University of Glasgow at TREC 2014: Experiments with Terrier in Contextual Suggestion, Temporal Summarisation and Web Tracks

Richard McCreadie, Romain Deveaud, M-Dyaa Albakour, Stuart Mackie,
Nut Limsopatham, Craig Macdonald, Iadh Ounis, Thibaut Thonet*
and Bekir Taner Dinçer†
{firstname.lastname}@glasgow.ac.uk
School of Computing Science
University of Glasgow
Glasgow, UK

ABSTRACT

In TREC 2014, we focus on tackling the challenges posed by the Contextual Suggestion and Temporal Summarisation tracks, as well as enhancing our existing technologies to tackle risk-sensitivity as part of the Web track, building upon our Terrier Information Retrieval Platform. In particular, for the Contextual Suggestion track, we propose a novel bundled venue retrieval approach and experiment with text-based summarisation for building the venue description. For our participation to the Temporal Summarisation track, we propose a general framework for performing summarisation over time and two new real-time filtering approaches that leverage the semi-structured nature of news articles to enhance summary coverage. For the TREC Web track, we investigated a novel risk-sensitive learning to rank approach that is based on hypothesis testing and examined approaches that selectively apply different retrieval techniques based upon the query, with the aim of minimising risk.

1. INTRODUCTION

In TREC 2014, we participate in the Web adhoc and risk-sensitive tasks, the Contextual Suggestion track "entertain me" task and the Temporal Summarisation sequential update summarisation task. Our focus is the development of effective and efficient approaches to these tasks, building upon our open-source Terrier Information Retrieval (IR) platform [23] and extensive experience working with machine learned models [13]. Our Web track participation further develops upon the core data-driven ranking models and infrastructure within Terrier v4.0,[1] and in-line with the Terrier vision [11]. Meanwhile, our Contextual Suggestion and Temporal Summarisation participations revolve around the development of new supervised real-time streaming applications and technologies building upon Terrier.

In the Contextual Suggestion track, we have two main goals. First, to investigate how to effectively diversify venue suggestions for a given user profile. Venue diversification is an important problem, since users are unlikely to want to be recommended only a single type of venue to visit. To tackle this challenge, we propose a new venue bundling method that uses Foursquare's venue category tree to diversify the venues suggested to the user and combine it with an estimate of venue popularity from within the city. Our second goal is to examine how to generate effective venue descriptions for the end-user. Indeed, the description of the venue provided is a key factor that users leverage to decide whether they want to visit a venue. We experiment with a novel venue summarisation approach that aims to help the user to better understand why a particular venue has been suggested with respect to its popularity and relevance to them.

We also participate in the sequential update summarisation task of the Temporal Summarisation track. In contrast to our 2013 participation to this track, we developed and experimented with a new fully real-time filtering framework, that aims to eliminate summarisation latency. Indeed, within the Temporal Summarisation track task, there exists an implicit effectiveness/latency trade-off between batch-orientated 'rank-then-select' style approaches that delay the issuing of sentences until more evidence is available, in comparison to fully real-time solutions that aim to provide immediate updates. We experiment with both 'rank-then-select' and real-time filtering approaches in our participation. Furthermore, one of the core challenges to temporal summarisation is identifying updates that have no semantic overlap with the event description (query). We develop two new real-time filtering approaches that leverage the semi-structured nature of news articles when performing sentence selection to tackle this challenge. In particular, these approaches use sentence proximity to identify additional sentences that are on-topic, but do not share any semantic overlap with the event description.

In our participation to the Web track, we participated in both the adhoc and risk-sensitive tasks. Our participation in this track aims to build upon the data-driven learning infrastructure released in version 4.0 of the Terrier IR platform. In the adhoc task, we investigate how both traditional and risk-sensitive learning to rank techniques can impact upon retrieval effectiveness. Meanwhile for the risk-sensitive task, we experiment with two enhanced approaches for reducing risk during retrieval, namely: automatic selec-

*Visitor from IRIT, Paul Sabatier University, Toulouse, France. Email: thibaut.thonet@irit.fr
†Visitor from Dept. of Statistics & Computer Engineering, Mugla University, Mugla, Turkey. Email dtaner@mu.edu.tr
[1] http://terrier.org

tion of the best retrieval model via statistical analysis of 115 features; and our recently proposed risk-sensitive learning to rank technique based on hypothesis testing. We investigate how transfer learning could be used to increase the robustness of our learning to rank techniques.

