Recent Developments in Language Modeling Techniques and their Applications

Berlin Chen (陳柏琳)
Professor, Department of Computer Science & Information Engineering
National Taiwan Normal University
Fall, 2013

Outline

- **Introduction (*n*-gram)**
- Topic Modeling (LSA, NMF, PLSA, LDA, WTM)
- Discriminative Language Modeling
- Neural Network Language Modeling
- Relevance Language Modeling
- Positional Language Modeling
- Conclusions

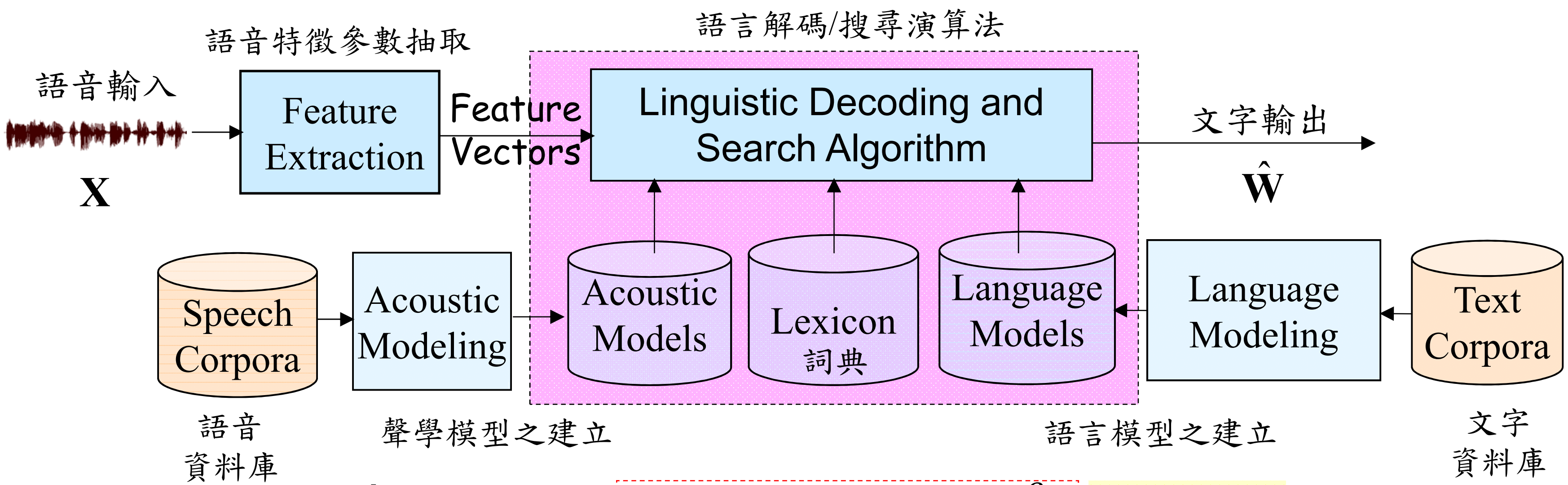
Introduction

- Language is unarguably the most nuanced and sophisticated medium to express or communicate our thoughts
  - A natural vehicle to convey our thoughts and the content of all wisdom and knowledge
- Language modeling (LM) is a **mathematical description** of **language phenomena** (a kind of uncertainty situations/observations)
  - **Compositions (samples)**:
    - Classes/clusters, documents, paragraphs, sentences/passages, phrases, etc.
  - **Units (instances):**
    - Words, sub-words (phones/graphemes/syllables), syntactic/semantic tags, etc.
  - **Relationships** among/between compositions and units:
    - Occurrence/co-occurrence (0/1, counts), proximity (0/1, counts) , structure, etc.
  - **Application Tasks** (deduce some properties/information of interest)

1. T. Hofmann, "ProbMap - A probabilistic approach for mapping large document collections," *IDA*, 2000.
2. B. Chen, "Word topic models for spoken document retrieval and transcription," *ACM TALIP*, 2009.

Introduction: LM for Speech Recognition

- LM can be used to capture the regularities in human natural language and quantify the acceptability of a given word sequence, has long been an interesting yet challenging research topic in the speech recognition community



$$\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} \; p(\mathbf{X} \mid \mathbf{W})^{\alpha} P(\mathbf{W})^{\beta}$$

Decoding

Acoustic Modeling

Language Modeling

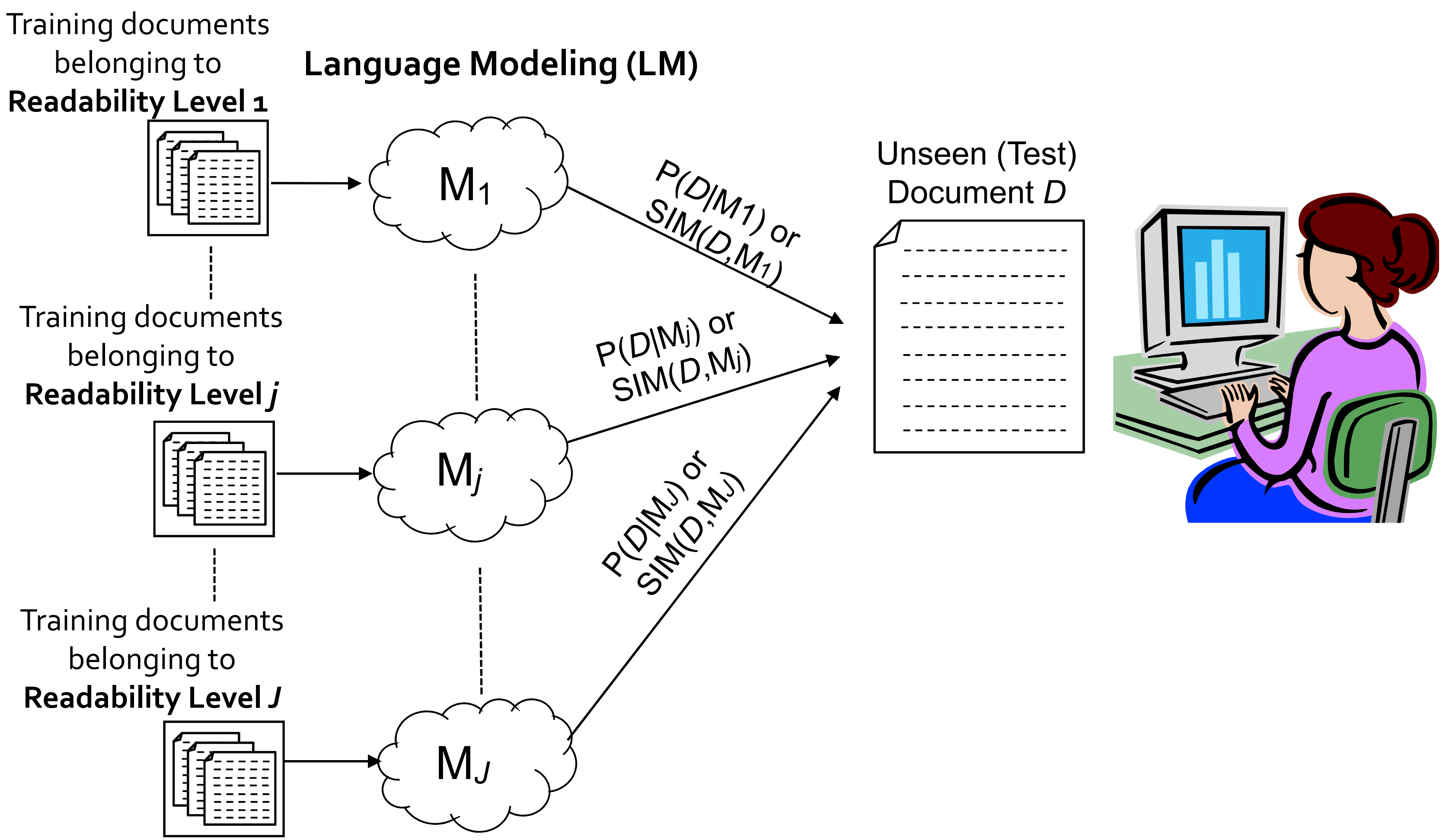
M.J.F. Gales and S.J. Young. *The Application of Hidden Markov Models in Speech Recognition. Foundations and Trends in Signal Processing*, 2008

Introduction: Other Applications

- Recently, LM also has been introduced to a wide spectrum of natural language processing (NLP) problems, and provided an effective and theoretically attractive (statistical or probabilistic) framework for building application systems
    - What is LM Used for (apart from speech recognition)?
        - Information retrieval
        - Machine translation
        - Summarization
        - Document classification and routing
        - Spelling correction
        - Handwriting recognition
        - Optical character recognition
        - ...

• G. Tur and R. D. Mori (eds.), *Spoken Language Understanding - Systems for Extracting Semantic Information from Speech*, John Wiley and Sons, 2011.

Exemplar: LM for Readability Classification



Can we leverage various language modeling techniques for readability classification?

Introduction: *n*-gram

- The *n*-gram language model that determines the probability of an upcoming word given the previous *n*-1 word history is the most prominently used

$$
\begin{aligned}
P(\mathbf{W} = w_1, w_2, \ldots, w_m) \\
&= P(w_1) P(w_2 | w_1) P(w_3 | w_1, w_2) \ldots P(w_m | w_1, w_2, \ldots, w_{m-1}) \\
&= P(w_1) \underbrace{\prod_{i=2}^{m} P(w_i | w_1, w_2, \ldots, w_{i-1})}_{\text{Multiplication of Conditional Probabilities}}
\end{aligned}
$$

Chain Rule

  - *n*-gram assumption

$$P(w_i | w_1, w_2, \ldots, w_{i-1}) \approx P(w_i | \underbrace{w_{i-n+1}, w_{i-n+2}, \ldots, w_{i-1}}_{\text{History of length } n\text{-1}})$$

$$P(w_i | w_1, w_2, \ldots, w_{i-1}) \approx P(w_i | w_{i-2}, w_{i-1})$$ Trigram

$$P(w_i | w_1, w_2, \ldots, w_{i-1}) \approx P(w_i | w_{i-1})$$ Bigram

$$P(w_i | w_1, w_2, \ldots, w_{i-1}) \approx P(w_i)$$ Unigram

R. Rosenfeld, "Two decades of statistical language modeling: Where do we go from here?," *Proceedings of IEEE*, 2000.

Known Weakness of n-gram Language Models

- Shortcomings are at least two-fold
  - Sensitive to changes in the style or topic of the text on which they are trained
  - Assume the probability of next word in a sentence depends only on the identity of last *n*-1 words
    - Capture only local contextual information or lexical regularity (word ordering relationships) of a language

$$P(w_i | w_1, w_2, \ldots, w_{i-1}) \approx P(w_i | w_{i-2}, w_{i-1})$$ e.g., trigram LM

- Ironically, *n*-gram language models take no advantage of the fact that what is being modeled is language
  - Frederick Jelinek said "*put language back into language modeling*" (1995)

F. Jelinek, "The dawn of statistical ASR and MT," Computational Linguistics, 35(4), pp. 483-494, 2009.

Introduction: Typical Issues for LM

- Evaluation
  - How can you tell a good language model from a bad one
  - For example, in the context of speech recognition, we can run a speech recognizer or adopt other statistical measurements
- Smoothing
  - Deal with data sparseness of real training data
  - Various approaches have been proposed
- Caching/Adaptation
  - If you say something, you are likely to say it again later
  - Adjust word frequencies observed in the current conversation
- Clustering
  - Group words with similar properties (similar semantic or grammatical) into the same class
  - Another efficient way to handle the data sparseness problem

