Resorting Relevance Evidences to Cumulative Citation Recommendation for Knowledge Base Acceleration[*]

Jingang Wang
School of Computer Science
Beijing Institute of Technology
Beijing, China
bitwjg@bit.edu.cn

Dandan Song[†]
School of Computer Science
Beijing Institute of Technology
Beijing, China
sdd@bit.edu.cn

Chin-Yew Lin
Knowledge Mining Group
Microsoft Research
Beijing, China
cyl@microsoft.com

Lejian Liao
School of Computer Science
Beijing Institute of Technology
Beijing, China
liaolj@bit.edu.cn

Yong Rui
Knowledge Mining Group
Microsoft Research
Beijing, China
yongrui@microsoft.com

ABSTRACT

Most knowledge bases (KBs) can hardly be kept up-to-date due to time-consuming manual maintenance. Cumulative Citation Recommendation (CCR) is a task to address this problem, whose objective is filtering relevant documents from a chronological stream corpus and recommending them as candidate citations with different relevance levels (i.e., vital, useful, neutral and garbage) to target entities in knowledge bases. The biggest challenge of CCR is how to separate relevant documents into different levels, especially vital and useful. We mainly evaluate three kinds of relevance evidences, i.e., entities' profile pages, existing citations in KBs, and temporal signals, in two CCR tasks. In addition, a novel pseudo citation generation strategy is proposed to supplement annotation data for less popular entities. The effectiveness and robustness of these evidences are validated by integrating them into different approaches, including query expansion, classification and learning to rank. All the approaches outperform their own baselines. These evidences are not of the same significance in different CCR tasks, and our analysis reveals that classification approach has more potential.

1. INTRODUCTION

Knowledge Bases (KBs), such as Wikipedia and DBpedia, have shown great power in many applications including question answering, entity linking and entity retrieval. With the explosion of information on the web, it becomes critical to detect relevant documents and assimilate new information to entities in KBs in a timely manner. However, most KBs are maintained manually by volunteer editors, which are hard to keep up-to-date because of the limit number of editors and the huge volume of entities in KBs. As reported in [?], the median time lag between the publishing date of cited articles and the date of the citations created in Wikipedia is almost one year. Moreover, some less popular entities in KBs do not attract enough attentions from the editors, which makes the maintenance more challenging. This gap could be reduced if relevant documents could be automatically found as soon as they are published online and then recommended to the editors with different priorities. Because not all found relevant documents are of the same significance to the target entity, the editors could focus on the documents with high relevance levels firstly, evaluate them and supplement newly found vital information by citing the most relevant documents.

To address the above problem, Cumulative Citation Recommendation (CCR) was launched by the Text REtrieval Conference (TREC) Knowledge Base Acceleration (KBA) [1] Track from 2012. CCR is defined as filtering a time-ordered stream corpus for documents related to a predefined set of entities in KBs, such as Wikipedia and Twitter, and assigning a relevance level for each document-entity pair. **??** shows the overview of a CCR system.

Unlike traditional information retrieval and filtering tasks, CCR not only need to retrieve relevant documents from the stream, but also need to distinguish relevant documents according to their citation-worthiness to the target entities. However, there not exist an acknowledged regulation for editors to refer to decide whether a document be cited into the knowledge base. Therefore, A document judged as vitally relevant by an editor may be treated as relevant by another one. Although KBA has defined 4 different relevance levels for CCR: *vital*, *useful*, *neutral* and *garbage* to represent various citation-worthiness, the community has not come to an agreement on their explicit definitions and differences except some heuristic guidelines. KBA recognizes a vital document as the document which contributes timely update to a target entity, but it's still quite difficult to draw an obvious borderline between vital and useful documents. For example, for a famous person in Wikipedia, only the documents

[*] This work was done when the first two authors were visiting the Knowledge Mining Group of Microsoft Reseach.
[†] Corresponding Author

[1] http://trec-kba.org/

Figure 1: Overview of a CCR System

that trigger an update of her profile can be annotated as vital. However, if the person is a less noteworthy person from Twitter, a document simply describe how she spends her time may be treated as vital documents. Even in the ground truth data of KBA 2013, 361 instances of all 7413 ones contain inconsistent annotations by different annotators. In a word, it's a great challenge to define and verify the differences between the different relevance levels.

In this paper, we resort three extra evidences to differentiate them, including the profile pages of the target entity, existing citations in KBs and temporal signals in timeline. These evidences contain rich information we can utilize to distinguish documents with different levels of relevance. For a target entity, a relevant document would more or less repeat or supplement the content in its profile page. Besides, the documents possess same patterns with the existing citations are more likely to be relevant documents. In addition, most of the relevant documents appear around the time period in which the target entity are spotlighted.

CCR are divided into two sub-tasks: (i)**vital + useful:** distinguishing relevant (*vital + useful*) and irrelevant (*neutral + garbage*)documents, and (ii)**vital only:** distinguishing between *vital* and *useful* documents further. After taking these evidences into account, all the test approaches achieve the state-of-the-art performances in both tasks. Lots of approaches has been applied in the two tasks, such as query expansion, text classification and learning to rank, the proposed evidences indeed improve the distinguishing capabilities of them. Our analysis reveals that classification approach has more potential than the other approaches.

