HELSINKI UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION AND NATURAL SCIENCES
DEPARTMENT OF INFORMATION AND COMPUTER SCIENCE

Teemu Ruokolainen

Topic adaptation for speech recognition in multimodal environment

Master's thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Technology

Espoo, 4th August 2009

Supervisor: Prof. Erkki Oja
Instructor: Mikko Kurimo, D.Sc. (Tech.)

HELSINKI UNIVERSITY OF TECHNOLOGY Faculty of Information and Natural Sciences Degree Programme of Computer Science and Engineering			ABSTRACT OF MASTER'S THESIS
Author	Teemu Ruokolainen	Date	4th August 2009
		Pages	5 + 58
Title of thesis	Topic adaptation for speech recognition in multimodal environment		
Professorship	Computer and information science	Code	T-61
Supervisor	Prof. Erkki Oja		
Instructor	Mikko Kurimo, D.Sc. (Tech.)		

Automatic speech recognition system consists of two basic elements, the acoustic model and the language model. In topic adaptation of the language model, we take into account the underlying topic of speech by elevating the probabilites of the subvocabulary characteristic to its topic. Via topic adaptation, we aim at improving the recognition of topically important words.

The potential benefit of topic adaptation relies on the success of retrieving the underlying topic correctly. Given a sufficiently large amount of keywords related to the topic, we can be confident that the retrieved topic is accurate. Traditionally, the keywords are extracted from a textual document or the transcription provided by the recognizer itself. However, due to the development of multimodal interfaces, we are interested in a scenario where the keywords are provided by an abstract modal source and no guarantees of the sufficient size or reliability of the keywords can be assumed.

In this work, we discuss the prospect of topic adaptation using small-sized and potentially unreliable topical keyword lists. The topic retrieval and speech recognition results are evaluated in large vocabulary continuous speech recognition task with English newswire data. The results indicate that successful topic retrieval using small-sized cues is feasible. However, topic adaptation did not either improve or degrade the speech recognition performance on the whole.

Keywords

topic adaptation, topic retrieval

TEKNILLINEN KORKEAKOULU Informaatio- ja luonnontieteiden tiedekunta Tietotekniikan koulutusohjelma/tutkinto-ohjelma		DIPLOMITYÖN TIIVISTELMÄ	
Tekijä	Teemu Ruokolainen	Päiväys	4. Elokuuta 2009
		Sivumäärä	5 + 58
Työn nimi	Puheentunnistuksen aiheadaptaatio multimodaalisessa ympäristössä		
Professuuri	Informaatiotekniikka	Koodi	T-61
Työn valvoja	Prof. Erkki Oja		
Työn ohjaaja	TkT Mikko Kurimo		

Automaattinen puheentunnistusjärjestelmä koostuu kahdesta peruskomponentista, akustisesta ja kielimallista. Kielimallin aiheadaptoinnilla otetaan huomioon puheen aihe nostamalla aiheelle tyypillisten sanojen todennäköisyyksiä. Aiheadaptoinnin avulla pyritään parantamaan aiheen kannalta oleellisten sanojen tunnistamista.

Aiheadaptoinnin mahdollinen hyöty riippuu oikean aiheen haun onnistumisesta. Mikäli käytettävissä oleva, aiheeseen liittyvä avainsanalista on riittävän suuri, voidaan olettaa, että aihehaku tapahtuu onnistuneesti. Yleensä avainsanat on saatu tekstimuotoisista dokumenteista tai puheentunnistimen itsensä tuottamasta tunnistustuloksesta. Multimodaalisten käyttöliittymien kehittymisen myötä on kuitenkin kiinnostavaa tutkia tilannetta, jossa avainsanat ovat peräisin yleiseltä modaaliselta lähteeltä. Tällöin avainsanalistan riittävää kokoa tai luotettavuutta ei voida olettaa.

Tässä työssä käsitellään aiheadaptointia käyttäen pienikokoisia ja mahdollisesti epäluotettavia aihekohtaisia avainsanalistoja. Aihehakujen onnistumista ja puheentunnistustuloksia arvioidaan suuren sanaston jatkuvan puheen tunnistuksessa käyttäen englanninkielistä uutisaineistoa. Tulokset osoittavat, että onnistunut aihehaku on mahdollista tehdä pienellä avainsanamäärällä. Aihehaku ei kuitenkaan vaikuttanut parantavasti tai huonontavasti puheentunnistustulokseen kokonaisuudessaan.

Avainsanat

aiheadaptaatio, aihehaku

Foreword

This thesis was done in the Department of Information and Computer Science in TKK during the years 2008 and 2009 as a part of two projects, PinView and UI-ART, on multimodal interfaces. I thank my supervisor Erkki Oja and instructor Mikko Kurimo for the work opportunity and valuable corrections.

Teemu Ruokolainen
Otaniemi, 4th August, 2009

Contents

Symbols and abbreviations

W	Word sequence
O	Observation sequence
S	State sequence
w_τ	Word at time instant τ
o_τ	Observation at time instant τ
s_τ	State at time instant τ
w_i^n	Word sequence from i to n
$c(.)$	Number of instances in corpus
D	Document collection (corpus)
V	Vocabulary
F	Feature subset of V
N	Noise subset of V
$\|X\|$	Size of X
d	Document
$\mathbf{x}, \mathbf{y}$	Document feature vector
$\hat{t}$	Retrieval result (topic estimate)
q	Topic cue
$l(.)$	Likelihood
λ	Mixture coeffificent
B	Background corpus

BMU	Best matching unit
HMM	Hidden Markov model
LM	Language model
ppl	Perplexity
sim	Similarity
SOM	Self organizing map
tfidf	term frequency inverse document frequency
TER	Term error rate
WCR	Word change rate
WER	Word error rate

Chapter 1

Introduction

In automatic speech recognition, we aim at providing a textual transcription for a given speech signal. The modern approach to accomplish this relies on statistical methods to determine the text sequence which best matches the speech signal phonetically and makes the most sense lingually. As to lingual sensibility, we make use of the notion of topics as in differing discussion topics we typically have differing vocabulary. For example, in sports news articles, we are likely to encounter words such as *score*, *match*, and *league*, whereas finances news is more likely to contain words such as *funds*, *investment*, or *stocks*. Furthermore, we are interested in the means of inferring the underlying topic of discussion. As to speech communication, it is natural to make conclusions about the topic based on the words heard in the discussion itself. However, humans are multimodal beings observing their environment through vision, touch, tastes and smells, as well as hearing. Therefore, topic deduction is also affected by information provided by modalities other than speech.

Modern statistical speech recognition systems consist of three basic elements, the acoustic model, the language model and the decoder. The first two assign probabilities to phoneme and word sequences, respectively. In combination, they result in multiple word sequence hypotheses of which the decoder selects the best as the final transcription. The importance of language modeling in speech recognition can be verified easily by removing the language model block from the recognizer and watching the recognition performance plunge.

As to capturing the topical essence of the speech, it is desirable to raise the probability of capturing the topically prominent words. In order to achieve this, we can incorporate the language model with a topic adaptation procedure. In topic adaptation, we take into account the prevailing topic of the speech by elevating the probability of the subvocabulary characteristic to the topic. The information enabling the retrieval of the suitable topic is referred to as a topic cue. The topic cue has traditionally been provided using the recognition history of the speech recognizer itself.

Due to the development of multimodal applications, we are additionally interested in studying a speech recognition scenario where the topic cue for the recognizer is provided by a general modal source. It is expected that the topic cues provided by speech and differing modalities have differing intrinsic characteristics. Mainly, speech is best described as highly dependent word sequences whereas the cue words provided by other modalities are expected to be of much more fragmented and uncorrelated form. In addition, and importantly so, the multimodal topic cue is expected to be of small size, i.e. words instead of sentences. Consequently, these properties of the topic cues may set additional requirements for the topic adaptation procedure and particularly for the topic retrieval scheme.

In pursuit of maximal reductions in recognition errors, we would prefer to be provided with as much topic-specific data as possible to estimate the correct topic at hand. However, in multimodal environment, we expect the topic cues to be small in size. Therefore, in this work we focus on the problem of topic adaptation using small-sized topic cues in topic retrieval and cover the following two fundamental research questions.

1. Do small-sized topic cues enable a reliable topic retrieval?

2. Is successful/failed topic retrieval significantly beneficial/harmful to the recognition performance?

Related to the first question, we additionally examine if the source of the topic cue or the choice of the topic retrieval criterion have a significant effect on the topic retrieval.

This master's thesis was conducted as a part of the speech group of the Computer and Information Science laboratory (presently the department of Information and Computer science) in TKK. During the last decade, the research in the group has been spread on a wide focus on the speech recognizer decoder implementation [1], acoustic modeling [2, 3] and language modeling [4, 5, 6, 7, 8]. This thesis extends the work on language modeling with a specific focus on the development of relevantly novel field of multimodal interfaces. The topic adaptation procedure discussed in this work is based largely on the approach described in [9].

The rest of the work is organized as follows. In chapter 2, we describe the fundamental statistical approach to language modeling, motivate the use of topic adaptation techniques, and introduce the procedure implemented in this work. In chapter 3, alternative implementation techniques of the adaptation procedure are discussed. In chapter 4, we present the adaptation experiments and the results with analysis and discussion. Finally, the conclusions on the work are presented in chapter 5.

Chapter 2

Fundamentals of speech recognition

2.1 Statistical speech recognition

An automatic speech recognizer aims at providing a textual transcription for a given speech signal. Modern statistical speech recognition systems comprise five fundamental components: feature extraction, acoustic models, language models, lexicon, and decoder. The flowchart of the system is presented in Figure 2.1. In the following, we describe the components in brief. A thorough description of the system can be found in [1]. Basic literature on the subject can be found, for example, in [10].

Feature extraction

In feature extraction, we acquire acoustic features from the speech signal. The features are in the form of mel-frequency cepstral coefficients (MFCC) and their 1st and 2nd derivatives extracted from short time windows. With the Mel-scale, we execute a non-linear transformation for frequencies in order to take into account the varying resolution of human auditory system in frequency domain. Furthermore, with discrete cosine transformation (DCT), we map the coefficients to the cepstral domain. The observed acoustic features at time instant τ, $O = \{o_\tau\}$, are then fed forward to acoustic models.

Acoustic models

The acoustic models map the acoustic information from the MFCC features to sequences of some basic units of speech, for example, phonemes. This

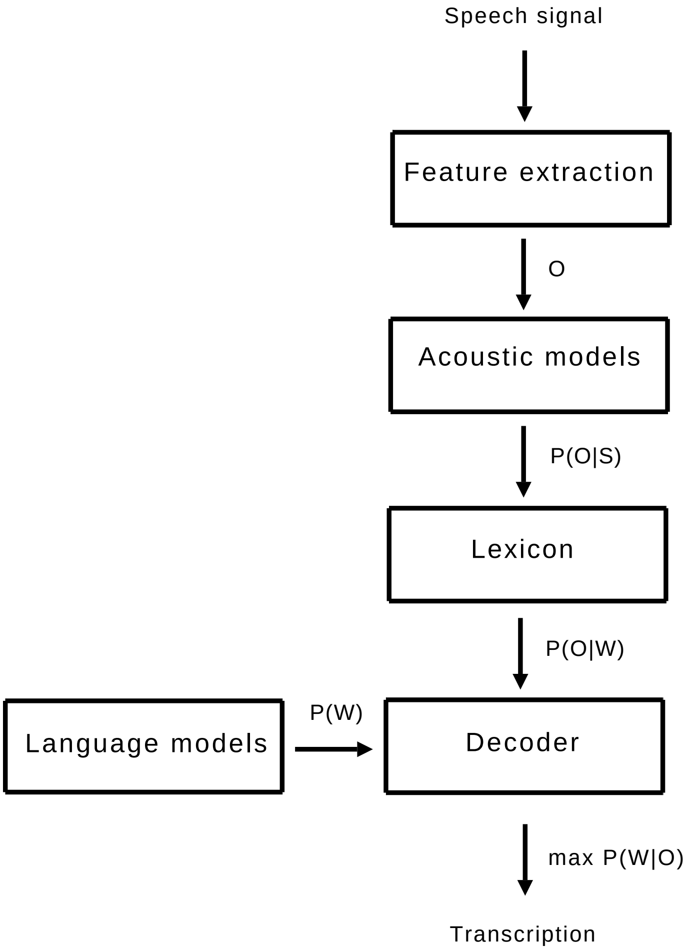


Figure 2.1: Speech recognition system.

is done using Hidden Markov Models (HMMs) [11] defined by prior state probabilities, transition probabilities between the states and emission probability distributions. Here the hidden states correspond to the phonemes and the emitted observations to the features presented above. The output of the acoustic model is the set of likelihoods of different state sequences $S = \{s_\tau\}$ generating the given observation sequence, i.e. $P(O|S)$. As acoustic models were needed to map the acoustic information from the feature vectors to states, lexicon, in turn, defines a mapping from the states to the language model vocabulary.

Language models

Essentially, language models assign probabilites for word sequences $W = \{w_1, \ldots, w_N\}$, i.e. $P(W)$. Most commonly, this is accomplished using n-gram models. The n-gram models are further discussed in section 2.2.

Decoder

Given the output of the acoustic and language models, $P(O|W)$ and $P(W)$, the decoder finds the best hypothesis for the transcription. This is equivalent to finding the maximum a posteriori, i.e. $\arg\max_W P(W|O) \propto P(O|W)P(W)$. In HMM-based system solutions this is accomplished using the Viterbi algorithm [12].

2.2 N-gram models

In modern speech recognition systems the standard choice for statistical language modeling is the n-gram model [13]. The n-gram model is a fine example of a linguistically ill-posed and over simplified method which nonetheless has proven to be extremely efficient in practical use.

