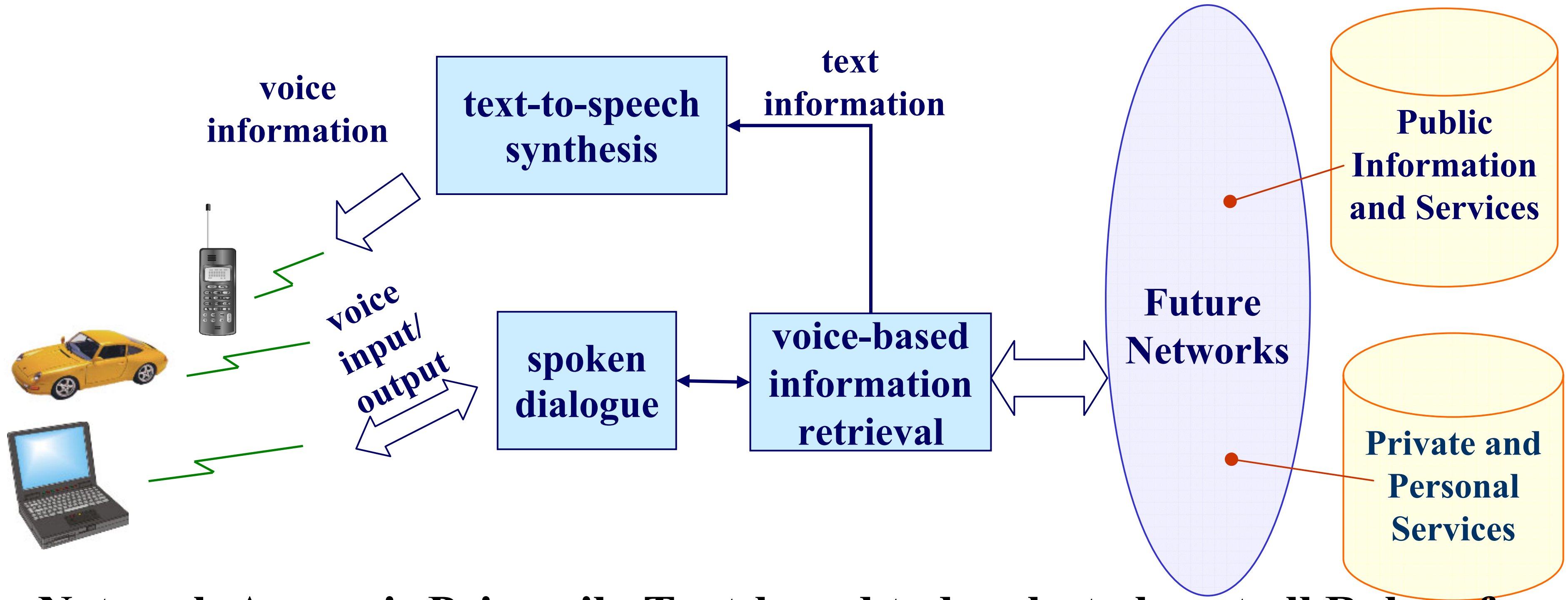
13.0 Speech-based Information Retrieval

References: 1. “ Speech and Language Techniques for Audio Indexing and Retrieval ”, Proceedings of the IEEE, Aug 2000