In addition, CCR are more challenging to less popular entities than popular ones because popular entities usually have enough annotations that can provide some clues for relevant document discovery. Nevertheless, for less popular entities, such as Twitter entities, there exist few annotations by assessors because of their few appearance in stream, which makes it difficult to capture relevance information for them. We propose a novel pseudo-citation generation method to supplement the training data.

To the best of our knowledge, this is the first work to verify the different relevance levels in CCR by exploring additional relevance evidences. The contributions are summarized as follows:

1. We implement a straightforward but effective filtering approach to discard irrelevant documents as many as possible beforehand, which makes our approach more efficient and practical.
2. We investigate three extra evidences and evaluate different roles they play in two CCR tasks.
3. We propose a novel pseudo-citation generation method to supplement training data for less popular entity with few annotation, relieving the impact of lack of training data for less popular entities in CCR.

The rest of this paper is organized as follows. After discussing related work in **??**, **??** formulates the problem and details the dataset used. Then, **??** introduces an effective and high-recall filtering strategy to reduce the workload to process the volume stream corpus. In **??**, we first present the relevance evidences utilized to distinguish documents with different relevance levels, then depicts our approaches in detail. This is followed by experimental results and further discussions in **??**. Finally, we conclude this paper and introduce future work in **??**.

2. RELATED WORK

We discuss several research topics which are related to CCR in this section.

Information Filtering.

Information filtering (IF) is a long-standing problem to split (usually large) data stream into useful and unuseful components and direct the useful part to interested end users, such as personal email filters based on personal profiles. Early researches treat IF as a text classification problem and mainly utilize content-based features [**?**, **?**, **?**] to address it. Currently, most works in IF focus on sociology filtering by incorporating relationships between users [**?**]. An IF system is fed with queries based on topics described by a set of keywords or short descriptions, while CCR adopts entities as queries, which possess strongly typed attributes and relationships.

Entity Linking.

Entity linking describes the task of linking a textural name of an entity to a knowledge base entry [**?**]. In [**?**], cosine similarity was utilized to rank candidate entities based on the relatedness of the context of an entity mention to a Wikipedia article. [**?**] proposed a large-scale named entity disambiguation approach through maximizing the agreement between the contextual information from Wikipedia and the context of an entity mention. [**?**] implemented a system named as WikiFy! to identify the important concepts in a document and automatically link these concepts to the corresponding Wikipedia pages. Similarly, [**?**] parsed the Wikipedia to extract a concept graph, measuring the similarity by means of the distance of co-occurring terms to candidate concepts. Besides identifying entity mentions in documents, CCR is required to evaluate the relevance levels between the document and the mentioned entity.

Query Expansion.

Query expansion (QE) is the process of supplementing a basic query with additional related terms to improve retrieval performance in information retrieval system [**?**]. The key idea of QE is expanding the query terms to the direction maximizing the similarity between the query terms and the terms in relevant documents. The existing query expansion approaches can be classified into two categories: automatic relevance feedback approaches and log-based query methods. Automatic relevance feedback methods, including explicit relevance feedback and pseudo relevance feedback, mine relevant terms through analyzing a subset of initial search results [**?**, **?**]. Log-based QE methods derive expansion terms for a query from the document set identified by user clicks recorded in search logs [**?**, **?**, **?**]. QE is adopted as a unsupervised baseline for comparison in this work.

Knowledge Base Acceleration.

TREC has hosted the CCR track in 2012 and 2013. In the past tracks, there are three mainstream approaches submitted by the participants: query expansion [**?**, **?**], classification, such as SVM [**?**] and Random Forest classifier [**?**, **?**, **?**], and ranking-based approaches [**?**, **?**]. In [**?**], a graph-based entity filtering method is implemented though calculating the similarity of word co-occurring graphs between an entity's profile and a document. [**?**] developed a time-aware evaluation paradigm to study time-dependent characteristics of CCR. However, most of these approaches only work well for entities with abundant annotation and are not suitable for less popular entities with few annotation.

3. PROBLEM STATEMENT AND DATASET

Given a target entity E from KBs (e.g., Wikipedia and Twitter) and a document D, our goal is to generate a confidence score $r(E, D) \in (0, 1000]$ for each document-entity pair, representing the citation-worthiness of D to E. The higher $r(E, D)$ is, the more likely D should be considered as a citation for E.

We use the stream corpus[2] of KBA 2013 in this paper. The stream corpus are composed of document collection and the target entity set.

Document Collection.

The document collection, stream corpus, is a time-ordered document collection provided by the TREC KBA 2013. The stream corpus contains nearly 1 billion documents published between Oct. 2011 and Feb. 2013.

Target Entity.

The target entity set is composed of 141 entities, 121 of them come from Wikipedia and the remaining ones come from Twitter. These entities consist of 98 persons, 24 facilities and 19 organizations. It's worth noting that the entity set contains a lot of less popular entities, especially some Twitter entities. For example, the entity *Danville Engineering (@danvillekyengr)* from Twitter only has 7 tweets and 17 followers in total.

Annotation.

The documents from Oct. 2011 to Feb. 2012 are annotated as training data and the remainder from Mar. 2012 to Feb. 2013 as testing data. Each document-entity pair is annotated as one of the four relevance levels as follows.

