Building an Entity-Centric Stream Filtering Test Collection for TREC 2012

John R. Frank[1], Max Kleiman-Weiner[1], Daniel A. Roberts[1]
Feng Niu[2], Ce Zhang[2], Christopher Ré[2],
Ian Soboroff[3]
1 KBA Organizers, Massachusetts Institute of Technology, jrf@mit.edu
2 University of Wisconsin-Madison
3 National Institute of Standards and Technology Gaithersburg, MD ian.soboroff@nist.gov

ABSTRACT

The Knowledge Base Acceleration track in TREC 2012 focused on a single task: filter a time-ordered corpus for documents that are highly relevant to a predefined list of entities. KBA differs from previous filtering evaluations in two primary ways: the stream corpus is >100x larger than previous filtering collections, and the use of entities as topics enables systems to incorporate structured knowledge bases (KB), such as Wikipedia, as external data sources. A successful KBA system must do more than resolve the meaning of entity mentions by linking documents to the KB: it must also distinguish **centrally relevant** documents that are worth citing in the entity's WP article. This combines thinking from natural language processing (NLP) and information retrieval (IR). Filtering tracks in TREC have typically used queries based on topics described by a set of keyword queries or short descriptions, and annotators have generated relevance judgments based on their personal interpretation of the topic. For TREC 2012, we selected a set of filter topics based on Wikipedia entities: 27 people and 2 organizations. Such named entities are more familiar in NLP than IR. We also constructed an entirely new stream corpus spanning 4,973 consecutive hours from October 2011 through April 2012. It contains over 400M documents, which we augmented with named entity classification tagging for the ~40% of the documents identified as English. Each document has a timestamp that places it in the stream. The 29 target entities were mentioned infrequently enough in the corpus that NIST assessors could judge the relevance of most of the mentioning documents (~91%). Judgments for documents from before January 2012 were provided to TREC teams as training data for filtering documents from the remaining hours. Run submissions were evaluated against the assessor-generated list of citation-worthy documents. We present peak F_1 scores averaged across the entities for all run submissions. High scoring systems used a variety of approaches, including simple name matching, names of related entities from the knowledge base, and support vector machines. Top scoring systems achieved F_1 scores in the high 30s or low 40s depending on score averaging techniques. We discuss key lessons learned at the end of the paper.

Categories & Subject Descriptors: H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval – Information Filtering; H.3.m [Information Storage and Retrieval]: Miscellaneous – Test Collections; I.2.7 [Natural Language Processing] Text analysis – Language parsing and understanding
General Terms: Experimentation, Measurement

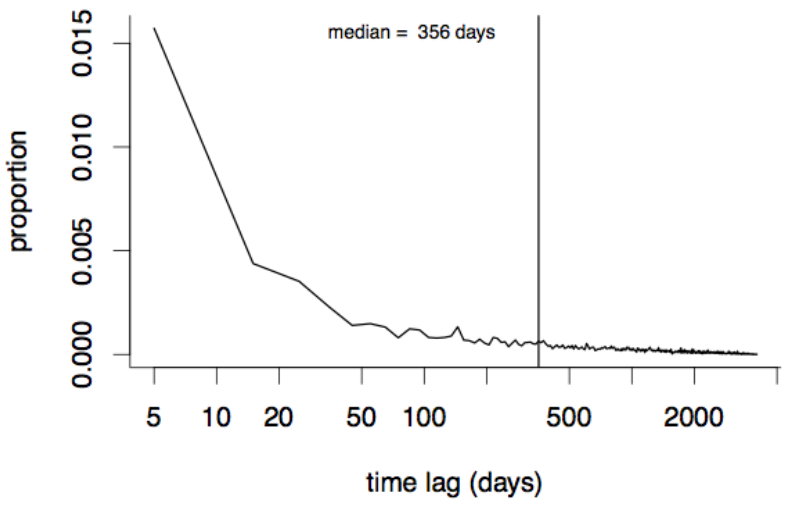
1. Introduction

We organized TREC KBA to help accelerate the construction and maintenance of large KBs. KBA builds on ideas from the TAC KBP evaluation [Ji 2011], which has shown that entity linking is an increasingly well understood problem. KBA pushes entity linking algorithms to larger data sizes and pushes beyond coreference resolution to focus on the user task of building a KB. KBA seeks cross-pollination of ideas between NLP and IR.

As a filtering task, KBA systems must decide on each document using only the currently available data at that point in the stream. At the start of the evaluation, participants were given "urlname" identifiers of 29 target entities and relevance rating judgments generated by NIST assessors for documents before 2012, i.e.

stream_time.epoch_ticks<1325376000 seconds. Participants' systems generated lists of highly relevant documents after the 2011-to-2012 cutoff.

The entity-centric nature of this evaluation has similarities with both topic filtering and adaptive filtering [Soboroff 2002]. While participants' systems did not have access to the judgments after January 2012, the query entities were certainly active and evolving in the world. At each hour of the stream corpus, teams could use external data sources that came into existence at that hour or earlier. This matches the real world task of monitoring an entity in streams of unstructured content, which is a common task amongst financial analysts and other industries.