The n-gram models are used to predict the probability of a word given its immediately preceding words. Intuitively, unigram and higher order n-grams form word probability distributions and word sequence probability distributions, respectively. Formally, n-grams are nth order Markov chains [13] with a Markov property referred to as *the limited horizon*:

$$p(w_i|w_1^{i-1}) = p(w_i|w_{i-n+1}^{i-1})$$

where w_i denotes word at time instant i, $i \geq n$, and w_{i-n+1}^{i-1} the $n-1$ words preceding it. Moreover, we define that, in case of $i < n$, $p(w_1|w_1^0) = p(w_1)$, $p(w_2|w_1^1) = p(w_2|w_1)$, etc. This property is important in that it tells us that the current state of the system (s_i) depends only on the $n-1$ previous states (s_{i-n+1}^{i-1}), i.e. any word w_i is dependent only on the $n-1$ previous words w_{i-n+1}^{i-1}. The limited horizon property essentially makes the n-gram models very effective tool at capturing the local dependencies of the language. Increasing the order of model can improve performance of the model up to the order of 5 or 6 [14]. However, as this increase leads to rapid growth of memory requirements, n is typically in the range 2-4. The training of n-grams is discussed in section 3.4.

2.3 Long distance dependencies and topic information

As stated previously, n-gram models are effective at capturing the local dependencies between words. However, a real language only partially agrees with the limited horizon assumption. For instance, let's examine a scenario including the following sentences.

The cat meowed.
The cat, more or less surprisingly, meowed.

Now, the probability of encountering the word *meowed* following the word *cat* and the sequence *the cat, more or less surprisingly,* is likely to be roughly the same. However, in the latter case, any feasible n-gram model would be unable to capture this dependence since the limited length of word history does not endure over the fragment *more or less surprisingly.* In fact, an unreasonable n-gram of order 6 should be applied here in order to make the prediction plausible.

Let's then consider a second scenario rising from the development of multimodal interfaces, for instance an image annotating tool using speech. From other modalities included in the interface, we can in principle derive topical information for the language modeling. That is, a combination of image and eye movement data can offer us information about not only *what* is on the screen, but also *what is it there the user is likely to be interested in.* Let's assume we know the user has focused his attention at a screen, specifically on a picture depicting a *harbour.* Now, when commenting on his view, it is presumable that he will more likely utter the word *ship* than *chip.* If we can utilize this shift in probability distributions of words in our language model, it may make a crucial difference for the recogniser when it is deciding between such acoustically similar words.

To approach the above two scenarios, we need to merge knowledge of long distance dependencies between words, i.e. the shift in the probability distributions of words, into our language models. This can be accomplished using topic adaptation principles.

2.4 Topic adaptation procedure

The topic adaptation procedure followed in this work is depicted in Figure 2.2. Intuitively, we wish to obtain a large language model to capture the general features of the language and smaller models to take into account the shift in word frequencies due to topical changes. The previous is obtained from all the training data available, i.e. the background corpus, and the latter from subcorpuses derived by topical clustering of the background corpus. The final adapted model is a combination of the two types. In the following, we take a look at the main features of the procedure.

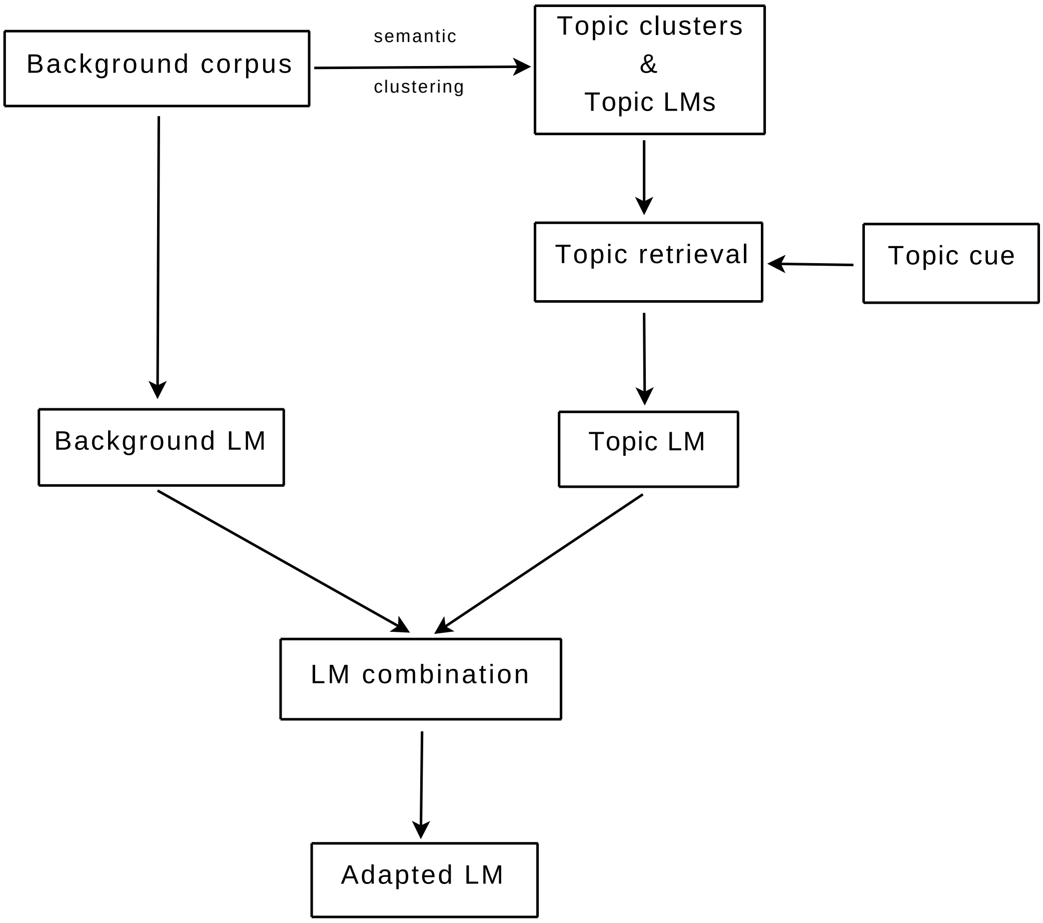


Figure 2.2: Topic adaptation procedure. LM stands for language model.

First, we assume that the background corpus consists of documents. Here, we assume a document to consist of a group of sentences of one or more topics. Topical clustering of text corpora is an extensively studied problem and an overview on the subject is presented in section 3.1. Here we would like to point out that, essentially, there is no single correct way to partition documents by topics, since topics may overlap inside documents and topic definition is ambiguous. Therefore, we are interested in obtaining in some sense well-founded partitioning of the data. If given a properly preprocessed corpus and executed correctly, it is justified to expect that all the state-of-the-art algorithms result in a sufficiently appropriate clustering. In this work we use self-organizing maps [15] to accomplish the clustering task. The clustering procedure using SOM is discussed in more detail in section 3.2.

After the background corpus has been partitioned into topic clusters, we are introduced to a topic cue provided by external information sources. In the case of multimodal applications, the sources are modalities including the recognition history provided by the speech recognition system itself. Typically, the topic cue is in the form of a keyword list. With the topic cue, we focus our attention on the topic currently at hand. This means that we wish to find *the cluster most similar to the topic cue* or, nearly equivalently but not quite, *the topic language model which has most likely generated the topic cue*. In this work, the topic retrieval process is of our prime interest and is discussed further in section 3.3.

After obtaining the wanted topic, we acquire the final adapted model by combining the language models trained based on background corpus and cluster corpus corresponding the retrieved topic. Again, language model combining is an extensively studied problem and a great amount of algorithms have been developed to accomplish this task. Combination methods, including the method used in this work, namely mixture models, are discussed further in section 3.6.

Chapter 3

Topic adaptation

3.1 Topical clustering of textual corpus

In clustering, we assign objects (data points) to clusters in an unsupervised manner so that similar objects are assigned to the same cluster and dissimilar objects to different clusters. As to clustering, we generally use the terms *objects* or *data points* but as we are interested in clustering of textual corpus, it is straighforward to refer to them as *documents*.

For the clustering of the corpus we present the documents using a vector-space model [13]. The general starting point is a document-word co-occurrence matrix $W \in \mathbf{R}^{|D| \times |V|}$, where $|D|$ and $|V|$ are the number of documents in corpus and words in vocabulary, respectively. Each document is presented as a word histogram in which the information of word order has been eliminated. This is known as the bag-of-words approach. Furthermore, it is beneficial to weight the words according to their frequency counts in individual documents to reflect their topical importance.

One of the most popular weighting schemes is the tfidf (term frequency inverse document frequency) weighting [13] scheme. We define tfidf weight for a given word-document pair as

$$\text{tfidf}_{ij} = \frac{|w_i \in d_j|}{|d_j|} \times \log \frac{|D|}{|d : w_i \in d|} \tag{3.1}$$

where $|w_i \in d_j|$ is the number of instances of word w_i in document d_j, $|d_j|$ is the total number of words in document d_j, $|D|$ is the total number of documents in corpus, and $|d : w_i \in d|$ is the number of documents where word i appears. In calculations, we use 10-based logarithms.

Another common weight [16] based on information theory for a given word-document pair is defined as

$$\mathrm{it}_{ij} = (1 - \epsilon_i)\frac{|w_i \in d_j|}{|d_j|} \tag{3.2}$$

where $|w_i \in d_j|$ is again the number of instances of word w_i in document d_j and $|d_j|$ the total number of words in document d_j, and ϵ_i the normalized entropy of w_i in the corpus D. The expression for ϵ_i is

$$\epsilon_i \; = \; -\frac{1}{\log|D|}\sum_{j=1}^{|D|}\frac{|w_i \in d_j|}{|w_i \in D|}\log\frac{|w_i \in d_j|}{|w_i \in D|} \tag{3.3}$$

where $|D|$ is the total number of documents in D, $|w_i \in d_j|$ the number of words w_i in document d_j, and $|w_i \in D|$ the total number of words w_i in the corpus D. Consequently, the weighting $(1 - \epsilon_i)$ describes the way the word w_i is distributed among the documents in the corpus D. As can easily be seen $0 \leq \epsilon_i \leq 1$, where the first equality holds when $|w_i \in d_j| = |w_i \in D|$. Respectively, the second equality holds when $|w_i \in d_j| = \frac{|w_i \in D|}{|D|}$. Consequently, a value of ϵ_i near 0 indicates that word w_i appears focused only in a few documents, whereas a value of ϵ_i near 1 indicates that word w_i is spread evenly among the documents in D.

Categorical data describes data where the data attributes (dimensions) do not have numerical values but rather qualitative interpretations. A traditional example of categorical data is the purchases of market customers where the attributes would correspond to items such as *milk*, *cheese*, *bread*, and *ice cream*. The attributes have binary values (zero or one) depending on if the customer has purchased the item in question or not. Textual data can be transformed into a categorical representation by simply replacing all the non-zero elements in the document-word matrix W with ones. The elements (w_{ij}) with value one in W can then be interpreted as "the word w_i appears in the document d_j at least once".

Depending on the choice of word weighting scheme, a variety of clustering algorithms can be applied to the corpus. In this section, we describe four examples of clustering methods, namely, Latent Semantic Analysis (LSA) [17], Agglomerative Clustering [18], Information-Theoretic Co-Clustering [19], and ROCK: Robust Clustering Algorithm for Categorical Attributes[20].

Latent Semantic Analysis

Latent Semantic Analysis (LSA) [17] is an effective means of determining document and term similarity in textual data. It is based on the notion of *concept space* obtained by singular value decomposition (SVD) of the document-word matrix. Latent Semantic Indexing (LSI) [21] utilizing LSA is a fundamental technique in Information Retrieval (IR) applications. As we are interested in document clustering, we will present LSA as a means of dimension reduction as follows.

We start with a document-word co-occurrence matrix $W \in \mathbf{R}^{|D|\times|V|}$, where $|D|$ and $|V|$ are the number of documents in corpus and words in vocabulary, respectively. Words in W are usually weighted using weighting schemes similar to (3.1) or (3.2). After word weighting, we rewrite W using singular value decomposition as

$$W = U\Sigma V^T \tag{3.4}$$

where $U \in \mathbf{R}^{|D|\times|D|}$ and $V \in \mathbf{R}^{|V|\times|V|}$ are orthonormal matrixes and $\Sigma \in \mathbf{R}^{|D|\times|V|}$ a pseudo-diagonal matrix holding the singular values. For W it holds

$$WW^T = U\Sigma V^T(U\Sigma V^T)^T = U\Sigma V^T V\Sigma^T U^T = U\Sigma\Sigma^T U^T \tag{3.5}$$

and

$$W^T W = (U\Sigma V^T)^T U\Sigma V^T = V\Sigma^T U^T U\Sigma V^T = V\Sigma^T\Sigma V^T \tag{3.6}$$

since for orthonormal matrixes it holds $V^TV = U^TU = I$. Therefore, as $\Sigma\Sigma^T$ and $\Sigma^T\Sigma$ are diagonal, we see that the columns of U contain the eigenvectors for $\Sigma\Sigma^T$ and columns of V the eigenvectors for $\Sigma^T\Sigma$. By selecting the k largest singular values from Σ and their corresponding eigenvectors from U and V, we result in k rank approximation W_k of the original matrix W, that is

$$W = U_k\Sigma_k V_k^T \tag{3.7}$$

where k corresponds to the wanted number of concepts extracted from W. Using Σ and V, we can now transform the original document vectors $\mathbf{d}_j \in \mathbf{R}^{|D|}$ into the new concept space $\hat{\mathbf{d}}_j \in \mathbf{R}^k$, $k \ll |D|$, with

$$\hat{\mathbf{d}}_j = \Sigma_k^{-1} V_k^T \mathbf{d}_j. \tag{3.8}$$

The documents can subsequently be clustered in this new lower dimensional space using suitable distance measures., e.g., distance

$$\text{distance}(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x}-\mathbf{y})^T(\mathbf{x}-\mathbf{y})}. \tag{3.9}$$

and the Agglomerative Algorithm [18] described in the next subsection.

Agglomerative Clustering

The Agglomerative Algorithm (presented e.g. in the context of Clustering Aggregation [18]) is an example of a simple means to obtain a partitioning for data. The algorithm is initialized by assigning each document into a singleton cluster. Then, we consider the pair of clusters with the largest average similarity. The similarity measure is defined as seen suitable depending on the used word weighting measure. Average similarity between two clusters is defined as the average of pairwise similarities between the documents in the two clusters. If the average similarity is greater than a predetermined threshold β, the two clusters are merged into a single cluster. These cluster comparisons and mergings are then iterated until there is no pair of clusters with average similarity greater than the threshold. In that case, the algorithm stops and gives the latest clusters as an output.