- Garbage: No information about the target entity could be learnt from the document, e.g. spam.
- Neutral: Informative but not citable, e.g. tertiary source like Wikipedia article itself not relevant.
- Useful: possibly citable but not timely, e.g. background bio, primary or secondary source.
- Vital: timely info about the entity's current state, actions, or situation. This would motivate a change to an already up-to-date knowledge base article.

Even given the heuristic definition of different relevance levels, there still exist annotation disagreements in the ground truth data, especially between vital and useful.

Table 1: Inconsistent Annotation in ground truth data from Oct. 2011 to Feb. 2012

	Vital	Useful	Neutral	Garbage
Vital		179	57	19
Useful	179		55	17

The details of the annotations for Wikipedia and Twitter entities are shown in **??** separately. In the target entity set, only 131 entities have been annotated. And according to **??**, only 32.8% entities are labeled with more than 10 *vital* instances in training data.

Table 2: Annotation statistics for Wikipedia and Twitter entities from Oct. 2011 to Feb. 2012

	Vital	Useful	Neutral	Garbage
Wikipedia	2096	2257	1162	1756
Twitter	182	326	72	569

Table 3: *Vital* Annotation Statistics in Training Data

Vital Annotation #	Entity #	Percentage
0	31	23.7%
$1 \sim 5$	37	28.2%
$6 \sim 10$	20	15.3%
> 10	43	32.8%

4. FILTERING

This section introduces our filtering strategy, which aims to reduce the size of the stream corpus through discarding obviously irrelevant documents. Remember that there are 141 entities in the target entity set and nearly 1 billion documents in the stream corpus, it is too time-consuming and laborious to process all the documents in the stream corpus for each entity.

According to the annotation analysis shown in **??**, we can speculate that the majority of vital and useful documents mention the target entity explicitly. Therefore, we implement a filtering step to remove the documents that do not mention target entities at all after indexing all the documents with ElasticSearch[3]. To achieve this goal, we construct a high-recall phrase query for each entity to ensure that retrieved document must mention the target entity at

[2] http://trec-kba.org/kba-stream-corpus-2013.shtml

[3] http://www.elasticsearch.org/

least once, either exactly by its name or other surface forms. For example, *Barack Hussein Obama*, the current president of the U.S., can be referred to by multiple surface forms (e.g., *Barak Obama* and *Obama*) in texts. Therefore, the prerequisite of filtering is expanding as many reliable surface forms as possible for each target entity.

For each Wikipedia entity, we treat the redirect[4] names as its surface forms. For instance, *Geoffrey E. Hinton*, who is a computer scientist in machine learning field, has the following redirect names in Wikipedia: *Geoffrey Hinton*, *Geoff Hinton*, and *Geoffrey Everest Hinton*. For each Twitter entity, we add its display name into its surface form set. Take *@AlexJoHamilton* for example, we acquire its display name *Alexandra Hamilton* via Twitter's APIs.

For a given entity E_t, the surface form set of entity E_t is $Rel(E_t) = \{E_i | i \in [1, M]\}$, the phrase query for E_t is

$$E_t \vee \; E_1 \vee E_2 \vee \cdots \vee E_M, \tag{1}$$

where the $\vee$ operator ensures that at least one operand is true, which means the corresponding term is matched in the document. This query Is named as basic query.

Table 4: Annotation Details of Training Data

	Garbage	Neutral	Useful	Vital
Mentions	1516	1226	2543	2271
Zero Mention	809	8	40	7

5. RELEVANCE ESTIMATION

In this section, we first describe three relevance evidences that we employ to distinguish documents with different relevance levels. Next, we introduce our relevance level estimation approaches in detail, including query expansion and two supervised approaches, classification and learning to rank.

5.1 Evidences of Relevance

Profile Page.

All the target entities possess profile pages either in Wikipedia or on Twitter. The profile page includes the basic information of an entity such as name, birth date, profession, related entities etc. These information could be leveraged to differentiate the documents with different levels of relevance given a target entity.

These profile page are utilized in different ways in different approaches. In query expansion, we extract related entities from the profile page as expanding terms for a target entity. Furthermore, we develop some features based on the profile pages for supervised approaches. Given a document-entity pair, one solution to capture their relevance is calculating similarity between the entity's profile page and the document. In Wikipedia, the profile page is organized as a list of sections. Each section introduces a specific aspect of the target entity. Vital documents for a target entity are possibly highly related with a few of these sections rather than all of them. Therefore, we calculate the similarity (cosine and jaccard) between the documents with different sections respectively instead of the whole profile page.

Existing Citations.

For a Wikipedia entity, there usually already exists a list of citations in its entry page. As we know, Wikipedia is collaboratively maintained by volunteer editors all around the world. The collective intelligence ensures low-quality citations are replaced by better ones persistently. Therefore, the remaining citations after several edits are extremely valuable in identifying vital or useful documents for the target entity.

However, few citation exists in the profile pages for most Twitter entities and some less popular Wikipedia entities. We create pseudo citations for them using pseudo feedback from public search engines as **??** shows.