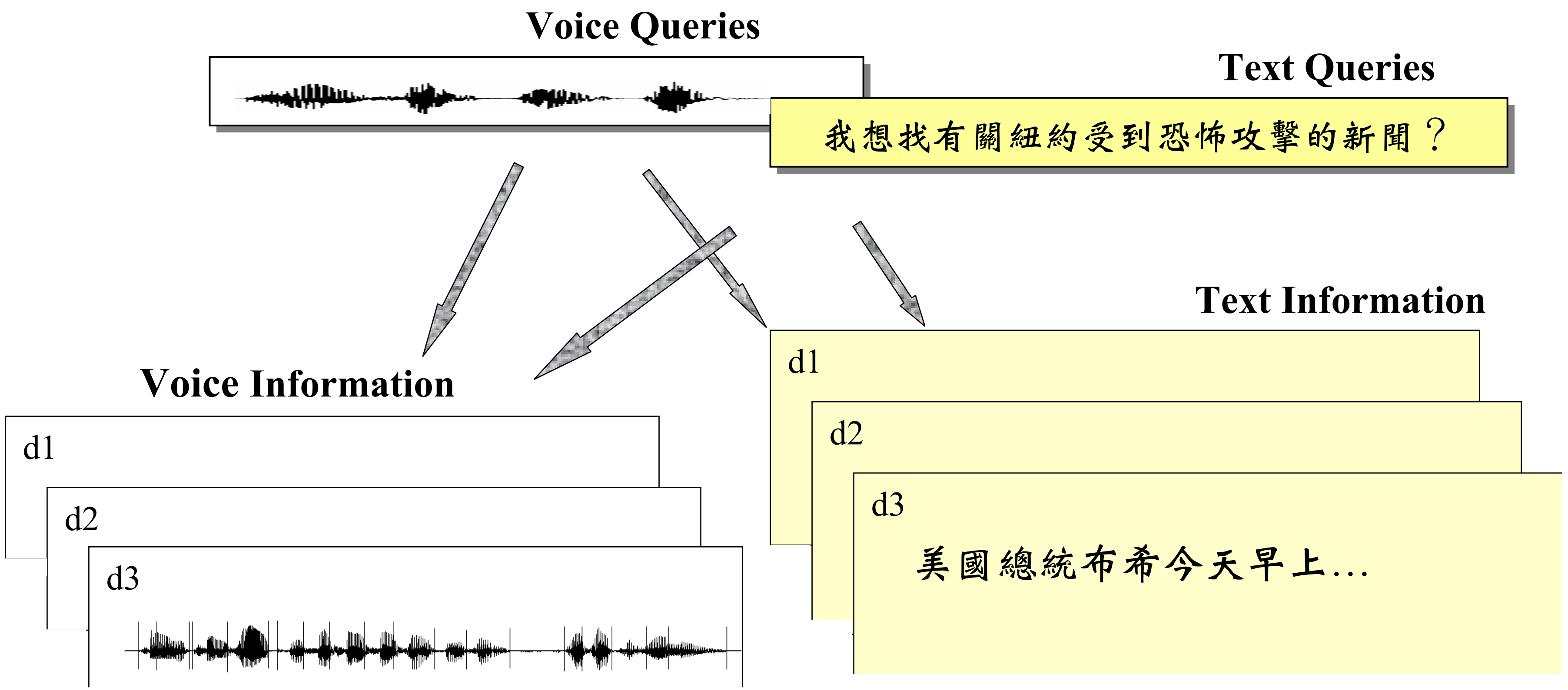
2. Baeza-Yates & Ribeiro Neto, “ Modern Information Retrieval”, ACM Press, 1999
3. ACM Special Interest Group on Information Retrieval, http://www.acm.org/sigir
4. “ A Hidden Markov Model Information Retrieval System”, ACM SIGIR, 1999
5. “ Probabilistic Latent Semantic Indexing”, ACM SIGIR, 1999

Voice –enabled Web-based Applications



- Network Access is Primarily Text-based today, but almost all Roles of Texts can be Replaced by Voice in the Future
- Human-Network Interactions can be Accomplished by Spoken Dialogues
- Voice-based Information Retrieval needs to be integrated with Spoken Dialogues
- More Multi-media Information including Voice but not including Enough Text will be Available on the Web in the Future

