING NEURAL NETWORKS

Christoph Neukirchen, Daniel Willett, Gerhard Rigoll

Department of Computer Science
Faculty of Electrical Engineering
Gerhard-Mercator-University Duisburg, Germany
e-mail: {chn,willett,rigoll}@fb9-ti.uni-duisburg.de

ABSTRACT

The paper deals with the problem of extracting topic information from news show stories by statistical methods. It is shown that the traditional topic-dependent n-gram language modeling approach can be decomposed in order to apply neural networks for topic indexing. Two specific problems in training of these networks are addressed: a very sparse data distribution in the stories and the superposition of different topics in a story. The first problem is tackled by an integrated smoothing approach in the backpropagation method; an expansion of the neural network structure can be used to cope with topic mixtures in stories. Due to the efficient parameter sharing the application of neural networks results in a small improvement in topic indexing performance on a small corpus of broadcast news compared to the traditional topic-dependent n-gram method.

1. INTRODUCTION

The automatic extraction of topic information from stories as they appear e.g. in broadcast news shows is useful for information retrieval, categorization and adaptation of speech recognition systems. The common approaches for topic indexing make either use of selecting keywords for each possible topic or they construct statistical models that describe the generation of stories probabilistically [5]. In this paper, the probabilistic approach is extended by the incorporation of neural networks for the approximation of topic probabilities. The neural networks are trained discriminatively and and the number of trainable weight parameters can be adjusted to the amount of training data.

When the task is to extract one single topic (out of a set of J different topics) from a given story the selection of the most probable topic yields the minimum expectation of false assignment. In a Bayesian approach the conditional topic probability is decomposed as usual:

$$P(Topic_j|Story) = P(Topic_j) \cdot \frac{P(Story|Topic_j)}{P(Story)} \qquad (1)$$

Suppose a corpus of stories with topic labels is given; then a maximum likelihood model for the prior $P(Topic_j)$ that a story that is related to the j-th topic occurs can be derived from the relative frequency counts of this topic. The structure of the model for the conditional likelihood $P(Story|Topic_j)$ depends on the type of elements that are used to represent the story. When the story is given as a waveform of a spoken utterance a suitable representation can be formed by a sequence of acoustic feature vectors, vector quantized discrete labels of these features, or by the sequence of (sub-)phonetic classes like the phonemes or the states of an acoustic HMM [8]. Since the topic of a story is in general coded in the meaning of the words of the story the application of speech recognition technology is a reasonable intermediate step to generate word transcriptions [6]. If the story is given as text (either obtained by a recognition system or given as a printed article) the string of words or of the sentences form suitable elements.

In the following, we assume that a story is given as a string $W_1^T = (w_1, \ldots, w_T)$ of T items; although the items w_t are referred to as words they can be actually any of the elements mentioned above. Then the conditional story likelihood can be written as:

$$P(Story|Topic_j) = P(W_1^T|j) \qquad (2)$$

1.1. Topic-dependent n-gram language models

A common way to describe the story probability in Eqn. (2) is the application of topic-dependent n-gram language modeling, that assumes that a topic-dependent word probability only depends on the sequence W_{t-n+1}^{t-1} of the previous $n-1$ words. In this case, Eqn. (2) is approximated by the product of n-gram probabilities:

$$P(W_1^T|j) = \prod_{t=1}^{T} P(w_t|W_{t-n+1}^{t-1}, j) \qquad (3)$$

In general, the estimation of a large number of reliable n-gram probabilities suffers from the problem of sparse data and never occurring events in the training corpus; this situation is even more dramatic when n-grams have to be constructed for each of the J different topics. Smoothing methods like model interpolation are a suitable means to avoid zero probabilities: in [8] n-gram topic models are interpolated with topic models of smaller history dependency; in [5] topic-dependent language models are interpolated with topic-independent models.

