WIDIT in TREC-2007 Blog Track: Combining Lexicon-based Methods to Detect Opinionated Blogs

Kiduk Yang, Ning Yu, Hui Zhang
School of Library and Information Science, Indiana University, Bloomington, Indiana 47405, U.S.A.
{kiyang, nyu, hz3}@indiana.edu

1. INTRODUCTION

In TREC-2007, Indiana University's WIDIT Lab[1] participated in the Blog track's opinion task and the polarity subtask. For the opinion task, whose goal is to "uncover the public sentiment towards a given entity/target", we focused on combining multiple sources of evidence to detect opinionated blog postings. Since detecting opinionated blogs on a given topic (i.e., entity/target) involves not only retrieving topically relevant blogs but also identifying those that contain opinions about the target, our approach to the opinion finding task consisted of first applying traditional IR methods to retrieve on-topic blogs and then boosting the ranks of opinionated blogs based on combined opinion scores generated by multiple opinion detection methods. The key idea underlying our opinion detection method is to rely on a variety of complementary evidences rather than trying to optimize a single approach. This fusion approach to opinionated blog detection is motivated by our past experience that suggested no single approach, whether lexicon-based or classifier-driven, is well-suited for the blog opinion retrieval task. To accomplish the polarity subtask, which requires classification of the retrieved blogs into *positive* or *negative* orientation, our opinion detection module was extended to generate polarity scores to be used for polarity determination.

2. RESEARCH QUESTIONS

Having explored the topical search problem over the years (Yang, 2002; Yang et. al, 2005), we focused on the question of how to adapt a topical retrieval system for opinion finding task. The intuitive answer was to first apply IR methods to retrieve blogs about a target (i.e., on-topic retrieval), and then identify opinionated blogs by leveraging various evidences of opinion.

Therefore, our primary research question centers on the evidences of opinion, namely what they are and how they can be leveraged. To maximize the total coverage of opinion evidence, we considered the following three complementary sources of opinion evidence:

- Opinion Lexicon: An obvious source of opinion evidence is the set of terms commonly used in expressing opinions (e.g., "Skype *sucks*", "Skype *rocks*", "Skype is *cool*").
- Opinion Collocations: One of the contextual evidence of opinion comes from collocations used to mark adjacent statements as opinions (e.g., "*I believe* God exists", "God is dead *to me*").
- Opinion Morphology: When expressing strong opinions or perspectives, people often use morphed word form for emphasis ("Skype is *soooo* buggy", "Skype is *bugfested*").

Because blogs are generated by content management software (i.e. blogware) that allows authors to create and update contents via a browser-based interface, they are laden with non-posting content for navigation, advertisement, and formatting display. Thus, our secondary research question concerns how such blogware-generated noise influences opinion detection.

[1] Web Information Discovery Integrated Tool (WIDIT) Laboratory at the Indiana University School of Library and Information Science is a research lab that explores a fusion approach to information retrieval and knowledge discovery.

3. METHODOLOGY

WIDIT approach to blog opinion retrieval task consisted of three main steps: initial retrieval, on-topic retrieval optimization, and opinion identification. Initial retrieval was executed using the standard WIDIT retrieval method, on-topic retrieval optimization was done by a post-retrieval reranking approach that leveraged multiple topic-related factors, and opinion identification was accomplished by a fusion of five opinion modules that leveraged multiple sources of opinion evidence. To assess the effect of noise on retrieval performance, we explored various noise reduction methods with which to exclude non-English blogs and non-blog contents from the collection. The overview of WIDIT blog opinion retrieval system is shown in Figure 1.