The algorithm can be illustrated with the following example. We will interpret the document-word matrix W as categorical data and, consequently, we introduce the Jaccard similarity coefficient [13]. Jaccard similarity coefficient for documents d_i and d_j is determined as the percentage of non-differing indexes between their binary feature vectors $\mathbf{d}_i, \mathbf{d}_j \in \mathbf{R}^M$, that is

$$J(\mathbf{d}_i, \mathbf{d}_j) = 1 - \frac{(\mathbf{d}_i - \mathbf{d}_j)^T(\mathbf{d}_i - \mathbf{d}_j)}{M}. \tag{3.10}$$

Consequently, the average similarity between two clusters C_k and C_l containing F and G documents $\{d_{k1}\,,\, d_{k2}\,,\,\dots\,,\, d_{kF}\}$ and $\{d_{l1}\,,\, d_{l2}\,,\,\dots\,,\, d_{lG}\}$ with corresponding feature vectors $\{\mathbf{d}_{k1}, \mathbf{d}_{k2}, ..., \mathbf{d}_{kf}, ..., \mathbf{d}_{kF}\}$ and $\{\mathbf{d}_{l1}, \mathbf{d}_{l2}, ..., \mathbf{d}_{lg}, ..., \mathbf{d}_{lG}\}$, is

$$J_{avg}(C_k, C_l) = \frac{\sum_{f=1}^{F} \sum_{g=1}^{G} J(\mathbf{d}_{kf}, \mathbf{d}_{lg})}{FG}\,. \tag{3.11}$$

Furthermore, since $0 \le J(\mathbf{d}_i, \mathbf{d}_j) \le 1$ for all (i, j) and, consequently, $0 \le J_{avg}(C_k, C_l) \le 1$ for all (k, l), a suitable threshold is $\beta = 0.5$.

We start with a document-word co-occurrence matrix $W \in \mathbf{R}^{|D|\times|V|}$, where $|D| = 4$ and $|V| = 5$ are the number of documents in corpus and words in vocabulary, respectively.

$$\mathbf{W} = \begin{pmatrix} 10 & 7 & 0 & 0 & 0 \\ 7 & 0 & 5 & 0 & 0 \\ 8 & 5 & 3 & 1 & 0 \\ 0 & 4 & 0 & 0 & 2 \end{pmatrix}$$

Let's form a new binary matrix W' by assigning value 1 to each non-zero element in W.

$$\mathbf{W'} = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$$

As a starting point, all the documents form their own singleton cluster.

$$\begin{array}{c|ccccc} \mathrm{C}_1 & 1 & 1 & 0 & 0 & 0 \\ \mathrm{C}_2 & 1 & 0 & 1 & 0 & 0 \\ \mathrm{C}_3 & 1 & 1 & 1 & 1 & 0 \\ \mathrm{C}_4 & 0 & 1 & 0 & 0 & 1 \end{array}$$

The pairwise average similarities for clusters C_1, C_2, C_3, and C_4 in W' can be gathered to a symmetric matrix where element (k, l) is the average Jaccard similarity of clusters C_k and C_l, i.e. $J_{avg}(C_i, C_j)$:

$$\begin{pmatrix} 1.0 & & & \\ 0.6 & 1.0 & & \\ 0.6 & 0.6 & 1.0 & \\ 0.6 & 0.2 & 0.2 & 1.0 \end{pmatrix}$$

The diagonal holds similarities 1.0 but these elements are ignored since a cluster can not be merged with itself. Now we see that at least $J_{avg}(C_1, C_2)$ exceeds the threshod $\beta = 0.5$ so clusters C_1 and C_2 can be merged together. The new clustering is therefore

$$\begin{array}{c|ccccc} \mathrm{C}_1 & 1 & 1 & 0 & 0 & 0 \\ \mathrm{C}_1 & 1 & 0 & 1 & 0 & 0 \\ \mathrm{C}_2 & 1 & 1 & 1 & 1 & 0 \\ \mathrm{C}_3 & 0 & 1 & 0 & 0 & 1 \end{array}$$

and the new average pairwise similarities are

$$\begin{pmatrix} 1.0 & & \\ 0.6 & 1.0 & \\ 0.4 & 0.2 & 1.0 \end{pmatrix}.$$

Again, we see that clusters C_1 and C_2 can be merged. Consequently we get the final clustering

$$\begin{array}{c|ccccc}
\mathrm{C}_1 & 1 & 1 & 0 & 0 & 0 \\
\mathrm{C}_1 & 1 & 0 & 1 & 0 & 0 \\
\mathrm{C}_1 & 1 & 1 & 1 & 1 & 0 \\
\mathrm{C}_2 & 0 & 1 & 0 & 0 & 1
\end{array}$$

since the new pairwise similarities

$$\begin{pmatrix} 1.0 & \\ 0.33 & 1.0 \end{pmatrix}$$

do not lead to any new merges.

It is worth noticing that the Agglomerative Algorithm has at least one advantage in addition to the simplicity of implementation: it does not need the number of clusters as an input parameter. This follows from the fact that the algorithm stops its iterations when there are no more clusters similarity of which exceeds the given similarity threshold.

Information-Theoretic Co-clustering

In the LSA framework, the object was to yield improved document clustering by grouping the words in a conceptual manner. This means that if the words are semantically similar, they should be treated similarly as to the document clustering by the clustering algorithm. In [19], this idea for document clustering was presented in an information-theoretic framework in the context of *co-clustering*, i.e. the simultaneous clustering of documents and words.

In the following notation, we use x and y to denote rows and columns, respectively. Let's assume random variables X and Y which take values in the sets $\{x_1, x_2, ..., x_{|D|}\}$ and $\{y_1, y_2, ..., y_{|V|}\}$ where $|D|$ and $|V|$ are the number of documents in corpus and words in vocabulary. Their joint distribution $p(x, y)$ is a $|D| \times |V|$ matrix which can be empirically derived by normalizing the sum of the document-word co-occurrence matrix $W \in \mathbf{R}^{|D| \times |V|}$ to 1. The clusterings are denoted as

$$C_x : \{x_1\,,\, x_2\,,\, \ldots\,,\, x_{|D|}\} \;\rightarrow\; \{\hat{x}_1\,,\, \hat{x}_2\,,\, \ldots\,,\, \hat{x}_K\}$$

and

$$C_y : \{y_1\,,\, y_2\,,\, \ldots\,,\, y_{|V|}\} \;\rightarrow\; \{\hat{y}_1\,,\, \hat{y}_2\,,\, \ldots\,,\, \hat{y}_L\}$$

where $\{x_1\,, x_2\,, \ldots\,, x_{|D|}\}$ denote rows, $\{y_1\,, y_2\,, \ldots\,, y_{|V|}\}$ columns, $\{\hat{x}_1\,, \hat{x}_2\,, \ldots\,, \hat{x}_K\}$ clustered rows, and $\{\hat{y}_1\,, \hat{y}_2\,, \ldots\,, \hat{y}_L\}$ clustered columns. K and L are the number of row and column clusters, respectively, given as input parameters. The co-clustering is the solution to the following optimization problem.

Find a co-clustering (C_x, C_y) which minimizes the objective function

$$I(X;Y) - I(\hat{X};\hat{Y}) \tag{3.12}$$

for given K and L.

In the previous, $I(X';Y')$ denotes the mutual information [13] between random variables X' and Y' defined as

$$I(X';Y') = \sum_{x'\in X'} \sum_{y'\in Y'} p(x',y') \log \frac{p(x',y')}{p(x')p(y')}. \tag{3.13}$$

Furthermore, it holds that

$$I(X;Y) - I(\hat{X};\hat{Y}) = D_{KL}(p(x,y)||q(x,y))$$

where $D_{KL}(.|.)$ is the Kullback-Leibler divergence [13]. Since the Kullback-Leibler divergence is always non-negative, minimizing $I(X;Y) - I(\hat{X};\hat{Y})$ is equivalent to finding the probability distribution which is most similar to $p(x,y)$. Distribution $q(x,y)$ has the expression

$$q(x,y) = q(\hat{x},\hat{y})q(x|\hat{x})q(y|\hat{y})\,, \qquad \text{where } x \in \hat{x}, y \in \hat{y} \tag{3.14}$$

The core of the algorithm lies in this equality and its full derivation can be found in [19]. Furthermore, in [19], they present a local search algorithm which finds $q(x,y)$ by monotonically decreasing the value of the objective function (3.12).

The sensiblity of the solution $(C_x.C_y)$ obtained by minimizing equation (3.12) is illustrated by the following example used originally in [19]. Let's assume a $|D| \times |V|$ document-word matrix

$$\mathbf{W} = \begin{pmatrix} 5 & 5 & 5 & 0 & 0 & 0 \\ 5 & 5 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5 & 5 & 5 \\ 0 & 0 & 0 & 5 & 5 & 5 \\ 4 & 4 & 0 & 4 & 4 & 4 \\ 4 & 4 & 4 & 0 & 4 & 4 \end{pmatrix}$$

where $|D| = 6$ and $|V| = 6$. The empirical joint distribution is consequently

$$p(x,y) = \begin{pmatrix} 0.05 & 0.05 & 0.05 & 0 & 0 & 0 \\ 0.05 & 0.05 & 0.05 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.05 & 0.05 & 0.05 \\ 0 & 0 & 0 & 0.05 & 0.05 & 0.05 \\ 0.04 & 0.04 & 0 & 0.04 & 0.04 & 0.04 \\ 0.04 & 0.04 & 0.04 & 0 & 0.04 & 0.04 \end{pmatrix}.$$

By looking at $p(X,Y)$, it is easy to see that a smart clustering for rows and columns would be $\hat{x}_1 = \{x_1 , x_2 \}$, $\hat{x}_2 = \{x_3 , x_4 \}$, $\hat{x}_3 = \{x_5 , x_6 \}$, and $\hat{y}_1 = \{y_1 , y_2 \ , \ , y_3\}$, $\hat{y}_2 = \{y_4 , y_5 , y_6\}$, respectively. The resulting joint distribution is

$$q(\hat{x},\hat{y}) = \begin{pmatrix} 0.3 & 0 \\ 0 & 0.3 \\ 0.2 & 0.2 \end{pmatrix}.$$

Indeed, using this co-clustering, the loss in the mutual information in (3.12) is only 0.0957. Furthermore, it can be verified that no other clustering results in a lower loss.

Robust Clustering Algorithm for Categorical Attributes (ROCK)

Most clustering methods group documents based on the similarity between the objects themselves. However, it might be beneficial to define the similarity between two objects according to how similar their neighborhoods are to one another. This idea was utilized in a clustering method for categorical data in ROCK [20] where the clustering is based on *links* between data points.

As to clustering, a common daunting task is to separate two very close clusters. In these situtations, there can be two documents which, although being adjacent to each other, belong to different clusters. However, although being neighbors, it is presumable that these documents do not possess a large number of *common* neighbors. This intuition is utilized in the definition of *links* between two documents. Let's start by assuming a document-word co-occurrence matrix $W \in \mathbf{R}^{|D|\times|V|}$, where $|D|$ and $|V|$ are the number of documents in corpus and words in vocabulary, respectively. We define that two documents d_i and d_j with feature vectors $\mathbf{d}_i$ and $\mathbf{d}_j$ are neighbors if their similarity exceeds a given threshold value α. Similarity is defined as the Jaccard coefficient in equation 3.10. The number of links between two documents d_i and d_j, $links(d_i, d_j)$, is defined as the number of their common neighbors. Consequently, if $links(d_i, d_j)$ is large, d_i and d_j are likely to belong to the same cluster.

As can be easily deduced, a good clustering for a corpus D is such that, within the clusters, the documents share as many common neighbors as possible, meanwhile documents in different clusters share as little common neighbors as possible. In [20], this was formulated as an objective function to maximize for given number of clusters k.

$$O_{ROCK} = \sum_{i=1}^{k} |C_i| \times \sum_{d_m, d_n \in C_i} \frac{link(d_m, d_n)}{|C_i|^{1+2f(\phi)}} \tag{3.15}$$

where C_i is the ith cluster with size $|C_i|$ and $link(d_m, d_n)$ the links between the documents d_m and d_n. $|C_i|^{1+2f(\phi)}$ is an estimate for the total number of links in cluster C_i where $f(\phi)$ is a function dependent on data set, e.g. $f(\phi) = \frac{1-\phi}{1+\phi}$, ϕ being a constant chosen based on the data set. Additionally, a *goodness measure* $g(C_i, C_j)$ is defined to describe how beneficial it is to merge two clusters C_i and C_j into one.

$$g(C_i, C_j) = \frac{link(C_i, C_j)}{(|C_i| + |C_j|)^{1+2f(\phi)} - |C_i|^{1+2f(\phi)} - |C_j|^{1+2f(\phi)}} \tag{3.16}$$

where $|C_i|$ and $|C_j|$ are the cluster sizes and $f(\phi)$ the predetermined function dependent on the data set. The number of links between two clusters C_i and C_j, $link(C_i, C_j)$, is defined as $\sum_{d_m \in C_i, d_n \in C_j} link(d_m, d_n)$.

In [20], a local maximum for the objective function O_{ROCK} in (3.15) is obtained as follows. First, each document is assigned to its own singleton cluster. Then, at each iteration step, the pair of clusters with the highest goodness measure in equation (3.16) are merged together. The merging of clusters stops and the latest clustering is given as the output when the given number of clusters k is reached .

3.2 Topical clustering of corpus using self organizing map

To obtain the topical clustering of the background corpus, we use the procedure described in [22] which is depicted in Figure 3.1. Successful implementation of the scheme particularly in the area of topic adaptation was presented in [9, 23]. The procedure comprises of extracting features of the textual data, dimension reduction and finally clustering using self organizing map.