The remainder of this paper is structured as follows. In Section 2, we describe our participation in the Contextual Suggestion track. Section 3 details our participation in the Temporal Summarisation track. In Section 4, we describe our Web track adhoc and risk-sensitive task participations. Conclusions are provided in Section 5.

2. CONTEXTUAL SUGGESTION TRACK

The main aim of our participation in the TREC 2014 Contextual Suggestion track is to extend and refine novel contextual retrieval models, which we have developed upon our Terrier IR platform to address emerging information needs in smart cities, such as the "entertain me" zero-queries tackled in this track.

Building on our findings from last year's track [16], we rely on Location-based Social Networks (LBSN's) such as Foursquare[2] to obtain information concerning venues that we suggest to the users. We adopted a more data-driven approach this year, where we leveraged the wealth of structured information available within Foursquare and combined them in order to learn an effective ranker. We also explored two other research questions that are central to the problem of Contextual Suggestion: the diversity of the suggested venues, and the quality of the venue description provided for each suggestion. We tackled the first question by introducing a novel bundled venue retrieval approach (BVR), which jointly ranks venues with respect to their popularity (derived from Foursquare) and their similarity to the user profile. Concerning the second question, we improved the quality of the textual description that accompanies each venue by implementing a TextRank-based summarisation method that displays the most relevant sentences extracted from positive Foursquare reviews. Also, space permitting, the method augments the summary with a list of the most similar venues from the user's profile, highlighting why it is similar and hence potentially suitable.

At the heart of our two approaches lies Foursquare, an LBSN that can be seen as a comprehensive directory of venues in the entire world. For all the contexts (i.e. cities) of the TREC 2014 Contextual Suggestion, we started by crawling all their venues from Foursquare, thus allowing us to obtain a comprehensive representation of these cities as well as a deep pool of venues that would ideally suit a wide range of users (i.e. high recall). Apart from its completeness, the main rationale behind the choice of Foursquare is that it provides several attributes about venues within a city that we can use to augment venue recommendation. For instance, using Foursquare we are able to estimate a venue's popularity, using metrics such as the number of "check-ins", the number of photos that have been taken in the venue and the number of "likes". Moreover, Foursquare provides a fine-grained categorisation of venues[3] that we can use in order to understand the venue type (e.g. "whisky bar", "creperie", "national park") and hence why a user might want to visit them.

[2] `http://foursquare.com`

[3] `http://developer.foursquare.com/categorytree`

We consider the recommendation of venues to be comprised of two main components. First, the ranking of venues for a user and a city. Second, the generation of the venue description for each venue. This year, we experimented with two very different approaches to perform the first ranking component using Foursquare data, namely: Learning-to-rank for Venues; and Bundled Venue Retrieval (BVR). Meanwhile, we propose a novel method for generating the venue description, referred to as Venue Summarisation. We describe these approaches below.

Learning-to-rank for Venues: Our first approach relies on supervised learning, for which we learn a ranking model using the LambdaMART learning to rank technique. Our aim was to build upon our recent findings [6], and to confirm that the strong results obtained on last year's data were generalisable. For each pair of context/user profile, we retrieve an initial personalised set of venues using our uogTrCFP method that already obtained above-median results in last year's track [16]. Then, we compute 64 different features for each venue based upon this initial set, before reranking them using a LambdaMART learning-to-rank model. This model was trained using the 2013 Contextual Suggestion dataset. To train this model, we defined four main feature groups:

- Category features relate to the high-level categories of the suggested venue and to their similarity to the categories of venues that the user likes in his profile.
- City features relate to the overall activity of the city, obtained by aggregating Foursquare social information over all the venues.
- User features relate to the preferences and interests of the user, expressed through the categories of venues that he likes; we also integrated features accounting for the categories of venues that the user dislikes.
- Venue features relate to the venue itself.

Bundled Venue Retrieval (BVR): For our second approach, we experimented with a novel technique that builds bundles of venues, with the goal of suggesting coherent (i.e. bundles that contain venues of the same category, or that are very similar) and relevant packages of venues to a user visiting a city. Building upon our recent findings showing that diversification is an important element when suggesting contextual venues [1], we adapted the aforementioned bundle venue retrieval approach (BVR) to suggest only the most central venues of each bundle in order to promote diversity, while fitting to the Contextual Suggestion track guidelines. Bundles were created by using two main criteria: the overall popularity of their venues (inferred from their number of Foursquare "likes") and the similarity between the user's profile and the venue. The latter has been computed using a tree-matching technique that compares two trees of categories, thus allowing to compute a finer-grained category similarity.