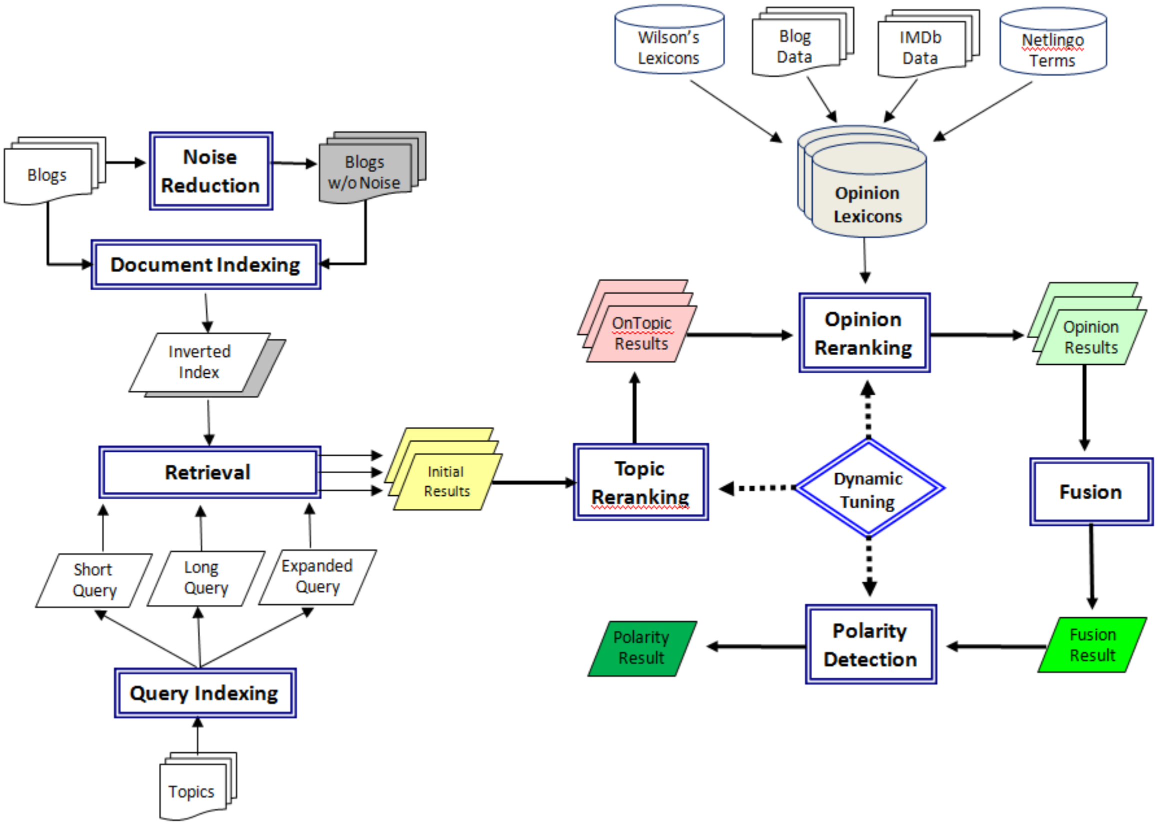


Figure 1. WIDIT Blog Opinion Retrieval System Architecture

3.1. Noise Reduction

To effectively eliminate the noise in blog data without inadvertently excluding valid content, we constructed *Non-English Blog Identification (NBI)* module that identifies non-English blogs for exclusion, and *Blog Noise Elimination (BNE)* module that excludes non-blog content portion of the blog. *NBI* leverages the characteristics of non-English (NE) blogs, which contain a large proportion of NE terms, and/or high frequency of NE stopwords. *NBI* heuristic, which scores documents based on NE content density and frequencies of stopwords (both English and non-English), was tuned by iteratively examining the NE blog clusters identified by the module to find false positives and adjusting the NE threshold until no false positives were found. *BNE* module, which uses markup tags to differentiate blog content (e.g., post, comments, etc.) from non-blog content (e.g., scripts, style texts, forms, sidebar, navigation, profile, advertisement, header, footer, etc.), was constructed by examining all unique markup tags in the blog collection to identify patterns to be captured by regular expressions.

3.2. Initial Retrieval

The initial retrieval is executed by the WIDIT retrieval engine, which consists of document/query indexing and retrieval module. After removing markup tags and stopwords, WIDIT's indexing modules applies a modified version of the simple plural remover (Frakes & Baeza-Yates, 1992).[2] The stopwords consisted of non-meaningful words such as words in a standard stopword list, non-alphabetical words, words consisting of more than 25 or less than 3 characters, and words that contain 3 or more repeated characters. Hyphenated words were split into parts before applying the stopword exclusion, and acronyms and abbreviations were kept as index terms[3].

In order to enable incremental indexing as well as to scale up to large collections, WIDIT indexes the document collection in fixed-size subcolllections, which are searched in parallel. The whole collection term statistics, derived after the creation of the subcollections, are used in subcollection retrievals so that subcollection retrieval results can simply be merged without any need for retrieval score normalizations.

Query indexing module includes query expansion submodules that identify nouns and noun phrases, expand acronyms and abbreviations, and extract non-relevant portion of topic descriptions with which to formulate various expanded versions of the query.

The retrieval module implements the probabilistic model using the Okapi BM25 formula. The simplified version of the Okapi BM25 relevance scoring formula (Robertson & Walker, 1994) is used to implement the probabilistic model.

$$d_{ik} = \log\left(\frac{N - df_k + 0.5}{df_k + 0.5}\right)\frac{f_{ik}}{k_1((1-b) + b \cdot (\frac{dl}{avdl})) + f_{ik}} \qquad q_k = \frac{(k_3+1)f_k}{k_3 + f_k} \qquad (1)$$

3.3. On-topic Retrieval Optimization

Optimizing the ranking of the initial retrieval results is important for two reasons. First, on-topic retrieval optimization is an effective strategy for incorporating topical clues not considered in initial retrieval (Yang, 2002; Yang & Albertson, 2004; Yang et. al, 2007). Second, our two-step strategy for targeted opinion detection consists of minimalistic initial retrieval that favors recall followed by post-retrieval reranking to boost precision.

Our on-topic retrieval optimization involves reranking of the initial retrieval results based on a set of topic-related reranking factors, where the reranking factors consist of topical clues not used in initial ranking of documents. The topic reranking factors used the study are: *Exact Match*, which is the frequency of exact query string occurrence in document, *Proximity Match*, which is the frequency of padded[4] query string occurrence in document, *Noun Phrase Match*, which is the frequency of query noun phrases occurrence in document, and *Non-Rel Match*[5], which is the frequency of non-relevant nouns and noun phrase occurrence in documents. All the reranking factors are normalized by document length. The on-topic reranking method consists of following three steps:

1) Compute topic reranking scores for top N results.
2) Partition the top N results into reranking groups based on the original ranking and a combination of the most influential reranking factors. The purpose of reranking groups is to prevent excessive influence of reranking by preserving the effect of key ranking factors.

[2] The simple plural remover was chosen to speed up indexing time and to minimize the overstemming effect of more aggressive stemmers.

[3] Acronym and abbreviation identification was based on simple pattern matching of punctuations and capitalizations.

[4] "Padded" query string is a query string with up to *k* number of words in between query words.

[5] Non-rel Match is used to suppress the document rankings, while other reranking factors are used to boost the rankings.

3) Rerank the initial retrieval results within reranking groups by the combined reranking score.

The objective of reranking is to float low ranking relevant documents to the top ranks based on post-retrieval analysis of reranking factors. Although reranking does not retrieve any new relevant documents (i.e. no recall improvement), it can produce high precision improvement via post-retrieval compensation (e.g. phrase matching). The key questions for reranking are what reranking factors to consider and how to combine individual reranking factors to optimize the reranking effect. The selection of reranking factors depends largely on the initial retrieval method since reranking is designed to supplement the initial retrieval. The fusion of the reranking factors can be implemented by a weighted sum of reranking scores, where the weights represent the contributions of individual factors. The weighted sum method is discussed in more detail in the fusion section of the methodology.

3.4. Opinion Detection

Determining whether a document contains an opinion is somewhat different from classifying a document as opinionated. The latter, which usually involves supervised machine learning, depends on the document's overall characteristic (e.g., degree of subjectivity), whereas the former, which often entails the use of opinion lexicons, is based on the detection of opinion evidence. At the sentence or paragraph level, however, the distinction between the two becomes inconsequential since the overall characteristic is strongly influenced by the presence of opinion evidence.

For opinion mining, which involves opinion detection rather than opinion classification, opinion assessment methods are best applied at subdocument (e.g., sentence or paragraph) level. At subdocument level, the challenges of machine learning approach are compounded with two new problems: First, the training data with document-level labels is likely to produce a classifier not well suited for subdocument level classification. Second, the sparsity of features in short "documents" will diminish the classifier's effectiveness.

Our opinion detection approach, which is entirely lexicon-based to avoid the pitfalls of the machine learning problems, relies on a set of opinion lexicons that leverage various evidences of opinion. The key idea underlying our method is to combine a set of complementary evidences rather than trying to optimize the utilization of a single source of evidence. We first construct opinion lexicons semi-automatically and used them in opinion scoring submodules to generate opinion scores of documents. The opinion scores are then combined to boost the ranks of opinionated blogs in a manner similar to the on-topic retrieval optimization.