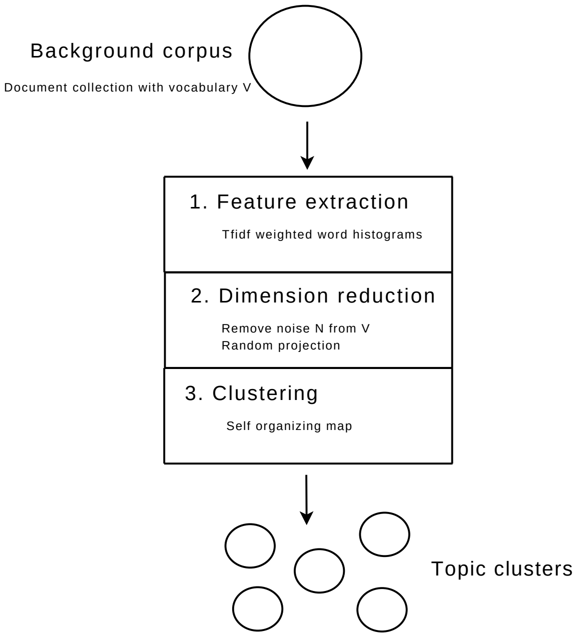


Figure 3.1: Scheme for topical clustering of background corpus.

We start with a document-word co-occurrence matrix $W \in \mathbf{R}^{|D|\times|V|}$, where $|D|$ and $|V|$ are the number of documents and words in vocabulary, respectively. To weight words according to their topical importance, we use tfidf weighting presented in equation (3.1).

As a result of the vector-space model we have obtained feature vectors for the documents. However, the data is in its current form of excessively high dimension. Therefore, we first divide the vocabulary in two sections, subsets

F and N, i.e.

$$V = F \cup N. \tag{3.17}$$

Subset F includes all words that have appeared in the corpus for more than Θ (e.g. $\Theta = 100$) times and subset N correspondingly the rest of the vocabulary. Intuition here is that words in N have appeared such a small number of times that they do not contain substantial discriminative power as to clustering and are therefore considered noise. When we use words in F as features for the documents, we have already reduced the dimensionality drastically. In order to reduce the dimensionality further to a practical magnitude, we utilize the random projection technique [24, 25]. In random projection we obtain a new vector $\mathbf{x}_i \in \mathbf{R}^m$ for each data vector $\mathbf{y}_i \in \mathbf{R}^n$, $m \ll n$, using equation

$$\mathbf{x}_i = R\mathbf{y}_i \tag{3.18}$$

where the columns of $R \in \mathbf{R}^{m \times n}$ are normally distributed orthogonal vectors of unit length. Implemented in a document clustering task in [25], random projection was shown to result in a prominent reduction in computational load while causing only minor loss in discrimination power of the data.

Let's define similarity between documents d_i and d_j in the vector space as

$$\text{sim}(\mathbf{y}_i, \mathbf{y}_j) = -\sqrt{(\mathbf{y_i} - \mathbf{y_j})^T(\mathbf{y_i} - \mathbf{y_j})} \tag{3.19}$$

i.e. the negatively signed euclidean distance between the document feature vectors $\mathbf{y}_i, \mathbf{y}_j \in \mathbf{R}^m$. Assuming the bag-of-words approach, identical documents have similarity value of zero and dissimilar documents values below zero.

Finally, the document collection is clustered using a self-organizing map (SOM) [26, 15]. SOM is a neural network used in an unsupervised manner to represent high dimensional data in low dimensional space. A map consists of nodes which have weight vector representations in the high dimensional data space and low dimensional vector representations on the map lattice. In training, at iteration τ, we find the node with the most similar weight $\mathbf{w}_{BMU}(\tau)$ to input data vector $\mathbf{x}(\tau)$ in data space. The most similar node is referred to as the best matching unit (BMU). Weight vectors $\mathbf{w}_k$ for all nodes k are updated according to

$$\mathbf{w}_k(\tau + 1) = \mathbf{w}_k(\tau) + \alpha(\tau)\gamma(\tau, \delta_{BMU_\tau,k})(\mathbf{x}(\tau) - \mathbf{w}_k(\tau)) \tag{3.20}$$

where $\alpha(t)$ is a monotonically decreasing function and $\gamma(\tau, \delta_{BMU_\tau,k})$ is the neighborhood function dependent on the distance δ between node k and

BMU in the low dimensional map space. In consequence of δ being defined in the low dimensional map space instead of the high dimensional data space, the map captures the topological structure of the data in the sense that proximity on the map indicates similarity between map nodes in the data space. The clustering of the background corpus is then obtained by assigning documents to their BMUs. The result is hard clustered data, i.e. every data point is assigned to exactly one node. Furthermore, we note that from here on, each node k corresponds to one topic t_k, i.e. the topics are defined using SOM.

Additonally, it is important to be aware of the pitfalls of the topic clusters obtained as described above. First, the corpus used for training may be overly homogenous in that it does not contain sufficiently clear topical structure to begin with. Second, assuming the underlying topic structure exists, we are obliged to decide the number of the latent topics rather randomly. Third, after obtaining the clustering, we lack the means of analyzing the result properly. A common procedure is to view the clusters manually to see if they consist of topically similar documents and words. However, due to the massive number of documents involved in the training, this overview is bound to be superficial with no well-founded means to validate the result.

3.3 Topic retrieval

At the topic retrieval stage of the adaptation scheme, we approach the broader field of information retrieval (IR). For instance, see [27] for a quick overview on the subject. In information retrieval, we search a database for documents given an information query from the user. Similarly, in topic retrieval, we are interested in finding the underlying topic at hand given a topic cue.

3.3.1 Definitions

Let's first recap the definition of the term *topic*. Assuming a background corpus B divided to K subsets through topical clustering, topic t_k corresponds to the subset k. This means that the topics are defined as SOM nodes. Furthermore, *the retrieval result* $\hat{t}$ (also referred to as *the topic estimate*) is considered correct if the subsequent language model adaptation is considered successful. Adaptation is considered successful if, given a test corpus, the perplexity score given by the adapted model is lower than the perplexity score given by the unadapted baseline model. The perplexity measurements is further discussed in section 3.5. A graphical interpretation of a successful and failed topic retrieval is presented in Figure 3.2.

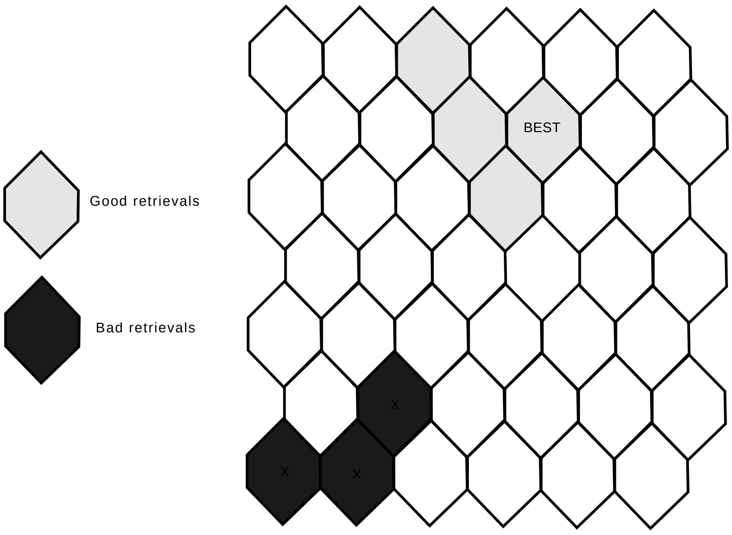


Figure 3.2: A graphical interpretation of topic retrieval with a SOM. Since the map is topologically organized, proximity of nodes indicate similar topics among them. Our topic retrieval is likely to be successful if it is close to the best possible topic estimate (BEST) on the map. The BEST retrieval is acquired executing the retrieval using a full document.

3.3.2 Topic cues as queries

Let's consider a background corpus with vocabulary V and topics $t_k, k \in 1, .., K$. Each topic t_k has been assigned with a subvocabulary $V_k \subset V$ in the topical clustering scheme, i.e. V_k comprises the words found in the documents of topic k. Let's define the topic cue as a keyword list $q = \{w_l^{(q)} : l \in 1, .., L\}$, $w_l \in V$, sampled from the underlying topic $t \in \{t_k\}$. Now, q operates as a query for the topics t_k. Naturally, the gained benefit of topic adaptation relies heavily on the success of this query. By weighting the word distributions towards an ill-retrieved topic, we are likely to degrade our recognition performance.

Let's divide the words in a keyword list q in three groups.

1. Topical words coherent with underlying topic. The relative frequency of these words is higher in documents of this topic than in the rest of the corpus.

2. Topical words incoherent with underlying topic (topical outliers). The relative frequency of these words in rest of the corpus is higher than in documents of this topic.

3. Topically neutral words. These words are found uniformly everywhere in the corpus.

Words belonging to groups 1 and 2 determine the estimate $\hat{t}$ for underlying topic for q. A uniformly sampled set of words from document follows the same distribution as the original document. However, as to the retrievals, the topical importance of words within the groups 1 and 2 differ. Therefore, two (small-sized) samples with the same size may lead to very different kinds of retrieval results. Consequently, the effect of varying the size of query L on the retrieval success is difficult to predict.

Focusing further on small-sized ques, queries of identical size L may have varying temporal dependecies among the words. First, the query may consist of a sentence-like word segment where all the words are dependent on the preceding word history. This is a valid assumption if the cue is provided, for example, as speech by a human user. Second, the query may comprise a set of keywords, in which case it is convenient to assume the word set to be temporally independent. This is the expected case with other modalities. In combination with the topic retrieval criteria presented in section 3.3.3, the varying dependencies within topic cue words may have an impact on the topic retrieval performance.

3.3.3 Topic retrieval criteria

Let's then discuss three decision criteria in retrieval of the best topic estimate $\hat{t}$ for the query q. The overall retrieval schemes are depicted in Figure 3.3. The criteria, from now on referred to as *retrieval criteria 1, 2* and *3*, are

1. Similarity of q and cluster k using words in vocabulary subset F as query.

2. Similarity of q and cluster k using words in vocabulary V as query.

3. Likelihood of topic language model p_k generating observed word probability distribution p_q .

In the first approach, we treat q as a document and find the most similar topic cluster. Similarity between a document and a topic cluster is the negatively signed euclidean distance between the feature and the weight vectors of the document and the cluster, respectively. Furthermore, as discussed in section 3.2, the vocabulary of the background corpus is divided into two subsets, F and N, before the clustering. Therefore, only the words in F are used in calculating feature vectors and affect the resulting partitioning. In consequence, it is justified to consider words in N unreliable keywords and filter them out from the queries before extracting features and executing retrievals. In brief, we divide q in two sections, $q = \{q_F, q_N\}$, where $q_F \subset F$ and $q_N \subset N$. Therefore, given q, retrieval criterion 1 is

$$\hat{t} = \arg\max_{t} \ \mathrm{sim}(\mathbf{q}_F, \mathbf{w}_t) \tag{3.21}$$

where $\mathbf{q_F}$ is the feature vector corresponding to q_F and $\mathbf{w}_t$ the feature vector corresponding to the topic cluster t.

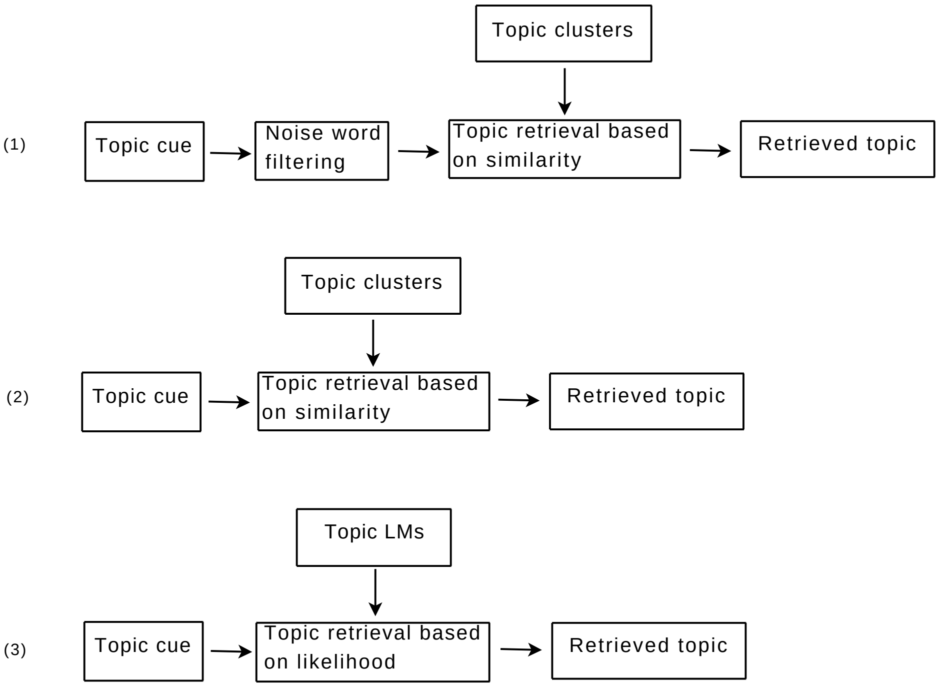


Figure 3.3: Topic retrieval schemes using topic retrieval criteria 1, 2 and 3.

In the above approach, we regarded N as noise. However, the words in

N are highly correlated with the words in F. This means that subsets of words in N which occur coherently with subsets of F are clustered correspondingly during the topic partitioning, and consequently make them potentially valuable keywords. Therefore, in our second approach, we restore N back to our keyword vocabulary after clustering the background corpus with F. This is done by replacing each cluster j with a pseudo document. The pseudo documents are obtained by adding together the word counts of documents belonging to j and calculating tfidf weights for the resulting document-word matrix. Given q, retrieval criterion 2 is

$$\hat{t} = \arg\max_{t} \ \mathrm{sim}(\mathbf{q}, \mathbf{w_t}) \tag{3.22}$$

where $\mathbf{q}$ is the feature vector corresponding to q and $\mathbf{w}_t$ the feature vector corresponding to the topic cluster t.

The two criteria above retrieved the topic utilizing similarity between the query and the clusters. In the third approach, we train topic language models (see section 3.4) with the topic clusters and retrieve the model that has most likely generated the query. The likelihood of topical language model t generating q is

$$l_t(q) = \prod_i p_t(w_i|w_{i-n+1}^{i-1})\,, \qquad w_i \in \{q\}. \tag{3.23}$$

However, it is common practice to make independence assumption between the words, i.e. $p_j(w_i|w_{i-n+1}^{i-1}) = p_j(w_i)$, as q is considered a keyword list. Thus, retrieval criterion 3 in log-likelihood form is

$$\hat{t} = \arg\max_{t} \sum_i \log p_t(w_i)\,, \qquad w_i \in \{q\}\,, \qquad q \subset V. \tag{3.24}$$

Notice that, as in approach 2, we use the whole vocabulary as keywords.

