
Context-sensitive Semantic Smoothing for the Language Modeling Approach to Genomic IR

Xiaohua (Davis) Zhou

xiaohua.zhou@drexel.edu

College of Information Science & Technology
Drexel University

Overview

- **Context-sensitive Semantic Smoothing**
 - Model the semantic relationships between query terms and indexing terms.
 - Incorporate contextual and sense information into the semantic relationship modeling.
- **Genomic Information Retrieval**
 - Biomedical literature retrieval
 - TREC Genomic Track 2004 and 2005

Statistical Translation Model

- Basic Idea
 - Searching behavior may be different from indexing behavior (e.g., query: auto indexing: car)
 - Document model can be translated to query model to make up the mismatches between query terms and indexing terms (Berger and Lafferty 1999)

$$p(q_j | d) = \sum_w t(q_j | w)l(w | d)$$

Statistical Translation Model

- Translation Probability Estimate: $t(q|w)$
 - Pseudo query-document pairs (Berger and Lafferty 1999)
 - Title-document pairs (Jin, Hauptmann and Zhai 2002)
 - Markov Chain Model (Lafferty and Zhai 2001)
 - Cooccurrence with Semantic Constraint (Cao et al. 2005)
 - Use WordNet relationship
- The weakness
 - The word-word translation results are often fairly general and contain mixed topics because a single word is ambiguous in many cases.
 - (movie star, star war); (computer mouse, animal mouse)

An Ideal Translation Model

- Topic-based Translation Model
 - A topic incorporates contextual and sense information and thus its translation to individual terms will be more specific.
 - It is easy to decompose a document into a set of topics and to compute the weight (influence) of each topic in the document.

$$p_t(w | d) = \sum_k p(w | \theta_{t_k}) p(t_k | d)$$

Cluster Language Model

■ Motivation

- Have more similar documents to estimate a more accurate and smoothed model (Liu and Croft 2004)

■ Retrieval Model

- Treat each cluster as a topic.
- A document belongs to only one cluster, i.e. one topic. This assumption does not hold very well for real data.

$$p(w | d) = \frac{N_d}{N_d + u} p_{ML}(w | d) + \left(1 - \frac{N_d}{N_d + u}\right) p(w | cluster)$$

HAL-based Information Flow

- Basic Idea (Song and Bruza, 2003)
 - Each concept is represented by HAL vector
 - Multiple concepts can be combined (referred to as combined concept) and represented by a HAL vector.
 - Concept similarity is derived by computing the HAL vector similarity.
- Retrieval Use
 - The translation from combined concept to individual concept is context-sensitive
 - It does not provide a mechanism of finding out combined concepts from documents. Thus it is good for query expansion, but not for document expansion.

Latent Topic Models

■ Basic Idea

- Statistical latent topic models will find out latent topics (distribution over words) and document models (distribution over topics) simultaneously.
- Latent Topic Models: LSI, pLSI, and LDA

■ Retrieval Use

- Efficiency of computing topic models is a big concern
- LDA-based document models (Wei and Croft, 2006)

Topic Signature Representation

Definition 1 A **topic signature** (t) is defined with two order-free components as in $t(w_i, w_j)$, where w_i and w_j are two concepts related to each other syntactically and semantically.

Definition 2 A **concept** (w) is a unique meaning in a domain. It represents a set of synonymous terms in the domain.

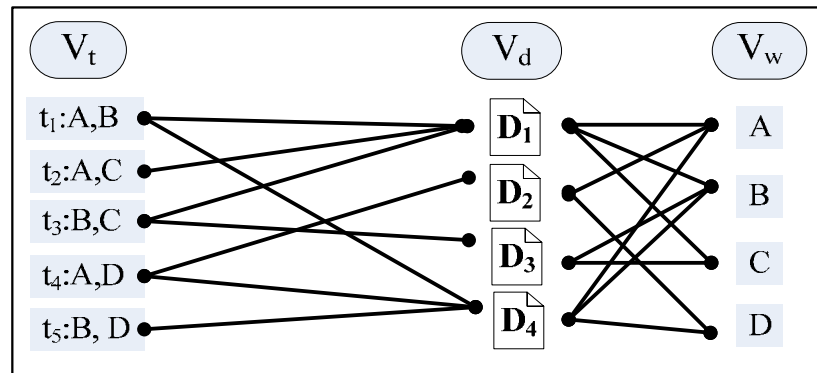


Figure 1. Illustration of document indexing. V_t , V_d and V_w are topic signature set, document set and concept set, respectively.

