Relevance in Microblogs: Enhancing Tweet Retrieval using Hyperlinked Documents

Richard McCreadie, Craig Macdonald
{richard.mccreadie,craig.macdonald}@glasgow.ac.uk

School of Computing Science
University of Glasgow
G12 8QQ, Glasgow, UK

ABSTRACT

Twitter serves over 1.6 billion searches each day, ranking tweets for display to the user in reverse-chronological order. However, finding relevant tweets can be a challenging task, since the relevance of a tweet is dependant both on its content and whether it links to a useful document. In this paper, we investigate how the content of documents hyperlinked from a tweet can be used to better estimate that tweet's relevance. In particular, we propose three approaches for incorporating the content of hyperlinked documents when ranking tweets. Within the context of the TREC 2011 and 2012 Microblog Tracks, we thoroughly evaluate to what extent hyperlinked documents can aid tweet retrieval effectiveness. Our results show that the application of hyperlinked documents can improve retrieval effectiveness over using the tweet content alone as well as using the presence of a URL within the tweet as a feature.

Categories and Subject Descriptors: H.3.3 [Information Storage & Retrieval]: Information Search & Retrieval

General Terms: Experimentation, Performance

Keywords: Twitter Search, Learning to Rank, Hyperlinked documents

1. INTRODUCTION

Twitter[1] is an information sharing platform using which users can broadcast short messages – known as tweets – not exceeding 140 characters to the pool of users that have subscribed to them. Moreover, Twitter provides a search facility, where by recent tweets can be retrieved for the user query and are displayed in reverse-chronological order.

However, effective tweet retrieval is a challenging task, due to the short length and unique characteristics of tweets. In particular, restricting a tweet's length to 140 characters increases the chance of term-mismatch between relevant tweets and a user query [20]. Meanwhile, to fit the message into the required length, term contractions such as acronyms are common [29]. Furthermore, tweets often contain Twitter-specific terminology such as hashtags [11], mentions [10] and hyperlinks to related documents [16].

Prior works in the field of tweet search have identified that the presence of a hyperlink to a related document is a positive indicator of relevance when retrieving tweets [8, 16]. However, these works have only considered the hyperlink itself, and not the document that it links to. In this paper, we propose to use both the content of the tweet and the content of any hyperlinked documents to better estimate the relevance of a tweet to a query. In particular, we propose three alternative approaches for incorporating a hyperlinked document into the scoring process for a tweet, namely: virtual document integration; field-based integration and learning to rank. We thoroughly evaluate the performance of each approach to determine whether they can more effectively retrieve tweets when testing upon the TREC 2011 and 2012 Microblog track topic sets. From our results and subsequent analysis, we show that integrating the scores for the hyperlinked documents for a query can significantly increase real-time tweet retrieval effectiveness.

The remainder of this paper is structured as follows. Section 2 discusses prior work in the field of Twitter search. In Section 3, we describe our four proposed approaches to integrate hyperlinked documents into the tweet ranking process. Section 4 describes our experimental setting for real-time tweet search and the TREC datasets that we use. In Section 5, we concretely state the research questions that we investigate in this paper, while in Section 6, we report on our results. We provide concluding remarks in Section 7.

2. RELATED WORK

Twitter is at the time of writing the largest dedicated English microblogging service. Twitter enables anyone to sign-up and publicly post messages about any topic. Indeed, it is highly popular with over 100 million active users. Ranking in a microblog setting differs markedly from traditional information retrieval search tasks. In particular, rather than ranking in order of relevance [15], microblogs are often returned in reverse chronological order [23]. The reason behind this different ranking approach is that information needs posed to microblog search engines have a strong temporal component. Indeed, tweet search holds similarities to search of the blogosphere, where users are interested in finding up-to-date information about current news stories [22, 28]. In general, search in a microblog setting can be considered as answering the question '*find me the most recent information about X*'.

[1] `http://twitter.com/`

OAIR'2013, 22nd May, 2013, Lisbon, Portugal

Tweet search also introduces new problems for information retrieval systems to tackle. Firstly, tweets are by design very short, which may make document weighting models such as BM25 [27] less effective. In particular, the term frequency component of such a model provides much less information than in a Web search setting, since each term is likely to only appear once in a tweet. Furthermore, the shorter tweet length may make vocabulary mismatch between the query and relevant tweets more acute, reducing the recall of the tweet rankings produced with standard document weighting models. Next, the tweets themselves contain Twitter-specific vocabulary. For instance, when Twitter users post, hashtags, i.e. words beginning with the character '#' are often used to denote topics and concepts. These are used to link together many tweets about the same topic. Indeed, it has been reported that over 15% of tweets contain hashtags [9]. Similarly, mentions, i.e. user names prefixed with the '@' symbol are used to indicate replies or direct messages to the user in question. Additionally, Twitter allows a user to retweet another's tweets, i.e. post an exact copy of another user's tweet, normally with a reference to the source user [6]. However, the key characteristic of tweets that we investigate in this paper is the inclusion of links to related content. In particular, we investigate whether by leveraging the content of documents hyperlinked from a tweet, real-time tweet ranking effectiveness can be improved.

Notably, in 2011 and 2012, the Text REtrieval Conference (TREC) ran the Microblog track that investigated adhoc tweet search [25].[2] The aim of this task was to find the most relevant tweets for the user query in a real-time setting, i.e. to retrieve tweets on or before a point in time. To facilitate this track, the first legally redistributable Twitter test collection, named Tweets2011, was developed through collaboration between TREC and Twitter [18, 29]. We use the TREC Microblog Track 2011 and 2012 test collections (including Tweets2011) to evaluate our proposed approaches for integrating hyperlinked document content into the tweet ranking process.