2. NEURAL NETWORKS FOR TOPIC CLASSIFICATION

The topic-dependent n-gram probabilities that are multiplied in Eqn. (3) can be decomposed by making use of Bayes rule into:

$$P(w_t|W_{t-n+1}^{t-1}, j) = \frac{P(j|W_{t-n+1}^{t})}{P(j|W_{t-n+1}^{t-1})} \cdot P(w_t|W_{t-n+1}^{t-1}) \qquad (4)$$

The rightmost expression in Eqn. (4) is the traditional n-gram probability that is not related to any topic. The nominator represents the probability that the j-th topic occurs given the history of the past n words (including the current word w_t); the denominator depends on the same history without the current word (i.e. $n-1$ words).

According to Eqn. (1) the index of the most probable topic can be determined in this case by:

$$\arg\max_j \left\{ P(Topic_j) \cdot \prod_{t=1}^{T} \frac{P(j|W_{t-n+1}^{t})}{P(j|W_{t-n+1}^{t-1})} \right\} \qquad (5)$$

Neural networks (NN) are well known tools that can be used to generate class probabilities for a given feature vector $\mathbf{x}$ [9]. If the NN outputs are denoted by $O_j(\mathbf{x})$ ($1 \leq j \leq J$, for J different pattern classes) and a suitable training criterion (mean squared error, cross entropy) is optimized it can be shown that the outputs approximate the class posterior probabilities $O_j(\mathbf{x}) = P(j|\mathbf{x})$ [1].

For topic indexing tasks two different NNs can be applied to approximate the probabilities $P(j|W_{t-n+1}^{t})$ and $P(j|W_{t-n+1}^{t-1})$. The first NN uses a suitable representation of the n-word sequence W_{t-n+1}^{t} as network input, the second one uses the sequence W_{t-n+1}^{t-1}. For training these NNs the targets are taken from the topic labeling of the training stories. A simple way to represent a n-word sequence with the words taken from a vocabulary of size V is to use a $V \cdot n$ component binary feature vector with only n bits nonzero; these bits indicate the index of each word in the sequence. Although this forms a very large vector backpropagation's gradient computation can still be performed fast because a large number of features are zero, they do not influence the gradient. The advantageous aspect of this NN approach for topic indexing is the ability to scale the NN complexity with respect to the amount of training data by adjusting the number of hidden nodes. Thus, a robust way of probability parameter sharing can be incorporated in the NN structure.

2.1. Unigram based neural networks for topic classification

An additional common way to cope with the problem of limited training data that can be applied to the proposed NN approach as well as to traditional topic-dependent models is to reduce the n-gram order down to $n = 1$ (i.e. a unigram). Here, the NN that models the probabilities in the nominator of Eqn. (5) makes use of the single word w (e.g. represented by a V-component binary vector) as network input; the NN output approximates the topic posterior probability given this word, i.e. $O_j(w) = P(j|w)$.

In the unigram case, the denominator in Eqn. (5) collapses into $P(j)$, i.e. the prior probability that a word occurs in a story that is related to the j-th topic. Given the NN model for $P(j|w)$ the prior probability can be obtained by averaging the probability over all words that are contained in the stories of the training corpus.

$$P(j) = \sum_{w \in Vocab.} P(j|w) \cdot P(w) \approx \frac{1}{T} \cdot \sum_{t=1}^{T} O_j(w_t) \quad (6)$$

Since all these prior probabilities can be stored in a table there is no need to construct a second NN in this case. It must be noted that in general $P(j)$ differs from $P(Topic_j)$, because $P(j)$ refers to the words in stories related to $Topic_j$, hence it is influenced by the length of the stories (i.e. the number of contained words), and the probability $P(Topic_j)$ refers to the occurrence of stories without taking their length into account.

Now, the most probable story topic can be determined by

$$\underset{j}{\operatorname{argmax}} \left\{ P(Topic_j) \cdot \prod_{t=1}^{T} \left(\frac{O_j(w_t)}{P(j)} \right)^{\alpha} \right\} \quad (7)$$

As in [7], the tuning parameter α is used to compensate for the incorrect unigram independence assumption.