In examining citations in Wikipedia (WP), we observed a considerable time lag between the publication date of cited news articles and the date of an edit to WP creating the citation. Figure 1 plots this time lag for a sample of ~60,000 web pages cited by WP articles in the Living_people category. The median time is over a year and the distribution has a long and heavy tail.

In selecting entities, we studied correlations between WP edit frequency versus mention frequency in the news stream. Figure 2 shows samples of WP entities that are frequently mentioned in the news. The vertical axis shows how frequently their WP articles were updated during the same time period. While some entities have very frequent mentions in the stream and correspondingly many edits in WP, the majority experience updates to their WP article much less frequently than their mention frequency.

Such stale entries are the norm in any large knowledge base (KB), because the number of humans maintaining the knowledge base is far fewer than the number of entities in the KB. Further, the number of mentions is much larger than the number of entities. This mismatch between human maintainers and the large stream of mentions to entities of interest is effectively a definition of a "large" KB.

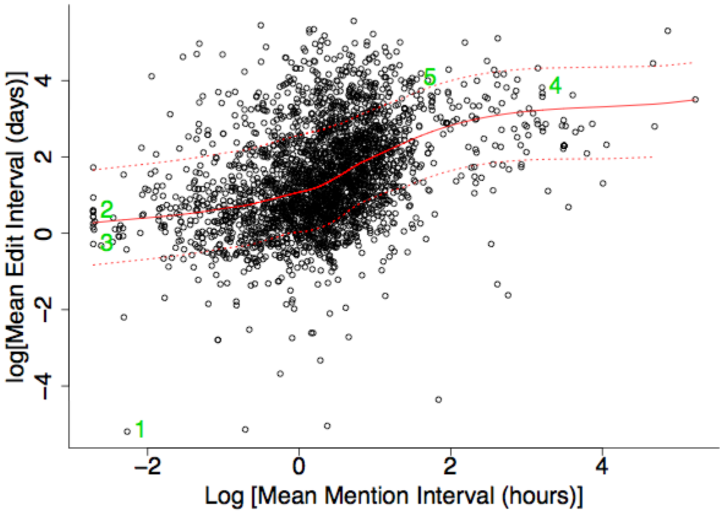


Figure 2: Edit interval vs. mention interval in the stream corpus shows a complex relationship between mentions in the news and edits in Wikipedia.   (1) Death of Michael Jackson, (2) Muammar Gaddafi, (3) Barack Obama, (4) Aung Myint Oo, (5) Allan Asher. The averages used in this plot are from all edits and all exact-match mentions in a five week window overlapping September and October 2011. Edits were gathered directly from the WP

APIs using the pywikipediabot library, and mentions were measured in a precursor to the news stream used for the kba-stream-corpus-2012. Both axes use natural logarithms. WP redirects were included, so pages like Death of Michael Jackson have boosted mention rates from phrases like “Ed Chernoff,” which redirects to that page. The leftmost points occur at the four-minute interval mark, which is the refresh rate of the underlying feed. The paucity of points below approximately one-day edit intervals (zero on vertical axis) appears to correspond to the locking down of WP pages experiencing edit wars or vandalism.

The remainder of this paper is organized as follows: the KBA stream corpus is described in Section 2; assessor judgments are described in Section 3; KBA task structure details in Section 4; entities and recall of annotation in Section 5; evaluation measurements in Section 6; and **lessons learned** are discussed in Section 7.

2. Corpus

To enable this task, we wanted a larger and longer stream corpus than had been previously released, so we built a new corpus, called the kba-stream-corpus-2012. The corpus is approximately 9TB of 'raw' text. To help TREC teams jump quickly into the data, we produced two forms of community metadata: 1) a ‘cleansed’ version showing only visible text of apparently English documents, and 2) the output of Stanford NER running over the cleansed text, which provided tokenization, sentence chunking, lemmatization, and named entity labeling on a per-token basis. After XZ compression, the full data set is 1.9TB. It was serialized into flat files using Thrift, and tools for interacting with the corpus have been released in an open code repository https://github.com/trec-kba/kba-corpus/ The corpus is encrypted with GPG and stored in Amazon’s Public Dataset hosting http://trec.nist.gov/data/kba.html, which has proven to be very useful.

	news	**linking**	**social**
number of documents	134,625,663	5,400,200	322,650,609
size of 'raw'	8072GB	350GB	531GB
number with 'cleansed' & 'ner'	53,245,364	5,343,568	309,071,598
size of 'ner'	1753GB	222GB	1723GB