To rank tweets, a variety of approaches have been proposed. For example, Amati *et al.* [1] proposed a new DFReeKLIM retrieval model from the divergence from randomness framework that accounts for the very short nature of tweets. The most effective approaches submitted to the TREC 2011 Microblog Track focused purely on relevance [25]. In particular, the approach by Metzler *et al.* [19] combined learning to rank with pseudo-relevance feedback to find the 30 most relevant tweets to the user query and returned only those 30 tweets. These approaches focus solely on the text of the tweet, including in some cases whether it contains a hyperlink to an external document. However, these approaches do not examine content of the documents linked to. In contrast, in this paper, we propose three approaches to integrate this additional content into the tweet scoring process and experientially evaluate their performance on both the TREC 2011 and TREC 2012 Microblog track datasets.

3. INTEGRATING HYPERLINKED DOCUMENTS

The motivations for making use of the content of hyperlinked documents when ranking tweets is two fold. First, the tweets themselves are very short, which increases vocabulary mismatch issues. By leveraging the content of hyperlinked documents, we may be better able to identify those tweets that are relevant even if the tweet appears to be loosely related to the query from its text alone. Secondly, some tweets may only be relevant because they link to a relevant article. For example, consider the tweet "RIM, Nokia news http://dlvr.it/2gcW3Y". This tweet is ambiguous and largely uninformative, however, following the hyperlink leads to a detailed article about how RIM and Nokia have settled their patent disputes - which might make the tweet useful to users searching on the topic.

However, while the motivations for using the hyperlinked documents are clear, how they can be integrated has not yet been examined. Indeed, given the large differences between a tweet and a Web page/article that is linked to, achieving an effective integration that will still retrieve tweets both with and without hyperlinks is challenging.

Formally, for a query Q and a time t, we want to rank a set of tweets $d \in D$, where tweets in D were published before t. However, a tweet d may contain a hyperlink to an additional document d_l. If d_l exists, then we need to score d with respect to the query Q, the time t and the hyperlinked document d_l. On the other hand, if d does not contain a hyperlink, then we score it with respect to the query Q and time t only. Hence, our target ranking function can be expressed as follows:

$$score(Q,t,d,d_l) = \begin{cases} score_{hyperlink}(Q,t,d,d_l) & \text{if } Exists(d_l) \\ score(Q,t,d) & otherwise. \end{cases} \tag{1}$$

where $score_{hyperlink}(Q,t,d,d_l)$ is the score for a tweet containing a hyperlink and $score(Q,t,d)$ is the score for a document without a hyperlink. The $score(Q,t,d)$ can be calculated using a traditional document weighting model, such as BM25 [27] or a more tailored one such as DFReeKLIM [1]. However, how $score_{hyperlink}(Q,t,d,d_l)$ should be calculated is unclear. We propose three different methodologies for integrating the content from hyperlinked documents into the scoring process, based on existing techniques in the field of information retrieval. These three approaches are: virtual document; field-based weighting; and learning to rank. We describe each in the following three sub-sections.

3.1 Virtual Document

The simplest approach for integrating hyperlinked document content into the scoring process is to directly append that content to the tweet. In this manner, any tweets that contain a hyperlink will be enlarged by the number of tokens within the hyperlinked document, creating a new virtual document. This approach is similar to works in the field of aggregate search, which rank objects (e.g. experts) are represented by multiple individual documents by combining those documents into a large virtual document [4, 7, 13]. We adapt these approaches for the real-time tweet ranking environment such that a tweet containing a hyperlink is scored as follows:

$$score_{hyperlink}(Q,t,d,d_l) = score(Q,t,d+d_l) \tag{2}$$

where $d + d_l$ is a virtual document comprised of the terms from both d and d_l. However, it is of note that this approach causes there to be effectively two classes of documents being ranked, i.e. d and $d+d_l$, where the virtual $d+d_l$ documents

[2] https://sites.google.com/site/microblogtrack/

are many times the size of the d documents (tweets without hyperlinks). This has the potential to cause problems during scoring, since much emphasis is placed on the document weighting model's ability to effectively normalise for document length. A document weighting model that uses a poor document length normalisation will tend to promote $d + d_l$ documents over d documents regardless of their relevance, simply because they are more likely to match the query terms and match them more often.

3.2 Field-Based Weighting

Our second approach considers the documents d and d_l to be two fields of the same document. In this case, for each tweet, a new document is created d_f, which contains two fields; f_d (the terms in the tweet) and f_l (the terms in the hyperlinked document). These new documents can then be scored using a field-based document weighting model, such as BM25F [26, 31] or PL2F [14] as follows:

$$score_{hyperlink}(Q, t, d, d_l) = score_f(Q, t, d_f, \mathbf{C}, \mathbf{W}) \qquad (3)$$

where $score_f()$ is a field-based document weighting model and d_f is the new (two-field) document. **C** is a vector of field normalisation parameters, one per field. For instance, in the context of BM25F, **C** defines the b (term frequency non-linearity) parameter used when scoring each field. **W** is a vector of field weights, i.e. the weight assigned to each field. The advantage of using a field-based document weighting model over the virtual document approach is two-fold. First, it enables d and d_l to be combined in a principled manner, while enabling different levels of emphasis to be placed upon each, i.e. via the weight vector **W**. Second, it allows for term frequencies to be normalised differently according to the type of document that the term came from. This second point is of great importance in a Twitter setting, since the term frequency of a term in a tweet is typically uninformative as the vast majority of terms appear only once in a tweet. Meanwhile, this is not true in the case of the hyperlinked document.

3.3 Learning to Rank

Finally, the third method that we propose to integrate the content of linked documents into the tweet scoring process is via a machine learned model, produced using a learning to rank technique [12]. Learning to rank techniques are machine learning algorithms, which take as input a set of features describing each document and learn a weight for each feature within an information retrieval (IR) system. The goal of learning to rank is to find the combination of these features, referred to as a model, which results in the most effective document ranking. The idea behind using learning to rank is that we can express both the relatedness of the tweet to the query and the relatedness of any hyperlinked document to the query as features. The learning to rank technique will then find an effective combination of these features (model) with which to rank.