By comparing approach 1 with 2 and 3, we can find out if expanding keyword vocabulary from F to V has an impact on the retrieval performance. As discussed above, with approach 1, we can be confident that our queries are executed with an optimal set of keywords. However, this optimality is gained at the cost of throwing away a major part of our vocabulary obtained from the background corpus. As for approaches 2 and 3, we can highlight the fact that both schemes are essentially about smoothing word distributions within topics. The difference between the two is that in the likelihood-based approach we divide the probability mass among the whole vocabulary whereas in the similarity-based approach the non-zero tfidf scores are localized on observed words only. Therefore, with approaches 2 and 3, we can essentially compare the difference in retrieval using grained and heavily smoothed word distributions within topics. Additionally, it should be noted that approaches 1 and 2 use the bag-of-words representation to queries whereas approach 3 preserves the word order information. This makes a notional difference for the approaches but is, in practice, irrelevant because of the independence assumption made in equation (3.24).

3.4 Language model training

N-gram models are simply trained from a text corpus by counting word sequence occurrences and calculating maximum likelihood estimates based on them as follows:

$$p(w_i|w_{i-n}^{i-1}) = \frac{c(w_{i-n}^{i})}{c(w_{i-n}^{i-1})} \tag{3.25}$$

where $c(.)$ denotes the number of instances found in the training data, w_{i-n}^{i} a sequence of n words, and w_{i-n+1}^{i-1} the word sequence preceding word w_i. However, let's consider a vocabulary V of size $|V|$. Now, the number of all the possible n-grams is $|V|^n$. As the size of the vocabulary grows, it rapidly becomes impossible to acquire enough training data to estimate the n-grams reliably, and majority of the n-grams will not be seen in the data. This phenomenom is called the sparse data problem and to counter-act it, we use a technique called smoothing.

In smoothing, we essentially move probability mass away from the seen events to unseen events, therefore making the probability distribution more uniform, i.e. smoother. Moreover, as to retrieval criterion 3, it is important to note that this smoothing results in a distribution where *all* the words in the model vocabulary have a non-zero probability to be observed. Ample methods related for smoothing exist, e.g. Good-Turing- [28] and Witten-Bell- [29] discounting, and Kneser-Ney- [30] and Katz-smoothing [31]. A commonly used and, as verified with empirical tests in [32], the best performing technique, is the Kneser-Ney backing-off method [30]. In this work, all the models in the experiments are smoothed using the Kneser-Ney technique. We describe the method in the following.

The smoothing includes three basic steps; discounting word observations, establishing the low order n-grams, and combining the n-grams of various orders. With discounting the probability mass is shifted from the seen events to unseen events by subtracting observations from seen events. Furthermore, combinations of n-grams can, in general, be done in two ways (e.g. [32]), by backing-off or interpolation. Backing-off methods are defined by the recursive equation

$$p_{smooth}(w_i|w_{i-n+1}^{i-1}) = \begin{cases} p(w_i|w_{i-n+1}^{i-1}) & \text{if } c\,(w_{i-n+1}^{i}) > 0 \\ \gamma(w_{i-n+1}^{i-1})p_{smooth}(w_i|w_{i-n+2}^{i-1}) & \text{if } c\,(w_{i-n+1}^{i}) = 0 \end{cases} \tag{3.26}$$

where $p(w_i|w_{i-n+1}^{i-1})$ is the n-gram estimated from the discounted counts and $\gamma(w_{i-n+1}^{i-1})$ is coefficient set to make the probabilities sum up to one. The idea behing the backing off is that if the word sequence w_{i-n+1}^{i} is seen in the data, we use $p(w_i|w_{i-n+1}^{i-1})$. Otherwise we back off to a lower order n-gram $p_{smooth}(w_i|w_{i-n+2}^{i-1})$. The recursion tends that we keep on backing off until we have an observation.

Interpolation methods are defined as a linear mixture

$$p_{smooth}(w_i|w_{i-n+1}^{i-1}) = \lambda_0 p(w_i) + \sum_{j=1}^{n-1} \lambda_j p(w_i|w_{i-j}^{i-1}) \tag{3.27}$$

where $p(.|.)$ are the probabilities estimated from the discounted counts and λ_j, $\sum_j \lambda_j = 1$, the mixture coefficients. In brief, in the interpolation approach, the high order n-grams ending with the word w_i are mixtures of lower order n-grams ending with w_i.

Kneser-Ney method is of the previous, backing off, approach using *absolute discounting*, where we subtract a fixed count Δ from all the each non-zero counts. The model is defined as

$$P_{KN}(w_i|w_{i-n+1}^{i-1}) = = \begin{cases} \frac{max\{c\,(w_{i-n+1}^{i})-\Delta,0\}}{\sum_{w_i} c\,(w_{i-n+1}^{i})} & \text{if } c\,(w_{i-n+1}^{i}) > 0 \\ \gamma(w_{i-n+1}^{i-1})p_{KN}(w_i|w_{i-n+2}^{i-1}) & \text{if } c\,(w_{i-n+1}^{i}) = 0 \end{cases} \tag{3.28}$$

where $c\,(w_{i-n+1}^{i})$ is the number of times word sequence w_{i-n+1}^{i} has been seen in the data, $\gamma(w_{i-n+1}^{i-1})$ coefficient set to make the probabilities sum up to one, and Δ a constant estimated in [30] from the data as

$$\Delta = \frac{c_1}{c_1 + c_2} \tag{3.29}$$

where c_1 and c_2 are the number of n-grams with exactly one or two counts, respectively. The lower order distribution is chosen so that the marginals of the smoothed higher order distribution (left side) match the lower order marginals of the training data (right side):

$$\sum_{w_{i-n+1} \in V} p_{KN}(w_i | w_{i-n+1}^{i-1}) \;=\; \frac{c(w_{i-n+2}^{i-1})}{\sum_{w_i} c(w_{i-n+2}^{i})} \tag{3.30}$$

where V denotes the vocabulary, and $c(.)$ again the count for word sequence. In [30] it is shown that the probabilities $p_{KN}(w_i|w_{i-n+2}^{i-1})$ in equation (3.28) are of the form

$$p_{KN}(w_i|w_{i-n+2}^{i-1}) \;=\; \frac{N_1 + \hat{w}_{i-n+2}^{i}}{N_1 + \hat{\hat{w}}_{i-n+2}^{i-1}} \tag{3.31}$$

where

$$N_1 + \hat{w}_{i-n+2}^{i} \;=\; |\{w_{i-n+1} : c\,(w_{i-n+1}^{i}) > 0\}| \tag{3.32}$$

and

$$N_1 + \hat{\hat{w}}_{i-n+2}^{i-1} \;=\; |\{(w_{i-n+1}, w_i) : c\,(w_{i-n+1}^{i}) > 0\}| \;=\; \sum_{w_i} \left(N_1 + \hat{\hat{w}}_{i-n+2}^{i}\right). \tag{3.33}$$

These formulas mean that the low-order n-grams are highly affected by the number of contexts they follow. This approach distinguishes the Kneser-Ney method from the other smoothing techniques which usually rely on occurrence frequencies when calculating the low-order n-grams.

In addition to smoothing the n-gram distributions, the background vocabulary is extensively large to be used as such. Therefore, it is necessary to determine a fixed size for the active language model vocabulary and train the language model using only these most frequently occurred words. A practical size of language model for large vocabulary speech recognition for English is approximately from 20k to 60k.

3.5 Language model evaluation

The most popular evaluation measure for n-gram language models is perplexity [13] defined as

$$ppl(p_j, w_1^M) \;=\; \sqrt[M]{\prod_{i=1}^{M} \frac{1}{p_j(w_i|w_{i-n+1}^{i-1})}} \tag{3.34}$$

where M is the number of words in the test data sequence and p_j the n-gram probabilities of the tested model. Perplexity measures how well the model predicts the given test data, larger values indicating worse performance.

As can be seen in equation (3.34), the perplexity measure corresponds to the inverse of the geometrical mean of the likelihood. Therefore, as to topic retrieval criterion in equation (3.24), topic LM maximizing the likelihood of a query is equivalent to LM minimizing the perplexity. Moreover, we can cover yet a third common measure for probability distribution similarity, namely Kullback-Leibler divergence [13] defined as

$$D_{KL}(q||p) = \sum_{x_i} q(x_i) \log \frac{q(x_i)}{p(x_i)} \tag{3.35}$$

for discrete probability distributions q and p. Returning to equation (3.24), we wish to find the topical language model $p_t(w_i)$ that minimizes the Kullback-Leibler divergence with word probability distribution $p_q(w_i)$ corresponding to a given query q. Here, $p_q(w_i)$ can be estimated from the keyword list, for example, simply using maximum likelihood estimate. As can easily be seen

$$\begin{aligned} \arg\min_t D_{KL}(p_q||p_t) &= \arg\min_t \sum_{w_i} p_q(w_i) \log \frac{p_q(w_i)}{p_t(w_i)} \\ &= \arg\min_t \sum_{w_i} \log \frac{1}{p_t(w_i)} \\ &= \arg\max_t \sum_{w_i} \log p_t(w_i) \end{aligned}$$

that is, $p_t(w_i)$ minimizing the Kullback-Leibler divergence is again, assuming uniform probabilities among the words in q, equal to $p_t(w_i)$ maximizing the likelihood of q .

Additionally, let's take note of two properties of the perplexity measure. First, as the vocabularies obtained from the training data are of limited sizes, our models naturally can not comprise all the possible words or word forms. Therefore, in previously unseen data, our models will always encounter so called out-of-vocabulary (OOV) words. For all the unseen words in training data, the maximum likelihood estimate gives a probability of zero. As seen in equation (3.34), $p(w_j|w_{j-n+1}^{j-1}) = 0$ for any given j is enough to make the perplexity go to infinity, which naturally makes the measure useless. This is taken into account by brutally skipping the unknown words. Second, decrease in perplexity rarely indicates significant reduction in recognition error since the overall performance of a speech recognition system depends on multiple different factors. In contrast however, increase in perplexity almost surely indicates either no improvement or degration in the recognition error.

3.6 Language model combining methods

Language model combination methods have been a subject of extensive research during the past two decades. A broad outlook at different combining methods can be found, for example, in [33]. In the following, we divide the techniques into two categories, the interpolation methods and the non-interpolation methods. We describe a few examples of both classes, namely mixture models [34]), cache models [35], trigger models [36, 37], Mimimum Discrimination Information (MDI) [38, 39], and exponential models [40].

Interpolation methods

Interpolation techniques rely essentially on mixture model approach where the adapted model is a weighted linear combination of the component models, that is

$$p_{adapted}(w_i|w_{i-n+1}^{i-1}) = \sum_k \lambda_k \, p_k(w_i|w_{i-n+1}^{i-1}) \, . \tag{3.36}$$

The λ_k, $\sum_k \lambda_k = 1$, are the mixture coefficients commonly optimized using a hand-held data set if available (see e.g. [34]). Mixture models are often a good combination technique choice for their robustness and small computational costs.

In this work, we use two types of mixture models. First type comprises of the background model and one topic model. Second type comprises of the background model and a neighborhood of topic models. As we are simulating an adaptation scenario with little adaptation data, we use static mixture coefficients. The neighborhood is conveniently determined by the nearby topic nodes on the SOM lattice.

As a historical remark, one of the first attempts to combine topic information with the background model was the use of cache models [35]. In cache models this is done by simply elevating the probabilities of words seen in the recognition history. Models adapted using caches are of the form

$$p_{adapted}(w_i|w_{i-n+1}^{i-1}) = \lambda \, p_{backgr}(w_i|w_{i-n+1}^{i-1}) + (1-\lambda) \, p_{cache}(w_i|w_{i-n+1}^{i-1}) \quad (3.37)$$

where $p_{backgr}(.)$ is a static model trained with the training corpus. The cache model $p_{cache}(.)$ is dynamically adapted with the cumulating word history so that the probability of a seen word is increased. The resulting model $p_{adapted}(.)$ is a linear combination of the two, where λ is again the mixing coefficient (optimized with held-out data if available). Effectiveness of the method, despite its obvious simplicity, is due to the fact that words tend to appear repeatedly within a topic course. Therefore, cache models are a fine example of a simple model which nonetheless takes efficiently into account this one underlying property of language.

In trigger-based language models [36, 37], the idea of caches were expanded so that an encountered word w_k will temporarily raise the probability of another word w_l. These trigger pairs are defined beforehand with a training corpus. However, selecting such trigger pairs from the training data is in practise a difficult task, greatly due to the following reasoning. As for common words, the trigger pairs can be estimated since there are enough occurrences to make the correlation deduction reliable. Yet, in case of these common words and events, the data is dense enough for the baseline n-grams to work at adequate precision. Respectively, the rare words we have most difficulties predicting with the baseline n-gram form also the trigger pairs that can not be reliably selected from the data. With trigger models the most significant improvents in recognition results are gained with highly topic focused data, as in [41], where they were used in automatic meeting transcription task.

Non-interpolation methods

As a first example of the non-interpolation techniques we discuss the marginal adaptation methods. Marginal adaptation methods are based on extracting low order distributions, referred to as constraints, from the topic dependent data. Subsequently, the background model is adapted so that its marginals agree with the constraints. As to model performance, this approach often seems to be preferable to interpolation methods (stated e.g. in [42]). However, the possible improvement is gained at the cost of higher computational load and, naturally, no guarantees for improvement can be provided.