Algorithm 1 Pseudo Citation Generation

```
Input: Entity $e$, Citation Number $n$
Output: Pesudo Citation Set $C$
query $e$ in search engine
crawl the hit link list $L$
for all $l \in L$ do
    if $l$ is a dead link or advertisement link then
        continue
    else
        extract the document $c$ from $l$
        if $c \in C$ then
            continue
        else
            add $c$ into $C$
            if $|C| == n$ then
                break
            end if
        end if
    end if
end for
```

In query expansion, related entities are extracted from the existing and pseudo citations as query expanding terms. In supervised approaches, the similarity (cosine or jaccard) between each citation and the document are taken into account. To avoid taking documents from a future time, we take great care to ensure that all the citations crawled are generated before the publishing time of the stream document.

Temporal Signals.

As CCR is a sequential task, some temporal signals have been employed to in previous work [?, ?, ?]. The view statistics of Wikipedia pages is adopted as a useful signal to capture if something significant happen around the target entity at a given time point. Another signal is the fluctuation of occurrence of an entity in a stream. Based on our observations of Wikipedia entities, a sudden increase, or a burst, in daily page views of an entity entails an increase of the number of vital and useful documents in stream corpus. This phenomenon may be caused by the sudden increase of vital edits of entity page, triggering lots of visits from the web. **??** shows an example for entity *Nassim Nicholas Taleb*. All the obvious bursts of Wikipedia page views are accompanied with the burst of occurrences in the stream. Because the stream corpus is a sample of the whole web documents, in which some entities may occur too rarely to reflect the real web environment, we choose Wikipedia page view statistics as our temporal signal.

The magnitudes of Wikipedia page views of different enti-

[4] http://en.wikipedia.org/wiki/Wikipedia:Redirect

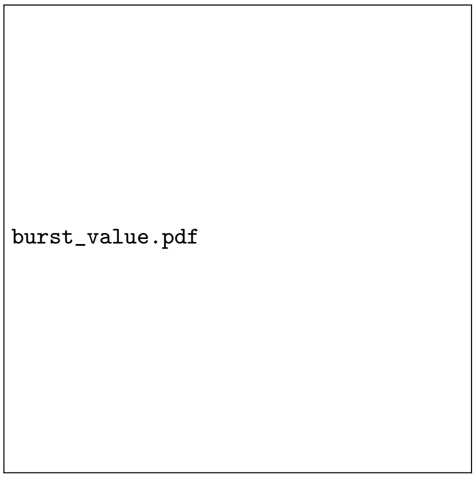

Figure 2: Wikipedia Page Views and Entity Occurrences in the Stream for *Nassim Nicholas Taleb*

ties varies sharply depending on their popularity. A popular entity may be viewed thousands of times every day, but an less popular entity can only attract a few views in the same period of time. We define a normalized burst_value for each document-entity pair as follows.

$$burst\_value = \frac{N * wpv(d_n)}{\sum_{i=1}^{N} wpv(d_i)} \quad (2)$$

where N is the total days the stream corpus covers. d_n means the document is published on the n_{th} day of the stream corpus. $wpv(d_i)$ is the views of the target entity's Wikipedia page during the i_{th} day of the stream corpus. For Twitter entities, page view statistics are not provided via Twitter APIs nor in any other forms, so we do not include temporal signals for Twitter entities in our experiments.

5.2 Approaches

5.2.1 Query Expansion

Query expansion is a common approach to retrieve relevant documents for a given query. In **??**, we have implemented a basic query to retrieve documents mentioning the target entities. While the base query neither can disambiguate ambiguous entities with a same name, such as *basic_element_(company)* and *basic_element_(music_group)*, nor can distinguish the relevance levels of the hit documents. In summary, the basic query can not provide sufficient evidence to select relevant documents effectively. Query expansion is an intuitive approach to address this problem by expanding the basic query with additional information.

We expand the basic query with contextual related entities extracted from the proposed evidences: the target entities' profile pages, and existing citations in KBs. For Wikipedia entities, we extract the anchor texts of inlinks in their Wikipedia pages as related entities. For Twitter entities, Stanford Named Entity Recognizer [**?**] is employed to recognize entities from their profile pages. Besides, the relevant (vital or useful) documents in the ground truth data is of great help to differentiate documents with different relevance levels. The related entities appear in a vital (useful) document can help us find more vital (useful) documents. So we also extract related entities for target entities from annotation data. Note that we only extract related entities from documents annotated as vital or useful.

For example, $\{E_i | i \in [1, N]\ \}$ is the related entity set we acquired for the target entity E_t, the query is formulated as follows:

$$Basic_{must} \wedge \{E_1 \vee E_2 \vee \cdots \vee E_N\} \quad (3)$$

Basic is the basic query demonstrated in **??**. The subscript *must* indicates this term is prerequisite in query. After extracting related entities, we incorporate them with basic query and then search against the built index. The hit documents are treated as relevant documents and the ranking scores returned by *ElasticSearch* are scaled to (0,1000] as the final confidence scores.

To validate the effectiveness of pseudo citations for Twitter entities, we implement two query expansion approaches: QE and QEP. The only difference between them is that we incorporate related entities extracted from pseudo citations into the query in QEP.

5.2.2 Supervised Methods

Classification and learning to rank approaches are two main approaches employed in previous work [**?**, **?**, **?**]. Most of them build a classifier or a ranking model for each target entity individually, which is not feasible in practical applications. A practical CCR system is required to process hundreds or even thousands of entities simultaneously, so it's impossible to label enough training data for each entity. Hence, we employ entity-independent supervised approaches by training a uniform supervised model with all the training instances.