Many different learning to rank techniques have been previously proposed. These techniques fall into one of three categories, dependant upon the loss function [12]. Point wise techniques learn on a per-document basis, i.e. each document is considered independently. Pair wise techniques optimise the number of pairs of documents correctly ranked. List wise techniques optimise an IR evaluation measure, like mean average precision, which considers the entire ranking list at one time. Prior work has indicated that list-wise techniques are often effective [12], hence we employ list wise learning to rank techniques here. Furthermore, these techniques can be further sub-divided into linear and tree-based learners. A linear learner will produce a model that linearly combines the feature scores for a tweet. Meanwhile, a tree-based learner builds a decision tree-like structure, where the branch nodes denote decisions based upon the features and each leaf node represents a final score to return. In this paper, we report performance using the linear Automatic Feature Selection (AFS) learner based upon simulated annealing [21]. The tree-based LambdaMART [30] learner was also tested but was less effective, likely due to insufficient available training data.

Feature Set	Summary	Number
TweetRet.	Retrieval scores for BM25, DPH, DirichletLM and DFReeKLIM on the tweet	4
ContainsURL	Does the tweet contain a URL	1
HyperlinkedRet.	Retrieval scores for the hyperlinked documents using BM25, DPH, DirichletLM and DFReeKLIM	4
HyperlinkedSpam	Five spam detection features [5]	5
Total		14

Table 1: Learning to Rank feature sets, descriptions and the number of features per set.

For illustration, we show how a linear model combines features extracted from both a tweet d and its hyperlinked document d_l into a single score below:

$$score(Q, t, d, d_l) = \sum_{0<i<|F^d|} weight(i) \cdot F_i^d + \sum_{0<i<|F_l^d|} weight(i) \cdot F_{li}^d \qquad (4)$$

where Q is the query, d is a tweet to be scored, F^d is the set of features extracted from d, $|F^d|$ is the number of features, F_i^d is the ith feature, d_l is the document linked to from d, F_l^d is the set of features extracted from d_l, $|F_l^d|$ is the number of features extracted from d_l, F_{li}^d is the ith feature from d_l and $weight(i)$ is the learned weight for a feature.

The key advantage that learning to rank approaches bring over either the virtual document and field-based weighting approaches is that it can integrate multiple features together, including those that are not solely dependant upon the query terms and tweets. We use learning to rank to generate a tweet ranking model that uses document ranking models to score both the tweets and their hyperlinked documents. Here, we represent each tweet as a vector of features, one per weighting model for both d and d_l. In this way, we integrate evidence from the hyperlinked documents with evidence from the tweets. We also use two additional types of features with learning to rank, specifically the presence of a URL in the tweet [8] that we use to create an additional baseline, and spam detection features extracted from the hyperlink documents [5] to provide an indicator of the quality of each page. The full list of features that we use in our subsequent experiments are provided in Table 1.

4. EXPERIMENTAL SETUP

To evaluate whether these approaches are able to increase tweet ranking effectiveness, we evaluate using the TREC Microblog track 2011 and 2012 topic sets. In particular, for

Data	Quality	Value
Tweets2011	Time Range	23/01/11 → 08/02/11
	# Tweets	13,205,709
	# Unique Terms	6,683,127
	# Tokens	101,249,658
Tweets2011_l	Time Range	23/01/11 → 08/02/11
	# Hyperlinked Documents	3,020,127
	# Unique Terms	7,675,225
	# Tokens	1,302,800,223

Table 2: Statistics of the Tweets2011 tweet and $\textbf{Tweets2011}_l$ hyperlinked document corpora.

each of the tweet ranking topics, the aim is to produce a rankings of tweets from the public Tweets2011 TREC Twitter corpus[3] for a point in time to the nearest second. This task simulates a real-time search environment, and as such, only tweets (and hyperlinked documents) posted before the query time can be used to rank for each topic.

Corpus: Tweets2011 is an approximately 16 million tweet sample from the period of the 23rd of January to the 8th of February 2011. Notably, Tweets2011 differs from a typical TREC corpus, in that it is not pre-provided by TREC, but is rather crawled by the participants [18]. Over time, the number of tweets in the corpus has decreased, e.g. as users have deleted their tweets. This reduction in corpus size is not a major issue however, since prior work has indicated that the overall ranking of approaches for a topic set is not adversely effected [29]. For these experiments, we use a crawl of the Tweets2011 corpus from May 2012. The Tweets2011 corpus is not provided with the hyperlinked documents. Hence, we independently extracted all of the hyperlinks from the tweets within Tweets2011 and crawled them separately, forming a second corpus, referred to as Tweets2011_l. Table 2 summarises the main statistics of both the Tweets2011 and Tweets2011_l corpora.

Indexing: To produce the rankings of tweets, we first indexed the Tweets2011 tweet corpus and the Tweets2011_l hyperlinked document corpus using the Terrier IR Platform [24]. Stopword removal and Porter stemming are applied in each case. For each query Q and point in time t, we re-produce a portion of both indices containing only those tweets/hyperlinked documents posted before t with term and collection statistics correct for t. Tweets are then ranked for Q only from this portion of the full index. Only the top ranked 1000 tweets are returned.

Topics and Assessments: For assessment, we use the 50 topics used during TREC 2011 and the 59 topics from TREC 2012. We refer to these topic sets as $\mathit{Microblog}_{2011}$ and $\mathit{Microblog}_{2012}$, respectively. Tweets pooled from the TREC Microblog participant systems were judged on a three point graded scale, Highly relevant, Relevant and Not Relevant, by human assessors. Assessments were made based upon both their text and any documents that they link to (i.e. assessors could also view the pages linked to from the tweets they were assessing). Hence, these assessments are suitable for evaluating whether the content of hyperlinked documents can be used to increase ranking effectiveness.