Venue Summarisation: In addition to these two ranking strategies, we focused this year on the generation of more meaningful and interesting descriptions for each venue suggestion, instead of simply returning the LBSN's description. Indeed, we hypothesise that the description for a venue provided can play a major role with respect to how users judge that venue's relevance. We propose a new approach that produces a summary of the venue using venue reviews as

	Submitted	P@5	MRR	TBG
TREC Median	-	0.3685	0.5350	1.3685
uogTrCFP	✖	0.0922	0.2000	0.2356
uogTrCsLtr	✔	0.3920	0.5207	1.9153
uogTrBunSum	✔	**0.4863**	**0.6653**	**2.1388**

Table 1: Results of our runs in the Contextual Suggestions track. Figures in bold represent the top performances.

follows. For each venue suggestion, we extracted the reviews that have been entered by Foursquare users for that venue. We only kept sentences that expressed a positive opinion according to the SentiStrength tool[4]. We then treated all these positive sentences as a single document and computed their TextRank [22] in order to identify the most central and salient sentences, before iteratively adding the top scoring ones to the description. However, since the guidelines specifically state that the description may be tailored to the user's preferences, we also added at the end of the description (within the 512 characters limit) a list of venues that the user has rated positively in his profile and that were the most similar to the suggested venue.

We devised three different runs to evaluate our approaches described above (uogTrCFP, uogTrCsLtr, and uogTrBunSum). Only the last two were submitted:

- **uogTrCFP**: This run serves as our baseline and was shown to be competitive in last year's track. Venues for each user profile and context pair are ranked using the similarity score between the user profile and the venue. This run also constitutes our initial sample for the uogTrCsLtr run.
- **uogTrCsLtr**: This run investigates the effectiveness of learning to rank techniques for suggesting venues. It uses a LambdaMART model learned with an ensemble of 64 features that represent both the venue and the preferences of the user [6]. All features are computed using the data that we obtained from Foursquare.
- **uogTrBunSum**: This run produces coherent and personalised bundles of popular venues, and suggests the most central venue of each bundle to the user, thus ensuring a high diversity in the suggestions. Venue popularity and personalisation are computed using venue information obtained from Foursquare. This run also implements our summarisation approach for generating meaningful descriptions.

Table 1 reports the performance of our two submitted runs and the non-submitted run together with the TREC Median using the official measures. First, we observe that our submitted runs achieve above median performance for all measures. In particular, the uogTrBunSum, which implements our bundle venue retrieval approach along with the summarisation of venues' reviews, achieves the best performance, markedly above the median. While this result suggests that combining a diversified approach with informative descriptions can help to achieve strong performance, we need further investigation to determine which of these two components provides the most added value. Our learning to rank run (uogTrCsLtr) shows that supervised learing can be very effective for venue recommendation, especially in comparison to the the lower performance achieved by our baseline run (uogTrCFP). However, we hypothesise that this supervised approach may be prone to overfitting, which might explain why it is outperformed by our uogTrBunSum approach.

[4]`http://sentistrength.wlv.ac.uk`

3. TEMPORAL SUMMARISATION TRACK

The aims of our participation in the second year of the Temporal Summarisation track [3] are two-fold. First, to transition from incremental 'rank-then-select' style summarisation approaches that issue updates each hour [16, 17], to approaches can issue updates as soon as new information arrives. Second, to investigate approaches to increase the coverage of the nuggets about an event, beyond those that can be semantically related to the event description. To this end, we develop an new modular real-time filtering framework that incorporates both filtering heuristics and supervised classification models to select sentences to issue as updates. Furthermore, using this framework as a base, we propose novel approaches that leverage sentence proximity to identify additional sentences to issue as updates that are on-topic, but do not share any semantic overlap with the event description, thereby enhancing coverage of the information nuggets about that event.