Outline

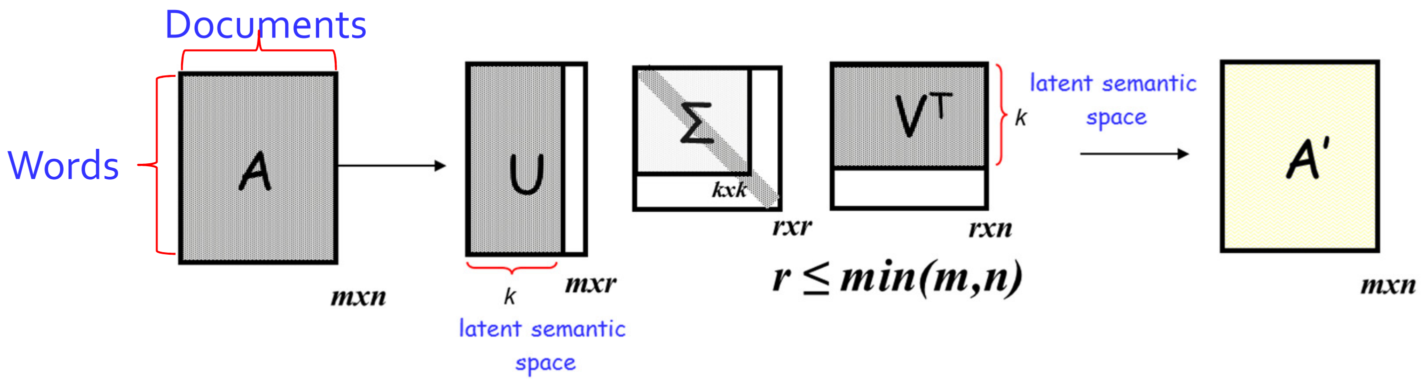
- Introduction (*n*-gram)
- **Topic Modeling (LSA, NMF, PLSA, LDA, WTM)**
- Discriminative Language Modeling
- Neural Network Language Modeling
- Relevance Language Modeling
- Positional Language Modeling
- Conclusions

Topic Modeling

- Topic language models have been introduced and investigated to complement the *n*-gram language models
  - A commonality among them is that a set of latent topic variables $\{T_1, T_2, \ldots, T_K\}$ is introduced to describe the "***word-document***" co-occurrence characteristics
- Models developed generally follow two lines of thought
  - Algebraic
    - Latent Semantic Analysis (LSA) (Deerwester et al., 1990), nonnegative matrix factorization (NMF) (Lee and Seung, 1999), and their derivatives
  - Probabilistic
    - Probabilistic latent semantic analysis (PLSA) (Hofmann, 2001), latent Dirichlet allocation (LDA) (Blei et al., 2003), Word Topic Model (Chen, 2009), and their derivatives

Latent Semantic Analysis (LSA)

- Start with a matrix describing the intra- and Inter-document statistics between all terms and all documents
- Singular value decomposition (SVD) is then performed on the matrix to project all term and document vectors onto a reduced latent topical space



$$\|A\|_F^2 = \sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}^2 \Rightarrow \|A\|_F^2 = \sigma_1^2 + \sigma_2^2 + \ldots + \sigma_r^2 \quad ?$$

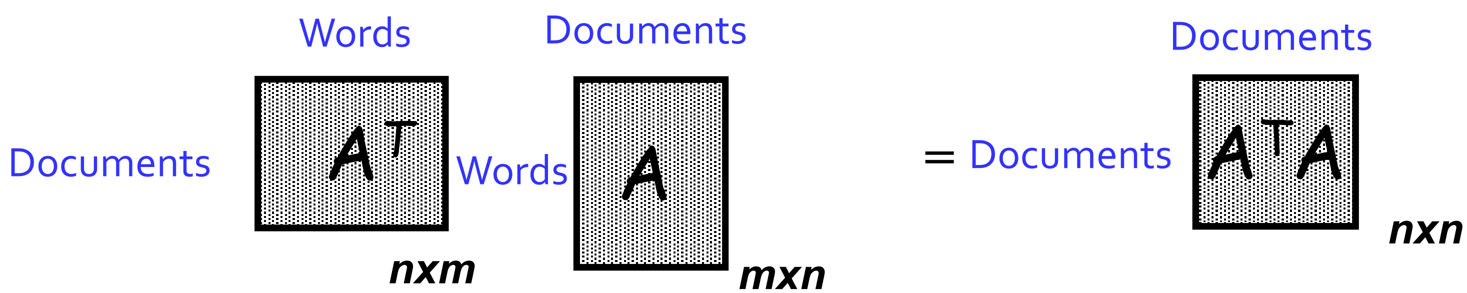
- In the context of IR, matching between queries and documents can be carried out in this topical space

1. G. W. Furnaset et al., "Information Retrieval using a Singular Value Decomposition Model of Latent Semantic Structure," *SIGIR1988.*
2. T. K. Landauer et al. (eds.) , *Handbook of Latent Semantic Analysis*, Lawrence Erlbaum, 2007.

LSA: Properties

- The latent space of LSA is derived on top of eigen-decomposition of the matrix $A^T A$
  - **Each entry of $A^T A$ represents the correlation (inner product; closeness relationship) between any document (vector) pairs**
- The column vectors v_j in V actually are eigenvectors of $A^T A$
  - $A^T A$ is symmetric and all its diagonal entities are positive
    - All eigenvalues λ_j are nonnegative real numbers
    - All eigenvectors v_j are orthonormal
    - Singular values σ_j in ∑ are the square roots of λ_j $\left(\sigma_j = \sqrt{\lambda_j}\right)$

$$\left(A^T A\right) v_i = \lambda_i v_i$$



LSA bears similarly to PCA (Principal Component Analysis), and has the aim of finding a subspace determined by the eigenvectors of $A^T A$ that preserves most of the relationships (a kind of simple structure information) between documents (compositions).

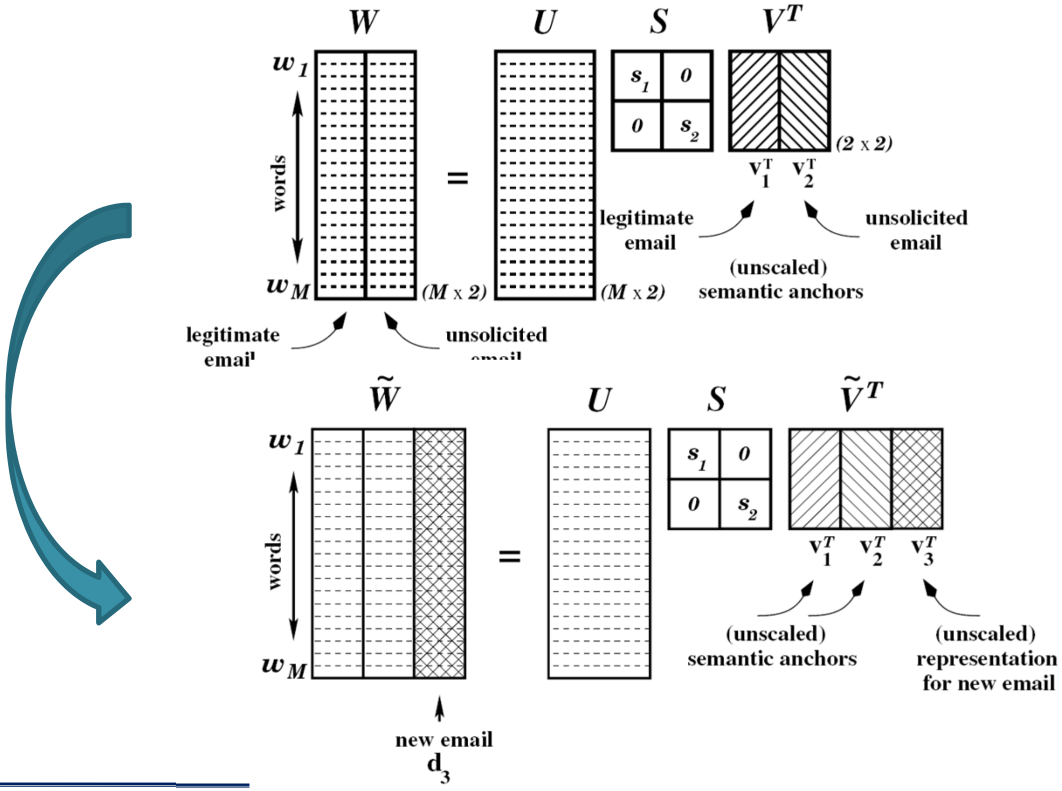
LSA: Properties

- Pro
  - A clean formal framework and a clearly defined optimization criterion (least-squares)
    - Conceptual simplicity and clarity
  - Handle synonymy problems ("heterogeneous vocabulary")
    - Replace individual terms as the descriptors of documents by independent "***artificial concepts***" that can specified by any one of several terms (or documents) or combinations
- Con
  - Contextual or positional information for words in documents is discarded (the so-called "***bag-of-words***" assumption)
  - High computational complexity (e.g., SVD decomposition)
  - Word and document representations have negative values
  - Exhaustive search are needed when compare among documents or between a query (word) and a document (cannot make use of inverted files ?)

LSA: Application to Junk E-mail Filtering

- One vector represents the centriod of all e-mails that are of interest to the user, while the other the centriod of all e-mails that are not of interest

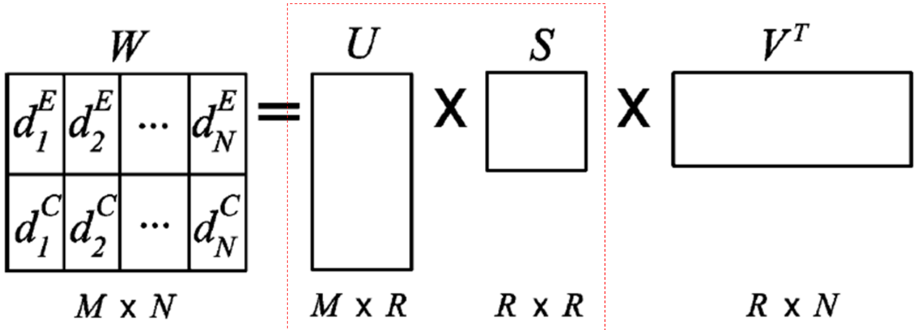
folding-in



J. R. Bellegarda, " Latent Semantic Mapping: Principles & Applications," *Synthesis Lecture on Speech and Audio Processing*, 3, 2007.