Training: Of the three approaches that we propose to integrate the contents of hyperlinked documents into the tweet scoring process, the field-based integration and learning to rank approaches require training. For field-based weighting, we train the normalisation parameters C and the field weights W in a cross topic-set manner, i.e. when ranking for the $\mathit{Microblog}_{2012}$ topics, we train C and W on the $\mathit{Microblog}_{2011}$ topics and vice versa. Meanwhile, to learn the feature combination under learning to rank, we experiment with two different training regimes, namely Cross-TopicSet and Per-TopicSet. In particular, under Cross-TopicSet training, we train using one set of topics (either $\mathit{Microblog}_{2011}$ or $\mathit{Microblog}_{2012}$) and then test upon the other set of topics and vice-versa (like we do for field-based weighting). Under Per-TopicSet training, we train and test on the same topic set using a cross-fold validation comprised of 5 training and test folds (i.e. 3 folds for training, 1 fold for validation and 1 fold for testing). In all cases we use MAP as the objective function [17].

Measures: To evaluate the performance of our tweet ranking approaches (that use the hyperlinked documents as evidence), we report the number of relevant tweets retrieved in the top ranks (precision@N and R-Precision) and mean average precision (MAP) over the top 1000 tweets retrieved. The official measures for the two years of the TREC Microblog tracks were precision@30 (P@30) and MAP.

5. RESEARCH QUESTIONS

In the next section, we investigate the following four research questions through experimentation, each in a separate sub-section:

- How effective are traditional document ranking models when ranking tweets and how do they compare to the systems deployed at TREC 2011 and 2012? (Section 6.1)

- Is the 'virtual document' approach effective when integrating content from hyperlinked documents into the tweet scoring process? (Section 6.2)

- Can field-based weighting models effectively integrate hyperlinked document content? (Section 6.3)

- Is learning to rank an effective paradigm for integrating evidence from hyperlinked documents into the tweet scoring process? (Section 6.4)

6. RESULTS

In this section, we report whether each of the three approaches that we propose to leverage the content of hyperlinked documents within the tweet scoring process are effective. In particular, in Section 6.1, we report the tweet ranking effectiveness of our baseline document weighting models. Section 6.2 examines the effectiveness of the proposed virtual document approach. In Section 6.3, we investigate two field-based weighting models for combining the tweet and hyperlinked documents, while Section 6.4 reports the performance of tweet ranking using models produced using learning to rank.

[3] http://trec.nist.gov/data/tweets/

Approach	Microblog$_{2011}$					
	P@5	P@10	P@20	P@30	MAP	R-Precision
TREC Median	—	—	—	0.2575	0.1426	—
TREC Best	—	—	—	0.4082	0.2078	—
BM25	0.4735	0.4449	0.3663	0.3381	0.2926▲	0.3376
DPH	0.5102	0.4673	0.4194	0.3707	0.3247▲▲	0.3718
DirichletLM	0.5061	0.4633	0.4204	**0.3864**	0.3388▲▲	0.3682
DFReeKLIM	**0.5265**	**0.4816**	**0.4214**	0.3850	**0.3390▲▲**	**0.3899**
	Microblog$_{2012}$					
	P@5	P@10	P@20	P@30	MAP	R-Precision
TREC Median	—	—	—	—-	0.1733	—
TREC Best	—	—	—	0.2701	0.2642	—
BM25	0.2102	0.1966	0.1873	0.1695	0.1360	0.1603
DPH	0.2576	0.2271	0.1805	0.1729	0.1514	0.1764
DirichletLM	0.2508	**0.2576**	**0.2136**	0.1746	0.1582	**0.1926**
DFReeKLIM	**0.2814**	0.2390	0.2000	**0.1774**	**0.1584**	0.1830

Table 3: Effectiveness of document weighting models for tweet ranking.

6.1 Baseline Ranking Effectiveness

We begin by examining the effectiveness of standard document weighting models for tweet ranking. This is important since prior works often report only BM25 [27] untrained (i.e. using the default parameter settings of b=0.75, k_1=1.2 and k_3=1000) as a baseline [8], not other potentially more effective baseline models. In contrast, we report tweet ranking performance using four different document weighting models, i.e. BM25 [27] from the best match family, DPH [3] from the Divergence from Randomness framework [2] a language modelling approach that uses Dirichlet smoothing [32] (denoted, DirichletLM) and the DFReeKLIM document ranking model that was highly effective at TREC 2011 [1].

Table 3 reports the performance of the BM25, DPH, DirichletLM and DFReeKLIM in terms of precision and MAP over both the *Microblog*$_{2011}$ and *Microblog*$_{2012}$ topic sets in comparison to the TREC Median and the highest performing TREC system (TREC Best). Statistically significant improvements over the TREC Best System (paired t-test $p<0.05$) are denoted ▲, while very significant improvements (paired t-test $p<0.01$) are denoted ▲▲. The highest performing document weighting model under each measure is highlighted in bold. From Table 3, we observe the following. First we see that on the *Microblog*$_{2011}$ topics, the basic retrieval models were effective for ranking tweets, achieving over 50% precision in the top 5 tweets returned. Second, we observe that the TREC median performance for this topic set is markedly lower than the performance of all the weighting models tested under P@30 and MAP, i.e. the median of TREC systems for this year is a weak baseline. Third, in comparison to the TREC Best system on the *Microblog*$_{2011}$ topics, our document weighting models outperform it by a statistically significant margin under MAP, but not P@30. This is because the top systems at TREC 2011 optimised for P@30, hence it is unfair to compare against this run under MAP. In contrast, on the *Microblog*$_{2012}$ topic set, the TREC Median and the TREC Best are much stronger baselines. Indeed, the TREC Median offers similar performance to common document weighting models, while the TREC best run outperforms them by a large margin.[4] Comparing the weighting models themselves, we observe that for tweet ranking, DFReeKLIM is highest performing under the majority of measures. As such, we use DFReeKLIM as our baseline in the subsequent experiments.