Eqn. (7) is directly related to the speech recognition approach using the hybrid system in [2] where phone posterior probabilities generated by a NN are divided by phone prior probabilities.

2.2. Neural network smoothing

Cross-validation is a common method to avoid parameter overfitting when training NNs as pattern classifier [2]. Various smoothing methods [4] can be used in general to produce robust probability estimates for (topic-dependent) language models. A simple smoothing method that avoids zero probabilities is to increase the frequency counts of all possible discrete events by one (one-plus smoothing [4]).

For backpropagation training of topic indexing NNs as they are used in Section 2.1. the one-plus smoothing method can be applied in a naive way: the training data set is augmented by all possible topic-word-pairs; this increases the training set (and training time) by $V \cdot J$ samples. In this case, the ideal target value at a fixed NN output node for a fixed word at the NN input will be zero $J-1$ times and 1.0 once. Since the usual NN training criteria (mean squared error, cross entropy) are additive for each sample the same smoothing impact on the gradients can be obtained faster as follows : two gradient sums are calculated for all of the V different words in the vocabulary by the backpropagation method: one for the target vector containing 0.0 components and one for the target vector containing 1.0 components. Finally, the first gradient term is added to the training data gradient with a weighting factor of $J-1$, the second expression is weighted by 1.0. Thus, the additional effort for smoothing can be reduced to $2 \cdot V$.

3. MULTIPLE TOPIC INDEXING

In general, stories like broadcast news texts deal with a couple of topics instead of one single topic; different words in a story are related to different topics and the major part of a story's words can not be assigned to any specific topic (general language words). Thus, in [7] the indexing task is extended to determine the most probable set of topics (out of I different sets) for a given story. A set-dependent unigram approach is adopted in [7]; each set-dependent unigram probability is modeled as a mixture of topic-dependent unigrams using all the topics being contained in the set:

$$P(w|Set_i) = \sum_{\substack{j \\ Topic_j \in Set_i}} P(j|Set_i) \cdot P(w|j) \quad (8)$$

A unigram model $P(w|0)$ for the $Topic_0 = general\ language$ is a component of each mixture model.

In order to apply the neural network approach of Section 2.1. that approximates $P(j|w)$ by $O_j(w)$ to the problem of topic set indexing Eqn. (8) must be transformed by using Bayes rule into:

$$P(Set_i|w) = \sum_{\substack{j \\ Topic_j \in Set_i}} P(Set_i|j) \cdot P(j|w) \quad (9)$$

Eqn. (9) motivates the expansion of the topic indexing neural network by one additional layer of weights in order to approximate the topic set posterior probabilities $P(Set_i|w)$ by the outputs $\tilde{O}_i(w)$ of the new network: The probabilities $P(Set_i|j)$ can be interpreted as weighting connections between the j-th output node (with $O_j(w)$) of the topic classifying network of Section 2.1. to the i-th output node of the new network (with $\tilde{O}_i(w)$). If the network outputs $O_j(w)$ and the new weighting connections are constrained to have the probability properties (being positive and summing up to unity e.g. by application of the softmax function [3]) it can be shown from Eqn. (9) that the new network outputs $\tilde{O}_i(w)$ will also have these properties. When the unconstrained output weight parameters $g_{j,i}$ are used for connecting the j-th node with the i-output node the constrained network outputs can be generated by:

$$\tilde{O}_i(w) = \sum_{\substack{j \\ Topic_j \in Set_i}} \frac{\exp(g_{j,i})}{\sum_{i'} \exp(g_{j,i'})} \cdot O_j(w) \quad (10)$$

The new weight parameters $g_{j,i}$ as well as the parameters in the traditional NN can be determined by backpropagation training. To be able to identify general language words the traditional NN makes use of an additional output node with $O_0(w)$ that is connected to all the topic set nodes (see Fig. 1). This approach allows the neural network to identify topic mixtures in the training stories in an unsupervised way; the distributions of the NN outputs are expected to be shaped much sharper compared to the approach of Section 2.1..