Opinion scoring modules used in this study are *High Frequency module*, which identifies opinions based on the frequency of opinion terms (i.e., terms that occur frequently in opinionated blogs), *Low Frequency module*, which makes use of uncommon/rare terms (e.g., "sooo good") that express strong sentiments, *IU module*, which leverages n-grams with IU (I and you) anchor terms (e.g., I believe, You will love), *Wilson's lexicon module*, which uses a collection-independent opinion lexicon composed of a subset of Wilson's subjectivity terms, and *Opinion Acronym module*, which utilizes a small set of opinion acronyms (e.g., imho). Each module computes two opinion scores for each lexicon used: a simple frequency-based score and a proximity score based on the frequency of lexicon terms that occur near the query string in a document. The generalized formula for opinion scoring can be described as

$$opSC(d) = \frac{\sum_{t \in L \cap D} f(t) \cdot s(t)}{len(d)} \qquad (2)$$

where *L* and *D* denote the term sets of a given lexicon and document *d* respectively, *len*(*d*) is the number of tokens in *d*, *s*(*t*) is the strength of term *t* as designated in the lexicon, and *f*(*t*) is the frequency function that returns either the frequency of *t* in *d* (simple score) or the frequency of *t* that co-occurs with the query string in *d* (proximity score). The proximity score, which is a strict measure that ensures the opinion found is on target, is liable to miss opinion expressions located outside the proximity window as well as those within the proximity of the target that is expressed differently from the query string. The simple score, therefore, can supplement the proximity score, especially when used in conjunction with the on-topic optimization. For polarity detection, positive and negative polarity scores are computed for each score type (i.e. simple, proximity). The generalized formula for computing opinion polarity scores is shown below.

$$opSC_{pol}(d)\frac{\sum_{t\in L_{pol}\cap D} f(t)\cdot s(t)}{len(d)} \qquad (3)$$

In equation 3, L_{pol} describes the lexicon term subset whose polarity is *pol* (positive or negative). The default term polarity from the lexicon is reversed if the term appears near a valence shifter (e.g., not, never, no, without, hardly, barely, scarcely) in *d*.

3.4.1. High Frequency Module

The basic idea behind the *High Frequency Modul*e (HFM) is to identify opinions based on common opinion terms. Since common opinion terms, which are words often used to express opinions, will occur frequently in opinionated text and infrequently in non-opinionated text, we create the candidate HF lexicon by identifying high frequency terms from the positive blog training data (i.e. opinionated blogs) and excluding those that also have high frequency in the negative blog training data. The resulting term set is then manually reviewed to filter out spurious terms and to assign polarity and strength of opinion.

3.4.2. Wilson's Lexicon Module

To supplement the HF lexicon, which is collection-dependent, we construct a set of opinion lexicons from Wilson's subjectivity terms (Wilson, Pierce, & Wiebe, 2003). *Wilson's Lexicon Module* (WLM) uses three collection-independent lexicons, which consists of strong and weak subjective terms extracted from Wilson's subjectivity term list, and emphasis terms selected from Wilson's intensifiers. Both strong and weak subjective lexicons inherit the polarity and strength from Wilson's subjectivity terms, but the emphasis lexicon includes neither the strength nor polarity values since the strength and polarity of an emphasis term depend on the term it emphasizes (e.g., *absolutely* wonderful). In computing opinion scores (equation 2), emphasis terms are assigned the strength of 1, which is the minimum value for term strength. No polarity scores are generated with the emphasis lexicon.

3.4.3. Low Frequency Module

While HFM and WLM leverage the obvious source of opinion, namely the standard opinion lexicon used in expressing opinions (e.g., "Skype *sucks*", "Skype *rocks*", "Skype is *cool*"), *Low Frequency Module* (LFM) looks to the low frequency terms for opinion evidence. LFM is derived from the hypothesis that people become creative when expressing opinions and tend to use uncommon or rare term patterns (Wiebe et. al, 2004). These creative expressions, or Opinion Morphology (OM) terms as we call it, may be intentionally misspelled words (e.g., luv, hizzarious), compounded words (e.g., metacool, crazygood), repeat-character words (e.g., sooo, fantaaastic, grrreat), or some combination of the three (e.g., metacoool, superrrrv).
Since OM terms occur infrequently due to their creative and non-standard nature, we start the construction of the OM lexicon by identifying low frequency (e.g., *df* < 100) terms in the blog collection. Words with three or more repeated characters in the low frequency term set are examined to detect OM patterns, which are encapsulated in a compilation of regular expressions. The regular expressions (OM regex) are refined iteratively by repeating the cycle described below:

1. Examine repeat-character patterns and compile regular expressions.
2. Apply regular expressions to the low frequency term set.
3. Modify regular expressions after comparing the result of step 2 with the repeat-character patterns in step 1.
4. Go to step 2.

To round out the OM regex, regular expressions that simulate misspellings by vowel substitutions (e.g., luv) as well as regular expressions for capturing compound morphing are constructed from HF and Wilson terms, applied to the LF term set, and refined iteratively in a manner similar to the repeat-character refinement steps describe above. LF terms not captured by OM regex but are nevertheless determined to be opinion terms form a basis for the OM lexicon. The final OM lexicon consists of the opinion terms flagged during the OM regex construction process as well as those identified during the final review of the LF term subset that excludes terms identified by the OM regex. The format of OM regex is consistent with other lexicons in that each entry is composed of a regular expression and associated polarity and strength.

3.4.4. IU Module

IU Module (IUM) is motivated by the observation that pronouns such as 'I' and 'you' appear very frequently in opinionated texts. IU collocations, which are n-grams with IU (I and you) anchor terms (e.g., I believe, you will love), mark adjacent statements as opinions (e.g., "*I believe* God exists", "God is dead *to me*"). In this regard, IU collocations provide yet another type of opinion evidence to complement the opinion lexicons of HFM/WLM and the opinion morphology of LFM.

For IU lexicon construction, we first extract n-grams that begin or end with IU anchors (e.g., I, you, we, my, your, our, me, etc.) from the positive blog training data and movie review data (obtained from http://www.cs.cornell.edu/people/pabo/movie-review-data/). Since IU collocations are not collection-dependent, movie reviews provids additional data to broaden and strengthen the IU n-gram harvest. The extracted n-grams are then manually filtered to create the lexicon of IU collocations (e.g., 'I believe', 'my assessment', 'good for you', etc.) with associated polarity and strength, after which the lexicon is expanded with additional IU collocations consisting of IU anchors and appropriate HF and Wilson terms (e.g., verbs).