Table 1: Corpus Size Statistics for each Substream

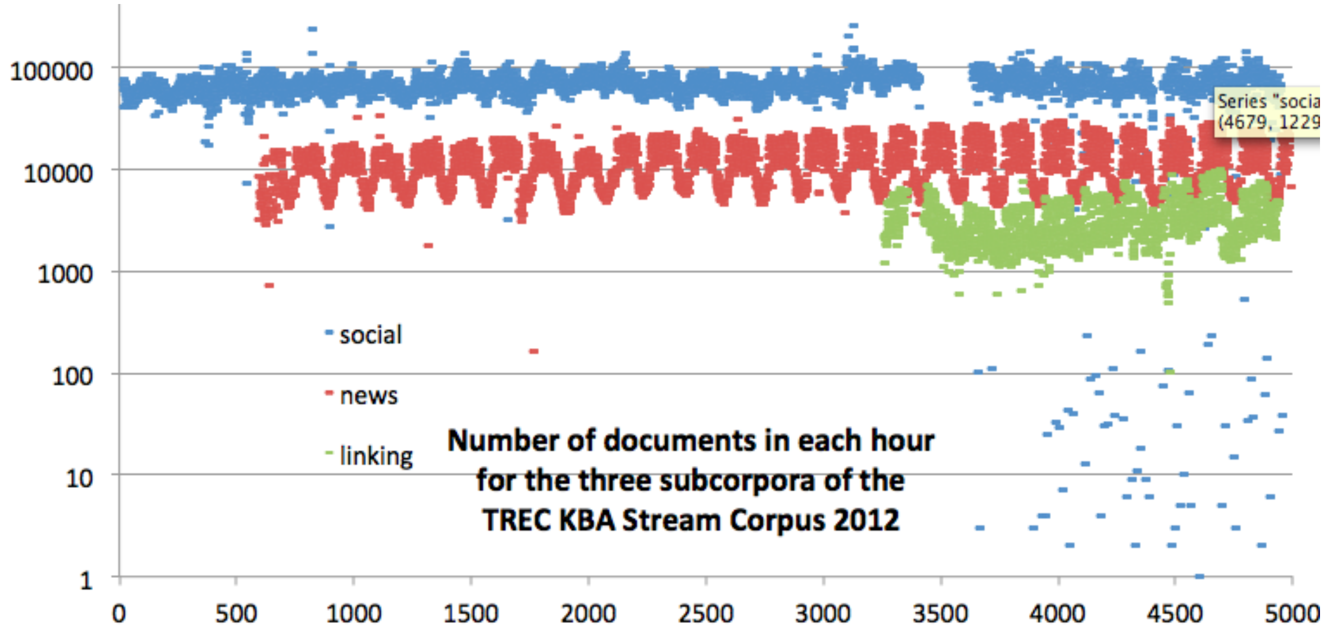


Figure 3: document counts in each substream over the 4,973 hours in the stream corpus.

There are three substreams in this corpus:

1 **Linking** substream**:** Brian Eoff and Hilary Mason at Bitly generously donated a list URLs that were shortened at bitly.com. Timestamp of the shortening event places it in the stream. To select a substream from Bitly's massive data stream, we designed a set of ~10k queries that Brian used to query their internal index of the full text of all pages. These ~10k queries are the Wikipedia page titles of the candidate topic entities and also titles of all of their in- and out-linking pages in an English Wikipedia snapshot from January 2012. The queries that matched a given text are in the 'source_metadata' property.

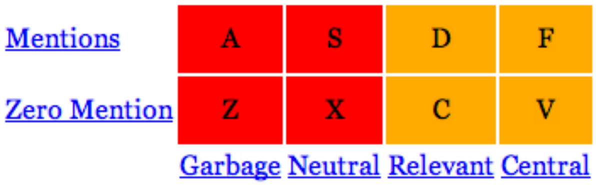
2 **Social** substream: an aggregated stream of blogs and forums with rich category metadata.

3 **News** substream: acquired a URLs (and timestamps) from public newswires, re-fetched content.

3. Annotation

To enable the KBA 2012 evaluation, we prepared a set of assessor tasks using a three different name matching techniques and sets of alternate names. As described in Section 5, we estimate the recall of this system to be ~91%, and 15% of the tasks were judged as citation worthy. The three techniques included thresholding of Jaro Winkler similarity scores between phrases labeled by the Stanford NER metadata and also general bigrams and trigrams of all tokens in the documents.

Assessors were instructed to "use the wikipedia article to identify (disambiguate) the entity, and then imagine **forgetting** all info in the WP article and asking whether the text provides any information about the entity."



Mentions	A	S	D	F
Zero Mention	Z	X	C	V
	Garbage	Neutral	Relevant	Central

Figure 4: Annotation Tool Grid. Letters correspond to keystrokes for rapid input.

Rows:

- **Mentions**: Document explicitly mentions target entity, such as full name, partial name, nickname, pseudonym, title, stage name.
- **Zero Mentions**: Document does not directly mention target. Could still be relevant, e.g. metonymic references like "this administration" --> "Obama". See also synecdoche. A document could also be relevant to target entity through relation to entities mentioned in document -- apply this test question: can I learn something from this document about target entity using whatever other information I have about entity?

Columns:

- **Garbage:** not relevant, e.g. spam.
- **Neutral:** Not relevant, i.e. no info could be deduced about entity, e.g., entity name used in product name, or only pertains to community of target such that no information could be learned about entity, although you can see how an automatic algorithm might have thought it was relevant.
- **Relevant:** Relates indirectly, e.g., tangential with substantive implications, or topics or events of likely impact on entity.
- **Central:** Relates directly to target such that you would cite it in the WP article for this entity, e.g. entity is a central figure in topics/events.

contains_mention	7991	3862	13971	7806
zero_mention	15367	163	61	0
	garbage	neutral	relevant	central

Table 2: Number of judgments

From manually examining a few hundred actual citations from WP articles in Category:Living_people, we observed a **non-mentioning citation fraction of about one-in-twenty.** Such articles can be very difficult to find, because instead of mentioning the entity directly, they mention related entities. The annotation data had zero non-mentioning+central and very few non-mentioning+relevant.