LSA: Application to Cross-lingual Language Modeling

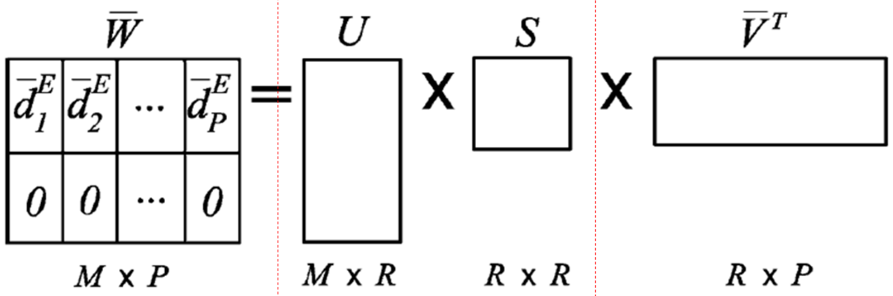
- Assume that a document-aligned (instead of sentence-aligned) Chinese-English bilingual corpus is provided



SVD of a word-document matrix for CL-LSA.

$$P_{\text{CL-LSA-Unigram}}\left(c \middle| d_i^E\right) = \sum_e P_T(c|e) P\left(e \middle| d_i^E\right)$$

$$P_T(c|e) \approx \frac{\text{sim}(\vec{c},\vec{e})^\gamma}{\sum_{c'} \text{sim}(\vec{c}\,',\vec{e})^\gamma} \qquad (\gamma >> 1)$$

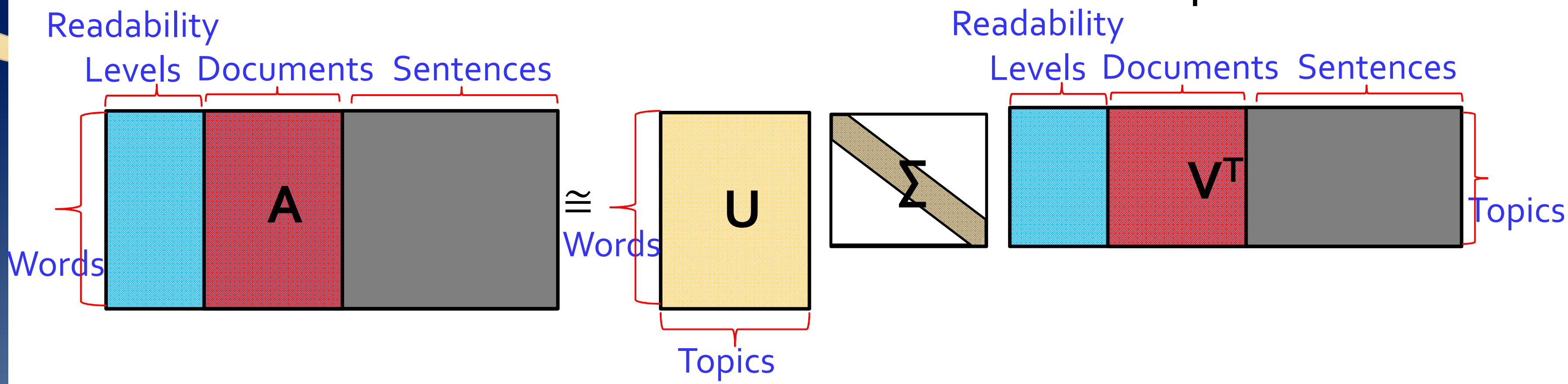


Folding-in a monolingual corpus into LSA.

W. Kim & S. Khudanpur, "Lexical triggers and latent semantic analysis for cross-lingual language model adaptation," *ACM Transactions on Asian Language Information Processing* (*TALIP*), 3(2), pp. 94 – 112, 2004.

LSA: Application to Readability Classification

- Aim to extract "word-readability level", "word-document" and "word sentence" co-occurrence relationships

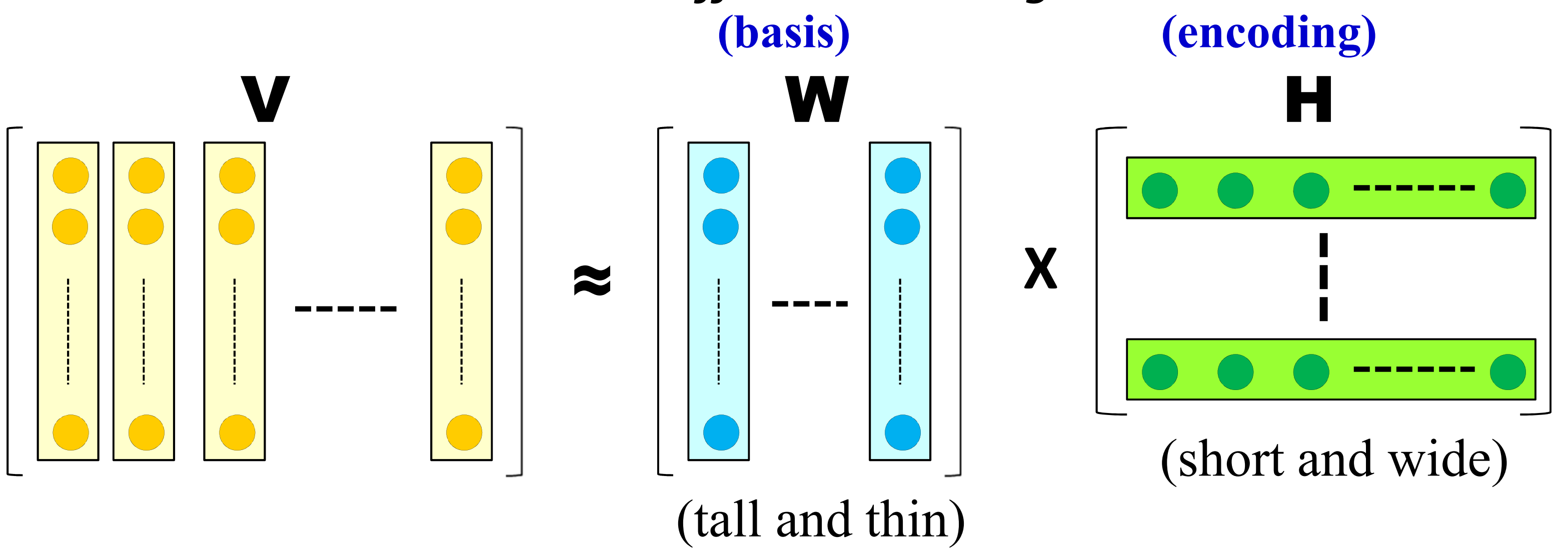


- Very Preliminary Results on Six-level Readability Classification (10-fold tests; w.r.t. classification accuracy (%))

	NHK98 (410 documents)	國編版 (265documents)
"word-readability level" relationship (dimensionality=6)	0.329	0.260
"word-readability level" & "word-document" relationships (dimensionality=20)	0.346	0.426

Nonnegative Matrix Factorization (NMF)

- NMF approximates data with an **additive and linear combination** of nonnegative components (or basis vectors)
  - Given a **nonnegative data matrix** $V \in R^{L \times M}$, NMF computes another two **nonnegative matrices** $W \in R^{L \times r}$ and $H \in R^{r \times M}$ such that $V \approx WH$
    - *r<< L and r<< M to ensure efficient encoding*

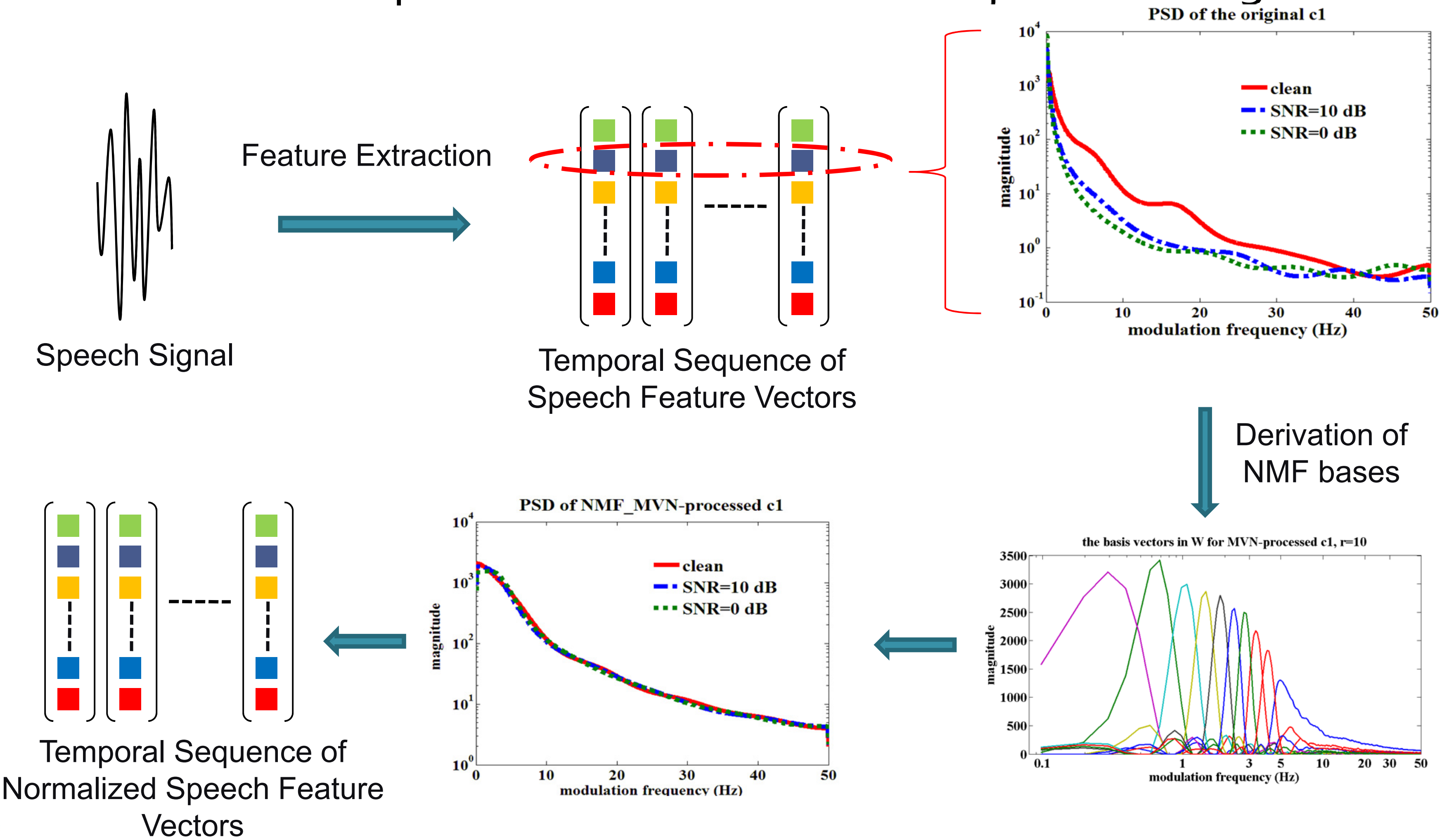


$$\mathbf{v} \approx \mathbf{W}\mathbf{h} = \sum_{r=1}^{R} h_r \mathbf{w}_r = h_1 \mathbf{w}_1 + \ldots + h_R \mathbf{w}_R$$

1. D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, 1999.
2. M. Shashanka et al., "Probabilistic Latent Variable Models as Non-Negative Factorizations," . *Computational Intelligence and Neuroscience*, 2008.

NMF: Application to ASR Robustness

- Modulation Spectrum Factorization for Speech Recognition



W.-Y. Chu, et al., "Modulation spectrum factorization for robust speech recognition," *APSIPA ASC*, 2011.

NMF: Application to ASR Robustness

- Word Error Rate (WER) Results on the Aurora-2 task

	Set A	Set B	Set C	Average
Baseline MFCC	45.13	51.13	36.05	45.71
NMF (DIM=5)	28.41	24.31	29.18	26.92
NMF (DIM=10)	28.80	24.35	29.56	27.17
NMF (DIM=20)	28.91	24.52	30.04	27.38
NMF (DIM=30)	28.58	24.42	29.54	27.11
NMF(DIM=5, sparse)	28.49	24.11	28.54	26.38
NMF(DIM=5)+CMVN	16.66	14.91	17.31	16.09
NMF(DIM=5, sparse)+CMVN	16.59	14.92	17.24	15.89
CMN	33.19	28.21	32.36	31.03
CMVN	24.07	23.24	23.18	23.56
HEQ	19.97	17.95	19.90	19.15
MVA	19.11	18.00	18.51	18.55
AFE	12.32	12.90	13.73	12.83

Probabilistic Latent Semantic Analysis (PLSA)

- Each document as a whole consists of a set of shared latent topics with different weights -- a document topic modeling (DTM) approach
  - Each topic in turn offers a unigram (multinomial) distribution for observing a given word

$$P_{\text{PLSA}}(w \mid D) = \sum_{k=1}^{K} P(w_i \mid T_k) P(T_k \mid D)$$

- LDA (latent Dirichlet allocation) differs from PLSA mainly in the inference of model parameters:
  - PLSA assumes the model parameters are fixed and unknown
  - LDA places additional a priori constraints on the model parameters, i.e., thinking of them as random variables that follow some Dirichlet distributions

1. T. Hoffmann, "Unsupervised learning by probabilistic latent semantic analysis," Machine Learning, 2001.
2. D. M. Blei et al., "Latent Dirichlet allocation," *Journal of Machine Learning Research*, 2003.