In order to accommodate the various forms of an IU collocation (e.g., I believe, I cannot but believe, I have always believed, etc.), IUM applies the IU lexicon in a slightly different manner than other modules. After preprocessing documents to compress certain prepositions, conjunctions, and articles (e.g., of, with, and, or, a, the, etc.), IU collocations in the lexicon are "padded" in such a way that document texts with up to two words in-between IU collocation words will be matched by IUM.

3.4.5. Opinion Acronym Module

The Opinion Acronym lexicon used by *Opinion Acronym Module* (OAM) complements the IU lexicon with common IU collocation acronyms (e.g., imho). The OA lexicon consists of a manually filtered subset of netlingo's chat acronyms and text message shorthand (http://www.netlingo.com/emailsh.cfm) in both acronym and expanded forms. Since opinion acronyms represent long phrases that serve as a clear indicator of opinion or sentiment, they are generally given higher opinion strength values than other lexicon entries. Like emphasis terms, no polarity is assigned to the OA lexicon.

3.5. Fusion

Topic reranking for on-topic retrieval optimization and opinion reranking for opinion detection generate multitudes of reranking scores that need to be combined. Two most common fusion methods are *Similarity Merge* (Fox & Shaw, 1995) and *Weighted Sum* (Bartell, Cottrell, & Belew, 1994). *Similarity Merge*, based on the assumption that documents with higher overlap are more likely to be relevant, multiplies the sum of fusion component scores for a document by the number of fusion components that retrieved the document (i.e. overlap),. Instead of relying on overlap, the *Weighted Sum* method sums the fusion component scores weighted by their relative contributions, which is usually estimated from training data.

In our earlier investigations (Yang, 2002, 2004), we found the normalized weighted sum formula to be most effective in combining fusion components that are dissimilar. The normalized weighted sum formula (equation 4) linearly combines the min-max normalized score of component *i*, (nSC_i) with fusion weight w_i, to generate the fusion score. In the min-max normalization (Lee, 1997), described in equation 5, $S_i(d)$ is the raw score of document *d* by component *i*, and $min\{S_i\}$ and $max\{S_i\}$ are the minimum and maximum scores by the component *i*.

$$FS(d) = \sum_{i=1}^{k} (w_i * NS_i(d)) \qquad (4)$$

$$NS_i(d) = \frac{S_i(d) - \min\{S_i\}}{\max\{S_i\} - \min\{S_i\}} \qquad (5)$$

In reranking, the original scores should be combined with fusion scores of reranking factors in a way that enables the documents with high reranking factors to float to the top without unduly influencing the existing document ranking. This reranking strategy can be expressed as

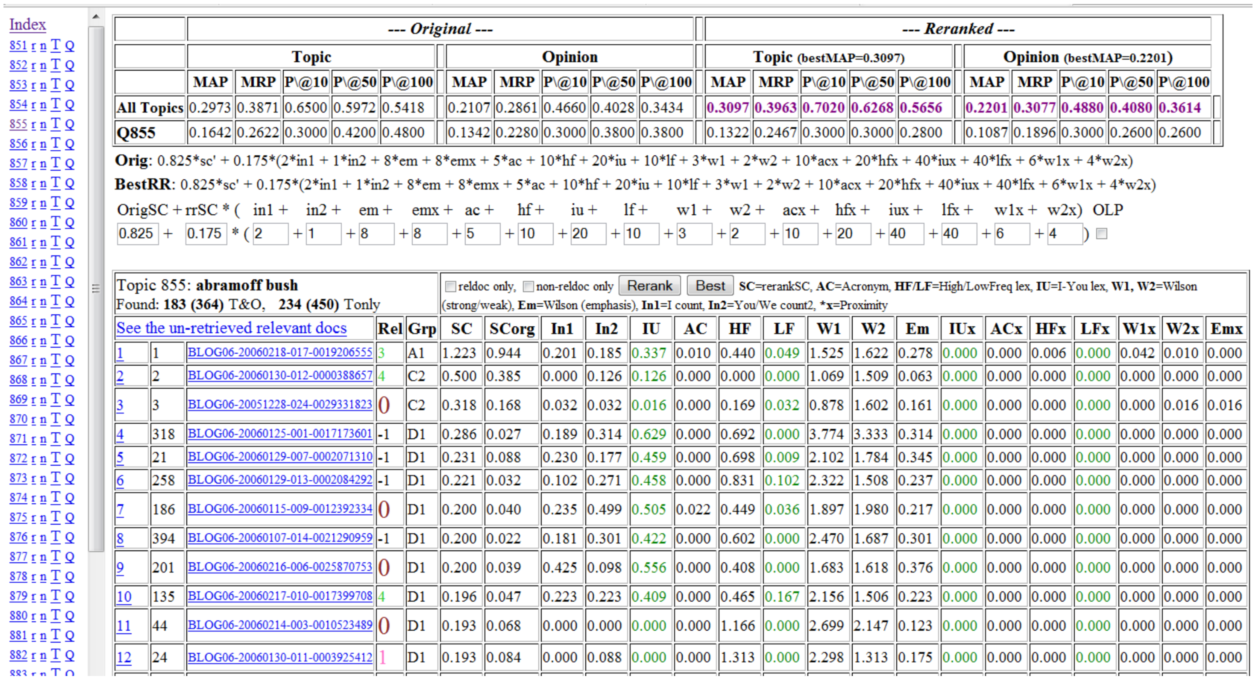
$$RS(d) = \alpha * NS_{orig}(d) + \beta * \sum_{i=1}^{k} (w_i * NS_i(d)) \qquad (6)$$

where $NS_{orig}(d)$ is the min-max normalized score of document *d* before reranking, and α and β are the weights that represent the estimated contributions of the original and combined reranking factor scores.

3.6. Dynamic Tuning

To optimize the reranking formulas, which involves determination of fusion weights (w_i) as well as original and reranking score weights (α and β), we implemented an interactive system tuning interface called *Dynamic Tuning*, which displays the effects of tuning parameter changes in real time to guide the human towards the system optimization state in a manner similar to bio-feedback. Dynamic Tuning, which is motivated by the idea of combining human intelligence, especially the pattern recognition ability, with the computational power of the machine, is implemented in a Web application that allows the human to examine not only the immediate effect of his/her system tuning but also the possible explanation of the tuning effect in the form of data patterns. By engaging in iterative dynamic tuning process that successively fine-tune the reranking parameters based on the cognitive analysis of immediate system feedback, system performance can be improved without resorting to an exhaustive evaluation of parameter combinations, which can not only be prohibitively resource intensive with numerous parameters but also fail to produce the optimal outcome due to its linear approach to factor combination.