Strength of Topic Signatures

- **Context-sensitivity**
 - Two related concepts help to determine context for each other. Thus, the translation from concept pair topic signature to individual terms would be more specific.
- **Domain Need**
 - Biomedical literature is full of various binary relationships which carry very useful information.
 - The density of binary relationships makes it possible to have a robust estimate of document models over topic signatures.

Topic Signature Extraction

■ Ontology-based Extraction

- Concept Extraction: MaxMatcher (Zhou et al. 2006)
- Topic Signature Extraction
 - Two concepts appear in the same clause of an English sentence;
 - their semantic types are compatible according to the domain ontology, e.g., protein-protein interaction.

■ Examples

A recent epidemiological study (C0002783, research activity) revealed that obesity (C0028754, disease) is an independent risk factor for periodontal disease (C0031090, disease).

Concept Index: C0002783, C0028754, C0031090

Topic Signature Index: (C0028754, C0031090)

Topic Signature Model

■ Basic Idea

- Denotes D_k the set of documents containing the topic signature t_k .
- Words in a document are either translated by the topic (topic signature t_k) or generated by the collection background model.

$$p(w | D_k) = (1 - \alpha)p(w | \theta_{t_k}) + \alpha p(w | C)$$

Where α is the coefficient controlling the influence of the translation component in the mixture language model.

Topic Signature Model

- Log likelihood of generating D_k

$$\begin{aligned}\log p(D_k | \theta_{t_k}, C) &= \sum_w c(w, D_k) \log p(w | D_k) \\ &= \sum_w c(w, D_k) ((1 - \alpha) p(w | \theta_{t_k}) + \alpha p(w | C))\end{aligned}$$

Where $c(w, D_k)$ is the document frequency of term w in D_k

- Parameter Estimating by EM

$$\hat{p}^{(n)}(w) = \frac{(1 - \alpha) p^{(n)}(w | \theta_{t_k})}{(1 - \alpha) p^{(n)}(w | \theta_{t_k}) + \alpha p(w | C)}$$

$$p^{(n+1)}(w | \theta_{t_k}) = \frac{c(w, D_k) \hat{p}^{(n)}(w)}{\sum_i c(w_i, D_k) \hat{p}^{(n)}(w_i)}$$

Document Model Smoothing

- Simple Language Model: DM0

$$p_b(w|d) = (1 - \alpha)p_{ml}(w|d) + \alpha p(w|C)$$

- Translation Language Model: DM1

$$p_t(w|d) = \sum_k p(w|\theta_{t_k})p_{ml}(t_k|d)$$

$$p_{ml}(t_k|d) = \frac{c(t_k, d)}{\sum_i c(t_i, d)}$$

$c(t_i, d)$ is the frequency of topic signature t_i in document d .

Document Model Smoothing

- *A Mixture Model: DM2
 - Because not all topics in a document will be expressed by topic signatures, we have to interpolate the topic-based translation model with a simple language model in order to recover the information loss caused by the topic model.

$$p_{bt}(w | d) = (1 - \lambda)p_b(w | d) + \lambda p_t(w | d)$$

The translation coefficient (λ) controls the influence of the translation component in the mixture model.

Pseudo-relevance Feedback

■ A Feedback Framework

$$p_{if}(w|q) = (1-\gamma)p_i(w|q) + \gamma p_f(w|q)$$

Zhai and Lafferty 2001. The feedback coefficient (γ) controls the influence of the feedback component in the mixture model.

■ A Heuristic Feedback Model: FM0

- Use topic signatures containing at least one query term to expand the query.
- Self-translation is applied.

$$p_f(w|q) = \sum_{k:q \cap t_k \neq \emptyset} p_s(w|t_k) \frac{c(t_k, D)}{\sum_{i:q \cap t_i \neq \emptyset} c(t_i, D)} \quad p_s(w|t_k) = \begin{cases} 0 & w \notin t_k \\ 1/|t_k| & w \in t_k \end{cases}$$

Pseudo-relevance Feedback

■ A Generative Feedback Model: FM1

- Assume topic signatures in feedback documents are generated by the feedback model or the collection background model

$$p(t_k | \theta_F, F, C) = (1 - \alpha) p(t_k | \theta_F) + \alpha p(t_k | C)$$

- The feedback model:

$$p_f(w | q) = \sum_k p_s(w | t_k) p(t_k | \theta_F)$$

- Parameter Estimate using EM:

$$\hat{p}^{(n)}(t_k) = \frac{(1 - \alpha) p^{(n)}(t_k | \theta_F)}{(1 - \alpha) p^{(n)}(t_k | \theta_F) + \alpha p(t_k | C)}$$

$$p^{(n+1)}(t_k | \theta_F) = \frac{c(t_k, F) \hat{p}^{(n)}(t_k)}{\sum_i c(t_i, F) \hat{p}^{(n)}(t_i)}$$