Approach	Microblog$_{2011}$					
	P@5	P@10	P@20	P@30	MAP	R-Precision
DFReeKLIM	**0.5265**	**0.4816**	**0.4214**	**0.3850**	**0.3390**	**0.3899**
BM25+VirtualDoc	0.4653	0.3918	0.3265	0.2776	0.2195	0.2550
DPH+VirtualDoc	0.3755	0.3388	0.2837	0.2544	0.2188	0.2656
DirichletLM+VirtualDoc	0.4816	0.4184	0.3561	0.3109	0.2624	0.3055
DFReeKLIM+VirtualDoc	0.5184	0.4571	0.4041	0.3646	0.3019	0.3621
	Microblog$_{2012}$					
	P@5	P@10	P@20	P@30	MAP	R-Precision
DFReeKLIM	**0.2814**	**0.2390**	**0.2000**	**0.1774**	**0.1584**	**0.1830**
BM25+VirtualDoc	0.2237	0.1797	0.1500	0.1435	0.1193	0.1389
DPH+VirtualDoc	0.2339	0.1983	0.1644	0.1492	0.1216	0.1509
DirichletLM+VirtualDoc	0.2203	0.1983	0.1551	0.1435	0.1322	0.1461
DFReeKLIM+VirtualDoc	0.2542	0.2322	0.1949	0.1678	0.1433	0.1780

Table 4: Effectiveness of the virtual document integration approach for tweet ranking.

6.2 Virtual Document Integration

We now examine the effectiveness of the first approach that we proposed, i.e virtual document integration. We report the tweet ranking performance of this approach when using the four document weighting models upon the tweets expanded with the content of any hyperlinked documents. If ranking performance increases, then this would indicate that simply merging the a tweet with its linked document is effective. On the other hand, if ranking performance decreases, then this would indicate that either the hyperlinked content is not useful or that the document weighting models are unable to make effective use of this additional evidence.

Table 4 reports the tweet ranking performance of BM25, DPH, DirichletLM and DFReeKLIM when the tweets being ranked are merged with the content of any linked documents. Performance is reported in terms of precision and MAP over both the *Microblog*$_{2011}$ and *Microblog*$_{2012}$ topic sets in comparison to our DFReeKLIM baseline. The highest performing approach under each measure and topic set is highlighted in bold. From Table 4, we observe that merging the content from hyperlinked documents into the tweets decreases ranking performance by a small margin under all retrieval models, measures and both topic sets.

Recall that in Section 3.1, we hypothesised that document weighting models might struggle to rank the new virtual documents due to poor document length normalisation, i.e. they might tend to promote documents with integrated content regardless of their relevance simply because they are more likely to match the query terms and match them more often. To investigate whether this is the case, we examine how the distribution of tweets containing hyperlinks changes between the BM25 and BM25+VirtualDoc rankings. Figure 1 reports the number of tweets retrieved by both BM25 and BM25+VirtualDoc that contain a hyperlink for the 50 topics within the *Microblog*$_{2011}$ topic set distributed into bins based upon the rank at which they were retrieved (a bin size of 10 ranks is used). From Figure 1, we see that the integration of content from the hyperlinked documents dramatically increases the number of these expanded documents that are retrieved, particularly within the top ranks. Indeed, within the top 10 results, over 33% more tweets with hyperlinks are retrieved. This shows that BM25+VirtualDoc strongly favours documents with hyperlinks over those without hyperlinks. However, it is unclear whether this is the reason for its reduced tweet ranking performance. To further examine this, Figure 2 illustrates the proportion of the relevant tweets retrieved by BM25 and BM25+VirtualDoc in the top 10 that contain hyperlinks. From Figure 2, we observe the following. First, examining the BM25 ranking, we see that the majority of relevant tweets retrieved in the top 10 have

[4] Note that the TREC Best systems use other techniques to improve performance, e.g. query expansion, tweet and/or temporal features that we do not consider in this work.

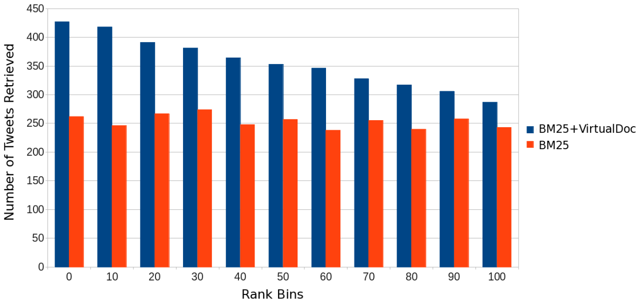

Figure 1: Comparison between the number of linked documents retrieved by both BM25 and BM25+VirtualDoc.

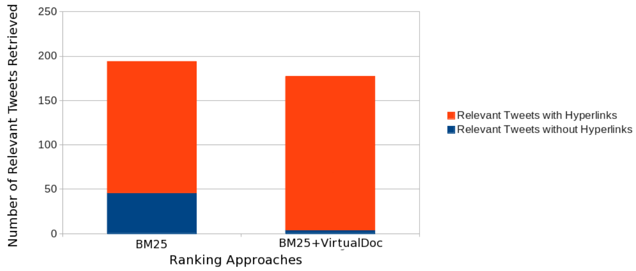

Figure 2: Proportion of relevant tweets either containing hyperlinks or not containing hyperlinks retrieved by BM25 and BM25+VirtualDoc.

hyperlinks. This shows that a proportion of the tweets that contain hyperlinks can already be ranked effectively. Secondly, we see that the BM25+VirtualDoc ranking actually retrieves more relevant tweets in the top ten that contain hyperlinks that BM25 alone. However, the majority of relevant tweets that do not contain hyperlinks are demoted by BM25+VirtualDoc out of the top ten results. These demoted tweets outnumber the additional relevant tweets promoted, explaining the reduced ranking performance. To answer our second research question, we conclude that simply integrating the content of hyperlinked documents into the linking tweet is not effective because it over-emphasises those tweets that link to documents containing the query terms, regardless of that hyperlinked document's relevance. Hence, a more fine-grained approach for integrating this content is needed.

6.3 Field-Based Integration

Having shown that the virtual document approach is not effective, we now examine the second of our proposed integration approaches, namely field-based weighting. Recall that this approach has two advantages in comparison to the virtual document approach: it enables different levels of emphasis to be placed upon the tweet and hyperlinked document fields; and it allows for term frequencies to be normalised differently for the two types of documents.