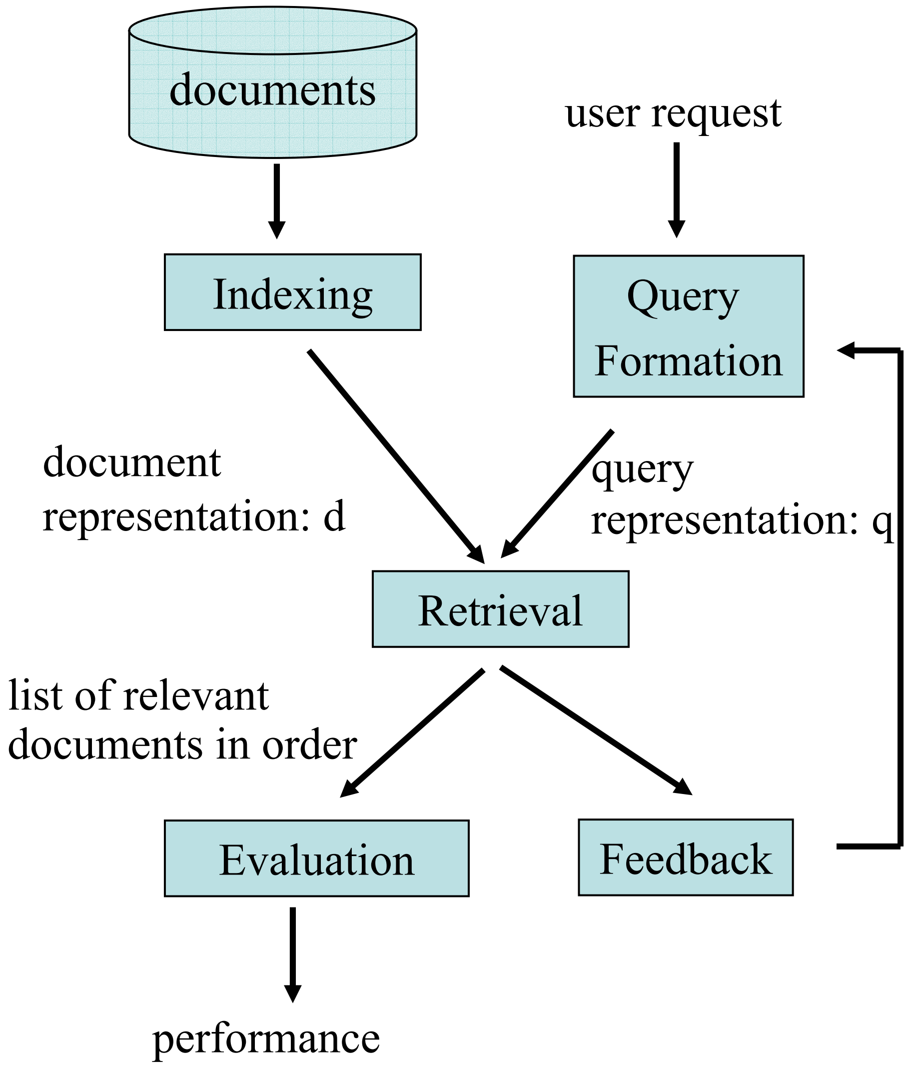
Speech-based Information Retrieval



- •Speech/Text Queries, Speech/Text Documents
- •Mobile/Office User Environments with Multi-modality
- •Speech may become a New Data Type, if the Difficulties in Browsing and Retrieval can be Overcome
- •Speech Provides Better User Interface in Wireless Environment

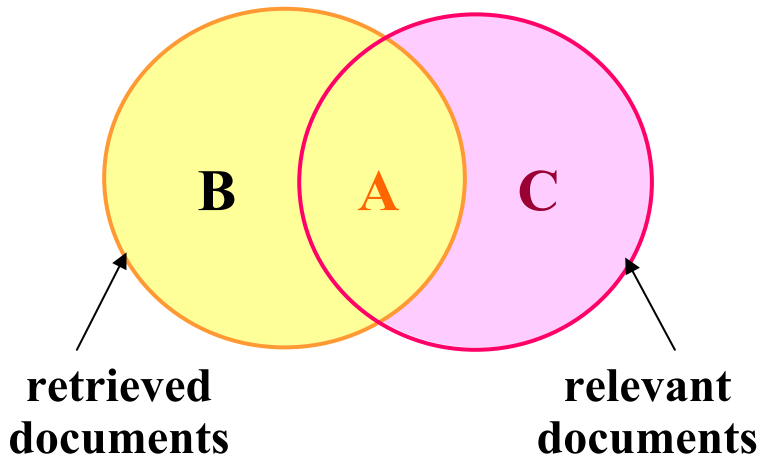
Information Retrieval Processes

- **Indexing**
  - Document representation :d
- **Query formation**
  - User request representation :q
- **Retrieval**
  - Matching query to documents
  - Returning relevant documents
- **Relevance feedback**
  - Assessing retrieved results
  - Modifying initial query
  - Iterated retrieval: automatic (blind)/manual
- **Performance evaluation**
  - Performance measure