Experiments

■ Collections

- TREC Genomic Track 2004 and 2005
- Use sub-collection (human relevance-judged pool)
 - 2004: 48,753 documents
 - 2005: 41,018 documents

■ Measures:

- Average Precision (AP), Recall

■ Settings

- Background coefficient (alpha) for SLM: 0.05
- Pseudo-relevance feedback: top 50 documents, expand 10 terms

Baseline Models

Table 1. Comparison of the baseline language model to the Okapi model. The Okapi formula is the same as the one in [10]. The number of relevant documents for TREC04 and TREC05 are 8266 and 4585, respectively. The asterisk indicates the initial query is weighted.

Collection	Recall			MAP		
	SLM	Okapi	Change	SLM	Okapi	Change
TREC04	6411	6662	+3.9%	0.345	0.363	+5.2%
TREC04*	6527	6704	+2.7%	0.364	0.364	+0.0%
TREC05	4084	4124	+1.0%	0.255	0.250	-2.0%
TREC05*	4135	4134	-0.0%	0.260	0.254	-2.3%

Results of Document and Query Model Smoothing

Table 2. The comparison of the baseline language model (DM0) to document smoothing model (DM2) and query smoothing model (FM1).

Collection		DM0	$\lambda=0.3$		$\gamma=0.6$	
			DM2	Change	FM1	Change
TREC04	MAP	0.345	0.395	+14.5%	0.451	+30.9%
	Recall	6411	6749	+5.3%	6929	+8.0%
TREC04*	MAP	0.364	0.414	+13.7%	0.460	+26.9%
	Recall	6527	6905	+5.8%	7039	+7.8%
TREC05	MAP	0.255	0.277	+8.6%	0.279	+9.4%
	Recall	4084	4167	+2.0%	4227	+3.5%
TREC05*	MAP	0.260	0.288	+10.8%	0.287	+10.4%
	Recall	4135	4214	+1.9%	4235	+2.4%

Effect of Document Smoothing

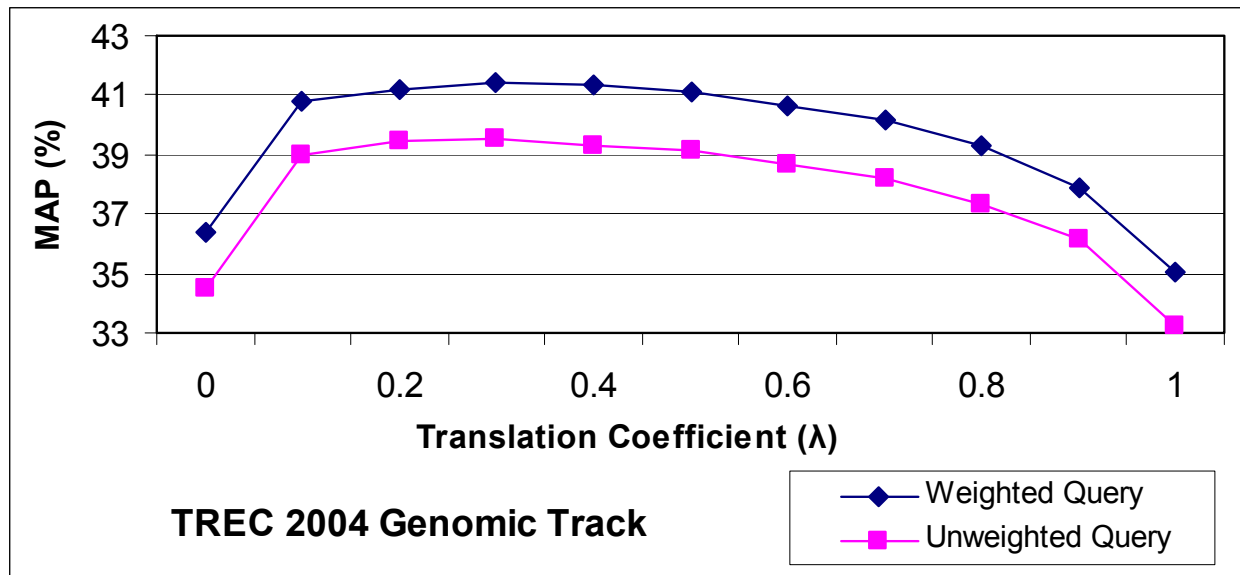


Figure 2. The variance of MAP with the translation coefficient (λ), which controls the influence of the translation model in DM2.