Table 5 reports the performance of two field-based weighting models, namely BM25F [26, 31] and PL2F [14], when ranking tweets in comparison to our DFReeKLIM baseline. In this case, the first field is the tweet text and the second field is the text from the hyperlinked document. Recall that we train the weights for the fields and the term normalisation parameters in a cross-topic set manner, i.e. for the *Microblog*$_{2011}$ topic set we train on the *Microblog*$_{2012}$ topics and vice-versa. The highest performing approach under each measure is highlighted in bold. Statistical significance was tested in comparison to our DFReeKLIM baseline, but neither field-based approach achieved a significance level of $p<0.05$ (paired t-test).

Approach	Microblog$_{2011}$					
	P@5	P@10	P@20	P@30	MAP	R-Precision
BM25	0.4735	0.4449	0.3663	0.3381	0.2926	0.3376
DFReeKLIM	0.5265	**0.4816**	0.4214	0.3850	0.3390	0.3899
BM25F	0.4735	0.4449	0.3663	0.3381	0.2926	0.3376
PL2F	**0.5347**	0.4796	**0.4327**	**0.3966**	**0.3442**	**0.3925**
	Microblog$_{2012}$					
	P@5	P@10	P@20	P@30	MAP	R-Precision
BM25	0.2102	0.1966	0.1873	0.1695	0.1360	0.1603
DFReeKLIM	**0.2814**	0.2390	0.2000	0.1774	0.1584	**0.1830**
BM25F	0.2237	0.1797	0.1500	0.1435	0.1193	0.1389
PL2F	0.2678	**0.2644**	**0.2110**	**0.1825**	**0.1607**	0.1807

Table 5: Effectiveness of Field-Based Document Weighting Models when the second field contents the text of any hyperlinked documents.

From Table 5, we see that comparing DFReeKLIM to BM25F, ranking effectiveness is decreased. This indicates that either the weights that BM25F learned for the two fields do not generalise between topic sets, or that any gain by adding hyperlinked document evidence is outweighed by superior ranking performance of DFReeKLIM in comparison to BM25. To examine this further, we compare the performance of BM25F to its non-field variant when ranking using the tweet text alone (BM25). When testing on the *Microblog*$_{2011}$ topic set, we see that the performance of BM25 and BM25F are identical. This is because when training (which was done on the *Microblog*$_{2012}$ topic set), the weight assigned to the hyperlinked document field was 0. In contrast, when we trained on the *Microblog*$_{2011}$ topics, a positive weight was assigned to the hyperlinked document field, which in turn resulted in BM25F outperforming BM25 when testing on the *Microblog*$_{2012}$ topics. This is a promising result, since it shows that there is some gain from using the content of the hyperlinked documents, and hence, with more training data, the performance of BM25F could be improved. Next, if we examine the performance of our second field-based document weighting model – PL2F – we see that it outperforms both BM25F and our DFReeKLIM baseline under the majority of measures (excepting the very high precision measures P@5 and R-Precision). The increased ranking effectiveness of PL2F indicates that it is better able to combine fields with very different characteristics.

For instance, one topic where PL2F outperformed DFReeKLIM was topic 48: "Egyptian evacuation". For DFReeKLIM, one of the relevant tweets was ranked 150'th: "Aussies to be evacuated from Cairo http://bit.ly/dRwUE3". Under DFReeKLIM, only the term 'evacuated' matched the query, resulting in a low score. However, PL2F ranked this tweet at rank 8 (a promotion of 142 ranks) because the hyperlinked document "http://bit.ly/dRwUE3" resolved to a relevant story, i.e. "Danny Southern and pregnant wife stuck in Cairo, amid evacuation call", which also contained the term Egypt. For comparison, this same tweet was promoted to only rank 49 using the BM25+VirtualDoc approach. The difference is that BM25+VirtualDoc also introduced many other irrelevant documents above it because it overemphasises the hyperlinked document text over the tweet text, which PL2F is less prone to do. In answer to our third research question, we conclude that using field-based weighting models to integrate hyperlinked document content into the tweet ranking process shows promise, however, the effectiveness gains observed are slight on the two TREC datasets that were tested.

6.4 Learning to Rank

Next, we evaluate the third of our approaches to integrate the evidence from hyperlinked documents into the tweet scoring process, namely using learning to rank. Recall that under this approach, we score each tweet and its hyperlinked document (if any) separately using the four baseline document weighting models for the query. The scores produced are treated as tweet ranking features, which the learning to rank algorithm combines into a score for the tweet. The full set of features that we use were summarised earlier in Table 1. Our hypothesis is that this approach will be more effective than either the virtual document and field-based approaches, since the learning to rank algorithm can combine evidence from multiple weighting models together, both when scoring tweet and the hyperlinked document.

Table 6 reports the tweet ranking performance of our learning to rank approach in comparison to our DFReeKLIM baseline over the Microblog$_{2011}$ and Microblog$_{2012}$ topic sets. Notably, we include two additional learning to rank baselines, one which combines the four document weighting models on the tweet text and the same model with the containsURL feature added, these represent the performance of a learning to rank model using evidence from the tweet only. The performance of each learned run is reported under both Cross-TopicSet and Per-TopicSet training regimes. The highest performing model is highlighted in bold. Statistically significant improvements ($p < 0.05$) over the DFReeKLIM baseline are denoted ▲.

From Table 6, we observe the following. First, comparing the most effective single document weighting model for tweet ranking tested (DFReeKLIM), to the learned combination of the four document weighting models tested, we see that performance improves over all measures, both topic sets and training regimes. This shows that while DFReeKLIM is effective, the inclusion of multiple ranking models can increase overall tweet ranking effectiveness. Second, if we compare the learning to rank baseline that combines the document weighting model baseline to the same model with the containsURL feature, we see that tweet ranking performance stays roughly constant, except when training on the Microblog$_{2012}$ topic set under Cross-Corpus training, where performance markedly improves. This is a surprising result, as it indicates that usefulness of the containsURL feature is not consistent, contrasting with observations by Duan *et. al.* [8]. However, this apparent contradiction may be accounted for when we consider that we are using a stronger learned baseline than BM25 used in their paper. Moreover, as we observed during our experiments with the virtual document approach in Section 6.2, many tweets containing hyperlinks retrieved for the Microblog$_{2011}$ topics are irrelevant.