Before taking the above proposed evidences into account, we develop a basic feature set for supervised learning approaches. The basic feature set is listed in the first block of **??**. These features are mainly designed to capture the intrinsic characteristics of a document and the occurrence distribution of the target entity in the document. Since not all entities' occurrences in documents are exactly the full name, we split entity name into different components as partial names and include them in features. We also demonstrate the extra features from the proposed relevance evidences in **??**.

Classification.

CCR is usually considered as a binary classification task which aims to classify documents into different relevance levels. In terms of the two sub-tasks of CCR, two classification problems need to be dealt with: relevant/irrelevant classification and vital/useful classification. So two classifiers are trained with different training data: (i) only *vital* documents are treated as positive instances, and (ii) both *vital* and *useful* documents are treated as positive instances. The former one is a vital classifier that classify documents into vital and non-vital categories. The latter one is a relevant/irrelevant classifier that classifies documents into relevant (vital + useful) and irrelevant (neutral + garbage) categories. In summary, we can view the classification step in differentiating vital from useful as an additional relevance classification step which aims to separate relevant documents identified in the first step further into "*very relevant*" (vital) and "*relevant*" (useful).

Based on our internal experiments, random forests classifier outperforms other classifiers, such as Support Vector Machine (SVM) and logistic regression. Therefore, we

Table 5: Feature set used in supervised approaches

Feature	Description
Basic Features	
$log(length)$	logarithm of document length
Source	document source
Weekday	post date of document
$N(E_{rel})$	# of entity E's related entities found in its profile page
$N(D, E)$	# of occurrences of E in document D
$N(D, E_p)$	# of occurrences of the partial name of E in D
$N(D, E_{rel})$	# of occurrence of the related entities in D
$FPOS(D, E)$	first occurrence position of E in D
$FPOS_n(D, E)$	$FPOS(D, E)$ normalized by document length
$FPOS(D, E_p)$	first occurrence position of E's partial name in D
$FPOS_n(D, E_p)$	$FPOS(D, E_p)$ normalized by document length
$LPOS(D, E)$	last occurrence position of E in D
$LPOS_n(D, E)$	$LPOS(D, E)$ normalized by document length
$LPOS(D, E_p)$	last occurrence position of E's partial name in D
$LPOS_n(D, E_p)$	$LPOS(D, E_p)$ normalized by document length
$Spread(D, E)$	$LPOS(D, E) - FPOS(D, E)$
$Spread_n(D, E)$	$Spread(D, E)$ normalized by document length
$Spread(D, E_p)$	$LPOS(D, E_p) - FPOS(D, E_p)$
$Spread_n(D, E_p)$	$Spread(D, E_p)$ normalized by document length
Profile Evidence Features	
$Sim_{cos}(D, S_i(E))$	cosine similarity between document and the i_{th} section of entity E's profile page
$Sim_{jac}(D, S_i(E))$	jaccard similarity between document and the i_{th} section of entity E's profile page
Citation Evidence Features	
$Sim_{cos}(D, C_i)$	cosine similarity between document D and the i_{th} citation
$Sim_{cos}(D, C_i)$	jaccard similarity between document D and the i_{th} citation
Temporal Evidence Feature	
$Burst\_Value(D)$	the *burst_value* of Document D, see details in **??**

employ random forests classifier implemented with Weka toolkit with default parameter settings [**?**]. The classifier outputs a probability for each classification operation, we scale this probability to (0, 1000] as the final confidence score. We evaluate these relevance evidences by integrating them into the feature set. Two different classification approaches are implemented as follows.

- UniClass: trains an uniform classifier for all entities using the basic feature set.
- UniClass+: train an uniform classifier for all entities with additional evidences proposed in **??** and the basic feature set.

Learning to Rank.

If we treat the different relevance levels as an ordered sequence, i.e., $vital > useful > neutral > garbage$, CCR becomes a learning to rank (LTR) problem. In [**?**], random forests LTR algorithm is reported as the best approach to complete last year's CCR tasks. In addition, to keep consistent with classification methods and facilitate comparisons, we also choose random forests ranking approaches. Same with classification approaches, two ranking models are trained for all the entities as follows.

- UniRank: trains an uniform ranking model for all entities using the basic feature set.
- UniRank+: trains an uniform ranking model for all entities with additional evidences proposed in **??** and the basic feature set together.

All the ranking approaches are implemented with RankLib[5] with default parameter settings. Also, the estimated ranking scores by rannking model are scaled to (0,1000] as the final confidence scores.

6. RESULTS AND DISCUSSIONS

6.1 Filtering Evaluation

In this section, we evaluate the performance of our filtering strategy. As indicated in **??**, the prerequisite of filtering is expanding as many surface forms as possible for target entities. We compare our expanding method with several methods proposed to expand surface forms in previous research. In [**?**], the official baseline system splits the entity name into different name tokens and manually select reliable ones as surface forms. In [**?**, **?**], DBpedia[6], a structural knowledge base extracted from Wikipedia, are used to extract name variants for target entities. in [**?**], google cross-lingual dictionary (GCLD) [**?**], mapping language-independent strings of words and Wikipedia articles bidirectionally, is used to expand variant strings for each target entity, and adopt these strings to query documents from the stream.