The following anecdotes from annotation give a flavor for the annotation.

- Mentions in border text fragments, often called "chome," are usually mentioning+garbage.
- Entity-is-concept like "this book's plot hints of [influence from] Alexander McCall Smith" is mentioning+neutral.
- Some articles about Putin and Moscow politics are non-mentioning+relevant for Boris Berezovsky (businessman).
- Some articles about ProBlogger but not specifically about Darren Rowse are non-mentioning+relevant. An article *by* ProBlogger describing his business strategy might be called non-mentioning+central by some assessors.
- Articles by a journalist entity, such as Charlie Savage, are only relevant unless it discusses the journalist himself, in which case they are central.

Generally, assessors form their own interpretations of the idea of citation worthiness. Instead of viewing one interpretation as more correct than others, we consider these different interpretations as representing variation fundamental to the IR task at hand. This differs from natural language processing assignments in which annotators label phrases in documents using guidelines that are intended to be universal and have a single correct interpretation. The tension between these viewpoints is a key part of KBA.

97.6% +/- 1.4% (N=5365)	coref				

69.5% +/- 2.7% (N=1352)	central			
70.9% +/- 2.0% (N=2403)		relevant		
58.4% +/- 3.4% (N=884)			neutral	
84.9% +/- 2.0% (N=2599)				garbage
82.6% +/- 1.8% (N=3200)	central	relevant		
89.0% +/- 1.7% (N=3551)	central	relevant	neutral	

Table 3: Interannotator agreement scores for each relevance rating level. Percentages are overlap amongst N duplicate annotation tasks. Some duplicate tasks were to the same assessor, most were to a different assessor. The full matrix of agreement scores per entity is available in the task definition tarball. In generating scores, we used the lowest rating for a document as the official judgment.

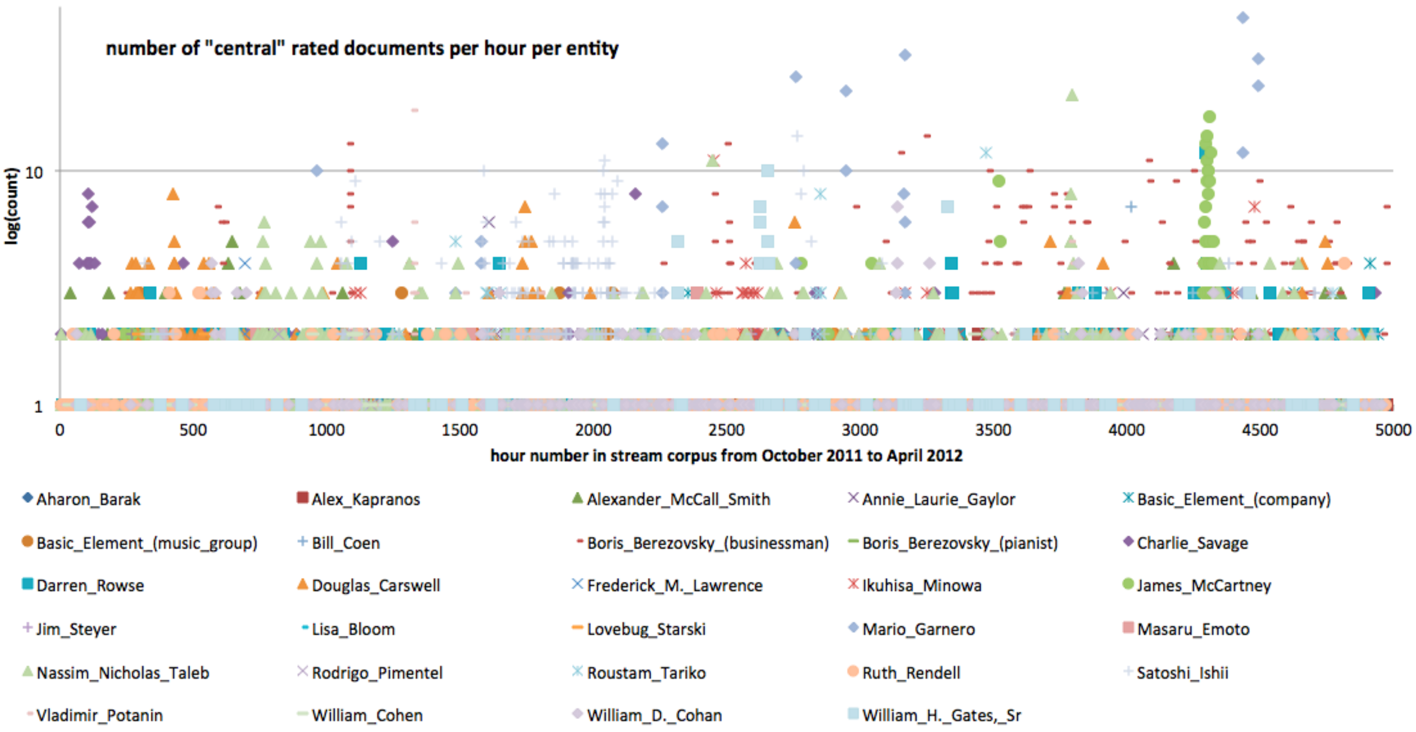


Figure 5: number of central-rated documents per hour. Several visible spikes correspond to events, such as James McCartney suggesting that the sons of The Beatles form "The Beatles -- the next generation."