The following example method referred to as Minimum Discrimination Information (MDI) [38, 39] summarizes well the idea behind the marginal adaptation. The constraints are now extracted from the retrieved topic clusters as unigram distributions $\hat{P}_A$. The joint distribution of the adapted model is the P_A which minimizes the Kullback-Leibler divergence [13] with the background distribution P_B while satisfying the constraints. Unigrams are beneficial in that they can be estimated reliably from the small-sized topic clusters. Moreover, the adapted model distribution conditioned on word history reduces (see [39] for full derivation) to form

$$P_A(w_i|w_{i-n+1}^{i-1}) = \frac{P_B(w_i|w_{i-n+1}^{i-1})\,\alpha(w_i)}{\sum_{\hat{w}_i \in V} P_B(\hat{w}_i|\hat{w}_{i-n+1}^{i-1})\,\alpha(\hat{w})} \tag{3.38}$$

where

$$\alpha(w) = \frac{\hat{P}_A(w)}{P_B(w)}. \tag{3.39}$$

Therefore, the adapted model is simply the background model multiplied with a scaling factor α. Additionally, as $\hat{P}_A(w) = 0$ or $P_B(w) = 0$ for any w would result in a zero probability for w in the adapted model, it is sensible to smooth both the background model and constraint unigrams with Kneser-Ney before combining.

Second example of the non-interpolation techniques is the exponential models [40]. Exponential models are of the form

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{1}{Z(w_{i-n+1}^{i-1})} exp\left(\sum_i f_i(w_{i-n+1}^{i})\mu_i\right) p_0(w_i|w_{i-n+1}^{i-1}) \tag{3.40}$$

where

$$Z(w_{i-n+1}^{i-1}) = \sum_{w_i} exp\left(\sum_i f_i(w_{i-n+1}^{i})\mu_i\right) p_0(w_i|w_{i-n+1}^{i-1})$$

is a normalization term, $p_0(w_i|w_{i-n+1}^{i-1})$ the *prior* probability, $f_i(w_{i-n+1}^{i})$ the *features* of the model, and μ_i the parameter associated with f_i.

The idea behind the exponential model is illustrated with the following example. First, the prior distribution p_0 corresponds to a general n-gram model (e.g. trigram) trained with the background corpus. Subsequently, topical information can be introduced to the model through the features f_i. Let's say we want to raise the probability of word w_i *mice* when its 2 preceding words w_{i-2}^{i-1} equals the word sequence *cat eats*. This can be done using a feature

$$f_1(w_{i-2}^{i}) = \begin{cases} 1 & \text{if } w_{i-2}^{i} = \{\mathit{cat\ eats\ mice}\} \\ 0 & \text{otherwise} \end{cases}$$

and by setting μ_1 so that the term e^{μ_1} equals how many times more probable the word *mice* becomes. The effect of this feature is that the probability of the word *mice* in prior p_0 is increased in the presence of context *cat eats* while the probabilities of other words are decreased due to the normalization term $Z(w_{i-n+1}^{i-1})$. On the other hand, in the absence of context *cat eats*, the prior distribution p_0 is left intact.

The problem with the exponential model is the large computational cost of obtaining the normalization terms $Z(w_{i-n+1}^{i-1})$. Consequently, an unnormalized version of the method was introduced in [40] where the probabilities in the model are replaced with scores. To keep the scores from exceeding value 1, the conditional probability $p(w_i|w_{i-n+1}^{i-1})$ in (3.40) is formulated as

$$p(w_i|w_{i-n+1}^{i-1}) = \frac{p_{aux}w_i|w_{i-n+1}^{i-1}}{1 + p_{aux}w_i|w_{i-n+1}^{i-1}} \tag{3.41}$$

where

$$p_{aux}(w_i|w_{i-n+1}^{i-1}) = exp\left(\sum_i f_i(w_{i-n+1}^{i})\mu_i\right)\frac{p_0(w_i|w_{i-n+1}^{i-1})}{1 - p_0(w_i|w_{i-n+1}^{i-1})}.$$

The term $\frac{p_0(w_i|w_{i-n+1}^{i-1})}{1-p_0(w_i|w_{i-n+1}^{i-1})}$ assures that $p(w_i|w_{i-n+1}^{i-1}) = p_0(w_i|w_{i-n+1}^{i-1})$ holds in the absence of features.

Chapter 4

Experiments

4.1 Data

The text data consists of news articles from the English Gigaword Corpus [43]. The Gigaword is an archive of newswire text data that has been acquired by the Linguistic Data Consortium (LDC) at the University of Pennsylvania. The corpus consists of articles provided by six distinct international new agencies. Of these, 101 000 documents published by Agence France-Presse, English Service during 1994-1997 and 2001-2002 were selected randomly. As a consequence to the selection of the documents, the expected topics found within the document corpus are roughly those associated with the daily news supply, such as finances, politics and sports.

The background corpus B used for training of language models (LM) consists of 100 000 documents. These LMs are used throughout the experiments. The average length of a document is 262 words. The background vocabulary V_B comprises 185641 words.

The experiments are divided into topic retrieval and speech recognition sections. The test set for topic retrieval experiments comprises the rest 1 000 news articles with an average length of 256 words. The number of out-of-vocabulary words per document in the test set compared to the background vocabulary is, on average, zero.

Data for speech recognition experiments is derived from Wall Street Journal database [44]. As we use acoustic models trained beforehand, we only obtain a test set for the speech recognition experiments. The test set consists of 166 sentences picked from 8 articles. Each of the articles, and consequently the sets of sentences, are of coherent topic spoken by an individual speaker. The articles consist of newswire data and, consequently, the topics found in the speech data are again those associated with the daily new supply.

4.2 System parameters

The background corpus B is clustered using the procedure described in section 3.2. The vocabulary V_B is divided into subsets F_B and N_B so that every word in F_B has appeared at least 50 times. Consequently, the subsets F_B and N_B are of sizes $|F_B| = 19$k and $|N_B| = 167$k. Additionally, 150 most frequently seen words are left out of V_B as topically neutral function words. The data dimension is reduced further from $|F_B|$ to $m = 500$ with random projection [25]. The trained self organizing map [15] consists of 42 nodes in a 7 × 6 hexagonal lattice. The U-matrix and cluster size histogram of the resulting map are presented in Figures 4.1 and 4.2, respectively. Additionally, Figure 4.3 shows ten highest tfidf-scoring words, i.e. the terms most characteristic to the topic, of each corner node of the map. The U-matrix and cluster size histogram in Figures 4.1 and 4.2 show that the documents have been spread evenly on the map whereas Figure 4.3 suggests that nodes far apart on the map indeed hold differing underlying topics. Therefore, the three figures confirm that the map has captured the topical structure of the background corpus successfully.

Language models (LM) are trained and combined as described in sections 3.4 and 3.6, respectively. The background LM is a bigram model based on the background corpus and topic LMs bigrams based on the topic clusters. The LM vocabulary consists of 60 000 words. All the models are trained using Kneser-Key backing off method [30]. Adapted models are mixture models of background LM and the retrieved topic LMs. We use two types of mixtures. First, using one cluster as the topic model and second, using a cluster neighborhood. The neighborhood is defined as the retrieved SOM node and nodes within the distance of one (see Figure 4.4). The mixture coefficients are 0.5 for the background model and 0.5 divided evenly for the topic models.

In summary, in the experiments we use three types of LMs which essentially differ in the use of node neighbourhood of the SOM map. The parameters of the LMs are presented in Table 4.1.

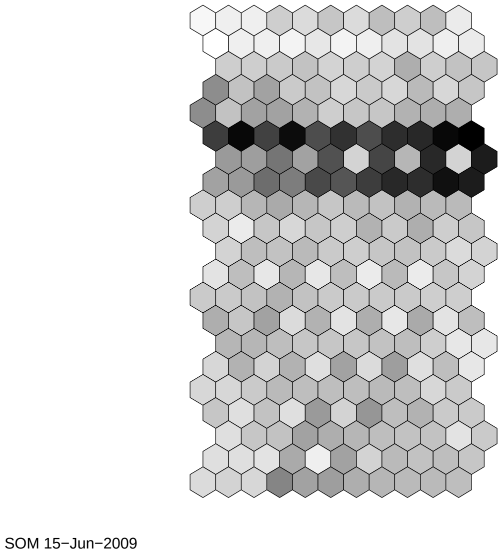


Figure 4.1: The U-matrix of the trained SOM. The nodes represent the distance in data space between adjacent nodes of the SOM. As the dark end of the color scale corresponds to large distances, the matrix is interpreted to show closely connected light areas separated by the dark areas.

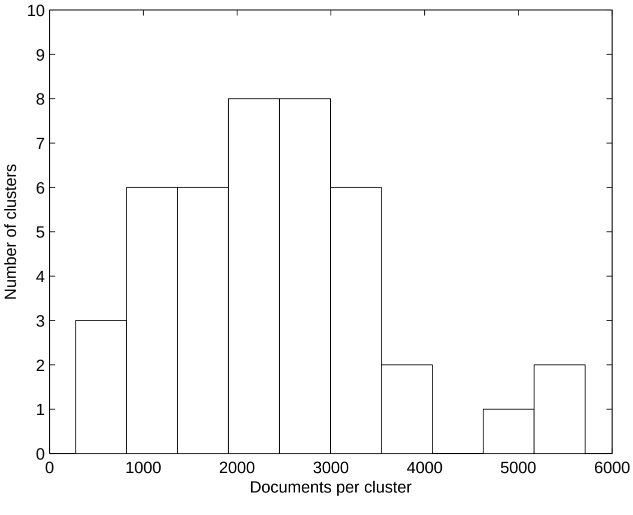
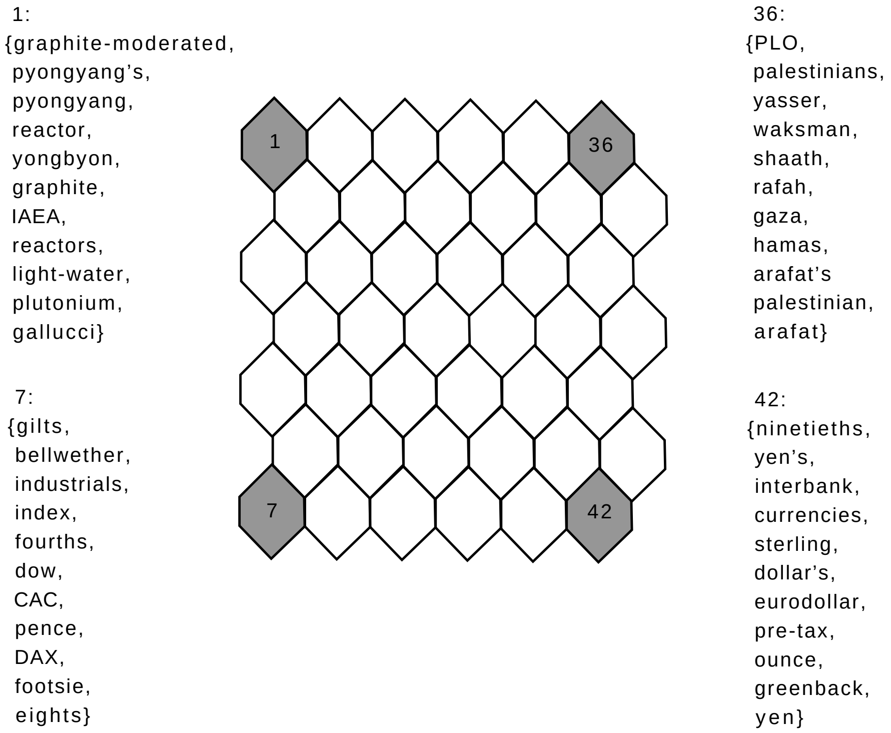


Figure 4.2: The histogram of cluster sizes as documents. The majority of the clusters are of sizes from 1000 to 4000 documents.



1:
{graphite-moderated,
pyongyang's,
pyongyang,
reactor,
yongbyon,
graphite,
IAEA,
reactors,
light-water,
plutonium,
gallucci}

7:
{gilts,
bellwether,
industrials,
index,
fourths,
dow,
CAC,
pence,
DAX,
footsie,
eights}

36:
{PLO,
palestinians,
yasser,
waksman,
shaath,
rafah,
gaza,
hamas,
arafat's
palestinian,
arafat}

42:
{ninetieths,
yen's,
interbank,
currencies,
sterling,
dollar's,
eurodollar,
pre-tax,
ounce,
greenback,
yen}

Figure 4.3: Ten highest tfidf-scoring words in clusters 1, 7, 36 and 42. The topics of nodes can be interpreted e.g. as *nuclear power*, *stock markets*, *palestinian politics* and *economy*, respectively.

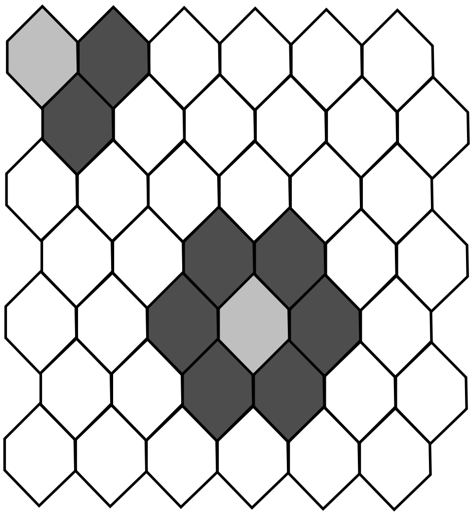


Figure 4.4: A SOM map in a 7×6 hexagonal lattice. Two 1-unit neighbourhoods of sizes 3 and 7 are presented with gray nodes.

Label	N-gram order	Vocabulary size	SOM node neighbourhood	Smoothing method
background	Bigram	60k	None	Kneser-Ney
background+1cluster	Bigram	60k	n = 1	Kneser-Ney
background+7clusters	Bigram	60k	n = 7	Kneser-Ney

Table 4.1: Language models

The speech recognition system used in the experiments has been developed in the laboratory of computer and information science in TKK. A thorough description of the system is given in [1]. The main properties of the acoustic model are presented in the following. Speech signal is sampled using 8 kHz sampling frequency and 16 bits. The signal is then represented with 12 MFCC (mel-frequency cepstral coefficients) and the log-energy along with their first and second differentials. Features are calculated in 16 ms windows with 8 ms overlap. Cepstral mean subtraction (CMS) and a maximum likelihood linear transformation, which is estimated in training, are applied to the features. For acoustic modeling we have state-clustered Hidden Markov triphone models constructed with a decision-tree method [45]. The model has 5062 states modeled with 32 Gaussians. State durations are modeled with gamma probability functions [46].