Performance Measures

- **Recall and Precision Rates**



$$\textbf{Precision rate} = \frac{A}{A+B}$$

$$\textbf{Recall rate} = \frac{A}{A+C}$$

- similar to missing/false alarm rates
- recall-precision plot similar to ROC curves
- recall rate may be difficult to evaluate, while precision rate is directly perceived by users

- **Non-Interpolated Average Precision**
  - Averaged at all relevant documents retrieved and over all queries
  - e.g. relevant documents ranked at 1, 5, 10, precisions are 1/1, 2/5, 3/10, non-interpolated average precision=(1/1+2/5+3/10)/3

Approaches to Speech-based Information Retrieval

- **Indexing Elements**
  - Words: Large-vocabulary Based
    - create text transcription of spoken documents/queries by speech recognition
    - use text retrieval methods
    - error propagation, out-of-vocabulary (OOV) problems, special terms
  - Subword Units: Subword Based
    - subword units: phones/syllables/something similar
    - a segment of one to a few subword units may carry some indexing information
    - not limited by the vocabulary
    - small size/handling some OOV/probably more ambiguity
  - Keywords: Keyword Based
    - based on a set of keywords
    - keyword selection: user specify/a prior/fixed/automatic generated
    - special terms for dynamic documents
  - Hybrid: Fusion of Information
- **Indexing Features**
  - a single element
  - different combinations of more than one elements
  - pre-defined, or automatically selected by data-driven approaches
  - each of such features is called an "indexing term"
- **Retrieval Model Examples**
  - vector space models
  - latent semantic indexing (LSI)
  - statistical (probabilistic) models
  - hidden Marcov model (HMM)
  - combinations/hybrid models

Vector Space Model

- **Vector Representations of query q and document d**
  - for each type j of indexing feature a vector is generated
  - each component in this vector is the weighted statistics z_{jt} of a specific indexing term t

$$z_{jt} = \underbrace{\left(1 + \ln[c_t]\right)}_{\text{Term Frequency (TF)}} \cdot \underbrace{\ln\left(N / N_t\right)}_{\text{Inverse Document Frequency (IDF)}}$$

c_t: frequency counts for the indexing term t present in the query q or document d (for text), or sum of normalized recognition scores or confidence measures for the indexing term t (for speech)

N: total number of documents in the database

N_t: total number of documents in the database which include the indexing term t

IDF: the significance (or importance) or indexing power for the indexing term t

- **The Overall Relevance Score is the Weighted Sum of the Relevance Scores for all Types of Indexing Features**

$$R_j(\vec{q}_j, \vec{d}_j) = \left(\vec{q}_j \bullet \vec{d}_j\right) \Big/ \left(\left\|\vec{q}_j\right\| \cdot \left\|\vec{d}_j\right\|\right)$$

$\vec{q}_j, \vec{d}_j$: vector representations for query q and document d with type j of indexing feature

$$R(q,d) = \sum_j w_j \cdot R_j(\vec{q}_j, \vec{d}_j)$$

w_j : weighting coefficients

Improved Retrieval Technique Examples

- **Blind Relevance Feedback**
  - the information from the relevant and irrelevant documents retrieved in the previous stage used to identify more helpful indexing terms
  - the initial query is reformulated accordingly:

    $\vec{q}' = \alpha \cdot \vec{q} + \beta \cdot \sum_{\mathbf{D_r}} \vec{d} - \gamma \cdot \sum_{\mathbf{D_{irr}}} \vec{d}$