4. KBA Task Structure

Participants in KBA were instructed to apply their systems to each hourly directory of corpus data in chronological order. For each hour, before processing the next hour, systems are expected to emit a list of assertions connecting documents (identified by stream_id) and entities (identified by urlname). The goal is to identify only central-rated documents. For each assertion, systems must generate a confidence score in the range (0, 1000]. Conceptually, the confidence score indicates the likelihood of a human agreeing that the document is citation worthy for that entity. There is no requirement that confidence scores be actual probabilities, however they must be normalized to be integers ranging from zero to a thousand.

The run submission format is a five-column text file with whitespace delimiters. First two columns indicate team ID and system ID; same for all rows. Third, fourth are stream_id, urlname. Last is confidence score:

```
MyTeam Sys1 1328057520-4e92eb721bfbfdfa0b1d9476b1ecb009 Bill_Coen 315
```

5. Entities and Recall of Annotation

Table 4 shows the number annotations per entity. To estimate the amount of recall loss in the assessor task generation process, we annotated all 113 such assertions made by thirty or more runs that were not previously judged. In these, we found 11 (10%) garbage, 16 (14%) neutral, 42 (37%) relevant, and 44 (39%) central. Approximately a third of the 44 "central" documents in this top-voted set were duplicates of the same syndicated news wire text echoing across different web sites.

To further analyze recall errors in tasks presented to assessors, we pooled assertions that appeared in fewer than thirty runs and more than eight and sampled twenty for each entity from this tier. The counts for each entity are shown in Figure 6. The recall loss averaged across entities is 9%. Weighting by the number of unjudged assertions, the recall loss is ~1%. For Annie_Laurie_Gaylor and Jim_Steyer, the recall loss was 50% and 80% respectively, indicating insufficient alternate names in the system that generated tasks for assessors.