Effect of Document Smoothing

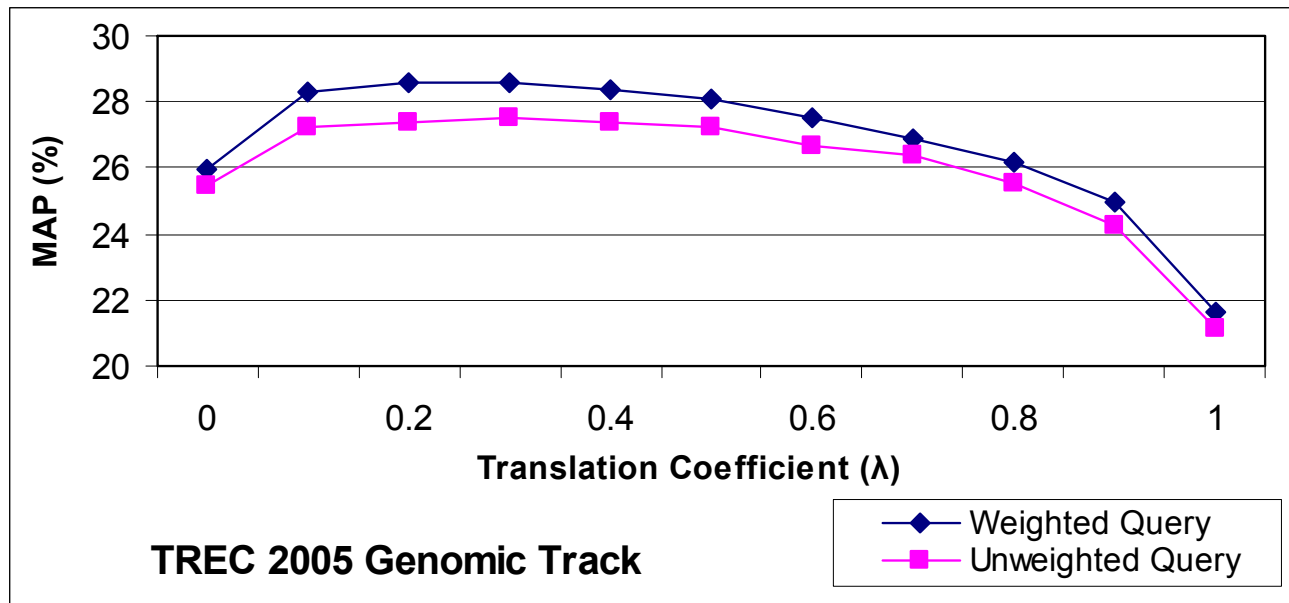


Figure 3. The variance of MAP with the translation coefficient (λ), which controls the influence of the translation model in DM2.