To evaluate the expanding effectiveness of these methods, we compare the max($macro\_avg(Recall)$) metrics in **??** through setting the cutoff value as 0, which means all the retrieved documents are considered as positive instances. Our redirect-based expanding method achieves the best overall recall in the *vital only* task. In the *vital + useful* task, the

[5] http://sourceforge.net/p/lemur/wiki/RankLib/
[6] http://dbpedia.org

$\max(macro\_average(R))$ of our expanding approach is very close to the best one. Our filtering step can retrieve 85% relevant documents and 72.8% vital documents from the volume stream corpus, proving our redirect-based surface form expanding method is effective in both CCR tasks..

Table 6: Recall measures of different expansion methods.

Filtering Method	$\max(macro\_average(R))$	
	Vital	Vital + Useful
Name Tokens	.713	**.856**
DBPedia	.601	.707
GCLD	.235	.581
Redirect	**.728**	.850

6.2 Relevance Level Estimation

6.2.1 Evaluation Metrics

A CCR system is fed with the stream documents in chronological order and outputs a confidence score in the range of (0,1000] for each document-entity pair. There are two evaluation metrics to evaluate the performance of a CCR system: $\max(F(avg(P), avg(R)))$ and $\max(SU)$. Scaled Utility (SU) is a metric introduced in filtering track to evaluate the ability of a system to separate relevant and irrelevant documents in a stream [?]. A cutoff value is varied from 0 to 1000 with some step size and the documents with the scores above the cutoff are treated as positive instances; while the documents with the scores below the cutoff are negative instances. The primary metric $\max(F(avg(P), avg(R)))$ is calculated based on the average precision and average recall of all entities. The calculation formulas of precision and recall are shown in **??** and **??** respectively.

$$Precision = \frac{\#(TP)}{\#(TP) + \#(FP)} \tag{4}$$

$$Recall = \frac{\#(TP)}{\#(TP) + \#(FN)} \tag{5}$$

TP represents the true positive instances, FP represents false positive instances, and FN represents false negative instances in the classification results. The primary metric $\max(F(avg(P), avg(R)))$ is calculated as follows: given a cutoff c and an entity E_i, $P_i(c)$ and $R_i(c)$ are calculated respectively, then macro-average them in all entities, $avg(P) = \frac{\sum_{i=1}^{N} P_i(c)}{N}$. $avg(R) = \frac{\sum_{i=1}^{N} R_i(c)}{N}$, N represents the quantities of entities in the target entity set. Therefore, F is actually a function of the relevance cutoff c, and we select the maximum F to evaluate the overall performance of a CCR system. In a similar manner, $\max(SU)$ are calculated as an auxiliary metric.

6.2.2 Approaches Comparison

All the results of our approaches are shown in **??**. It includes the overall metrics for all entities, the most primary metric of the performance. These measures are reported for Wikipedia and Twitter entities separately to evaluate the performance of these approaches in different entity sets. Please note that the cutoffs to reach maximum of F (or SU) for overall entity set and for separate entity set may be different, so the value of an overall metric is not always between the values of two separate measures.

For reference, the KBA official baseline, the 2nd and 3rd place approaches in KBA 2013 are also included. The KBA official baseline assigns a "*vital*" rating to every document that matches a surface form name of an entity and assigning a confidence score based on the length of the observed name [?]. The 2nd place approach (# 2) derives a sequential dependence retrieval model which scores the stream documents by frequency of unigrams, bigrams and windowed bigrams of the target entity name, taking document length and corpus statistics into account [?]. The 3rd place approach (# 3) pools related entities from the profile page of the target entity, estimate the weight of the related entities based on the training data, and apply the weighted related entities to estimate the confidence scores for stream documents [?]. The bottom block of **??** lists the overall mean and median of the results aggregated from all the submissions in KBA CCR 2013 track.

?? and **??** illustrate the macro-averaged recall and precision measures of all approaches listed in **??**. The parallel curves are contour lines of macro-averaged F-measure. Those approaches falling in upper right are better than the lower left ones.

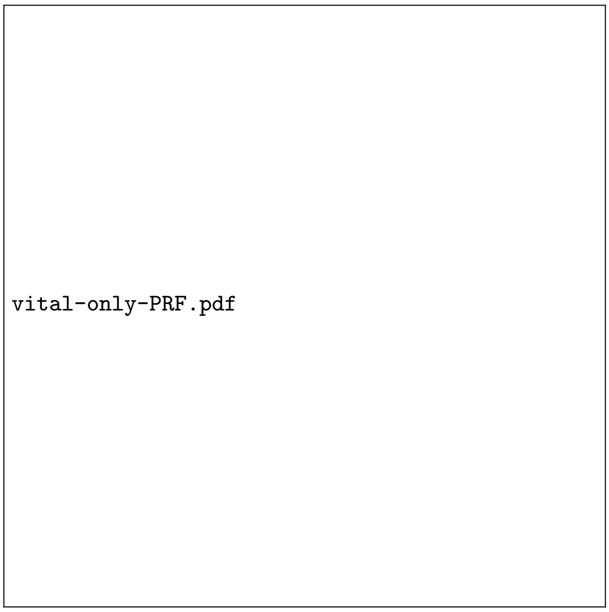

Figure 3: Macro-averaged recall versus precision with curves of constant F_1 in *vital only*

Overall Analysis.