This new method of NN training can be generalized to (semi-)supervised connectionist classifier design when there is uncertainty about the specific classes of the training patterns. As shown in Fig. 1 the traditional NN structure is expanded by an additional layer of weights (that must have properties of probabilities). For each pattern class a new output node is used; when the pattern class of a training sample is not known exactly a set node

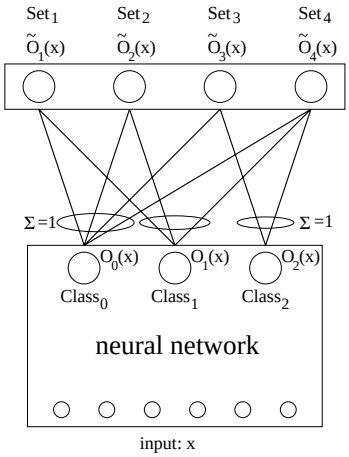

Figure 1. Expansion of the connectionist classifier architecture to cope with sets of pattern classes.

that contains all classes that might suit this sample is introduced. After training of this new structure the intermediate outputs $O_j(\mathbf{x})$ of the traditional NN can be used as approximated class posterior probabilities.

Now, by using the network outputs $O_j(\mathbf{x})$ the optimal topic set of a given story can be determined by

$$\underset{i}{\operatorname{argmax}} \left\{ P(Set_i) \cdot \prod_{t=1}^{T} \sum_{\substack{j \\ Topic_j \in Set_i}} P(j|Set_i)^{\beta} \cdot \frac{O_j(w_t)}{P(j)} \right\} \quad (11)$$

with the exponential constant β to compensate the independence assumptions [7].

In practice, the number of different topic sets is very large. This makes the estimation of reliable priors $P(Set_i)$ and mixture weights $P(j|Set_i)$ difficult, and the search for the best topic set becomes intractable. Thus, generating a topic set as a list of the N-best topics by considering each topic separately is an alternative approach that has proven to yield similar performance as Eqn. (11) in [7]:

$$\underset{j}{\operatorname{argmax}} \left\{ P(Topic_j) \cdot \prod_{t=1}^{T} \phi \left(P(j)^{\beta} \cdot \frac{O_j(w_t)}{P(j)} \right) \right\} \quad (12)$$

As mentioned above, in the mixture NN training framework the network outputs are expected to be quite sharp (many zero outputs); thus, in Eqn. (12) as in [7] the filter function $\phi(x) = x$ if $(x > \theta)$ or θ if $(x \leq \theta)$ is used to avoid zero probabilities for words in a story that are not related to the topic under consideration.

4. EXPERIMENTS

As a story database for the experiments the transcriptions of MarketPlace business radio broadcast news shows are used. The transcriptions are taken from the DARPA HUB4 1995 CD-ROMs.

The training data set consists of 105 stories from 10 different news shows, comprising 43,000 words. This is a very small corpus of stories for training topic models, but since audio data is available for all these texts in future experiments comparisons using transcriptions generated by speech recognizers can be obtained. The test sets are subdivided into 6 news shows as validation data and 5 news shows as evaluation test data. The validation part consists of 26,000 words in 62 stories; the test set consists of 22,000 words in 43 stories.

Each story is labeled by one up to twelve topic indices (an additional label for the *general language* topic is assigned to each story automatically). In total 62 different topics (excluding *general language*) are used for system training; on the average 5.6 topics are assigned to a story in the training data set. The out-of-topic rate is 4% and 7% in the validation and the test data, respectively.

Because the size of the training data set is quite small all the words in the transcriptions are preprocessed to generate word baseforms by removing common suffixes like "-ed", "-ing", etc. This yields a vocabulary size of approximately 5,000 word baseforms in the training transcriptions. The out-of-vocabulary rate in the validation and in the test sets is about 10%.