Figure 2 shows the dynamic tuning interface for optimizing the fusion formula that combines opinion reranking scores, and figure 3 shows the dynamic tuning interface for optimizing the combination of polarity scores. In both interfaces, the effect of manual setting of the fusion formula weights are immediately displayed in terms of retrieval performance averaged over all topics as well as for the given topic. Individual reranking factor scores are also displayed for each document so that the human user may detect patterns that can be reflected in another cycle of tuning to beat the best performance (displayed in purple).



	--- *Original* ---										--- *Reranked* ---									
	Topic					**Opinion**					**Topic** (bestMAP=0.3097)					**Opinion** (bestMAP=0.2201)				
	MAP	**MRP**	**P\@10**	**P\@50**	**P\@100**	**MAP**	**MRP**	**P\@10**	**P\@50**	**P\@100**	**MAP**	**MRP**	**P\@10**	**P\@50**	**P\@100**	**MAP**	**MRP**	**P\@10**	**P\@50**	**P\@100**
All Topics	0.2973	0.3871	0.6500	0.5972	0.5418	0.2107	0.2861	0.4660	0.4028	0.3434	**0.3097**	**0.3963**	**0.7020**	**0.6268**	**0.5656**	**0.2201**	**0.3077**	**0.4880**	**0.4080**	**0.3614**
Q855	0.1642	0.2622	0.3000	0.4200	0.4800	0.1342	0.2280	0.3000	0.3800	0.3800	0.1322	0.2467	0.3000	0.3000	0.2800	0.1087	0.1896	0.3000	0.2600	0.2600

Orig: 0.825*sc' + 0.175*(2*in1 + 1*in2 + 8*em + 8*emx + 5*ac + 10*hf + 20*iu + 10*lf + 3*w1 + 2*w2 + 10*acx + 20*hfx + 40*iux + 40*lfx + 6*w1x + 4*w2x)

BestRR: 0.825*sc' + 0.175*(2*in1 + 1*in2 + 8*em + 8*emx + 5*ac + 10*hf + 20*iu + 10*lf + 3*w1 + 2*w2 + 10*acx + 20*hfx + 40*iux + 40*lfx + 6*w1x + 4*w2x)

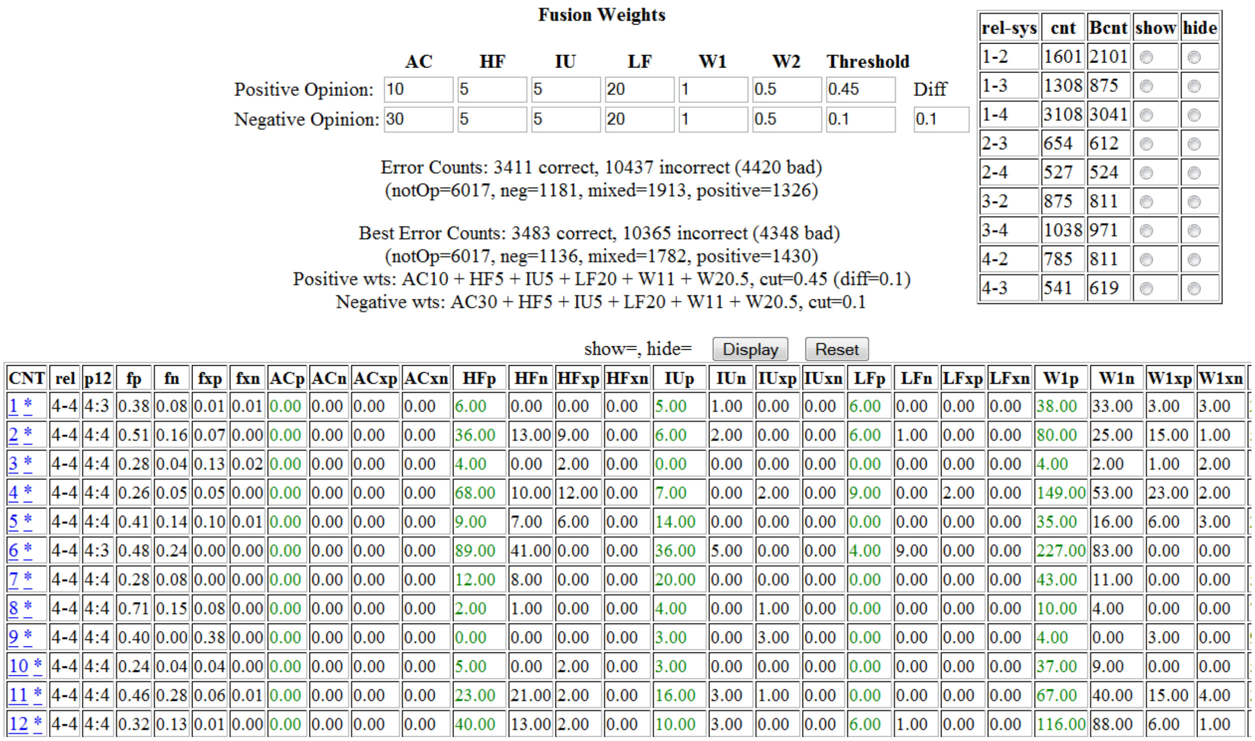
OrigSC + rrSC * (in1 + in2 + em + emx + ac + hf + iu + lf + w1 + w2 + acx + hfx + iux + lfx + w1x + w2x) OLP
0.825 + 0.175 * (2 + 1 + 8 + 8 + 5 + 10 + 20 + 10 + 3 + 2 + 10 + 20 + 40 + 40 + 6 + 4) □

Topic 855: **abramoff bush**
Found: **183 (364)** T&O, **234 (450)** Tonly

□ reldoc only, □ non-reldoc only Rerank Best **SC**=rerankSC, **AC**=Acronym, **HF/LF**=High/LowFreq lex, **IU**=I-You lex, **W1, W2**=Wilson (strong/weak), **Em**=Wilson (emphasis), **In1**=I count, **In2**=You/We count2, ***x**=Proximity