Word Topic Modeling (WTM)

- Each word of language is treated as a word topic model (WTM) for predicting the occurrences of other words

$$P_{\mathrm{WTM}}\left(w_i \mid \mathrm{M}_{w_j}\right) = \sum_{k=1}^{K} P\left(w_i \mid T_k\right) P\left(T_k \mid \mathrm{M}_{w_j}\right)$$

- The WTM $P_{\mathrm{WTM}}\left(w_i \mid \mathrm{M}_{w_j}\right)$ of each word can be trained with maximum likelihood estimation (MLE)
  - By concatenating those words occurring within a context window around each occurrence of the word, which are assumed to be relevant to the word, to form the training observation

Vicinity of a Word

$$\log L_{\mathbf{W}} = \sum_{w_j \in \mathbf{W}} \log P_{\mathrm{WTM}}\left(Q_{w_j} \middle| \mathrm{M}_{w_j}\right) = \sum_{w_j \in \mathbf{W}} \sum_{w_i \in Q_{w_j}} c\left(w_i, Q_{w_j}\right) \log P_{\mathrm{WTM}}\left(w_i \middle| \mathrm{M}_{w_j}\right)$$

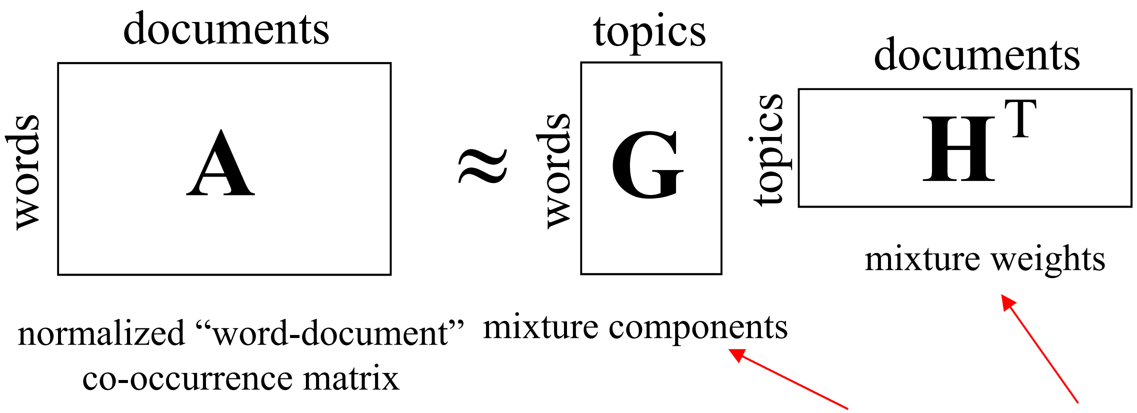
- $\mathbf{W}$: the set of words in the language

Can we model topical information using other compositions beyond "documents" ?

Comparison Between WTM and DTM

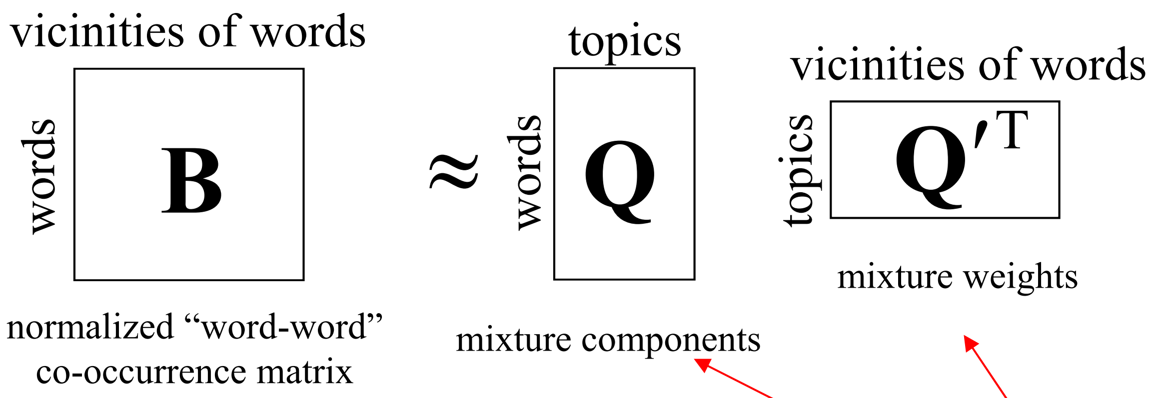
- Probabilistic Matrix Decompositions

PLSA/LDA



$$P_{\text{PLSA/LDA}}(w_i \mid D) = \sum_{k=1}^{K} P(w_i \mid T_k) P(T_k \mid D)$$

WTM



$$P_{\text{WTM}}(w_i \mid \mathrm{M}_{w_j}) = \sum_{k=1}^{K} P(w_i \mid T_k) P(T_k \mid \mathrm{M}_{w_j})$$

Example Topic Distributions of WTM

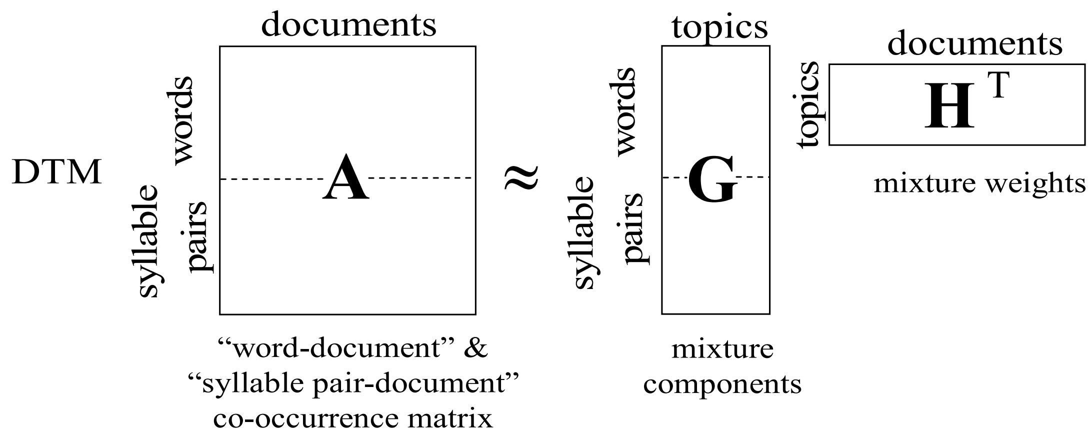
Topic 13	
word	**weight**
Vena (靜脈)	1.202
Resection (切除)	0.674
Myoma (肌瘤)	0.668
Cephalitis (腦炎)	0.618
Uterus (子宮)	0.501
Bronchus (支氣管)	0.500

Topic 14	
word	**weight**
Land tax (土地稅)	0.704
Tobacco and alcohol tax law (菸酒稅法)	0.489
Tax (財稅)	0.457
Amend drafts (修正草案)	0.446
Acquisition (購併)	0.396
Insurance law (保險法)	0.373

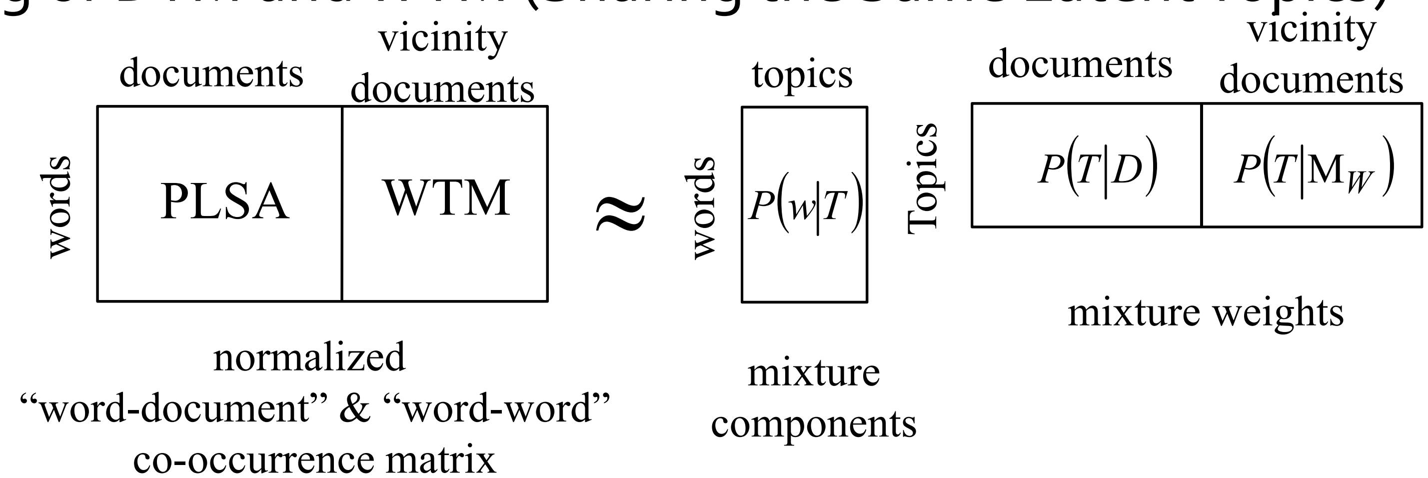
Topic 23	
word	**weight**
Cholera (霍亂)	0.752
Colorectal cancer (大腸直腸癌)	0.681
Salmonella enterica (沙門氏菌)	0.471
Aphtae epizooticae (口蹄疫)	0.337
Thyroid (甲狀腺)	0.303
Gastric cancer (胃癌)	0.298

Some Extensions of DTM and WTM

- Hybrid of Different Indexing Features for DTM/WTM



- Pairing of DTM and WTM (Sharing the Same Latent Topics)



Visualization of Document Collections with PLSA

- The original formulation of PLSA

$$P_{\text{PLSA}}(w \mid D) = \sum_{k=1}^{K} P(w_i \mid T_k) P(T_k \mid \mathrm{D})$$

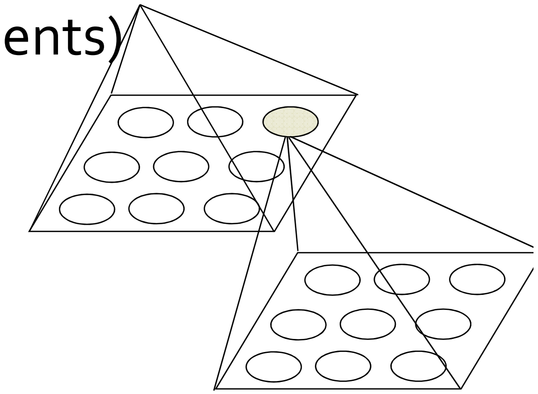
- ProbMap: PLSA additionally takes into account the relationships between topics

$$P_{\text{ProbMap}}(w \mid D) = \sum_{k=1}^{K} \left[\sum_{j=1}^{K} P(w \mid T_j) P(T_j \mid T_k) \right] P(T_k \mid \mathrm{D})$$