4.1. Neural network training

In the single topic training approach (Section 2.1.) two-layer NNs with 63 output nodes (for the 62 topics plus *general language*) and the softmax output function are trained to minimize the mean squared error criterion. The number of hidden nodes are set to 3, 10 and 50 corresponding to approximately 15,000, 50,000 and 250,000 weight parameters, respectively. The words in the stories that are related to several topics are used as multiple training patterns, yielding 300,000 training patterns totally.

As a baseline system a traditional topic dependent unigram language model (with 315,000 parameters) is created from the training stories.

For the experiments in the mixture of topics training framework (Section 3.) the NNs trained as explained above are used to initialize the new network structure. A new layer of weights that implements the mapping of Eqn. 10 is added to the network as shown in Fig. 1. The number of topic sets (i.e. the new network output layer size) is 103. After training of this new system the output weight layer is removed and the outputs $O_j(w)$ of the original NN are used as topic posteriors.

The tuning parameters α, β and θ are optimized on the validation set.

4.2. Results

The different behavior of NNs that are trained by the single topic assumption and of NNs that make use of a mixture of topics structure is shown in Tab. 3. For some characteristic words the most probable topic and the corresponding topic probability generated by the NNs are given. The traditionally trained NN yields low probabilities and *general language* is chosen frequently because of its high prior probability in the training set. In the mixture framework the NN generates very high probabilities in many cases what corresponds to the identification of specific key words to a topic. *General language* is assigned to words that occur in many different stories.

Since the test stories are labeled by multiple topics the performance assessment is based on sets of topics generated by the neural networks. N-best lists are generated for the traditionally trained systems by using the decoding rule Eqn. 7; for the NNs that are trained by integrating the topic mixture assumption Eqn. 12 is used. The at-least-one accuracy [7] (that is the fraction of stories to which at least one of the found topics is related) is shown in Tab. 1. The topic indexing precision [7] (that is the fraction of found topics that are in the set of annotated topics) is shown in Tab. 2.

For the NNs as well as for the traditional unigram model parameter smoothing improves the performance on the unseen test stories significantly. The large neural network with 50 hidden nodes yields higher precision and accuracy than the unigram model for a small number of topics per story. The precision of the top-choice is improved from 53% to 63%. When the size of the generated n-best list is increased both systems perform similar. In contrast to [7] the usage of topic models that are trained under the mixture of topic assumption does neither improve the precision nor the accuracy. Although the new models seem to provide a better discrimination on the level of single words (as shown in Tab. 3) the performance is worse on the level of whole stories. The fact that the mixture model performance even degrades on the training data gives evidence that the simplifications made in Eqn. 12 are a bad approximation of the optimal decoding rule in Eqn. 11; in particular, the usage of the global priors $P(j)$ instead of the different

model	num.	training set					validation set					test set				
	hidden	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
unigram	-	100	100	100	100	100	48	71	82	88	91	42	58	70	86	98
unigram (smoothed)	-	99	99	99	99	100	65	79	84	90	92	53	74	84	91	95
NN	3	63	79	88	90	92	48	68	79	82	89	42	60	77	81	84
NN	10	95	100	100	100	100	58	76	85	90	92	44	56	70	81	84
NN ("1+"-smoothed)	10	83	96	98	98	99	60	76	81	85	89	49	72	77	88	95
NN	50	100	100	100	100	100	56	79	85	87	87	44	60	79	95	98
NN ("1+"-smoothed)	50	98	99	99	99	99	66	77	84	85	89	63	81	86	91	95
mixture NN	50	74	84	88	92	94	48	66	68	79	84	42	49	56	81	84

Table 1. At-least-one accuracy (given in percent) for topic-dependent unigrams and for neural networks with varying hidden layer size. The number of recognized topics per story runs from 1 to 5.