See the un-retrieved relevant docs			**Rel**	**Grp**	**SC**	**SCorg**	**In1**	**In2**	**IU**	**AC**	**HF**	**LF**	**W1**	**W2**	**Em**	**IUx**	**ACx**	**HFx**	**LFx**	**W1x**	**W2x**	**Emx**
1	1	BLOG06-20060218-017-0019206555	3	A1	1.223	0.944	0.201	0.185	0.337	0.010	0.440	0.049	1.525	1.622	0.278	0.000	0.000	0.006	0.000	0.042	0.010	0.000
2	2	BLOG06-20060130-012-0000388657	4	C2	0.500	0.385	0.000	0.126	0.126	0.000	0.000	0.000	1.069	1.509	0.063	0.000	0.000	0.000	0.000	0.000	0.000	0.000
3	3	BLOG06-20051228-024-0029331823	0	C2	0.318	0.168	0.032	0.032	0.016	0.000	0.169	0.032	0.878	1.602	0.161	0.000	0.000	0.000	0.000	0.000	0.016	0.016
4	318	BLOG06-20060125-001-0017173601	-1	D1	0.286	0.027	0.189	0.314	0.629	0.000	0.692	0.000	3.774	3.333	0.314	0.000	0.000	0.000	0.000	0.000	0.000	0.000
5	21	BLOG06-20060129-007-0002071310	-1	D1	0.231	0.088	0.230	0.177	0.459	0.000	0.698	0.009	2.102	1.784	0.345	0.000	0.000	0.000	0.000	0.000	0.000	0.000
6	258	BLOG06-20060129-013-0002084292	-1	D1	0.221	0.032	0.102	0.271	0.458	0.000	0.831	0.102	2.322	1.508	0.237	0.000	0.000	0.000	0.000	0.000	0.000	0.000
7	186	BLOG06-20060115-009-0012392334	0	D1	0.200	0.040	0.235	0.499	0.505	0.022	0.449	0.036	1.897	1.980	0.217	0.000	0.000	0.000	0.000	0.000	0.000	0.000
8	394	BLOG06-20060107-014-0021290959	-1	D1	0.200	0.022	0.181	0.301	0.422	0.000	0.602	0.000	2.470	1.687	0.301	0.000	0.000	0.000	0.000	0.000	0.000	0.000
9	201	BLOG06-20060216-006-0025870753	0	D1	0.200	0.039	0.425	0.098	0.556	0.000	0.408	0.000	1.683	1.618	0.376	0.000	0.000	0.000	0.000	0.000	0.000	0.000
10	135	BLOG06-20060217-010-0017399708	4	D1	0.196	0.047	0.223	0.223	0.409	0.000	0.465	0.167	2.156	1.506	0.223	0.000	0.000	0.000	0.000	0.000	0.000	0.000
11	44	BLOG06-20060214-003-0010523489	0	D1	0.193	0.068	0.000	0.000	0.000	0.000	1.166	0.000	2.699	2.147	0.123	0.000	0.000	0.000	0.000	0.000	0.000	0.000
12	24	BLOG06-20060130-011-0003925412	1	D1	0.193	0.084	0.000	0.088	0.000	0.000	1.313	0.000	2.298	1.313	0.175	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Figure 2. WIDIT Dynamic Tuning Interface for Opinion Detection



Fusion Weights

	AC	**HF**	**IU**	**LF**	**W1**	**W2**	**Threshold**	
Positive Opinion:	10	5	5	20	1	0.5	0.45	Diff
Negative Opinion:	30	5	5	20	1	0.5	0.1	0.1

Error Counts: 3411 correct, 10437 incorrect (4420 bad)
(notOp=6017, neg=1181, mixed=1913, positive=1326)

Best Error Counts: 3483 correct, 10365 incorrect (4348 bad)
(notOp=6017, neg=1136, mixed=1782, positive=1430)
Positive wts: AC10 + HF5 + IU5 + LF20 + W11 + W20.5, cut=0.45 (diff=0.1)
Negative wts: AC30 + HF5 + IU5 + LF20 + W11 + W20.5, cut=0.1

rel-sys	**cnt**	**Bcnt**	**show**	**hide**
1-2	1601	2101	○	○
1-3	1308	875	○	○
1-4	3108	3041	○	○
2-3	654	612	○	○
2-4	527	524	○	○
3-2	875	811	○	○
3-4	1038	971	○	○
4-2	785	811	○	○
4-3	541	619	○	○

show=, hide= Display Reset

CNT	**rel**	**p12**	**fp**	**fn**	**fxp**	**fxn**	**ACp**	**ACn**	**ACxp**	**ACxn**	**HFp**	**HFn**	**HFxp**	**HFxn**	**IUp**	**IUn**	**IUxp**	**IUxn**	**LFp**	**LFn**	**LFxp**	**LFxn**	**W1p**	**W1n**	**W1xp**	**W1xn**
1 *	4-4	4:3	0.38	0.08	0.01	0.01	0.00	0.00	0.00	0.00	6.00	0.00	0.00	0.00	5.00	1.00	0.00	0.00	6.00	0.00	0.00	0.00	38.00	33.00	3.00	3.00
2 *	4-4	4:4	0.51	0.16	0.07	0.00	0.00	0.00	0.00	0.00	36.00	13.00	9.00	0.00	6.00	2.00	0.00	0.00	6.00	1.00	0.00	0.00	80.00	25.00	15.00	1.00
3 *	4-4	4:4	0.28	0.04	0.13	0.02	0.00	0.00	0.00	0.00	4.00	0.00	2.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	4.00	2.00	1.00	2.00
4 *	4-4	4:4	0.26	0.05	0.05	0.00	0.00	0.00	0.00	0.00	68.00	10.00	12.00	0.00	7.00	0.00	2.00	0.00	9.00	0.00	2.00	0.00	149.00	53.00	23.00	2.00
5 *	4-4	4:4	0.41	0.14	0.10	0.01	0.00	0.00	0.00	0.00	9.00	7.00	6.00	0.00	14.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	35.00	16.00	6.00	3.00
6 *	4-4	4:3	0.48	0.24	0.00	0.00	0.00	0.00	0.00	0.00	89.00	41.00	0.00	0.00	36.00	5.00	0.00	0.00	4.00	9.00	0.00	0.00	227.00	83.00	0.00	0.00
7 *	4-4	4:4	0.28	0.08	0.00	0.00	0.00	0.00	0.00	0.00	12.00	8.00	0.00	0.00	20.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	43.00	11.00	0.00	0.00
8 *	4-4	4:4	0.71	0.15	0.08	0.00	0.00	0.00	0.00	0.00	2.00	1.00	0.00	0.00	4.00	0.00	1.00	0.00	0.00	0.00	0.00	0.00	10.00	4.00	0.00	0.00
9 *	4-4	4:4	0.40	0.00	0.38	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	3.00	0.00	3.00	0.00	0.00	0.00	0.00	0.00	4.00	0.00	3.00	0.00
10 *	4-4	4:4	0.24	0.04	0.04	0.00	0.00	0.00	0.00	0.00	5.00	0.00	2.00	0.00	3.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	37.00	9.00	0.00	0.00
11 *	4-4	4:4	0.46	0.28	0.06	0.01	0.00	0.00	0.00	0.00	23.00	21.00	2.00	0.00	16.00	3.00	1.00	0.00	0.00	0.00	0.00	0.00	67.00	40.00	15.00	4.00
12 *	4-4	4:4	0.32	0.13	0.01	0.00	0.00	0.00	0.00	0.00	40.00	13.00	2.00	0.00	10.00	3.00	0.00	0.00	6.00	1.00	0.00	0.00	116.00	88.00	6.00	1.00