    $\vec{q}$, $\vec{d}$: vector representation for the query and documents

    D_r : selected set of relevant documents retrieved in the previous stage

    D_{irr}: selected set of irrelevant documents deleted in the previous stage

    $\vec{q}'$: new query representation

    α,β,γ: weighting coefficients
- **Query Expansion by Term Association**
  - the indexing terms co-occurring frequently in the same documents assumed to have some synonymity association
  - build an association matrix for each type of the indexing features, in which each entry *(i , j)* stands for the association between indexing terms t_i and t_j :

    $$A(i,j) = \frac{\hat{f}_{i,j}}{f_i + f_j - \hat{f}_{i,j}} \quad \text{as an example}, \quad 0 \le A(i,j) \le 1$$

    f_i, f_j : number of documents in the database including the indexing terms t_i, t_j

    $\hat{f}_{i,j}$: number of documents in the database including both indexing terms t_i and t_j
  - reformulate the query expression by adding indexing terms with higher synonymity

Difficulties in Speech-based Information Retrieval for Chinese Language

- **Even for Text-based Information Retrieval, Flexible Wording Structure Makes it Difficult to Search by Comparing the Character Strings Alone**
  - name/title 李登輝→李前總統登輝，李前主席登輝(President T.H Lee)
  - arbitrary abbreviation 北二高→北部第二高速公路(Second Northern Freeway)
  - similar phrases 中華文化→中國文化(Chinese culture)
  - translated terms 巴塞隆那→巴瑟隆納(Barcelona)
- **Word Segmentation Ambiguity Even for Text-based Information Retrieval**
  - 腦科(human brain studies) →電腦科學(computer science)
  - 土地公(God of earth) →土地公有政策(policy of public sharing of the land)
- **Uncertainties in Speech Recognition**
  - errors (deletion, substitution, insertion)
  - out of vocabulary (OOV) words, etc.
  - very often the key phrases for retrieval are OOV

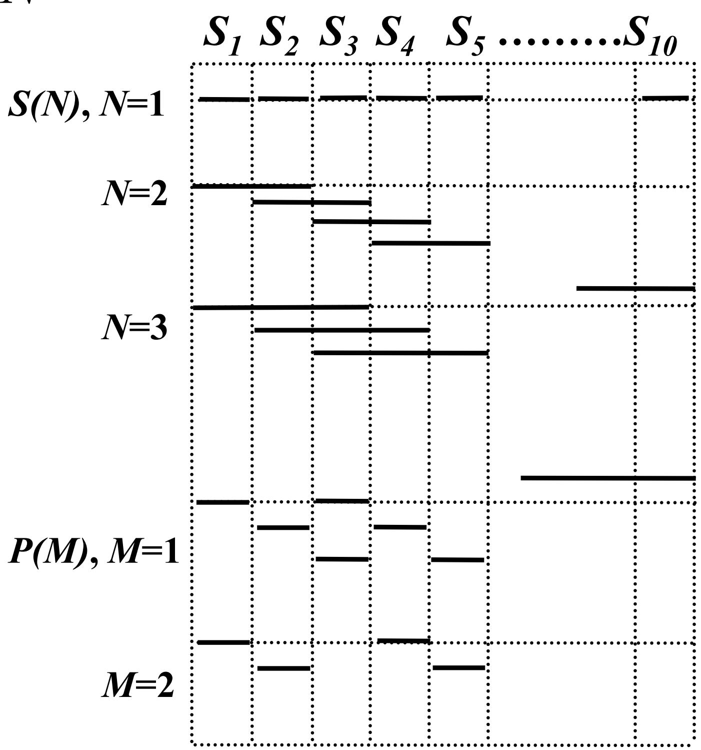
Syllable-Level Indexing Features for Chinese Language

- **A Whole Class of Syllable-Level Indexing Features with Complete Phonological Coverage and Better Discriminating Functions**
  - Overlapping syllable segments with length *N*