model	num.	training set					validation set					test set				
	hidden	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
unigram	-	100	100	98	94	88	48	48	45	44	41	42	42	40	42	40
unigram (smoothed)	-	99	98	95	90	84	65	57	52	52	46	53	52	45	44	41
NN	3	63	52	44	44	41	48	45	41	38	38	42	34	33	31	28
NN	10	95	95	89	83	77	58	56	53	50	45	44	40	37	40	39
NN ("1+"-smoothed)	10	83	80	76	71	64	60	54	46	42	39	49	49	43	41	37
NN	50	100	100	98	93	87	56	56	52	47	44	44	42	44	44	40
NN ("1+"-smoothed)	50	98	95	94	88	81	66	55	51	46	43	63	56	47	43	38
mixture NN	50	74	56	46	40	36	48	41	38	36	35	42	31	29	33	32

Table 2. Precision (given in percent) for topic-dependent unigrams and for neural networks with varying hidden layer size. The number of recognized topics per story runs from 1 to 5.

mixture weights $P(j|Set_i)$ is unreasonable in this case.

word	mixture training		traditional training	
	topic_{max}	$O_j(w)$	topic_{max}	$O_j(w)$
a	gen. language	0.91	gen. language	0.14
acre	animal	1.00	environment	0.16
amphitheater	entertainment	1.00	interview	0.38
apartheid	south-africa	0.93	gen. language	0.11
apex	conference	0.48	conference	0.10
are	gen. language	0.86	gen. language	0.14
arkansas	scandal	0.74	scandal	0.18
assassinate	crime	0.83	crime	0.13
bankrupt	business	0.99	company	0.28
before	gen. language	0.66	gen. language	0.13
bird	animal	1.00	environment	0.17
blackout	energy	1.00	gen. language	0.22
bootleg	china	1.00	asia	0.12
broker	stock-market	0.83	shopping	0.21

Table 3. A selection of several words and their most probable topic determined by the NNs (10 hidden nodes) using the traditional training and the mixture training method, the corresponding NN output value is given.

5. CONCLUSIONS

This paper presented an integration of neural networks in a probabilistic topic indexing framework. A method that allows a weight parameter smoothing within the backpropagation training procedure and a new approach to identify mixtures of topics in stories by a NN was proposed. These innovations can be applied to improve the performance of general NN based pattern recognition systems. Due to parameter reduction and smoothing the topic indexing performance of the NN is better than the traditional unigram approach for a small number of identified topics on a limited data set. The NN that is based on the topic mixture assumption is able to improve the discrimination at the word level but on the story level it performs worse. A larger corpus of stories will be used in the future to investigate the limitations of mixture NN approach in more detail.

REFERENCES

[1] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford: Clarendon Press, 1995.

[2] H. Bourlard, N. Morgan, *Connectionist Speech Recognition*, Amsterdam: Kluver, 1994.

[3] J. Bridle, "Probabilistic Interpretation of Feedforward Classification Network Outputs with Relationships to Statistical Pattern Recognition", *Neurocomputing: Algorithms, Architectures and Applications*, NATO ASI series, Berlin: Springer, 1990, pp. 227–236.

[4] S. Chen, J. Goodman, "An Empirical Study of Smoothing Techniques for Language Modeling", *Proceedings 34th Annual Meeting of the Association for Computational Linguistics*, 1996, pp. 310–318.

[5] J. McDonough, H. Gish, "Issues in Topic Identification on the Switchboard Corpus", *Proceedings International Conference on Spoken Language Processing (ICSLP)*, 1994, pp. 2163–2166.

[6] L. Gillick, et al., "Application of Large Vocabulary Continuous Speech Recognition to Topic and Speaker Identification Using Telephone Speech", *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1993, pp. 471–474.

[7] T. Imai, R. Schwartz, et al., "Improved Topic Discrimination of Broadcast News Using a Model of Multiple Simultaneous Topics", *Proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 1997, pp. 727–730.

[8] E. Nöth, S. Harbeck, et al., "A Frame and Segment Based Approach for Topic Spotting", *Proceedings European Conference on Speech Comm. and Technology (Eurospeech)*, 1997, pp. 257–278.

[9] M. Richard, R. Lippmann, "Neural Network Classifiers Estimate Bayesian A-Posteriori Probabilities", *Neural Computation*, Vol. 3, No. 4, 1991, pp. 461–483.