Figure 3. WIDIT Dynamic Tuning Interface for Polarity Detection

4. RESULTS

The effect of noise reduction is shown in Figure 4. Each bar in Figure 4 represents the average gain in performance by noise reduction. In order to isolate the influence of noise reduction, we computed the performance differences of all system pairs whose only differences were noise reduction and averaged the results. The upward direction of the bars indicates the gain in performance by noise reduction, which is consistent across evaluation metrics as well as types of performance (i.e., topical, opinion, polarity).

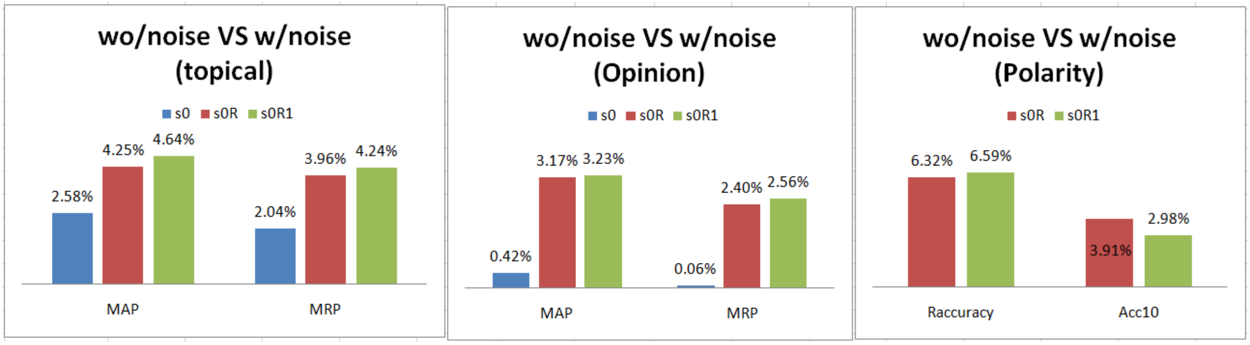


Figure 4. Noise Reduction Effect

As described in the methodology section, our approach to targeted opinion detection combined topic reranking to optimize the on-topic retrieval, opinion reranking to integrate opinion detection into on-topic retrieval, and fusion at various levels to affect the complementary combination of multiple methods as well as sources of evidence. Table 1 and 2 show the topic and opinion MAPs of the reranked results in comparison with the initial retrieval results (i.e. baseline). Topic MAP is the MAP computed using only the topical relevance (i.e., document is topically relevant), whereas opinion MAP is computed using the opinion relevance (i.e., document is topically relevant and opinionated).

Among the three main strategies, opinion reranking proved most useful (15% improvement in opinion MAP for short query, 11% improvement for long query over topic rerank results). Although topic reranking resulted in 16% improvement in topic MAP (short query) and 9% improvement (long query) over the initial retrieval results (Table 1), it reduced the opinion MAP by 2.4% for short query and showed only 5.9% improvement for long query.

Table 1. Topic MAP

	Short Query	Long Query	Fusion
Baseline	.3367	.3736	.3893 (+9.5%)
Topic Rerank	.3889 (+15.5%)	.4082 (+9.3%)	.4189 (+2.6%)

Table 2. Opinion MAP

	Short Query	Long Query	Fusion
Baseline	.2640	.2817	.2900 (+2.9%)
Topic Rerank	.2579 (-2.4%)	.2983 (+5.9%)	.3305 (+7.3%)
Opinion Rerank	.2959 (+14.7%)	.3303 (+10.7%)	.3343 (+1.2%)

In other words, topic reranking can hurt the opinion performance by over-boosting topically relevant documents to top ranks. , our The ability of opinion reranking method to compensate for the adverse effect of topic reranking is demonstrated in the short query column of Table 2, where opinion reranking shows 15% improvement over topic reranking compared to 12% improvement over baseline. The fusion columns of the tables show only marginal improvements over the best non-fusion results (i.e., long query) except for the topic rerank row in Table 2, which shows 7.3% improvement. This suggests the usefulness of fusion by illustrating its capacity to compensate for single method's poor performance.

Figure 5, which displays average performance improvements over baseline by topical (top chart) and opinion (bottom chart) relevance, shows the effect of topic reranking on the left side (r1-r0) and opinion reranking effect on the right side (r2-r0). The green bars (s0R1) indicate dynamic tuning results, while the red bars (s0R) represent results without dynamic tuning. The average performance improvement over baseline is computed by averaging the performance differences between all system pairs that are identical in all aspects except for the parameter in question (e.g., baseline vs. topic reranking with dynamic tuning). The fact that all the bars point upward shows that reranking is generally beneficial by all three measures (MAP, MRP, P@10). The markedly taller green bars on the right side suggests that dynamic tuning is quite effective in optimizing the opinion reranking formula.