Finally, considering the performance of our learning to rank approach when using features extracted from the hyperlinked documents (Baselines+HyperlinkScores), we see that it markedly outperforms our DFReeKLIM baseline under all measures, topic sets and training regimes. Indeed, under the MAP measure for three of the four settings tested, this improvement was statistically significant. Moreover, we see that in comparison to the most effective of the learned baselines (Baselines), our Baselines+HyperlinkScores is more effective under all measures for both topic sets using Per-TopicSet training and the Microblog$_{2012}$ topic set when using Cross-TopicSet training. This shows that our learning to rank approach is able to effectively use the content of the hyperlinked documents to improve tweet retrieval performance, answering our fourth research question. Furthermore, given that on these datasets, the containsURL feature was less effective, we can also conclude that for the Microblog$_{2011}$ and Microblog$_{2012}$ topic sets, we need to go beyond the presence of the URL in the tweet to improve the tweet ranking.

Feature Set	Feature	Microblog$_{2011}$ Weight	Microblog$_{2012}$ Weight
TweetRet.	BM25	0.0166	**0.0591**
	DPH	-0.0021	**-0.0487**
	DirichletLM	**0.0485**	**0.1111**
	DFReeKLIM	**0.0679**	**0.1284**
HyperlinkedRet.	BM25	0.0030	0.0054
	DPH	0.0011	-0.0066
	DirichletLM	**-0.0132**	**0.0270**
	DFReeKLIM	-0.0028	-0.0031
HyperlinkedSpam	Page Entropy	0.0020	0.0033
	% is stopwords	**-0.5531**	**-0.5286**
	% is table text	0.0027	0.0064
	Mean Token Length	**0.0615**	**0.0583**
	Stopword Coverage	**-0.2244**	-0.0136

Table 7: Summary of the features selected by LTR when training model on each topic set.

Also of interest is the features that our learning to rank approach selected on each topic set. Table 7 lists our features and the final weights learned for them when training on each topic set. The most influential features are highlighted. From Table 7, we that, as expected, the Tweet Retrieval feature set provides many influential features. In particular, all of the four document weighting models comprising this feature set receive a positive weight. However, surprisingly, we also observe that the learner is not placing much weight on those same models when scoring the hyperlinked documents. Instead, weight is predominantly being given to the HyperlinkedSpam feature set, specifically the stopword and mean token length features. This indicates that the general quality of the hyperlinked document is a more important feature than its relatedness to the query. Indeed, we believe that further investigation into linked document quality is a promising direction for future research, e.g. Page-rank.

7. CONCLUSIONS

In this paper, we have investigated how the content of documents hyperlinked from a tweet can be used to better estimate that tweet's relevance. We proposed three approaches for incorporating the content of hyperlinked documents when ranking tweets based upon prior works in the field of IR, namely: virtual document integration, field-based weighting and learning to rank. Through evaluation using the TREC 2011 and 2012 Microblog track topics, we empirically evaluated these approaches. In particular, we showed that the virtual document approach over-emphasises the content of the hyperlinked documents, leading to reduced overall performance. On the other hand, our results showed that the field-based approach using the PL2F field-based model and our learning to rank approach could improve retrieval performance over the baseline - highlighting the value leveraging the content of documents hyperlinked to from the tweets. For future work, we aim to further examine the topics where these three approaches harm performance, with a view toward developing enhanced approaches for real-time tweet search. We also aim to investigate further hyperlinked document quality features for tweet ranking.

Approach	Training	Four Feature Sets Used				Microblog$_{2011}$					
		TweetRet.	containsURL	HyperlinkedRet.	HyperlinkedSpam	P@5	P@10	P@20	P@30	MAP	R-Precision
DFReeKLIM	None	✗	✗	✗	✗	0.5265	0.4816	0.4214	0.3850	0.3390	0.3899
LTR	Per-TopicSet (5-folds)	✔	✗	✗	✗	0.5467	0.5002	0.4600	0.4202	0.3706	0.4056
LTR	Per-TopicSet (5-folds)	✔	✔	✗	✗	0.5307	0.4907	0.4541	0.4185	0.3733	0.4110
LTR	Per-TopicSet (5-folds)	✔	✗	✔	✔	**0.5557**	**0.5180**	**0.4637**	**0.4252**	**0.3810▲**	**0.4182**
						Microblog$_{2012}$					
DFReeKLIM	None	✗	✗	✗	✗	0.2814	0.2390	0.2000	0.1774	0.1584	0.1830
LTR	Per-TopicSet (5-folds)	✔	✗	✗	✗	0.3006	0.2610	0.2247	0.2005	0.1904	0.2151
LTR	Per-TopicSet (5-folds)	✔	✔	✗	✗	0.2903	0.2695	0.2332	0.2041	0.1995	0.2200
LTR	Per-TopicSet (5-folds)	✔	✗	✔	✔	**0.3106**	**0.2879**	**0.2458**	**0.2091**	**0.2112▲**	**0.2264**

Approach	Training	Four Feature Sets Used				Microblog$_{2011}$					
		TweetRet.	containsURL	HyperlinkedRet.	HyperlinkedSpam	P@5	P@10	P@20	P@30	MAP	R-Precision
DFReeKLIM	None	✗	✗	✗	✗	0.5265	0.4816	0.4214	0.3850	0.3390	0.3899
LTR	Cross-TopicSet	✔	✗	✗	✗	0.5224	0.5082	0.4786	0.4238	0.3735	0.4264
LTR	Cross-TopicSet	✔	✔	✗	✗	**0.5673**	**0.5388▲**	**0.4837**	**0.4429**	**0.3848▲**	**0.4347**
LTR	Cross-TopicSet	✔	✗	✔	✔	0.5265	0.4980	0.4449	0.4054	0.3461	0.3702
						Microblog$_{2012}$					
DFReeKLIM	None	✗	✗	✗	✗	0.2814	0.2390	0.2000	0.1774	0.1584	0.1830
LTR	Cross-TopicSet	✔	✗	✗	✗	0.2814	0.2644	0.2246	0.2000	0.1942	0.2142
LTR	Cross-TopicSet	✔	✔	✗	✗	0.3017	0.2661	0.2280	0.2000	0.2015▲	0.2200
LTR	Cross-TopicSet	✔	✗	✔	✔	**0.3051**	**0.2915▲**	**0.2297**	**0.2062**	**0.2087▲**	**0.2256**

Table 6: Tweet ranking effectiveness of learning to rank using features extracted from hyperlinked documents.