Syllable Segments	*Examples*
S(N), N=1	(s_1) (s_2) …(s_{10})
S(N), N=2	$(s_1\ s_2)$ $(s_2\ s_3)$…$(s_9\ s_{10})$
S(N), N=3	$(s_1\ s_2\ s_3)$ $(s_2\ s_3\ s_4)$…$(s_8\ s_9\ s_{10})$
S(N), N=4	$(s_1\ s_2\ s_3\ s_4)$ $(s_2\ s_3\ s_4\ s_5)$…$(s_7\ s_8\ s_9\ s_{10})$
S(N), N=5	$(s_1\ s_2\ s_3\ s_4\ s_5)$ $(s_2\ s_3\ s_4\ s_5\ s_6)$…$(s_6\ s_7\ s_8\ s_9\ s_{10})$

  - Syllable pairs separated by *M* syllables

Syllable Pair Separated by M syllables	*Examples*
P(M), M=1	$(s_1\ s_3)$ $(s_2\ s_4)$……$(s_8\ s_{10})$
P(M), M=2	$(s_1\ s_4)$ $(s_2\ s_5)$……$(s_7\ s_{10})$
P(M), M=3	$(s_1\ s_5)$ $(s_2\ s_6)$……$(s_6\ s_{10})$
P(M), M=4	$(s_1\ s_6)$ $(s_2\ s_7)$……$(s_5\ s_{10})$



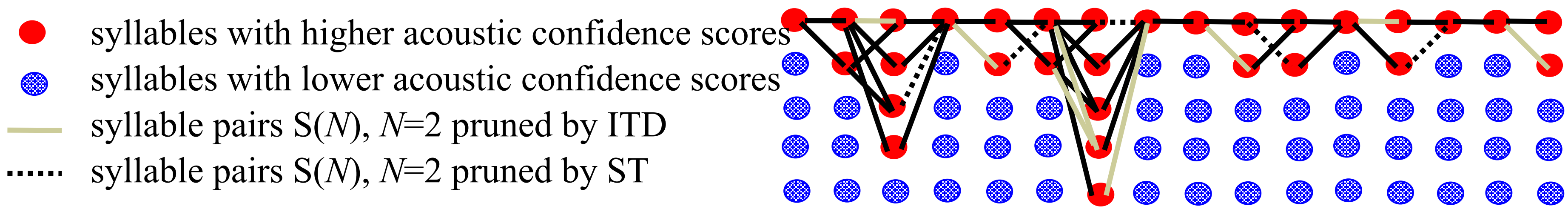
- **Character- or Word-Level Features can be Similarly Defined**

Syllable-Level Statistical Features

- **Singe Syllables**
  - each syllable usually shared by more than one characters with different meanings, thus causing ambiguity
  - all words are composed by syllables, thus partially handle OOV problem
  - very often relevant words have some syllables in common
- **Overlapping Syllable Segments with Length *N***
  - capturing the information of polysyllabic words or phrases with flexible wording structures
  - majority of Chinese words are bi-syllabic
  - not too many polysyllabic words share the same pronunciation
- **Syllable Pairs Separated by *M* Syllables**
  - tackling the problems arising from the flexible wording structure, abbreviations, and deletion, insertion, substitution errors in speech recognition

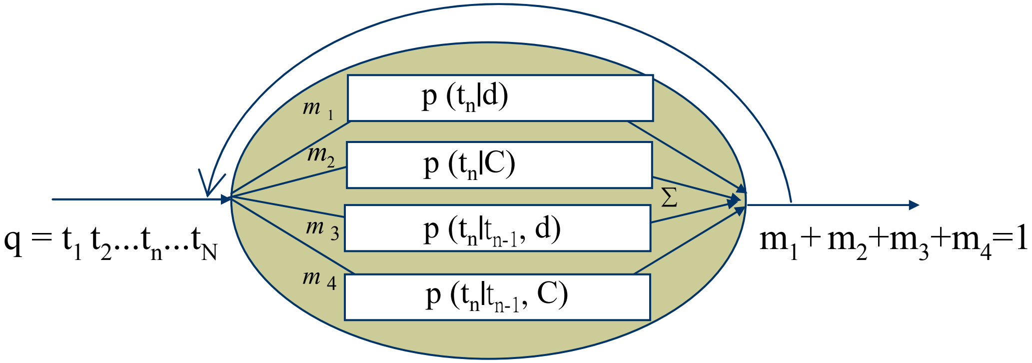
Improved Syllable-level Indexing Features