  - Where $P(T_j \mid T_k)$ has to do with the topological distance between any two topics (or clusters of documents)

$$E(T_l, T_k) = \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left[-\frac{dist(T_l, T_k)^2}{2\sigma^2} \right]$$

$$P(T_j | T_k) = \frac{E(T_j, T_k)}{\sum_{j'=1}^{K} E(T_s, T_k)}$$



Two-dimensional
Tree Structure for Organized Topics

T. Hofmann, "ProbMap - A Probabilistic Approach for Mapping Large Document Collections," *IDA*, 2000.

Visualization of Document Collections with PLSA

- Estimation of the Component Distributions (with EM algorithm)

$$\hat{P}(w \mid T_k) = \frac{\sum_{i=1}^{N} c(w, D_i) P_U(T_k \mid w, D_i)}{\sum_{j=1}^{M} \sum_{h=1}^{N} c(w_j, D_h) P_U(T_k \mid w_j, D_h)}$$

$$\hat{P}(T_k \mid D_i) = \frac{\sum_{j=1}^{M} c(w_j, D_i) P_V(T_k \mid w_j, D_i)}{\sum_{j'=1}^{M} c(w_{j'}, D_i)}$$

  - Where

$$P_U(T_k \mid w, D_i) = \frac{P(w \mid T_k) \cdot P(T_k \mid D_i)}{\sum_{m=1}^{K} P(w \mid T_m) \cdot P(T_m \mid D_i)}$$

$$P_V(T_k \mid w, D_i) = \frac{P(T_k \mid D_i) \sum_{k'=1}^{K} P(T_{k'} \mid T_k) P(w \mid T_{k'})}{\sum_{s=1}^{K} P(T_s \mid D_i) \sum_{l=1}^{K} P(T_l \mid T_s) P(w \mid T_l)}$$

- Selection of Representative Topic Words

$$S(w, T_k) = \frac{\sum_{i=1}^{N} c(w, D_i) P(T_k \mid D_i)}{\sum_{i'=1}^{N} c(w, D_{i'}) [1 - P(T_k \mid D_{i'})]}$$

L.-S. Lee and B. Chen, “Spoken document understanding and organization,” *IEEE Signal Processing Magazine*, 2005.

Outline

- Introduction (*n*-gram)
- Topic Modeling (LSA, NMF, PLSA, LDA, WTM)
- **Discriminative Language Modeling**
- Neural Network Language Modeling
- Relevance Language Modeling
- Positional Language Modeling
- Conclusions

N T N U

Discriminative Language Modeling (DLM)

- DLM for Speech Recognition
  - DLM takes a testing utterance X together with a set of top-scoring recognition hypotheses $\mathbf{GEN}(X)$, produced by the baseline speech recognition system, as the input
  - DLM selects the most promising hypothesis W^* out from $\mathbf{GEN}(X)$ through the following equation:

$$W^* = \mathrm{DLM}(X, \mathbf{GEN}(X)) = \underset{W \in \mathbf{GEN}(X)}{\arg\max}\, \mathbf{\Phi}(X, W) \bullet \boldsymbol{\alpha}$$

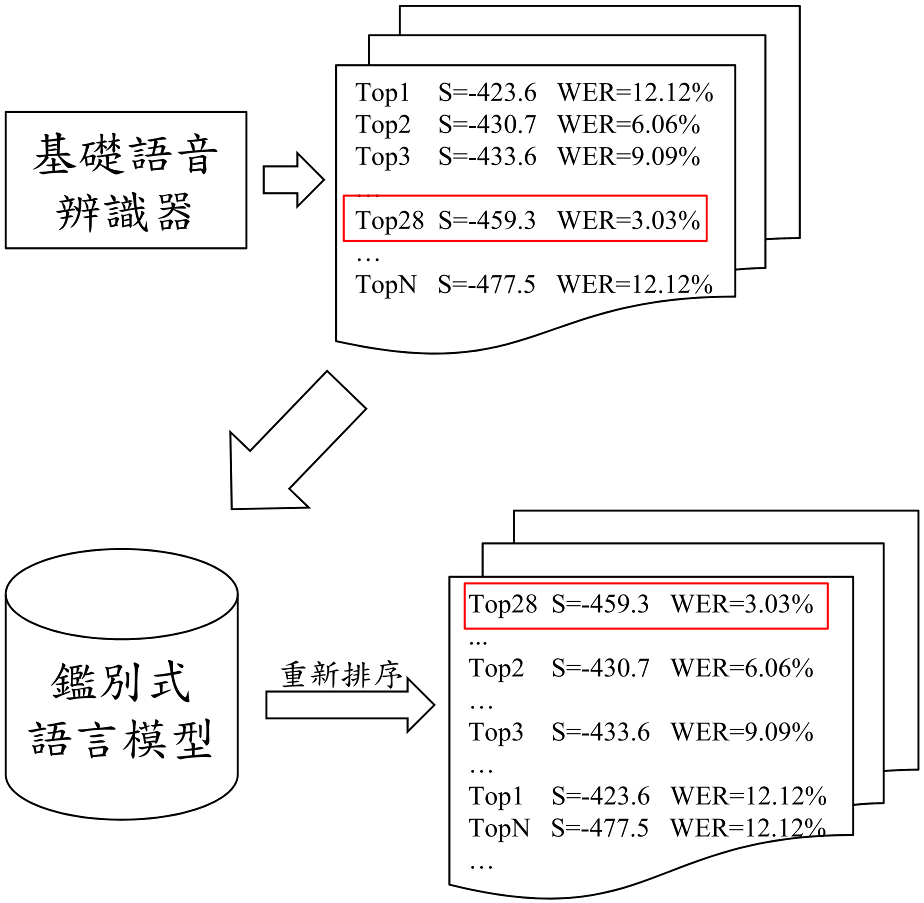
  - Where $\mathbf{\Phi}(X,W)$ is a feature vector used to characterize a recognition hypothesis W for X, and $\boldsymbol{\alpha}$ is the parameter vector of a DLM model

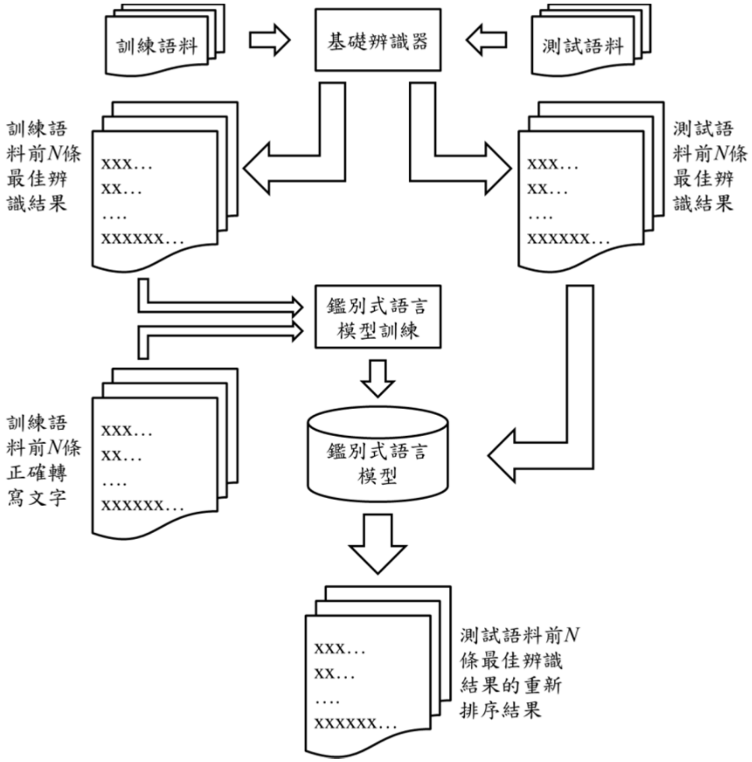
		word unigrams				word bigrams			
	$\log[P(W)P(W\|x)]$	w_p	w_q	...	w_t	$w_p w_k$	...	$w_j w_m$	$w_l w_m$
Feature Vector $\mathbf{\Phi}(X,W)$	-2602.62	1	3	$\cdots$	0	2	$\cdots$	1	0
Parameter Vector of DLM $\boldsymbol{\alpha}$	1	0.01	0.12	$\cdots$	-0.25	-0.03	$\cdots$	0.78	0.52

B. Roark et al., "Discriminative n-gram language modeling," *Computer Speech and Language, 21, 2007.*

Discriminative Language Modeling

- Schematic Illustration





Discriminative Language Modeling

- Training of a DLM model
  - Fulfilled by finding a parameter vector $\boldsymbol{\alpha}$ that minimizes a loss function having to do with the margin between the score of the reference transcript W_i^R and that of any other hypothesis W_i for each training utterance X_i

The Training Objectives of Various DLM Methods

Methods	Training Objectives
Perceptron	$F_{Perc}(\boldsymbol{\alpha}) = \frac{1}{2}\sum_{i=1}^{L}\left(\left(\boldsymbol{\Phi}(X_i, W_i^R) - \boldsymbol{\Phi}(X_i, W_i^*)\right)\bullet\boldsymbol{\alpha}\right)$
GCLM	$F_{GCLM}(\boldsymbol{\lambda}) = -\sum_{i=1}^{L}\log\frac{\exp\left(\boldsymbol{\Phi}\left(X_i, W_i^R\right)\bullet\boldsymbol{\alpha}\right)}{\sum_{W_i\in\mathbf{GEN}(X_i)}\exp\left(\boldsymbol{\Phi}\left(X_i, W_i\right)\bullet\boldsymbol{\alpha}\right)}$
WGCLM	$F_{WGCLM}(\boldsymbol{\lambda}) = -\sum_{i=1}^{L}\log\frac{\exp\left(\boldsymbol{\Phi}\left(X_i, W_i^R\right)\bullet\boldsymbol{\alpha}\right)}{\sum_{W_i\in\mathbf{GEN}(X_i)}\omega_{i,W_i}\exp\left(\boldsymbol{\Phi}\left(X_i, W_i\right)\bullet\boldsymbol{\alpha}\right)}$
MERT	$F_{MERT}(\boldsymbol{\lambda}) = \sum_{i=1}^{L}\sum_{W_i\in\mathbf{GEN}(X_i)}\frac{\varpi_{i,W_i}\exp\left(\boldsymbol{\Phi}\left(X_i, W_i\right)\bullet\boldsymbol{\alpha}\right)^{\beta}}{\sum_{W_s\in\mathbf{GEN}(X_i)}\exp\left(\boldsymbol{\Phi}\left(X_i, W_s\right)\bullet\boldsymbol{\alpha}\right)^{\beta}}$

1. B. Chen, J.-W. Liu, "Discriminative language modeling for speech recognition with relevance information," *ICME*, 2011
2. M.-H. Lai et al., "Empirical comparisons of various discriminative language models for speech recognition," *ROCLING*, 2011

DLM for Speech Summarization

- A global conditional log-linear model (GCLM) is used to establish the speech summarizer
  - GCLM will give a decision score to an arbitrary sentence S_i of a spoken document D_n to be summarized according to the posterior probability which is approximated by

$$P_{\text{GCLM}}(S_i|D_n) = \frac{\exp(X_i \bullet \zeta)}{\sum_{l=1}^{L_n} \exp(X_l \bullet \zeta)}$$

X_i is the M-dimensional feature vector of S_i
ζ is the M-dimensional parameter vector of GCLM
$X_i \bullet \zeta$ is the inner product of X_i and ζ
L_n is the total number of sentences in D_n

- Training objectives

$$F_{\text{GCLM -I}} = \sum_{n=1}^{N} \sum_{S_i \in \mathbf{Summ}_n} \log \frac{P_{\text{GCLM}}(S_i|D_n)}{\sum_{l=1}^{L_n} (1 - e(S_l, \mathbf{Summ}_n)) P_{\text{GCLM}}(S_l|D_n)}$$

$$F_{\text{GCLM-II}} = \sum_{n=1}^{N} \sum_{l=1}^{L_n} e(S_l, \mathbf{Summ}_n) P_{\text{GCLM}}(S_i|d_n)$$

B. Chen et al., "Extractive speech summarization using evaluation metric-related training criteria," Information Processing & Management, Vol. 49, No. 1, pp. 1-12, January 2013.