As shown in **??**, our approaches outperform the official baseline on all metrics. In the *vital + useful* task, both the 2nd and 3rd place approaches do not do as well as the official baseline. In the *vital only* task, the 2nd place approach beats the official baseline less than 1% on overall $\max(F)$ measure. All our approaches outperform the official baseline notably, which validates the effectiveness of the proposed basic feature set and relevance evidences in both CCR tasks. Moreover, our approaches not only perform outstandingly on overall measures, but also outperform

Table 7: Approaches comparison. All the measures are reported by the KBA official scorer with cutoff-step-size=10. Best scores are typeset boldface.

Run	Vital Only						Vital + Useful					
	$\max(F(avg(P), avg(R)))$			$\max(SU)$			$\max(F(avg(P), avg(R)))$			$\max(SU)$		
	Overall	Wiki	Twitter	Overall	Wiki	Twitter	Overall	Wiki	Twitter	Overall	Wiki	Twitter
QE	.281	.288	.257	.170	.178	.174	.645	.658	.567	.544	.557	.466
QEP	.281	.288	.274	.173	.178	.194	.645	.658	.600	.544	.557	.536
$UniClass$	.291	**.297**	.248	.219	.215	.240	.644	.659	.567	.544	.562	.466
$UniClass+$	**.300**	.296	.311	.222	.214	**.268**	**.660**	**.663**	.634	**.568**	**.570**	.586
$UniRank$	.285	**.297**	.275	**.260**	**.269**	.239	.644	.657	.588	.544	.557	.490
$UniRank+$	.290	.293	**.315**	.253	.257	.258	.651	.657	**.658**	.560	.557	**.600**
$OfficialBaseline$	.267	.276	.217	.174	.179	.154	.637	.646	.593	.531	.548	.427
#2 [?]	.273	.273	.289	.247	.251	.240	.610	.613	.594	.496	.507	.470
#3 [?]	.267	.270	.244	.158	.162	.138	.611	.619	.552	.515	.526	.444
$Median$	.174	.179	.164	.255	.259	.233	.406	.382	.333	.423	.433	.389
$mean$	.166	.172	.136	.137	.240	.224	.376	.433	.360	.425	.438	.364

`vital-useful-PRF.pdf`

Figure 4: Macro-averaged recall versus precision with curves of constant F_1 in *vital + useful*

the others on separate measures for Wikipedia and Twitter entities. This demonstrates that our entity-independent approaches indeed work for both popular (Wikipedia) entities and less popular (Twitter) entities. Please note that the official baseline is a strong baseline in which human annotators went through target entities and came up with a list of keywords for filtering.

Supervised vs. Unsupervised.

Supervised approaches are more potential than unsupervised approaches in both tasks. Even the classification and ranking approaches merely using the basic feature set, i.e., *UniClass* and *UniRank* in **??**, achieve comparative performance with unsupervised approaches including relevance evidences. After being augmented with additional relevance evidences, supervised approaches can achieve more promising results.

Query Expansion.

As illustrated in **??** and **??**, though our query expansion approaches can achieve the best recall measures among all the approaches, their precision measures are not satisfactory. This may be resulted from our equally weighting strategy for all expansion terms. The approach of **# 3** is similar to query expansion, which weights expansion terms with the help of training data. **# 3** has achieved a high precision in the *vital + useful* task. We believe our query expansion approach could be improved by introducing better weighting strategies.

Effect of Pseudo Citation.

According to the comparison between two query expansion approaches, although they perform similarly on the overall metrics, all the metrics for Twitter entities are improved. This proves the effectiveness of pseudo citations for Twitter entities. For the entities with few annotation, we could crawl pseudo citations to mitigate the impact of lack of annotation data. Pseudo citations could provide more training data. It is especially useful for the entity with few annotation data.

Classification.

UniClass+ outperforms **UniClass** on all metrics prominently as illustrated in??. The difference between them is whether integrating relevance evidences into the feature set. This proves that the proposed relevance evidences play an positive role in classification approaches to address the two CCR tasks. Furthermore, according to ?? and ??, we conclude that the relevance evidences help improve both precision and recall of classification approaches.

Learning to Rank.

Similar to classification approaches, in both tasks, the ranking model trained with relevance evidences, i.e.,**UniRank+**, achieves better results than the baseline ranking model trained with the basic feature set, i.e., **UniRank**. The proposed evidences also enhance the performance of learning to rank approaches.

Classification vs. Ranking.

Both classification and ranking approaches leveraging additional evidences outperform their own baselines in *vital only* and *vital + useful* tasks. Classification approaches achieve better results than ranking-based approaches on F-measure in both tasks. However, ranking approaches achieve a higher SU than classification approaches in *vital only* task, which reveals that the ranking methods show stronger filtering ability. However, ranking approaches is not stable as classification approaches. As shown in ?? and ??, although the **UniRank+** achieves higher precision than **UniRank**, the recall declines in the meantime.