- **Syllable Lattice and syllable-level utterance verification**
  - Including multiple syllable hypothesis to construct syllable-aligned lattices for both query and documents
  - Generating multiple syllable-level indexing features from syllable lattices
  - filtering out indexing terms with lower acoustic confidence scores
- **Infrequent term deletion (ITD)**
  - Syllable-level statistics trained with text corpus used to prune infrequent indexing terms
- **Stop terms (ST)**
  - Indexing terms with the lowest IDF scores are taken as the stop terms



Hidden Markov Model (HMM) for Speech-based Information Retrieval

- **Modeling the Query q as a Sequence of Input Observations (Indexing Terms), $q=t_1t_2...t_n...t_N$, and each Document d as a HMM (1-state at the moment) Composed of Distributions of N-gram Parameters**



$P(t_n|d)$, $p(t_n|t_{n-1},d)$
unigram/bi-gram trained from the document d
$P(t_n|C)$, $p(t_n|t_{n-1},C)$
unigram/bi-gram trained from a large corpus, specially helpful for missing terms in the documents

- **MAP Principle (as a simple example)**

$$d^* = \arg\max_d \text{Prob}(d \text{ is } R|q) = \arg\max_d \text{Prob}(q|d \text{ is } R)\text{Prob}(d \text{ is } R)$$

q: input query, d: all documents in the database
"is R": is relevant

$$d^* = \arg\max_d \text{Prob}(q|d \text{ is } R)$$

reduced to maximum likelihood without prior knowledge

- **Observation Probability in the HMM state (as a simple example)**

$$P(q|d \text{ is } R) = \left[m_1 P(t_1|d) + m_2 P(t_1|C)\right] \cdot \prod_{n=2}^{N}\left[m_1 P(t_n|d) + m_2 P(t_n|C) + m_3 P(t_n|t_{n-1},d) + m_4 P(t_n|t_{n-1},C)\right]$$

  – m_1,m_2,m_3,m_4 trained by EM/MCE

Latent Semantic Indexing (LSI) Model for Speech-based Information Retrieval

- **Term-Document Matrix**
  - M indexing terms $\{t_1,t_2,...t_M\}$ and N documents $\{d_1,d_2,....d_N\}$

    $W = [w_{ij}]_{M\times N}$
  - $w_{ij} = l_{ij} \cdot g_i$, l_{ij}: local weight

    g_i: global weight

    $w_{ij} = (\frac{c_{ij}}{n_j})(1-\varepsilon_i)$, normalized with document length and term entropy, or

    $w_{ij} = [1+\ln(c_{ij})]\ln(N/N_i)$, TF/IDF
- **Singular Value Decomposition (SVD)**

  $W \approx \hat{W} = USV^T$, S= diagonal with singular values
  - $\underline{u}_i = u_i S$ term vector