DLM for Speech Summarization

- Features X_i used to represent the sentences of a spoken document to summarized

SET 1
(**raw** features)

SET 2
(**more elaborate** features produced by unsupervised models)

Types	Description
Structural feature	1. Duration of the current sentence (S1)
Lexical features	1. Number of named entities (L1)
	2. Number of stop words (L2)
	3. Bigram language model scores (L3)
	4. Normalized bigram scores (L4)
Acoustic features	1. The 1st formant (F1-1 to F1-5)
	2. The 2nd formant (F2-1 to F2-5)
	3. The pitch value (P-1 to P-5)
	4. The peak normalized cross-correlation of pitch (C-1 to C-5)
Relevance features	1. Relevance score obtained by WTM
	2. Relevance score obtained by VSM
	3. Relevance score obtained by LSA
	4. Relevance score obtained by MRW

- Performance Evaluations (with erroneous speech transcripts)

		ROUGE-1	ROUGE-2	ROUGE-L
All	SVM	0.427	0.269	0.398
	Ranking SVM	0.449	0.283	0.418
	AdaRank	0.459	0.303	0.432
		(0.462)	(0.303)	(0.432)
	GCLM-I	0.477	0.325	0.451
	GCLM-II	0.456	0.294	0.425
SET 1	SVM	0.376	0.228	0.353
	Ranking SVM	0.407	0.243	0.380
	AdaRank	0.378	0.237	0.362
		(0.409)	(0.237)	(0.409)
	GCLM-I	0.408	0.264	0.390
	GCLM-II	0.401	0.247	0.377
SET 2	SVM	0.346	0.180	0.316
	Ranking SVM	0.417	0.255	0.380
	AdaRank	0.438	0.273	0.403
		(0.438)	(0.273)	(0.403)
	GCLM-I	0.429	0.262	0.398
	GCLM-II	0.431	0.266	0.396

(comparisons among various models)

The levels of agreement between the three subjects for important sentence ranking (10% summarization ratio) for the evaluation set.

	ROUGE-1	ROUGE-2	ROUGE-L
Agreement	0.675	0.645	0.631

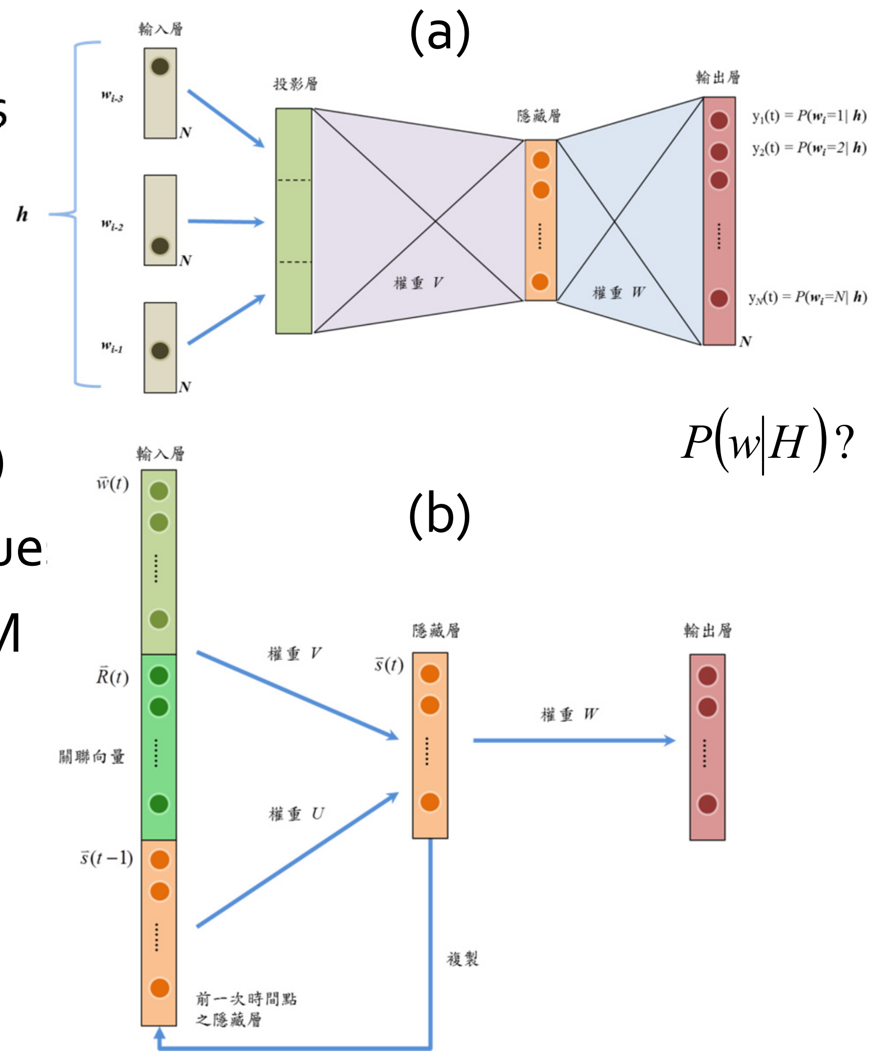
(the gold standard)

Outline

- Introduction (*n*-gram)
- Topic Modeling (LSA, NMF, PLSA, LDA, WTM)
- Discriminative Language Modeling
- **Neural Network Language Modeling**
- Relevance Language Modeling
- Positional Language Modeling
- Conclusions

Neural Network Language Modeling (NNLM)

- Schematic Illustrations
  - (a) Feed-forward neural networks
  - (b) Recurrent neural networks
- Research Issues
  - Encoding of words (and history)
  - Leveraging extra information cue
  - Discriminative training of NNLM
  - Exploring "**deep**" neural networks (DNN)



$P(w|H)$?

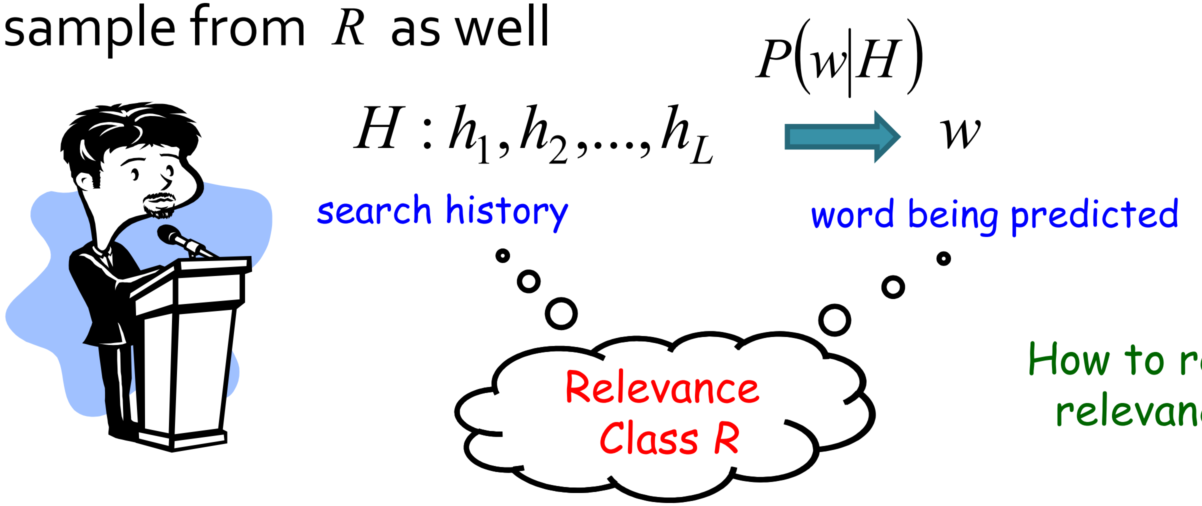
1. T. Mikolov et al., "Recurrent neural network based language model," *Interspeech 2010*
2. G. Hinton et al., "Deep Neural Networks for Acoustic Modeling in Speech Recognition- The Shared Views of Four Research Groups," *IEEE Signal Processing Magazine, 29(6), pp. 82-97, November 2012*

Outline

- Introduction (*n*-gram)
- Topic Modeling (LSA, NMF, PLSA, LDA, WTM)
- Discriminative Language Modeling
- Neural Network Language Modeling
- **Relevance Language Modeling**
- Positional Language Modeling
- Conclusions

Relevance Modeling (RM)

- Investigate a novel use of relevance information cues to dynamically complement (or adapt) the conventional *n*-gram models, assuming that
  - During speech recognition, a search history $H = h_1, h_2, \dots, h_L$ is a sample from a relevance class R describing some semantic content
  - Assume that a probable word w that immediately succeeds H is a sample from R as well



How to represent the relevance class *R* ?

B. Chen and K.-Y. Chen, "Leveraging relevance cues for language modeling in speech recognition," *Information Processing & Management*, 49 (4), pp. 807-816, July 2013.

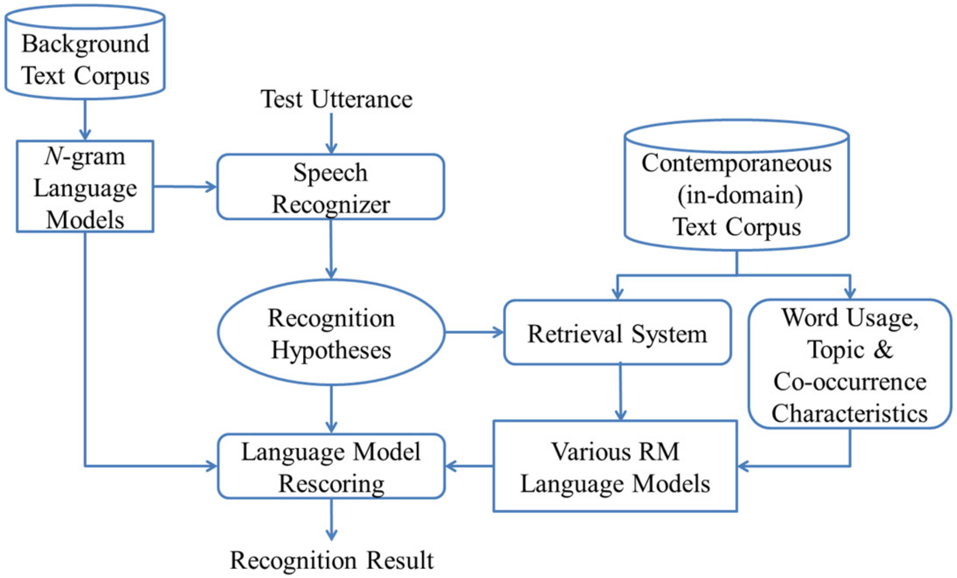
Relevance Modeling

- Leverage the top-*M* relevant documents of the search history to approximate the relevance class R
  - Take H as a query to retrieve relevant documents
  - **R**elevance **M**odel: Multinomial view (*bag-of-words modeling*) of R

$$P_{\mathrm{RM}}(w|H)=\frac{P_{\mathrm{RM}}(H,w)}{P_{\mathrm{RM}}(H)}$$

$$=\frac{\sum_{m=1}^{M}P(D_m)P(H,w\mid D_m)}{\sum_{m=1}^{M}P(D_m)P(H\mid D_m)}$$

$$=\frac{\sum_{m=1}^{M}P(D_m)P(w\mid D_m)\prod_{l=1}^{L}P(h_l\mid D_m)}{\sum_{m=1}^{M}P(D_m)\prod_{l=1}^{L}P(h_l\mid D_m)}$$



$$P_{\mathrm{Adapt}}(w|H)=\lambda\cdot P_{\mathrm{RM}}(w|H)+(1-\lambda)\cdot P_{\mathrm{BG}}(w|h_{L-1},h_L)$$