Effect of Query Smoothing

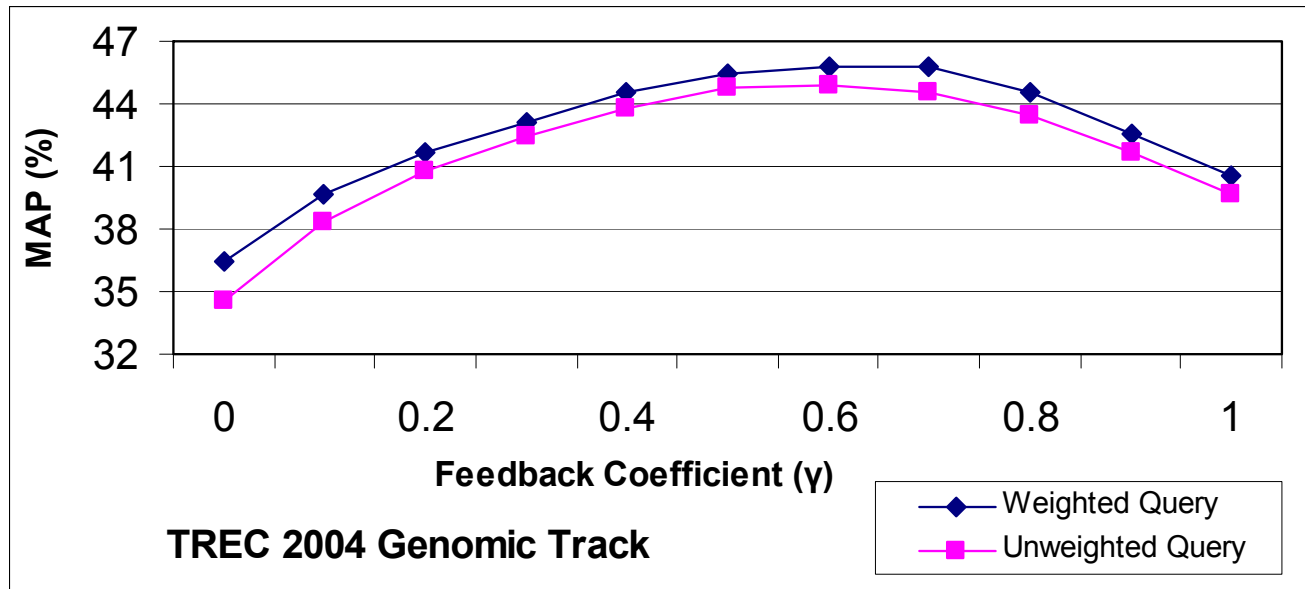


Figure 4. The variance of MAP with the feedback coefficient (γ), which controls the influence of the feedback model in blind feedback (i.e. DM0+FM1).

Effect of Query Smoothing

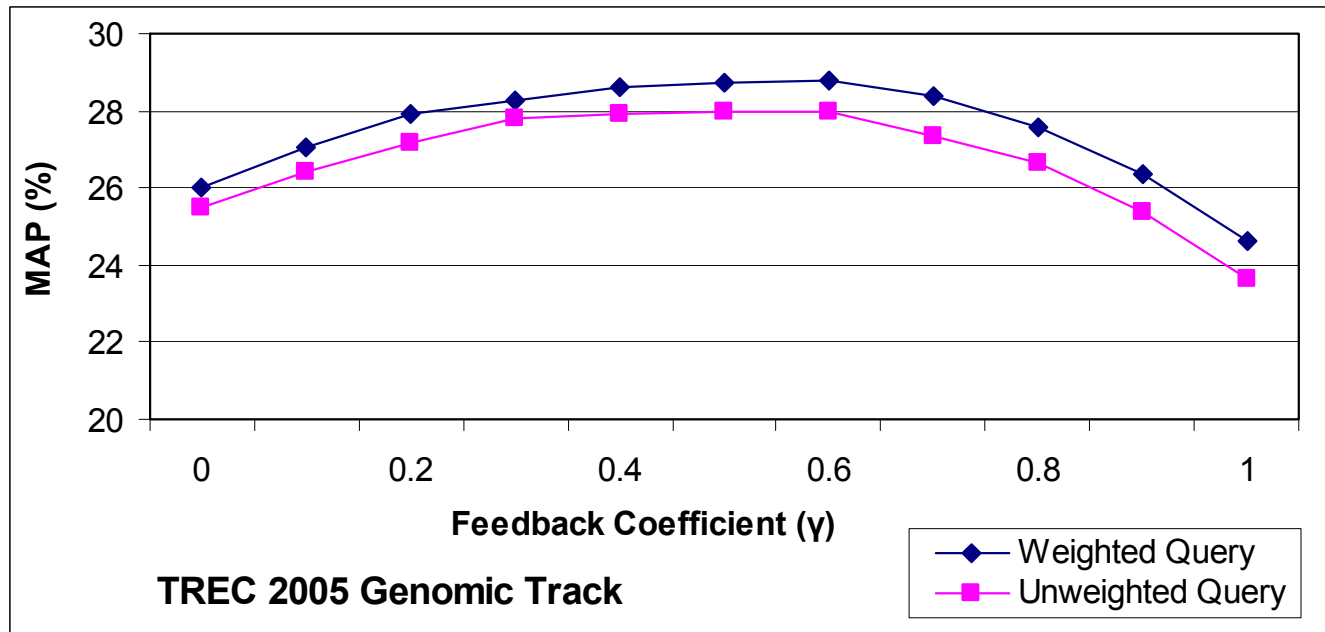


Figure 5. The variance of MAP with the feedback coefficient (γ), which controls the influence of the feedback model in blind feedback (i.e. DM0+FM1).