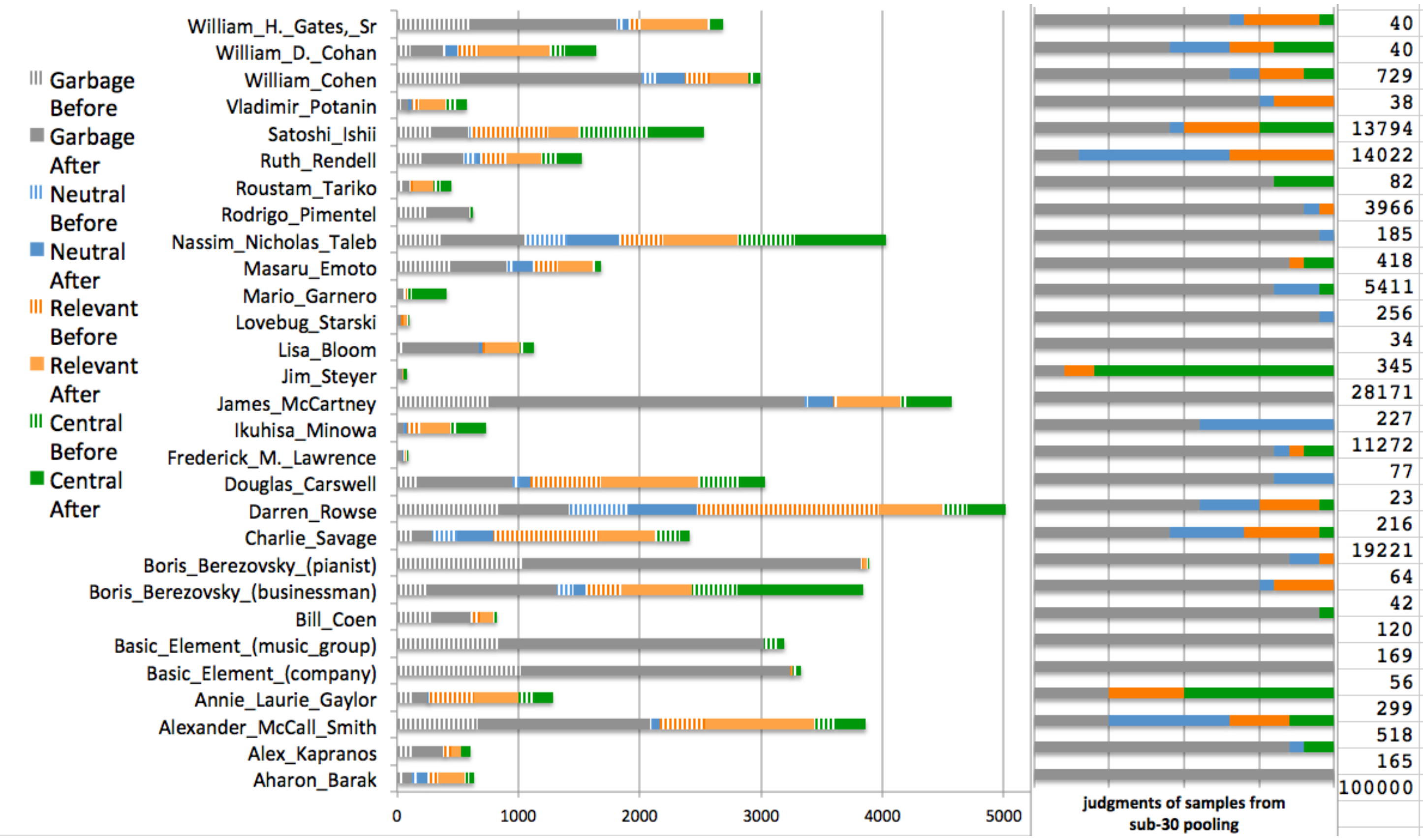


Figure 6. Number of judgments of each type per entity. Mentioning and non-mentioning are added. Fixed-width bar charts show quantity of each rating level found in a random sample of 20 assertions from the tier of unjudged assertions with fewer than 30 "votes," i.e. asserted by a run. Rightmost column shows quantity of assertions in top 100,000 pooled assertions, which means as few as 8 votes and as many as 35.

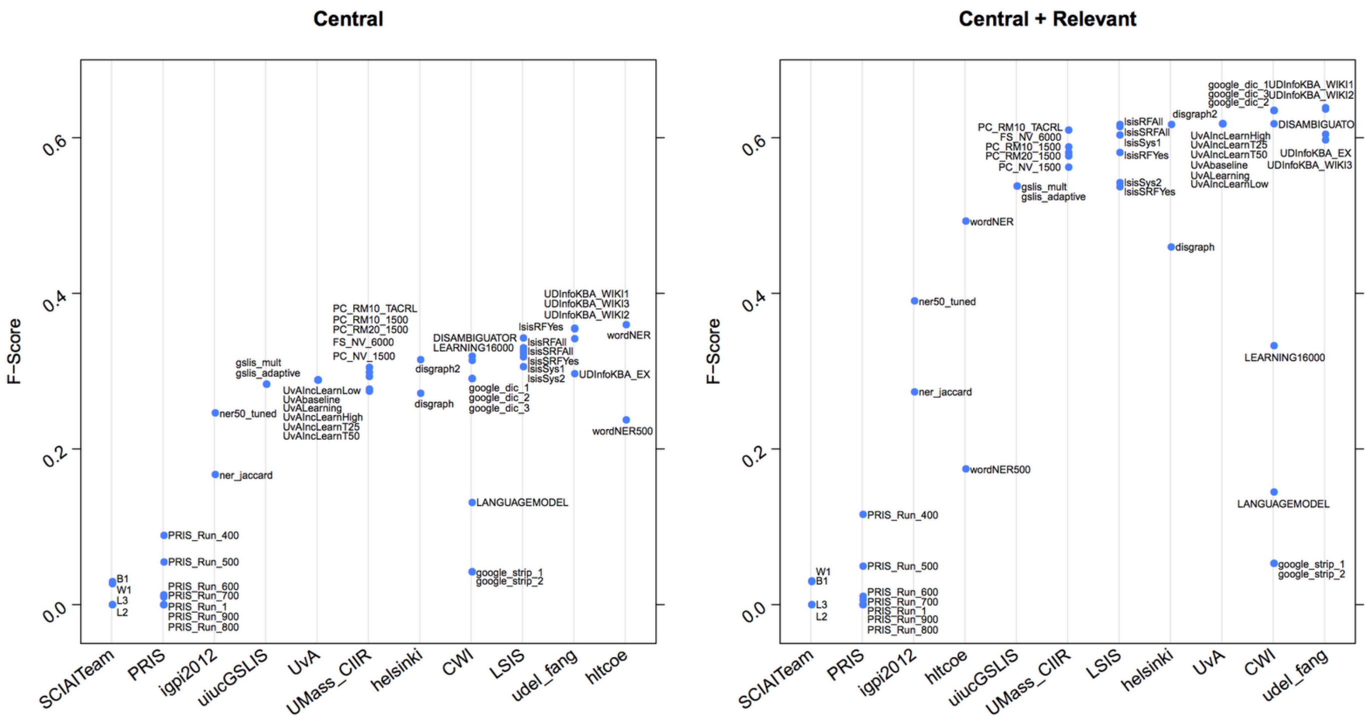


Figure 7a: Ranked by team's run with highest *average* F-score (averaged across the 29 target entities).