Variants of RM

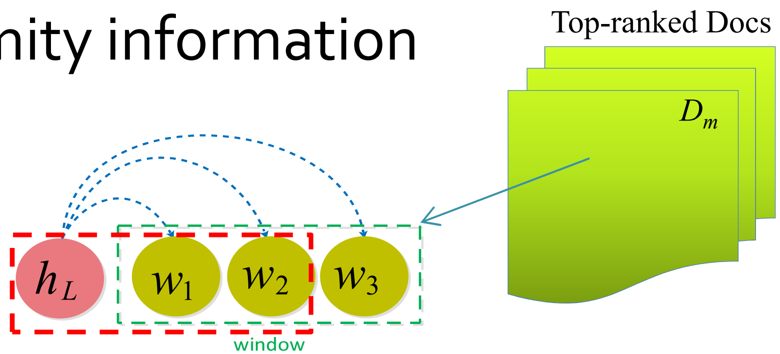
- Further incorporation of latent topic information
  - A shared set of latent topic variables $\{T_1, T_2, \ldots, T_K\}$ is used to describe "*word-document*" co-occurrence characteristics

$$P(w \mid D_m) = \sum_{k=1}^{K} P(w \mid T_k) P(T_k \mid D_m)$$

$$P_{\text{TRM}}(H, w) = \sum_{m=1}^{M} \sum_{k=1}^{K} P(D_m) P(T_k \mid D_m) P(w \mid T_k) \prod_{l=1}^{L} P(h_l \mid T_k)$$

- Further incorporation of proximity information

$$P(w \mid h_L, D_m) = \frac{C_\tau(h_{L,}, w, D_m)}{\sum_{w'} C_\tau(h_{L,}, w', D_m)}$$



$$P_{\text{PRM}}(H, w) = \sum_{m=1}^{M} P(D_m) P(h_1 \mid D_m) \left[\prod_{l=2}^{L} P(h_l \mid h_{l-1}, D_m) \right] P(w \mid h_L, D_m)$$

Y.-W. Chen et al., "Incorporating proximity information for relevance language modeling in speech recognition," *the 14th Annual Conference of the International Speech Communication Association (Interspeech 2013)*, Lyon, France, August 25-29, 2013.

RM: ASR Evaluations

- Tested on a large vocabulary broadcast new recognition task
  - Character error rate (CER) results (the lower the better)

Baseline Trigram	RM	PLSA	LDA	TBLM	RNNLM	DLM (MERT)	DLM (GCLM)	DLM (WGCLM)
20.22	19.21	19.28	19.22	20.09	19.10	19.74	19.89	19.62

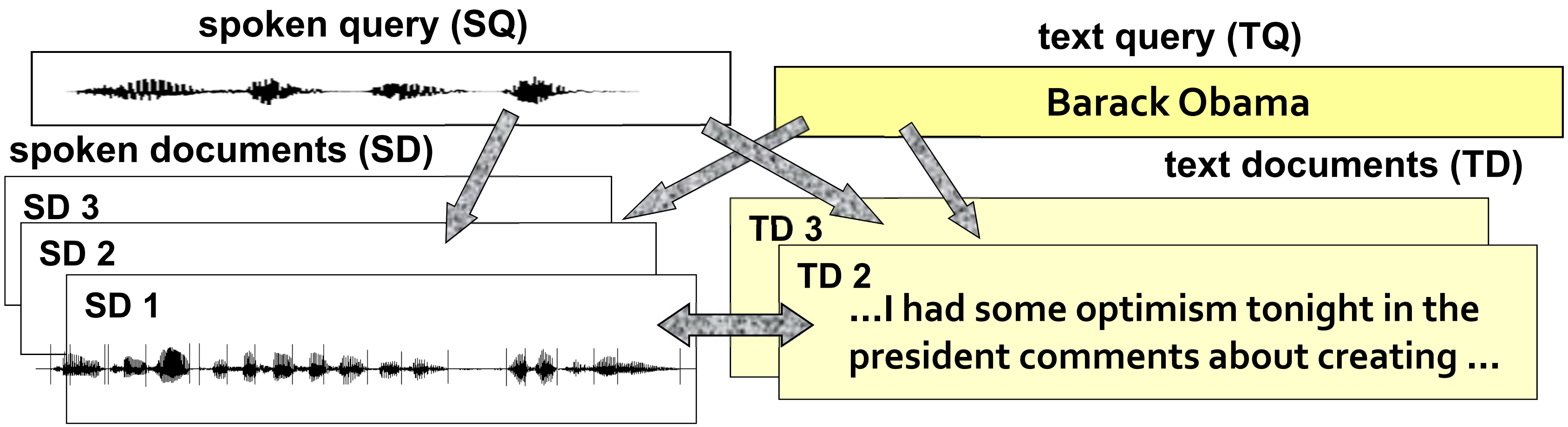
PRM (τ =2)	PRM (τ =3)	PRM (τ =4)	PRM (τ =5)	PRM (τ =6)
18.91	18.89	18.97	18.98	19.07

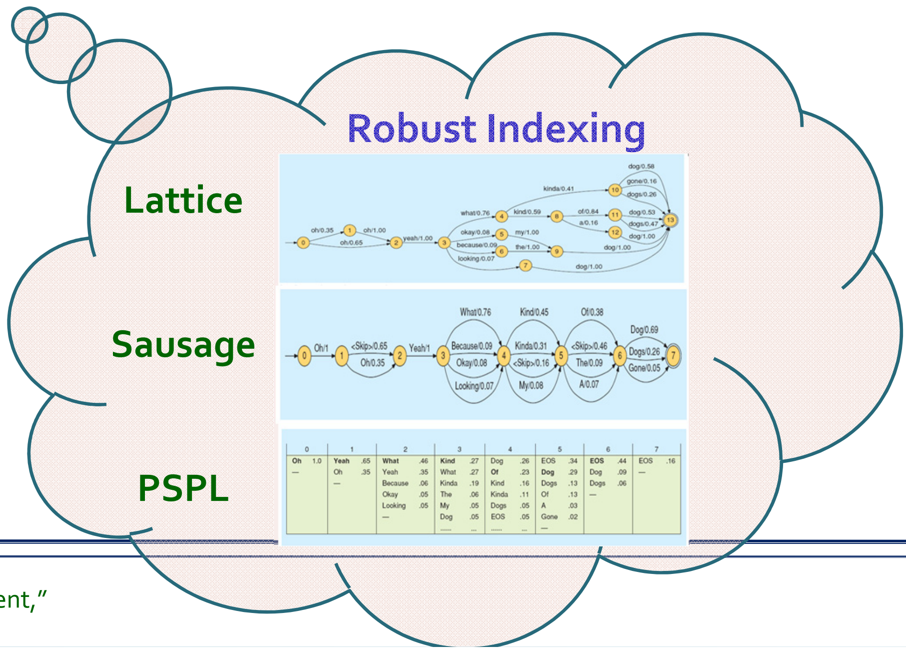
TRM	P-RM (τ =3) + TRM
19.18	18.84

  - The various RM models have been shown to be on par with, or even better than, PLSA, LDA (topic models), RNN and DLM
  - However, the "oracle" CER for the ASR word graphs of this task is 7.72 (something is still missing for language modeling)

Spoken Document Retrieval (SDR)

- Scenarios

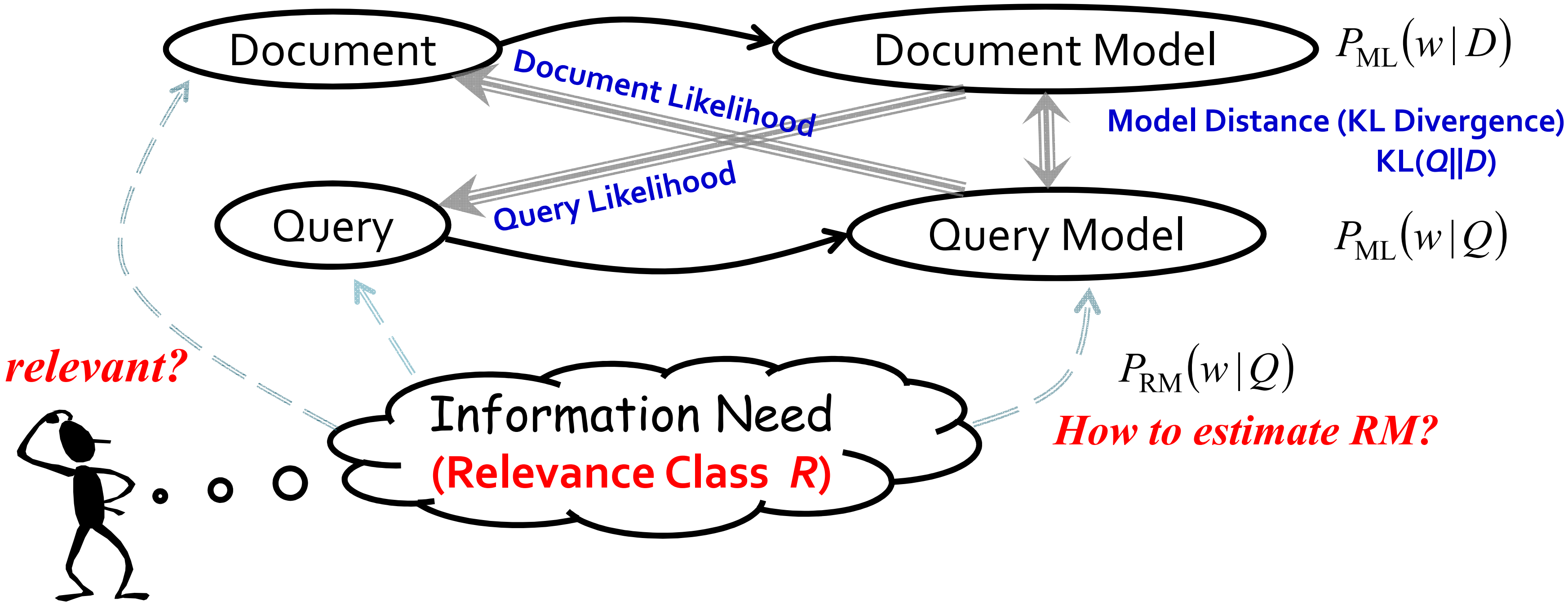




C. Chelba, T.J. Hazen, and M. Saraclar,
"Retrieval and browsing of spoken content,"
IEEE Signal Processing Magazine , 2008

Language Modeling for SDR (or IR)

- Schematic Illustration



1. C.X. Zhai, Statistical Language Models for Information Retrieval (Synthesis Lectures Series on Human Language Technologies), Morgan & Claypool Publishers, 2008.
2. B. Chen et al., "Spoken document retrieval with unsupervised query modeling techniques," *IEEE Transactions on Audio, Speech and Language Processing*, 20(9), 2012.