4.3 Experiment description

In chapter 1, we presented the following research questions.

1. Do small-sized topic cues enable a successful topic retrieval?
2. Is successful/failed topic retrieval significantly beneficial/harmful to the speech recognition performance?

Additionally, we examine if the source of the topic cue or the choice of the topic retrieval criterion have a significant effect on the topic retrieval.

We approach the first question by comparing the performances of LMs adapted using large and small-sized cues. To obtain these topic cues, each document in the test set (see section 4.1) is further processed to three different forms as follows.

1. Full document.

2. Word sequence of length n from a randomly selected location within document.

3. Randomly selected n words within document.

These three forms are used to simulate different topic cue scenarios. The full document contains all the data available for retrieving the topic. This retrieval result corresponds to the best estimate for underlying topic available. A word sequence of length n represents a topic cue obtained as a short segment of speech. Randomly selected n words represent a topic cue obtained from other modalities. In the experiments, n is set to 10. From now on, we refer to the topic cues 1,2, and 3 as *full cue*, *speech cue*, and *multimodal cue*, respectively. The difference between speech and multimodal cues lies in the expected strong correlation between the subsequent words in speech cue. Mainly, it is highly unlikely for speech cues not to contain any topic-specific words as for multimodal cues this is, in principle, possible. Additionally, it should be emphasized that the speech cue is merely a representation of a speech style text segment and differs fundamentally from a real speech transcription in that it does not include any recognition errors. An example of the three topic cue forms derived from a single document is presented in the following.

Full cue

- - - no decisions were taken at the one-and-a-half hour meeting bolger told reporters that among issues discussed were the possibility of taking legal action through the international court of justice support for a protest flotilla which is being planned to sail to mururoa the prospect of members of parliament taking part in the flotilla protest and action that could be taken by the south pacific forum bolger said it was not clear whether new zealand could mount a legal challenge but the party leaders had agreed the idea should be pursued - - -

Multimodal cue

told agreed meeting that the consideration would mururoa action leader

Speech cue

> bolger told reporters that among issues discussed were the possibility

The topic retrieval performance using the three types of topic cues is measured by LM perplexity scores using the Gigaword test set. In addition, we compare the results obtained using the three retrieval criteria:

1. Similarity of q and cluster k using words in vocabulary subset F as query.
2. Similarity of q and cluster k using words in vocabulary V as query.
3. Likelihood of topic language model p_k generating observed word probability distribution p_q .

If the topic retrieval can be done successfully, we should perceive significant improvement in the perplexity using adapted models compared to the unadapted baseline.

Second, we study the effect of successful and failed adaptation on speech recognition by adapting the background LM using properly retrieved and randomly selected topics, respectively. The results of these two scenarios are then compared with the unadapted baseline. The results are measured by perplexity scores, word error rates (WER) and term error rates (TER) [47] on the speech test set. Essentially, the WER and TER measures differ in that in WER we take into account all the words in transcriptions whereas in TER we stem the remaining words before calculating the error rate. The stemming equals to removing suffixes using the Porter algorithm [48]. In order to emphasize the effects of topic adaptation, in this work, we also remove non topical words, referred to as closed class words, from the transcription before executing the stemming. We define closed class words to include prepositions, determiners, conjunctions, and pronouns (see [49] for the word lists). The term order information is discarded and the transcriptions are treated as term histograms (bag-of-words approach). In short, TER is the fraction of differing term counts (tc) between the reference (ref) and recognized (rec) transcription histograms [47]:

$$TER = \frac{\sum_t \mid tc_{ref}(t) - tc_{rec}(t) \mid}{\sum_t tc_{ref}(t)}. \tag{4.1}$$

Additionally, we use word change rate (WCR) to show how many words, in percentages, differ between the transcription using background and adapted models. Again, we use the full articles to estimate the topic as well as possible. In consequence, adaptation using full cues results in 8 different adapted LMs. In addition, for each sentence, we pick randomly 10 words from the article corresponding to the sentence to be used as the small-sized topic cue. Therefore, adaptation using small-sized cues results in 166 different adapted LMs. In summary, we do the recognition for each of the 166 sentences using 1) no topic estimate, 2) topic retrieved with the whole article corresponding to the sentence, 3) topic retrieved with 10 words picked randomly from the article corresponding to the sentence, and 4) purely random topic estimate.

In the following, we use the Wilcoxon signed-rank test [50] to determine the statistical significance of the results. The null hypothesis is that the results are derived from identical distributions with equal medians. Therefore, rejection of the null hypothesis indicates that the difference in model performances is statistically significant. The null hypothesis is accurately rejected with confidence, i.e. probability, of α.

4.4 Results and analysis

The perplexity results for the adapted models using differing topic cues and the retrieval criteria are presented in Table 4.2. Since we are examining the effect of varying retrieval criteria and topic cue sources on the performance, the results are disected according to the applied LM.

The multimodal and speech topic cues result in equal adaptations with practically 100% confidence. This result holds for both background+1cluster and background+7clusters models. Moreover, we note that the adaptations using small-sized cues fall coherently between the full cues and random retrievals.

All pair-wise comparisons of the adapted models against the background model are presented in Table 4.3. For both LM types, background+1cluster and background+7clusters, the retrieval criteria performed in the following improving order: random, criterion 1, criterion 2 and criterion 3. The order was independent on the topic cue type. However, for the speech cue, the improvements were not statistically significant. For full topic cues, all the retrieval criteria (excluding random) and LM combinations resulted in a beneficially adapted models compared to the background model. With multimodal cues, the background+1cluster and background+7clusters LMs using the criterion 3 outperformed the background models with 99.69% and 99.19% confidence, respectively.

Language model	Retrieval criterion	Full	M.modal	Speech
Background	None	237	237	237
Background+1 cluster	Criterion 1	231	264	267
Background+1 cluster	Criterion 2	210	237	234
Background+1 cluster	Criterion 3	207	219	223
Background+1 cluster	Random	269	269	269
Background+7 clusters	Criterion 1	226	252	254
Background+7 clusters	Criterion 2	219	232	230
Background+7 clusters	Criterion 3	213	221	223
Background+7 clusters	Random	256	256	256

Table 4.2: Average perplexity results.

Compared models		Full	Multimodal	Speech
Backgr.	Backgr.+1cl+Crit.1	A	$\underline{B}$	$\underline{B}$
Backgr.	Backgr.+1cl+Crit.2	$\underline{A}$	A	A
Backgr.	Backgr.+1cl+Crit.3	$\underline{A}$	$\underline{A}$	A
Backgr.	Backgr.+1cl+Random	$\underline{B}$	$\underline{B}$	$\underline{B}$
Backgr.	Backgr.+7cls+Crit.1	A	$\underline{B}$	$\underline{B}$
Backgr.	Backgr.+7cls+Crit.2	$\underline{A}$	A	A
Backgr.	Backgr.+7cls+Crit.3	$\underline{A}$	$\underline{A}$	A
Backgr.	Backgr.+7cls+Random	$\underline{B}$	$\underline{B}$	$\underline{B}$

Table 4.3: Pair-wise comparisons of the unadapted background LM and the adapted LMs using different topic retrieval criteria and topic cues. Cell is marked with 'B' if the background model outperformed the adapted model and 'A' if the adapted model outperformed the background model. The letter is underlined if the result is statistically significant (with confidence of 99%).

Average recognition performances for backgound LM, LMs adapted using retrieval criterion 3, and LMs adapted using randomly retrieved topics are presented in Table 4.4. In the following, the confidences are determined on speaker-basis. As to perplexity scores, all the adapted LMs using topic retrieval criterion 3 outperfomed the unadapted background model with practically 100% confidence. In turn, the background model outperformed the LMs adapted using random topics with 100% confidence. As to WERs, the background model outperformed all the adapted models. However, the difference in WER between the background and any of the adapted LMs was not statistically significant. The column labeled WCR indicates the rate of words changed (in percentages) between the background and adapted transcriptions. As to TERs, the background model outperformed the background+1 cluster models with 100% confidence. The differences between background and background+7 clusters models were not statistically significant.

Language model	Ret. crit.	Topic cue	Perp	WER (%)	WCR (%)	TER (%)
Background	None	None	475	26.0	0	14.2
Background+1 cluster	Crit. 3	Full	456	26.2	20.4	16.0
Background+1 cluster	Crit. 3	Multim.	444	26.8	20.6	14.7
Background+1 cluster	Random	None	504	27.1	20.9	15.5
Background+7 clusters	Crit. 3	Full	432	26.7	20.8	16.7
Background+7 clusters	Crit. 3	Multim.	420	26.4	20.5	14.9
Background+7 clusters	Random	None	464	26.7	21.0	14.8

Table 4.4: Speech recognition results. The columns indicate used language model, retrieval criterion, type of topic cue, average perplexity score on transcription, word recognition error rate (WER), word change rate (WCR) between adapted and the background model, and term error rate (TER).

The topic retrieval histogram for speaker 1, 4 and 7, i.e. the values indicating the map nodes retrieved using different topic cues generated from the article corresponding to these speakers, is presented in Figure 4.5. The retrievals using small-sized cues are spread around the same general area at the maps as the retrievals executed using the full document cues.

4.5 Discussion

4.5.1 Topic retrieval

The perplexity results for multimodal and speech topic cues were identical with practically 100% confidence. This indicates that the cues are equally

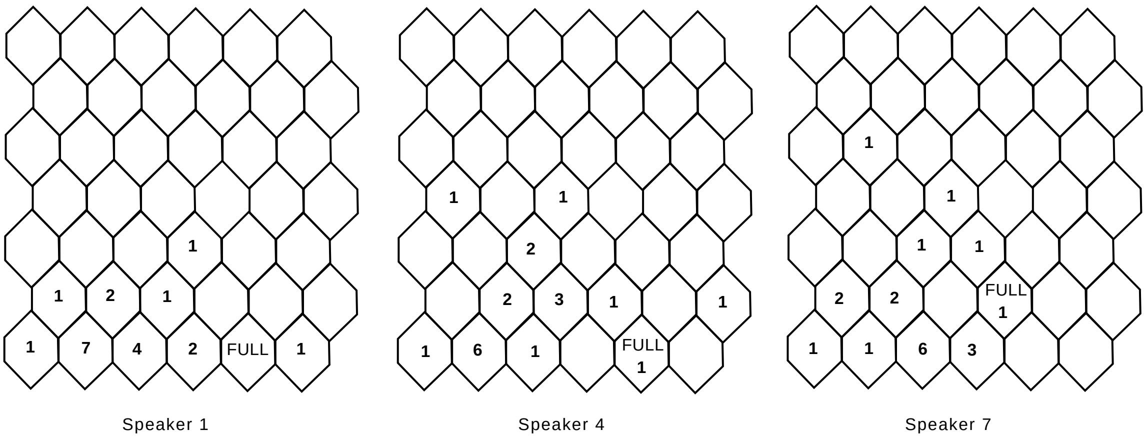


Figure 4.5: Topic retrieval histogram for speakers 1, 4 and 7. Each speaker had their individual articles. The 'FULL' nodes correspond to the nodes retrieved using the whole articles as topic cues. Then, 10 words were randomly picked from the articles and the retrieval was executed using these 10 words as a topic cue. This was repeated 20 times. The values on the nodes show the number of times the node was retrieved.

valuable as to topic retrieval. A priori to tests, the speech cue appeared to be more efficient since it is practically impossible for a sequence of 10 subsequent words not to contain topical words. In turn, for randomly selected 10 words this is, in principle, possible. However, as seen from the results, in large scale, the difference is non-existent.

As seen in Table 4.2, the topic retrieval schemes using document similarity and the whole vocabulary V_B as keywords (criteria 2 and 3) outperformed the scheme using only feature subvocabulary F_B as keywords (criterion 1). In fact, with small-sized topic cues, the latter performed as badly as if the topics would have been retrieved randomly. The result supports the notion that the words in subvocabulary N_B can, on average, be considered valuable keywords for retrieval although they do not play a role in the topical clustering of the background document corpus. This observation is supported by the fact that criterion 2 outperformed criterion 1 also when using full documents as topic cues. At any rate, the result strongly indicates that the potential loss of keyword optimality caused by expanding F_B is compensated by the gained benefit of enlargened keyword vocabulary.

Next comparison in performances is made between retrieval criteria 2 and 3, which, as discussed in section 3.3.3, represent a grained and heavily smoothed word distribution within topics, respectively. As seen in Table 4.2, criterion 3 outperforms criterion 2 when the size of the topic cue is decreased while the performances are identical using full cues. The result indicates that a heavily smoothed word distribution may be preferable to a grainier one, especially when the topic cues are of small size. This observation interestingly agrees with the intuitive conception that there is always a possibility, no matter how small, for all the known words to be associated with all the imaginable topics. Therefore, as to statistical modeling of topical information, no word should be assigned an observation probability of zero within any topic. This is exactly what is done with criterion 3 where the topical n-gram distributions are smoothed so that all the events have a non-zero probability. In turn, as noted in section 3.3.3, criterion 2 assigns a tfidf weight of zero to all unobserved events. Moreover, the conception is expected to be of greater importance in the presence of high uncertainty. This is supported by the fact that the perplexity results between the criteria differ in favor of criterion 3 specifically when the topic cues are small-sized.

As to mixture LMs, we note that, overall, using a single topic cluster and a neighborhood of 7 clusters has little impact on the performance of the adapted model compared to each other. However, the results indicate that the mixture models using neighborhoods slightly outperform the one topic mixtures when the topic cue is small-sized (multimodal cue). In contrast, the one topic mixture outperforms the neighborhood mixture when the topic cue is sufficiently large (full cues). This observation suggests that, given an unreliable topic cue, it is beneficial to broaden the topic estimation by its neighboring topics.

In summary, the perlexity results in Table 4.2 essentially hold the following information. (i) Given a large topic cue, all the topic retrieval criteria result in a beneficial adaptation. (ii) Given a small-sized cue, we can execute, on average, successful topic retrieval using retrieval criterion based on language model likelihoods (criterion 3). (iii) In the case of small-sized cues, the temporal dependencies between the words within the topic cue are not likely to have an impact on the topic retrieval. (iv) As to using mixtures in language model combination, given small-sized topic cues, the impacts of single topics and topical neighbourhoods did not differ significantly from each other.