6.3 Feature Analysis

In this section, we concentrate on the best classification approach **UniClass+** and explore the impacts of the relevance evidences in different tasks. We evaluate the effect of each evidence in classification by removing it from the feature set and observing whether $max(F)$ and $max(SU)$ decline in the same time. The results are exhibited in ??. Note that the negative percentages represent the corresponding measure arises. The full feature set contains both the basic feature set and relevance evidences. In general, **UniClass+** augmented with all relevance evidences achieved the best $max(F)$ and $max(SU)$ in both tasks. The $max(F)$ declines if burst evidence or citation evidence is removed in both figures This reveals that these two evidences are critical either in detecting vital and useful documents together or in detecting vital documents solely. The profile evidence is not of the same significance in two tasks. In *vital + useful*, the performance declines if the profile evidence is removed. Nevertheless, in *vitla only*, although $max(SU)$ decreases without the profile evidence, $max(F)$ seems not affected at all. This phenomenon can be explained from the characteristics of profile page. For an entity, either from Wikipedia or Twitter, its profile page usually focus on background information such as biography. This background evidences are not so essential in detecting vital documents.

To explore the different roles these evidences play in both tasks, we perform an analysis of the features with the help of information gain. ?? reports the information gains of the proposed features for two CCR tasks.

6.4 Error Analysis

We carry out qualitative and quantitative error analysis based on the classification results of the best classification approach **UniClass+**. In the *vital only* task, false negative instances are the actual vital documents that we miss in the filtering step. Most of them are the documents mentioning the target entity with a misspelled name or an unusual alias, both of which are not taken into account in our current approaches. We could address this problem by undertaking spelling correction before filtering or employ more effective resources to expand surface forms for entities. However, as the current best recall has reached 85%, this is not an urgent problem for CCR.

Table 9: Information gain values of the proposed features

	Information Gain	
Feature	vital only	vital+useful
$Burst\_Value(D)$	0.130	0.287
$avg(Sim_{cos}(D, S_i(E)))$	0.069	0.145
$avg(Sim_{jac}(D, S_i(E)))$	0.058	0.150
$avg(Sim_{cos}(D, C_i))$	0.028	0.083
$avg(Sim_{jac}(D, C_i))$	0.052	0.108
$N(E_{rel})$	0.121	0.175

$N(E_{rel})$ is the best feature in basic feature set according to the IG.

Compared with the high recall, we are more concerned about the poor precision, which is the bottleneck of current approaches. As shown in ??, even the best approach can not achieve a precision higher than 20%. Precision is mainly affected by the amount of false positive instances, such as the actual non-vital documents classified as vital documents in the *vital only* task. ?? shows the actual source distribution of false positive instances in the two tasks. Both in the *vital only* and the *vital + useful* tasks, garbage documents affect the classification performance apparently. Although our filtering step retrieves enough relevant documents, it also retains some garbage documents mentioning the target entity. Some spam documents camouflage themselves as vital or useful documents and mislead our classifiers. Applying a spam detection step in the filtering process is an optional solution for this problem. In the *vital + only* task, 44.3% of the false positive errors are caused by useful documents. Though effective evidences are utilized to distinguish between vital and useful documents, we have to admit that it's still a difficult problem. In some cases, even human beings can not easily decide whether a document is vital or useful.

Table 10: Source distribution of misclassification instances

Task	Garbage	Neutral	Useful
Vital Only	39.3%	16.4%	44.3%
Vital + Useful	63.4%	36.6%	

7. CONCLUSION AND FUTURE WORK

The objective of CCR is filtering vitally relevant documents from a time-ordered stream corpus. The key challenge is how to detecting vital documents accurately. Apart from annotation data, we leveraged to additional relevance evidences to improve our system performance. These additional evidences were mined from the profile page of an entity, existing citations and temporal signals, all of them can improve both unsupervised and supervised approaches. We also evaluated their significance in detecting only vital

Table 8: $\max(F)$ and $\max(SU)$ variations after removing evidence from feature set in *vital only* and *vital + useful*

Feature Set	Vital Only				Vital + Useful			
	$\max(F)$	$\swarrow \max(F)$	$\max(SU)$	$\swarrow \max(SU)$	$\max F$	$\swarrow \max(F)$	$\max(SU)$	$\swarrow \max(SU)$
$Full$	.300		.222		.660		.568	
$Burst^-$	.296	1.30%	.221	0.12%	.655	0.64%	.568	0%
$Citation^-$	.296	1.30%	.213	4.15%	.647	1.93%	.547	3.56%
$Profile^-$	.305	-0.13%	.214	3.50%	.657	0.45%	.560	1.38%
$Basic$	.291	2.94%	.219	1.44%	.644	2.37%	.545	4.03%

documents or both vital and useful documents together. Another challenge is lacking of training data for less popular entities, especially Twitter entities. We created pseudo citations for them and improved the overall performance. In addition, we developed an effective filtering step through executing a high recall query to handle the big stream corpus. Our classification approaches augmented with the proposed evidences achieved the state-of-the-art performance in the CCR task.

There is much room for improvement in the two tasks of CCR, especially discriminating vital documents from useful documents accurately. In future, we will focus on exploiting relevance evidences further to increase the precision of separating vital from useful documents. In addition, spam detection in a stream corpus is an interesting problem need to be addressed in the CCR task.