Kullback-Leibler (KL) Divergence

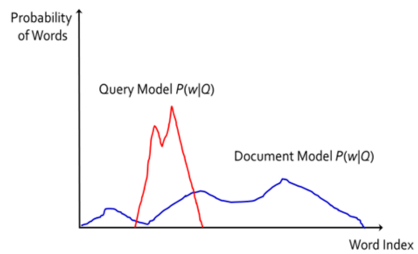
- KL-divergence measures the model distance between two probabilistic models (the smaller the more similar/relevant)
  - For example, in the context of information retrieval, we construct a query model (*Q*) and several document models (*D*)

$$KL(Q\|D)=\sum_w P(w|Q)\log\frac{P(w|Q)}{P(w|D)}$$

Query model

Document model

$$=\boxed{\sum_w P(w|Q)\log P(w|Q)}-\boxed{\sum_w P(w|Q)\log P(w|D)}$$



Negative entropy of the query model : the same for all document => can be disregarded

Cross entropy between the language models of a query and a document

Equivalent to ranking **in decreasing order** of

$$\sum_w P(w|Q)\log P(w|D)$$

Relevant documents are deemed to have lower cross entropies

$$\overset{\text{rank}}{=}\sum_w c(w,Q)\log P(w|D)=P(Q|D)$$

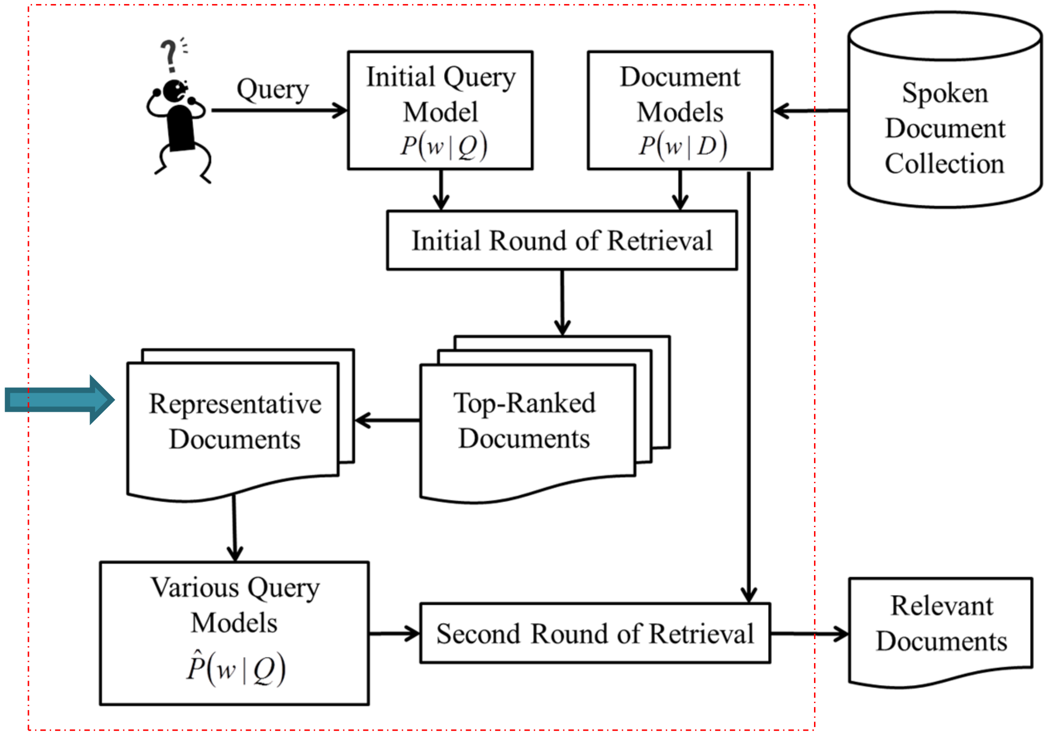
Query Likelihood Measure

S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, 22(1), pp. 79-86, 1951.

Effective Pseudo-relevance Feedback (PRF)

- How to effectively glean useful cues from the top-ranked documents so as to achieve more accurate relevance (query) modeling?

Considering **relevance**, **non-relevance**, **diversity** and **density** cues



$$D^* = \underset{D \in \mathbf{D}_{\mathrm{Top}} - \mathbf{D}_{\mathrm{P}}}{\arg\max} \left[(1-\alpha-\beta-\gamma) \cdot M_{Rel}(Q,D) + \alpha \cdot M_{NR}(Q,D) + \beta \cdot M_{Diversity}(D) + \gamma \cdot M_{Density}(D) \right]$$

Y.-W. Chen et al., "Effective pseudo-relevance feedback for spoken document retrieval," *the 38th IEEE International Conference on Acoustics, Speech, and Signal Processing* (ICASSP 2013), Vancouver, Canada, May 26-31, 2013.

Leveraging Indicative Cues for Effective PRF

- Relevance

$$M_{Rel}(Q,D) = -KL(Q \| D)$$
$$= -\sum_{w \in V} P(w|Q)\log\frac{P(w|Q)}{P(w|D)}$$
$$\stackrel{\text{rank}}{=} \sum_{w \in V} P(w|Q)\log P(w|D)$$

- Diversity

$$M_{Diversity}(D)$$
$$= \min_{D_j \in \mathbf{D}_\mathrm{P}} \frac{1}{2}\cdot\left[KL(D_j \| D) + KL(D \| D_j)\right]$$

- Non-relevance

$$M_{NR}(D) = KL(NR_Q \| D)$$
$$\cong -\sum_{w \in V} P(w|Collection)\log\frac{P(w|Collection)}{P(w|D)}$$

- Density

$$M_{Density}(D)$$
$$= \frac{-1}{\left|\mathbf{D}_\mathrm{Top}\right|-1}\cdot\sum_{\substack{D_h \in \mathbf{D}_\mathrm{Top} \\ D_h \neq D}}\left[KL(D_h \| D) + KL(D \| D_h)\right]$$

Query Reformulation with Effective PRF for SDR

- MAP Results on TDT-2 Spoken Document Collection
  - Baseline

	ULM	PLSA	LDA	RM	TRM	SMM
TD	0.371	0.418	0.401	0.421	0.456	0.415
SD	0.323	0.435	0.341	0.369	0.397	0.361

(the higher the value the better performance)

  - Simply use Top *N* documents for query reformulation

		RM	TRM	SMM
TD	Top 5	0.405	0.440	0.438
	Top 10	0.417	0.452	0.483
	Top 15	0.421	0.455	0.468
	Top 25	0.421	0.456	0.415
	Top 30	0.421	0.457	0.411
SD	Top 5	0.369	0.396	0.399
	Top 10	0.372	0.398	0.398
	Top 15	0.370	0.397	0.367
	Top 25	0.369	0.397	0.361
	Top 30	0.369	0.396	0.360

  - Use 5 "specially selected" documents for query reformulation

		RM	TRM	SMM
TD	Gapped	0.414	0.452	0.406
	Cluster	0.396	0.441	0.380
	Active-RDD	0.471	0.492	0.457
	Our Method	0.491	0.507	0.490
	Our Method +TW	0.523	0.522	0.496
SD	Gapped	0.357	0.391	0.333
	Cluster	0.378	0.395	0.325
	Active-RDD	0.437	0.461	0.403
	Our Method	0.448	0.475	0.424
	Our Method +TW	0.485	0.494	0.435

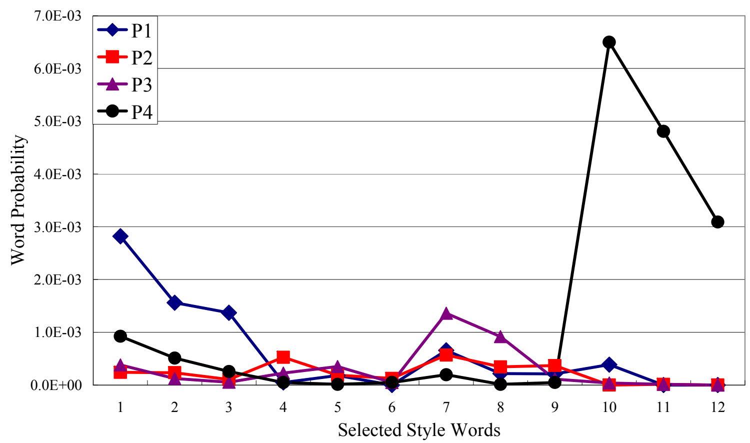
Outline

- Introduction (*n*-gram)
- Topic Modeling (LSA, NMF, PLSA, LDA, WTM)
- Discriminative Language Modeling
- Neural Network Language Modeling
- Relevance Language Modeling
- **Positional Language Modeling**
- Conclusions

Positional Language Modeling

- Are there any other alternatives beyond the above LMs?
- The table below shows the style words with higher rank of *TF-IDF* scores on four partitions of the broadcast news corpus
  - The corpus was partitioned by a left-to-right HMM segmenter

P1	P2	P3	P4
1繼續 Continue	4醫師 Doctor	7學生 Student	10公視 TV station name
2現場 Locale	5網路 Internet	8老師 Teacher	11綜合報導 Roundup
3歡迎 Welcome	6珊瑚 Coral	9酒 Rice wine	12編譯 Edit and translate



H.-S. Chiu et al., "Leveraging topical and positional cues for language modeling in speech recognition," *Multimedia Tools and Applications*, Published online: 19 April 2013.

Positional Language Modeling

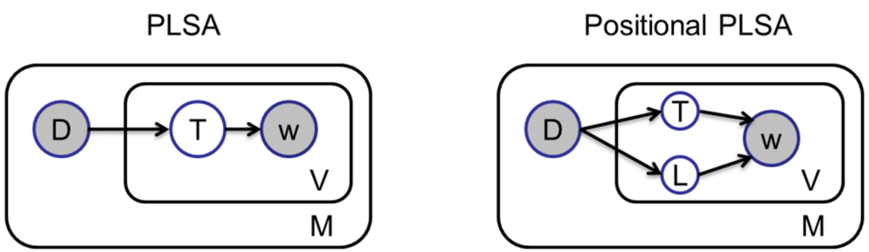
- Positional *n*-gram Model

$$P_{POS}(w_i \mid w_{i-2}, w_{i-1}) = \sum_{s=1}^{S} \alpha_s P(w_i \mid w_{i-2}, w_{i-1}, L_s)$$

  - Where S is the number of partitions, α_S is the weight for a specific position L_S

- Positional PLSA (Probabilistic Latent Semantic) Model

$$P_{PosPLSA}(w_i \mid H) = \sum_{s=1}^{S} \sum_{k=1}^{K} P(w_i \mid T_k, L_s) P(L_s \mid H) P(T_k \mid H)$$



Graphical Model Representations

Conclusions

- Various language modeling approaches have been proposed and extensively investigated in the past decade, showing varying degrees of success in a wide array of applications (cross-fertilization between speech, NLP and IR communities)
- Modeling and computation are intertwined in developing new language models ("simple" is "elegant"?)
- "*Put language back into language modeling*" remains an important issue that awaits further studies (our ultimate goal?)
- "*Automatic Speech Recognition then Understanding (ASRU)*" or "*Automatic Speech Understanding then Recognition (ASUR)*" ?
  - We start out to investigate "Concept Language Modeling"

D. Blei, "Probabilistic topic models," *Communications of the ACM*, 55(4):77–84, 2012.

Thank You!