4.5.2 Speech recognition

The discussion in previous section was based on the perplexity results. The results suggested that the topic retrieval can indeed be done, on average, successfully using small-sized cues. However, the improvements in perplexity scores usually appear in very small scale as for the whole speech recognition system. Therefore, we now analyze the speech recognition results in more detail.

First, by looking at Figure 4.5, we can confirm that the topics retrieved using small cues are spread nicely around the best possible retrievals in all three speaker cases. Although from 20 retrievals using small-sized topic cues none or only one coincided with the node retrieved using large cue, it seems that the topic estimations tend to be localized at the correct direction at the bottom section of the map. However, as seen in Table 4.4, by adapting the background model with valid topic cues, we achieved reductions in perplexity but no change in the word error rates (WERs). Additionally, in the worst case scenario, i.e. while using randomly retrieved topics, we perceived no significant change in the perplexity nor the WER. However, although after adapting the total WER remained intact, i.e. approximately every fifth word was recognized falsely, the WCR scores show that on average every fourth word in the recognition transcriptions was modified. Therefore, let's compare the transcriptions provided by unadapted background LM and adapted LM in more detail with four example cases.

In the first example, we observe that the topical word *department* is missed by the unadapted LM but captured by the adapted model.

Transcription
The *department* previously said jobs rose by four hundred forty-eight thousand in january.

Recognition using background LM
The *departing* previously said jobs rose by four hundred forty-eight thousand in january.

Recognition using adapted LM
The *department* previously said jobs rose by four hundred forty-eight thousand in january.

The above can be considered a model example of what we want to achieve by adapting. In the second example, both LMs result in identical, and correct, transcriptions. Therefore, changing the model did affect neither the recognition of the topical word *index* nor the non topical words.

Transcription
The index ended with a declining zero point three five point two one thousand two hundred seventy-two point one eighths.

Recognition using background LM
The index ended with a declining zero point three five point two one thousand two hundred seventy-two point one eighths.

Recognition using adapted LM
The index ended with a declining zero point three five point two one thousand two hundred seventy-two point one eighths.

In the third example, we encounter a similar event as above but this time the effect of adaptation is averse. The unadapted model recognizes the word *average* correctly but the adapted model suggests an acoustically similar word *everett* in its place. Additionally, the models cause errors at the terms *at* and *increased to.*

Transcription
The *average* rate on new thirteen week treasury bills *increased to* six point one two percent from five point nine seven percent *at* the previous auction last week.

Recognition using background LM
The *average* rate on new thirteen week treasury bills *to increase to* six point one two percent from five point nine seven percent *at* the previous auction last week.

Recognition using adapted LM
The *everett* rate on new thirteen week treasury bills *increased to* six point one two percent from five point nine seven percent *that* the previous auction last week.

The fourth example represents a situation where neither model is able to capture the correct transcription.

> **Transcription**
> What we don't know is *how much is price and how much is* volume.
>
> **Recognition using background LM**
> What we don't know is *calm are to spray sutton hama a truce* volume.
>
> **Recognition using adapted LM**
> What we don't know is *calm are to spray san hama choose* volume.

This result is caused by a distortion (speaker coughing) in the acoustic signal during the segment *how much is price and how much is*. Therefore, the background model suggests a word sequence which makes little sense. On the other hand, the adapted model can not do any better. However, this example is important in that it describes how the speech recognizer works as a whole. In the beginning of the sentence, the acoustic signal is of good quality and the system has no difficulty in making the transcription regardless of the LM. Then, as the acoustic signal becomes unreliable, the weight of the recognition is shifted on the LM. Therefore, after adapting, the transcription is affected at the segment where the distortion occurs. Unfortunately, as can be seen, the changes do not bring the result any closer to being correct.

The above discussion originated from the observation that we were not able to reduce WER by topically adapting LMs. However, as to the fundamental concept of topic adaptation, as we raise the probabilities of topical words, we consequently have to lower the probabilities elsewhere. Therefore, inherently, we may be improving our chances at recognizing single topical words while degrading the general properties of the LM. In consequence, it is difficult to evaluate the effectiveness of topic adaptation using purely the WER. Therefore, in addition to WER, we evaluated the recognition results using term error rates (TERs) in order to emphasize the effect of the topic adaptation procedure. After the removal of closed class words and stemming, the first example transcription above is as follows.

> **Transcription**
> department previous said jobs rose four hundred forti eight thousand january

This type of preprocessing is beneficial in that the subsequent error rate measurement is focused on the topical words and the effect of adapting the LM can be observed more clearly. As can be seen from the recognition results, the change in evaluation measure from WER to TER verifies further that the adaptation did not result in any improvement in the speech recognition results.

Chapter 5

Conclusions

In this master's thesis, we implemented a topic adaptation procedure for speech recognizer and studied its performance in a multimodal environment. In a multimodal environment, the main assumption suggested that the topic cues provided by modalities correspond to short lists of words. Therefore, we focused on studying the effect of these small-sized cues on topic retrieval in combination with differing topic retrieving criteria.

The adaptation procedure consisted of obtaining topical partitioning of background corpus, topic retrieval of the adaptation data, and obtaining an adapted model by combining the language model of the topic with the background language model. The topical partitioning was acquired using the vector space model for document representation, the random projection for dimensionality reduction, and the self organizing maps for accomplishing the clustering. Three topic retrieval criteria, namely document similarity using the feature subvocabulary, document similarity using the total vocabulary, and topic language model likelihood, were described and implemented. The adapted language models were combined using mixtures of bigram background and topical models. All the models used in the experiments were trained using the Kneser-Ney smoothing method.

The results of the experiments executed on analyzing the obtained language models are summarized as follows. First, assuming a small-sized topic cue, it seems unlikely that the topic retrieval would be affected by the temporal dependencies between the keywords. Second, the subvocabulary N consisting of rare words and not used as features in topical clustering of background corpus is to be, on average, considered valuable keywords instead of noise. Moreover, assuming topic retrieval using small-sized topic cues, heavily smoothed word distributions within topics seem to be preferable to fine-grained distributions. Finally, experiments suggest that topic retrieval using small-sized keyword lists as topic cues results, on average, in a successful topic estimation.

The results of the experiments executed on complete adapted speech recognition system are summarized as follows. The topic adapted LMs achieved consistently lower perplexity scores than the unadapted background model. However, the word error rates (WER) and the term error rates (TER) between topic adapted LMs and background model did not differ significantly. Importantly, this holds for well-retrieved topics as well as for intentionally ill-retrievied topics. In brief, the speech recognition results show that successful topic estimation and adaptation, i.e. improvements on perplexity, are feasible, while achieving reductions in word error rates appears to a much more tedious task.

In the experiments, the reception of topic cues from multimodal sources was based purely on a simulation. Therefore, the experiments in this work can be naturally extended, and the notes made verified, when real-life data from multimodal interfaces becomes available.

Bibliography

[1] T. Hirsimäki. A decoder for large-vocabulary continuous speech recognition. Master's thesis, Helsinki University of Technology, 2002.

[2] M. Varjokallio. Subspace Methods for Gaussian Mixture Models in Automatic Speech Recognition. Master's thesis, Helsinki University of Technology, 2007.

[3] U. Remes. Speaker-Based Segmentation and Adaptation in Automatic Speech Recognition. Master's thesis, Helsinki University of Technology, 2007.

[4] J. Kuusisto. Recognition of dialogue topics with learning methods. Master's thesis, Helsinki University of Technology, 2002.

[5] S. Broman. Combining methods for language models in speech recognition. Master's thesis, Helsinki University of Technology.

[6] M. Creutz. *Induction of the morphology of natural language: Unsupervised morpheme segmentation with application to automatic speech recognition.* PhD thesis, Helsinki University of Technology, 2006.

[7] V. Siivola. *Language Models for Automatic Speech Recognition: Construction and Complexity Control.* PhD thesis, Helsinki University of Technology, 2007.

[8] T. Hirsimäki. *Advances in unlimited-vocabulary speech recognition for morphologically rich languages.* PhD thesis, Helsinki University of Technology, 2007.

[9] M. Kurimo and K. Lagus. An efficiently focusing large vocabulary language model. *Lecture notes in computer science*, pages 1068–1073, Springer, 2002.

[10] L. Rabiner and B.H. Juang. *Fundamentals of speech recognition.* Tsinghua University Press, 1993.

[11] BH Juang and LR Rabiner. Hidden Markov Models for Speech Recognition. *Technometrics*, 33(3):251–272, 1991.

[12] G.D. Forney Jr. The Viterbi algorithm. *Proceedings of the IEEE*, 61(3):268–278, 1973.

[13] C.D. Manning, H. Schutze, and MIT Press. *Foundations of statistical natural language processing*. MIT Press, 1999.

[14] J.T. Goodman. A bit of progress in language modeling. *Computer Speech & Language*, 15(4):403–434, 2001.

[15] T. Kohonen. *Self-organizing maps*. Springer, 2001.

[16] J.R. Bellegarda. Exploiting latent semantic information in statistical language modeling. *Proceedings of the IEEE*, 88(8):1279–1296, 2000.

[17] T.K. Landauer, P.W. Foltz, and D. Laham. An introduction to latent semantic analysis. *Discourse processes*, 25:259–284, 1998.

[18] A. Gionis, H. Mannila, and P. Tsaparas. Clustering aggregation. *ACM Trans. Knowl. Discov. Data*, 1(1):4, 2007.

[19] I.S. Dhillon, S. Mallela, and D.S. Modha. Information-theoretic co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98. ACM New York, NY, USA, 2003.

[20] S. Guha, R. Rastogi, and K. Shim. ROCK: A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5):345–366, 2000.

[21] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407, 1990.

[22] T. Kohonen, S. Kaski, K. Lagus, J. Salojärvi, J. Honkela, V. Paatero, and A. Saarela. Self organization of a massive document collection. *IEEE transactions on neural networks*, 11(3):574–585, 2000.

[23] V. Siivola, M. Kurimo, and K. Lagus. Large vocabulary statistical language modeling for continuous speech recognition in Finnish. In *Seventh European Conference on Speech Communication and Technology*. ISCA, 2001.

[24] T. Honkela, S. Kaski, K. Lagus, and T. Kohonen. WEBSOM - self-organizing maps of document collections. In *Proceedings of WSOM*, volume 97, pages 4–6, 1997.

[25] S. Kaski. Dimensionality reduction by random mapping: fast similarity computation for clustering. In *Neural Networks Proceedings, 1998. IEEE World Congress on Computational Intelligence. The 1998 IEEE International Joint Conference on*, volume 1, 1998.

[26] T. Kohonen. The self-organizing map. *Neurocomputing*, 21(1-3):1–6, 1998.

[27] A. Singhal. Modern information retrieval: A brief overview. *IEEE Data Engineering Bulletin*, 24(4):35–43, 2001.

[28] IJ Good. The Population Frequencies of Species and the Estimation of Population Parameters. *Biometrika*, pages 237–264, 1953.

[29] IH WITTEN and TC BELL. The zero-frequency problem: estimating the probabilities of novel events in adaptive text compression. *IEEE transactions on information theory*, 37(4):1085–1094, 1991.

[30] H. Ney R. Kneser. Improved backing-off for m-gram language modeling. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 1:181–184, May 1995.

[31] S. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35(3):400–401, 1987.

[32] J. Goodman S.F. Chen. An empirical study of smoothing techniques for language modeling. *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, 1:310–318, 1996.

[33] J.R. Bellegarda. Statistical language model adaptation: review and perspectives. *Speech Communication*, 42(1):93–108, 2004.

[34] P.R. Clarkson and A.J. Robinson. Language model adaptation using mixtures and an exponentially decaying cache. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1997. ICASSP-97.*, volume 2, 1997.

[35] R. Kuhn and R. De Mori. A cache-based natural language model for speech recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6):570–583, 1990.

[36] S. Roukos R. Lau, R. Rosenfeld. Trigger-based language models: a maximum entropy approach. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2:45–48, May 1993.

[37] R. Rosenfeld. *Adaptive Statistical Language Modeling: A Maximum Entropy Approach.* PhD thesis, Carnegie Mellon University, 1994.

[38] R. Kneser, J. Peters, and D. Klakow. Language model adaptation using dynamic marginals. In *Fifth European Conference on Speech Communication and Technology.* ISCA, 1997.

[39] M. Federico. Efficient language model adaptation through MDI estimation. In *Sixth European Conference on Speech Communication and Technology.* ISCA, 1999.

[40] S.F. Chen, K. Seymore, and R. Rosenfeld. Topic adaptation for language modeling using unnormalized exponential models. In *Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on*, volume 2, 1998.

[41] T. Kawahara C. Troncoso. Trigger-based language model adaptation for automatic meeting transcription. *INTERSPEECH-2005*, 2005.

[42] C.H. Chueh and J.T. Chien. Reliable feature selection for language model adaptation. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2008. ICASSP 2008*, pages 5089–5092, 2008.

[43] D. Graff. English Gigaword. *Linguistic Data Consortium, Philadelphia*, 2003
http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2003T05,
July 1, 2009.

[44] D.B. Paul and Baker J.M. D.B. Paul, and J.M. Baker. The Design of the Wall Street Journal-based CSR Corpus. February, 1992. *Proceedings of the DARPA Speech and Natural Language Workshop*, February 1992.

[45] J.J. Odell. *The use of context in large vocabulary speech recognition.* PhD thesis, PhD thesis, Cambridge University Engineering Department, 1995.

[46] J. Pylkkönen and M. Kurimo. Duration modeling techniques for continuous speech recognition. In *Eighth International Conference on Spoken Language Processing.* ISCA, 2004.

[47] S.E. Johnson, P. Jourlin, G.L. Moore, K.S. Jones, and P.C. Woodland. The Cambridge University spoken document retrieval system. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999. ICASSP'99. Proceedings.*, volume 1, 1999.

[48] M.F. Porter. An algorithm for suffix stripping. *Program: electronic library and information systems*, 40(3):211–218, 2006.

[49] http://ucrel.lancs.ac.uk/bncfreq/flists.html. July 1, 2009.

[50] J.S. Milton and J.C. Arnold. Introduction to probability and statistics, McGraw-Hill. *New York, USA*, 1995.