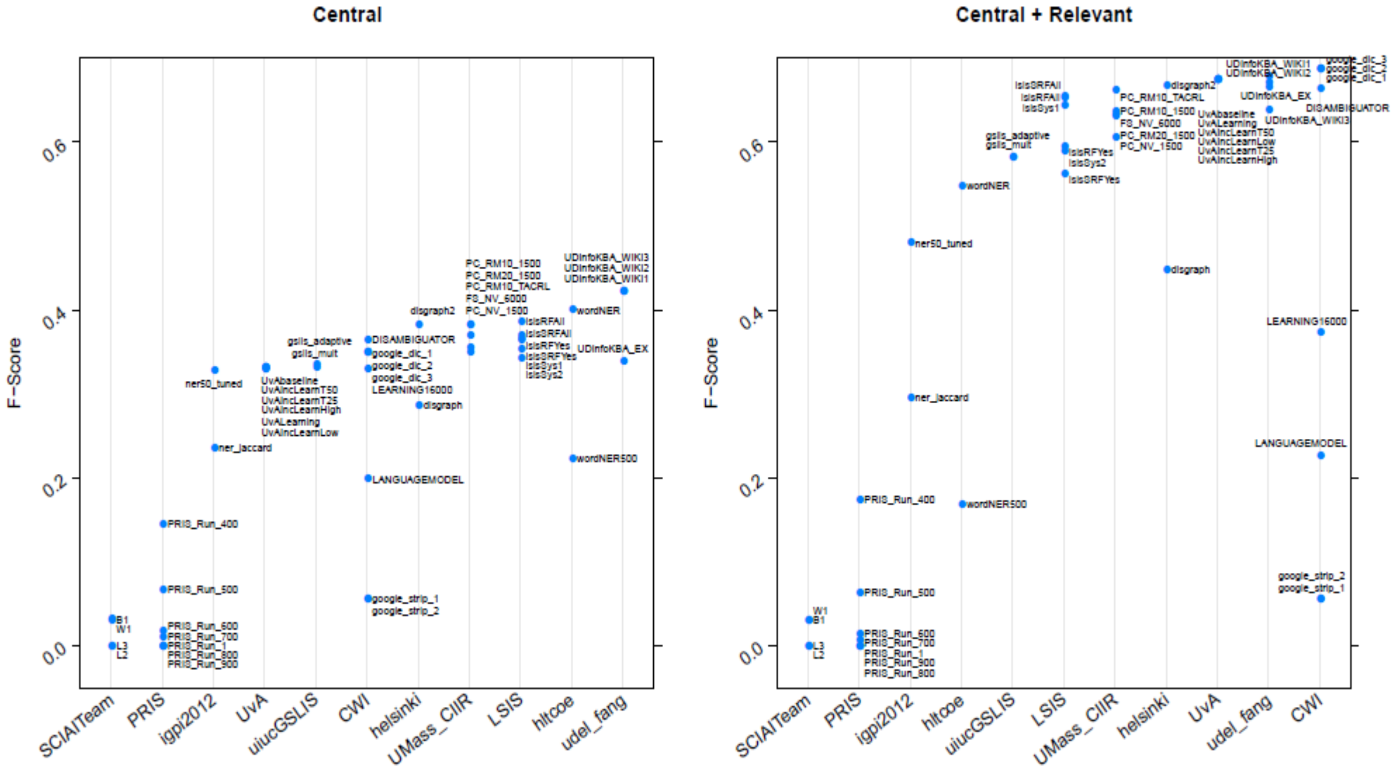


Figure 7b: Ranked by highest F-score computed from macro-averaged precision and recall.

6. Evaluation of the KBA Run Submissions

By default the KBA scoring tool treats central documents as positives and non-central as negatives. For each run submission, the scoring tool sweeps a confidence cutoff across all its assertions to count true-positives, false-positives and false-negatives. From this confusion matrix, the tools computes precision, recall and F-score for each entity. The tool includes an optional flag for treating the unannotated corpus as negatives so that any if a run asserted an unannotated document with a confidence score above the confidence cutoff, it is treated as a false positive. An additional flag causes relevant-rated documents to be included with central-rated as part of positive set. The tool resolves annotator disagreement by taking the lowest rating for a given (stream_id, urlname) pair.

Figure 7a&b show scores for all runs. These scores use macro-averaging rather than micro-averaging, i.e. we computed a run's score for each entity and then averaged across entities with equal weight per entity. 7a shows team ranking based on averaging the F-scores across the entities. 7b shows team ranking based on re-computing the F-score after macro-averaging the precision and recall. See Appendix for run descriptions.

7. Future Work & Lessons Learned

The KBA corpus enables a variety of future research opportunities. More than half of the corpus is non-English, and has not yet been explored. The timestamps could enable spike detection and other temporal correlation studies. There are many more entities to examine in the KBA corpus, including non-person entities, such as pharmaceutical compounds.

While assessors generated mentioning–versus–non-mentioning labels (coref) with 97% interannotator agreement, agreement on relevance ratings was ~70%. This indicates that it is more difficult for humans, and the scores in table 7 indicate that distinguishing "relevant" from "central" was very difficult for automatic systems. The definition of "citable" varies across entities. We plan to assemble richer tagging guidelines.