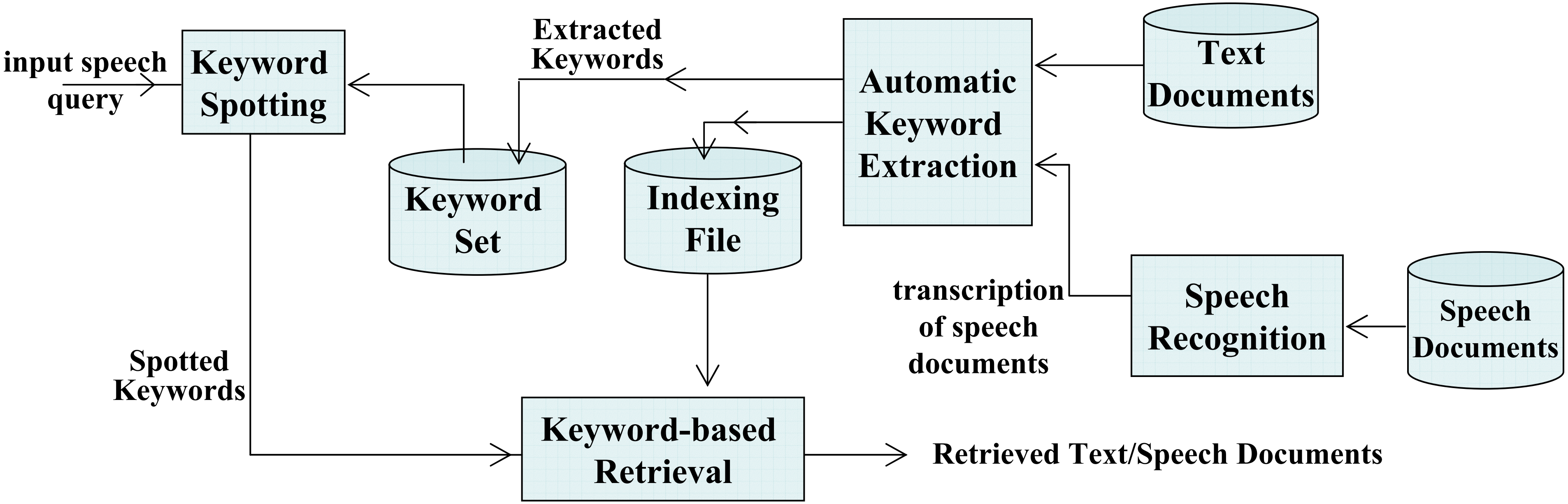
    $\underline{v}_i = \underline{v}_i S$ document vector
  - reduced to R-dimensional space of "latent semantic concepts"
- **Query q considered as a new document "folded-in"**

  $$\underline{v}_q = d_q^T U$$
  - relevance score:

    $$R(q,d) = \frac{\underline{v}_q \cdot \underline{v}_d}{\left|\underline{v}_q\right| \cdot \left|\underline{v}_d\right|}$$

Speech-based Information Retrieval by Keywords — An Example

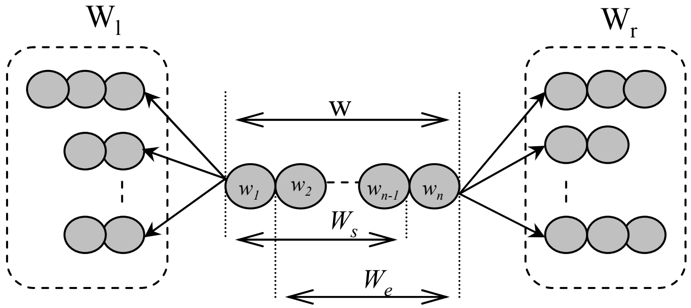
- **Automatic Keyword Extraction from Texts integrated with Keyword Spotting**



- **Integration with Other Approaches**

Automatic Keyword/Key Phrase Extraction from Texts for Chinese Language

- **Automatic Keyword Extraction from Texts is not too difficult for Alphabetic Languages**
  - words well defined by boundary blanks
  - proper nouns identified by capital letters
  - these are not true for Chinese



- **Two Steps: Complete Pattern Identification and Domain Significance Evaluation**
- **Complete Pattern Identification**
  - W: The Segment of Characters Being Considered
  - Within-segment Checking : f(W) , f(W_s) , f(W_e) , f(·): some function
    - example : if W_s,W_e always appear as a part of W,W is a a more “complete” pattern
  - Left/Right Context Checking : W_l , W_r , f(W) , f(W_l), f(W_r), W_l, W_r: some patterns on the left and right
    - example : if W always appears next to a fixed W_r , (W,W_r) is a more “complete” pattern
      if W appears freely with quite many different W_l and W_r ,W is a more “complete” pattern
- **Domain Significance Evaluation**
  - words/phrases commonly used in many documents deleted
  - a domain significance score evaluated with domain-specific PAT trees constructed by domain-specific training documents