Acknowledgements

The authors acknowledge financial support for the CROSS Project[5] from EPSRC grant EP/J020663/1.

8. REFERENCES

[1] G. Amati, G. Amodeo, M. Bianchi, G. Marcone, C. Gaibisso, A. Celi, C. De Nicola, and M. Flammini. FUB, IASI-CNR, UNIVAQ at TREC 2011. In *Proc. of TREC 2011.*

[2] G. Amati. *Probabilistic Models for Information Retrieval based on Divergence from Randomness.* PhD thesis, University of Glasgow, 2003.

[3] G. Amati, E. Ambrosi, M. Bianchi, C. Gaibisso and G. Gambosi. FUB, IASI-CNR and University of Tor Vergata at TREC 2007 Blog Track. In *Proc. of TREC 2007.*

[4] K. Balog and M. de Rijke. Finding experts and their details in e-mail corpora. In *Proc. of WWW 2006.*

[5] M. Bendersky, W.B. Croft, and Y. Diao. Quality-biased ranking of web documents. In *Proc. of WSDM 2011.*

[6] D. Boyd, S. Golder, and G. Lotan. Tweet, tweet, retweet: Conversational aspects of retweeting on Twitter. In *Proc. of HICSS 2010.*

[7] N. Craswell, D. Hawking, A. Vercoustre, and Wilkins P. Panoptic expert: Searching for experts not just for. In *Proc. of AusWeb-2004.*

[8] Y. Duan, L. Jiang, T. Qin, M. Zhou, and H.Y. Shum. An empirical study on learning to rank of tweets. In *Proc. of COLING 2010.*

[9] M. Efron. Information search and retrieval in microblogs. *American Society for Information Science and Technology Journal*, Volume 62, Issue 6, 2011.

[10] A. Java, X. Song, T. Finin, and B. Tseng. Why we twitter: understanding microblogging usage and communities. In *Proc. of WebKDD 2007.*

[11] Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon. What is twitter, a social network or a news media. In *Proc. of WWW 2010.*

[12] T.-Y. Liu. Learning to rank for information retrieval. *Foundations Trends Information Retrieval Journal*, Volume 3, Issue 3, 2009.

[13] X. Liu, W.B. Croft, and M. Koll. Finding experts in community-based question-answering services. In *Proc. of CIKM 2005.*

[14] C. Macdonald, V. Plachouras, B. He, C. Lioma, and I. Ounis. University of Glasgow at WebCLEF 2005: Experiments in per-field normalisation and language specific stemming. In *Proc. of CLEF 2005.*

[15] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval.* Cambridge Univ. Press, 2008.

[16] K. Massoudi, M. Tsagkias, M. de Rijke, and W. Weerkamp. Incorporating query expansion and quality indicators in searching microblog posts. *Information Retrieval Journal*, 2011.

[17] C. Macdonald, R. L. T. Santos and I. Ounis. The Whens and Hows of Learning to Rank. *Information Retrieval.* Springer, 2012.

[18] R. McCreadie, I. Soboroff, J. Lin, C. Macdonald, I. Ounis, and D. McCullough. On building a reusable Twitter corpus. In *Proc. of SIGIR 2012.*

[19] D. Metzler and C. Cai. USC/ISI at TREC 2011: Microblog track. In *Proc of TREC 2011.*

[20] D. Metzler, C. Cai, and E. Hovy. Structured event retrieval over microblog archives. In *Proc. of HLT-NAACL 2012.*

[21] D. Metzler. Automatic feature selection in the markov random field model for information retrieval. In *Proc. of CIKM 2007.*

[22] G. Mishne and M. de-Rijke. A Study of Blog Search. In *Proc. of ECIR 2006.*

[23] R. Nagmoti, A. Teredesai, and M. De Cock. Ranking approaches for microblog search. In *Proc. of WI-IAT 2010.*

[24] I. Ounis, G. Amati, V. Plachouras, B. He, C. Macdonald, and C. Lioma. Terrier: A high performance and scalable information retrieval platform. In *Proc. of OSIR 2006.*

[25] I. Ounis, C. Macdonald, and I. Soboroff. Overview of the TREC-2011 Microblog Track. In *Proc. of TREC 2011.*

[26] S. Robertson, H. Zaragoza, and M. Taylor. Simple BM25 extension to multiple weighted fields. In *Proc. of CIKM 2004.*

[27] S. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In *Proc. of TREC 1994.*

[28] R. L. T. Santos, C. Macdonald, R. McCreadie, I. Ounis and I. Soboroff. Information Retrieval on the Blogosphere. *Foundations and Trends in Information Retrieval*, Volume 6, Issue 1, 2012.

[29] I. Soboroff, D. McCullough, J. Lin, C. Macdonald, I. Ounis, and R. McCreadie. Evaluating Real-Time Search over Tweets. In *Proc. of ICWSM 2012.*

[30] Q. Wu, C. Burges, K. Svore, and J. Gao. Adapting boosting for information retrieval measures. *Information Retrieval Journal*, Volume 13, Issue 3, 2010.

[31] H. Zaragoza, N. Craswell, M. Taylor, S. Saria and S. Robertson. Microsoft Cambridge at TREC?13: Web and HARD tracks In *Proc. of TREC 2004.*

[32] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst. Journal*, Volume 22, Issue 2, 2004.

[5] `http://demeter.inf.ed.ac.uk/cross/`