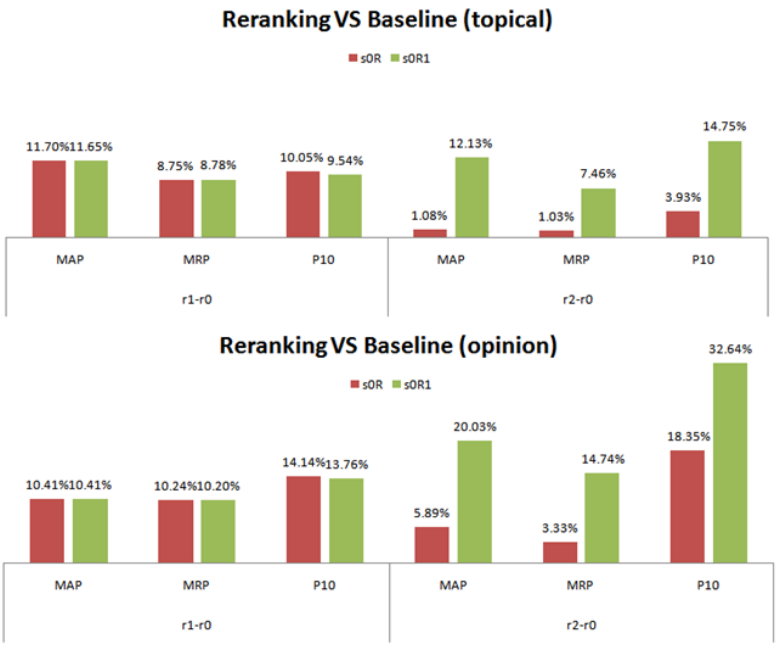


Figure 5. Dynamic Tuning Effect

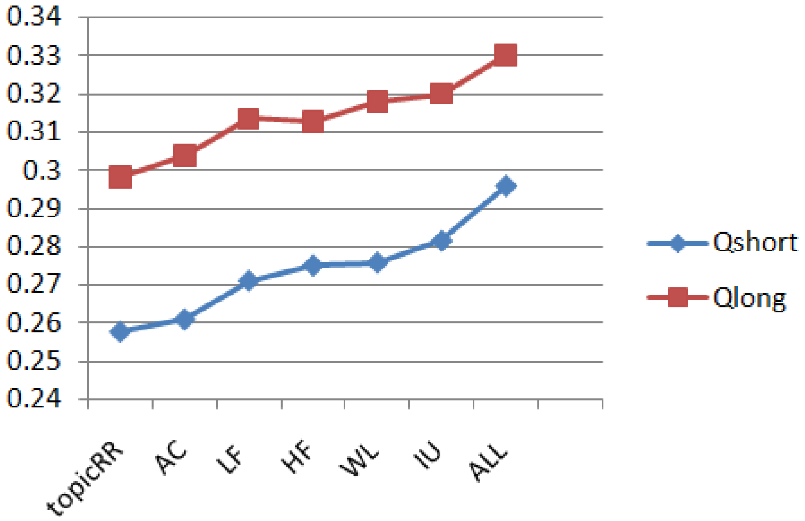


Figure 6. Opinion Reranking Factors

Figure 6 displays the performance chart of opinion reranking factors. Each plot points except “ALL” and “topicRR” represents the opinion MAP achieved by a single opinion module. ALL corresponds to the performance of the combined modules and topicRR, the topic reranked performance. The chart shows that all of the WIDIT opinion modules contribute to opinion detection and the combined result outperforms the best single method (IU), which once again demonstrates the advantage of fusion.

Opinion detection performances of TREC-2007 participants are shown in Figure 7. Each line, representing a TREC participant, starts on the left with the MAP of his/her baseline system and ends on the right with the MAP of the opinion detection system. The group of lines on the left shows participants whose opinion detection strategy had rather severely detrimental effects on performance, the middle lines show those with negligible opinion detection effects, and the lines on the right side show the ones with effective opinion detection methods. The fact that only one third of the TREC participants were able to devise effective opinion detection strategies reflects the difficulty associated with the targeted opinion detection task. Incidentally, the WIDIT performance line is the second from the top in the right side of Figure 7.

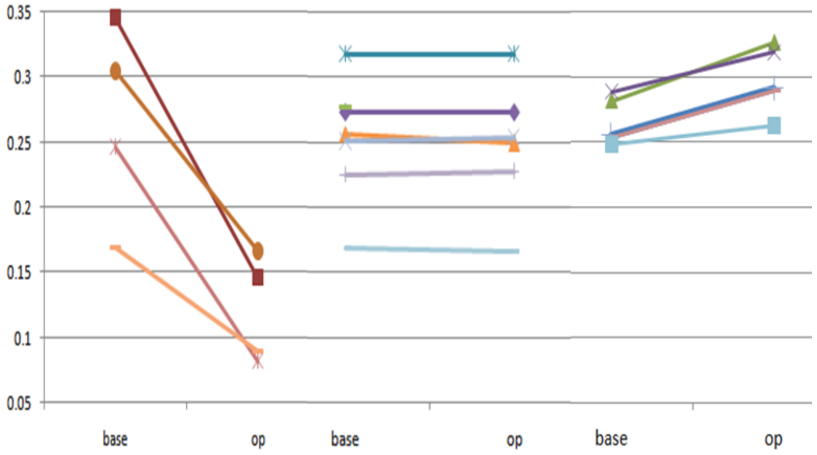


Figure 7. Opinion Detection Performances in TREC-2007

5. Conclusion

Our experimental results showed the effectiveness of combining multiple complementary lexicon-based methods for opinion detection. The analysis of the results also revealed that Dynamic Tuning is a useful mechanism for fusion, and post-retrieval reranking is an effective way to integrate topical retrieval and opinion detection as well as to optimize the baseline result by considering factors not used in the initial retrieval stage.

References

Bartell, B. T., Cottrell, G. W., & Belew, R. K. (1994). Automatic combination of multiple ranked retrieval systems. Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval.

Fox, E. A., & Shaw, J. A. (1995). Combination of multiple searches. Proceeding of the3rd Text Rerieval Conference (TREC-3), 105-108.

Frakes, W. B., & Baeza-Yates, R. (Eds.). (1992). Information retrieval: Data structures & algorithms. Englewood Cliffs, NJ: Prentice Hall.

Lee, J. H. (1997). Analyses of multiple evidence combination. Proceedings of the ACM SIGIR Conference on Research and Development in Information Retrieval, 267-276.

Robertson, S. E. & Walker, S. (1994). Some simple approximations to the 2-Poisson model for probabilistic weighted retrieval. Proceedings of the 17th ACM SIGIR Conference on Research and Development in Information Retrieval, 232-241.

Wiebe, J., Wilson,T., Bruce, R., Bell, M., & Martin, M. (2004). Learning subjective language. Computational Linguistics, 30 (3), 277–308.

Wilson, T., D. R. Pierce, and J. Wiebe (2003). Identifying opinionated sentences. Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, 33–34.

Yang, K. (2002). Combining Text- and Link-based Retrieval Methods for Web IR. Proceedings of the 10th Text Rerieval Conference (TREC2001), 609-618.

Yang, K. & Albertson, D (2004). WIDIT in TREC2003 Web track. Proceedings of the 12th Text Retrieval Conference (TREC2003), 328-336.

Yang, K., Yu, N., Valerio, A., & Zhang, H. (2007). WIDIT in TREC 2006. Proceedings of the 15th Text Retrieval Conference (TREC2006).

Yang, K., Yu, N., Wead, A., La Rowe, G., Li, Y. H., French, C., & Lee, Y. (2005). WIDIT in TREC2004 Genomics, HARD, Robust, and Web tracks. Proceedings of the 13th Text Retrieval Conference (TREC2004).