Interaction Effect

Table 3. The interaction effect of document smoothing (DM2) and query smoothing (FM1). “Max” is the maximum effect achieved by DM2 or FM1. “Both” is the result of DM2+FM1. “Change^[1]” is the improvement of DM2+FM1 over DM0. “Change^[2]” is the improvement of DM2+FM1 over “Max”.

Collection		DM0	Max	Both	Change ^[1]	Change ^[2]
TREC04	MAP	0.345	0.451	0.461	+33.6%	+2.2%
	Recall	6411	6929	7026	+9.6%	+1.4%
TREC04*	MAP	0.364	0.460	0.470	+29.1%	+2.2%
	Recall	6527	7039	7079	+8.5%	+0.6%
TREC05	MAP	0.255	0.279	0.295	+15.7%	+5.7%
	Recall	4084	4227	4273	+4.7%	+1.1%
TREC05*	MAP	0.260	0.288	0.313	+20.4%	+8.7%
	Recall	4135	4235	4317	+4.4%	+1.9%

Comparison of Feedback Models

■ Conclusions

- The generative model (FM1) consistently outperforms the heuristic model (FM0)

Table 4. Comparison of blind feedback model FM1 to FM0

Collection	Recall			MAP		
	FM0	FM1	Change	FM0	FM1	Change
TREC04	6808	6929	+1.7%	0.442	0.451	+2.0%
TREC04*	6811	7039	+3.3%	0.449	0.460	+2.4%
TREC05	4192	4227	+0.8%	0.270	0.279	+3.3%
TREC05*	4215	4235	+0.5%	0.279	0.288	+3.2%

Context-sensitive vs. Context-insensitive

- Context-insensitive Semantic Smoothing
 - Follow the method proposed by Cao et al. (2005)

$$p_t(w | d) = \sum_k p(w | w_k) p_{ml}(w_k | d)$$
$$p(w | w_k) = (1 - \alpha) \frac{c(t(w, w_k))}{\sum_i c(t(w_i, w_k))} + \alpha p(w | C)$$

$c(t(w, w_k))$ is the frequency count of $t(w, w_k)$ in the whole collection

- Conclusions
 - The context-sensitive semantic smoothing approach performs consistently better than context-insensitive approaches.

Context-sensitive vs. Context-insensitive

Table 5. Comparison of the context-sensitive semantic smoothing (DM2) to the context-insensitive semantic smoothing (DM2') on MAP. The rightmost column is the change of DM2 over DM2'.

Collection	DM0	DM2'		DM2		Change
	MAP	MAP	Change	Map	Change	
TREC04	0.346	0.367	+6.1%	0.395	+14.5%	+7.6%
TREC04*	0.364	0.384	+5.5%	0.414	+13.7%	+7.8%
TREC05	0.255	0.260	+2.0%	0.277	+8.6%	+6.5%
TREC05*	0.260	0.269	+3.5%	0.288	+10.8%	+7.1%

Comparison to Other Approaches

- Other Approaches
 - Simple language model
 - Local Information Flow (Song and Bruza 2003)
 - Context sensitive semantic smoothing
 - Can not incorporate domain knowledge
 - Model-based Feedback (Zhai and Lafferty 2001)
- Findings
 - The incorporation of domain knowledge did not help much when using simple language model
 - The context-sensitive semantic smoothing using topic signature provided an effective framework to incorporate domain knowledge.

Comparison to Other Approaches

Table 6. Comparison of the retrieval performance of six approaches on TREC genomic track 2004 and 2005. “Word” or “Concept” means the indexing unit used. The concept-based indexing is based on the UMLS Metathesaurus. All approaches are implemented by us.

IR Approaches	TREC 2004		TREC 2005	
	MAP	Recall	MAP	Recall
Simple Language Model (Word)	0.324	6328	0.258	4101
Simple Language Model (Concept)	0.345	6411	0.255	4084
Local Information Flow (Word)	0.378	6793	0.272	4220
Model-based Feedback (Word)	0.372	6742	0.279	4260
Model-based Feedback (Concept)	0.424	6896	0.290	4213
Topic Signature (Concept)	0.461	7026	0.295	4273

The Dragon Toolkit



- Descriptions
 - Written in pure Java
 - Initially designed for semantic-based language modeling and information retrieval.
 - The current version support text indexing, text retrieval, text clustering, text classification, and topic modeling.
- Download
 - <http://www.ischool.drexel.edu/dmbio/dragontool>

Future Work

- Apply this IR model to public domains
- Adopt other existing concept and relation extraction approaches which will not need domain ontology.
- Use context-sensitive topic signatures other than concept pairs.