Several teams commented on the difficulty of working with the large corpus. We are investigating possible centralized cluster resources for future KBA tasks. To encourage teams to jump into the data early, we may borrow from Kaggle by offering a web service that scores runs before the final submission deadline.

At the conference, we hope to gather more lessons learned from teams. We note the highest scoring systems in KBA 2012 were split between rich feature engineering from the KB versus focusing on machine learning tools, such as SVMs. In the future a combination of these approaches might score even higher.

References:

Building a Filtering Test Collection for TREC 2002, Ian Soboroff and Stephen Robertson
Overview of the TAC2011 Knowledge Base Population Track, Heng Ji, Ralph Grishman, Hoa Dang

Appendix: Run Descriptions

Key for Figures 7a & 7b. Condensed from texts submitted by teams. Read teams' papers for details.

hltcoe wordNER, wordNER500: Support vector machine using tokenized words and named entities as features. Features were bags of words and bags of named entities.

udel_fang UDInfoKBA_EX: If a document has an exact match with the query entity, the ranking score will be 1000. In other cases, the ranking score will be 0.

udel_fang UDInfoKBA_WIKI1, UDInfoKBA_WIKI2, UDInfoKBA_WIKI3: Exact match with query entity. For each entity, extract entities from internal link on its WP page; such entities also have WP page. For each of filtered doc, we then count occurrences of related entities from WP. Scores reflect the occurrences.

LSIS lsisRFAll, lsisRFYes, lsisSRFAll, lsisSRFYes, lsisSys1, lsisSys2: With help of WP, variant names have been found for each topics. As the process go through the stream, each index is queried with the topic's url_name as well as the variants. Then for each document, statistics are computed based on what can be found in the document, what can be found in the current day, and what has been seen on the previous days currently in the queue. Those statistics are used for training a RandomCommittee classifier which uses multiple RandomForrest classifiers. Two classifications are done as so, one to separate garbage from relevant and Central, the other one to choose between relevant and central.
CWI DISAMBIGUATOR: This method uses the words in the dbpedia page of the entities to disambiguate the ambiguous entities. A documents is considered central if it contains the label of the dbpedia entity and at least one word that occur in the dbpedia page of this entity.
CWI google_dic_1, google_dic_2, google_dic_3, google_strip_1, google_strip_2: This system uses google cross-lingual dictionary's strings and probabilities to represent the entities and searches the documents for a match. This dictionary has two probabilities: P(entity|string) and P(string|entity).
CWI LANGUAGEMODEL: A language models was built using only the central documents. Then this model was used to rank the test documents. We compare each document with the perplexity measure.
CWI LEARNING16000: Find only central documents using a supervised approach. It uses a list of query strings learned from the trained data. Documents retrieve are those that exact match a string in this list. For each entity, a list of strings are used as a query.
helsinki disgraph2: Relation to named entities is detected by looking at the overlap of named entity graphs and document word collocation graph.
helsinki disgraph: Collapses entity title strings and documents into sets of words and looks for fraction of exact match overlap with entity titles. Relevance is fraction of entity title words that appear in doc.
UMass_CIIR FS_NV_6000, PC_NV_1500: This run performs retrieval over the entire collection without wrt to time using entity name and simple variants. Galago sequential dependence based retrieval over entire document stream; dirichlet smoothing. Combines original topic name with name variants from Wikipedia
UMass_CIIR PC_RM10_1500, PC_RM20_1500: Initial original query from PC_NV_1500. Incorporates entity concepts from extracted NER tags, 10 top weighted plus up to 10 entities from Wikipedia link neighborhood (incoming and outgoing topic names).
UMass_CIIR PC_RM10_TACRL: This run applies a TAC entity linking approach to filter the stream of documents. For this approach, all documents returned from PC_RM10_1500 are converted into TAC EL queries. A supervised TAC EL ranker is applied with the topic entity as the candidate set. KBA documents are re-ranked by their linker score to the topic entity. Ranking model is a linear model optimized with Coordinate Ascent incorporating dozens of features including surface form & document similarity functions.
UvAbaseline: baseline 2012 run
UvALearning, UvAIncLearnHigh, UvAIncLearnLow, UvAIncLearnT25, UvAIncLearnT50: Learning to rerank run, with incremental learning with high or low threshold, top 25 or 50 instances
uiucGSLIS gslis_adaptive, gslis_mult: Initial queries consist of wikitext extracted from each entitys history. We impose a document prior favoring docs with high in-link count. Only English docs with near-exact name match on entities are ranked. Query is updated monthly, as which point the weights of features (but not features themselves) are recalculated based on previously retrieved docs.
igpi2012 ner50_tuned: Using the top 50 popular named entities to compute the jaccard coefficient between the entity list for each document and the entity list for the positive doucments from the annotated set. Each topic has its own threshold tuned from the annotated dataset.
igpi2012 ner_jaccard: Compute the jaccard coefficient between the entity list for each document and the entity list for the positive doucments from the annotated set.
PRIS_Run_1,...,900: Relevance Feedback is first applied to our system according to the annotation data. Then Jaccard coefficient weighting scheme is used to calculate the relevance
SCIAITeam B1, L2, L3, W1: Lucene, with and without query expansion on different subsets of the data