To perform summarisation, we first define a *basic real-time filtering framework* that describes a generic system for performing real-time summarisation. Under this framework, new documents are processed in real-time as they arrive, resulting in the selection of zero or more sentences from each document to issue immediately as updates to the user. Figure 1 illustrates the main components of this framework. In particular, the document is first classified as predominantly relating to the event of interest or not. On-topic documents are passed to the next component to be further processed, while the remaining documents are discarded. This initial filtering step is critical to the efficiency of the framework as a whole, since it is computationally expensive to process a document in depth (since a typical document contains hundreds of sentences), while the document stream for a given event can contain hundreds of thousands of documents. If a document is identified as being on-topic, the sentences within that document are extracted and then classified based upon the following criteria: whether they contain useful information about the event or not; whether they are well written; and whether they contain boilerplate content. Sentences that pass these classification criteria are considered as candidates to issue as updates. Finally, each candidate is compared against those previously issued as updates to avoid reporting redundant information. Those sentences identified as containing novel information are then issued as updates.

We instantiate each of the three main components of the framework as follows:

Document Filtering: Uses a machine learned document classifier trained on the TREC 2013 Temporal Summarisation topics. This classifier uses 43 features representing the similarity of the document to the initial event representation (query) and expanded event representations based upon Freebase and DBPedia.

Sentence Classification: Uses both a series of classification heuristics and a supervised sentence classification model

Run	Type	Approach	Expected Latency Gain (ELG)	Latency Comprehensiveness (LC)	ELG/LC Mean
TREC Average	N/A	N/A	0.0389	0.4840	0.0620
uogTr2A	Real-time Filtering	BWS(2)	**0.0571**	0.5564	**0.0986**
uogTr4A	Real-time Filtering	BWS(4)	0.0370	**0.6238**	0.0677
uogTr4AC	Real-time Filtering	BWS(4)+SPFS(4)	0.0451	0.5786	0.0793
uogTr4ARas	Rank-and-Select	BWS(4)	0.0500	0.4480	0.0772

Table 2: Performance of our submitted runs to the sequential update summarisation task.

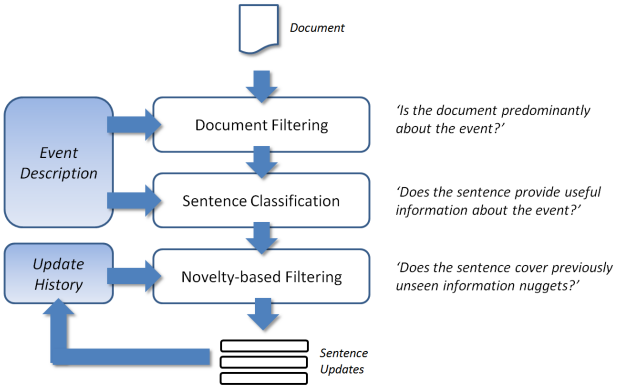

Figure 1: Overview of our basic real-time filtering framework.

to select sentences. In particular, sentences are filtered based upon their length (very short or long sentences are removed) and only sentences containing one or more named entities are considered. We then use a supervised classifier trained on a small manually annotated set of sentences extracted from the TREC 2013 Temporal Summarisation topics that aims to find well-written sentences. This classifier uses emergency-related term dictionaries, effective summarisation approaches from the literature [14, 15], sentence information mass features (e.g. sentence TF-IDF based on a background Wikipedia corpus) and quality features such as term capitalisation.

Novelty-based Filtering: We use a greedy cosine similarity heuristic to remove sentences that are overly similar to those already selected. Sentences that have a cosine similarity less than a threshold are emitted as updates. The threshold was trained on the TREC 2013 Temporal Summarisation topics.

Using this instantiation of the framework, we can automatically extract updates to issue to a user about a given event in real-time, without the hour's worth of latency introduced by the system we used last year [16]. However, another of the key challenges that we identified from our participation last year was the high degree of vocabulary mismatch that occurs between the event description (query) and the associated information nuggets for that event. For example, for the query 'buenos aires train crash', a highly relevant sentence might be 'The most likely cause was said to be brake failure', which exhibits no semantic overlap with the query. To tackle this issue, we developed two new real-time filtering approaches that leverage the semi-structured nature of news articles when performing sentence selection. In particular, our first approach, referred to as Browsing Window Selection (BWS) uses a variable-size sliding window within each document to select only sentences nearby those estimated to be relevant. Meanwhile, our second approach, which we refer to as Supervised Proximity-Focused Selection (SPFS), uses a supervised classifier that represents each sentence using a series of topicality, quality, and informativeness features extracted from other sentences in close proximity to it. In this way, we leverage the positional relationship of sentences to find additional relevant sentences that do not exhibit any semantic overlap with the event query.

Using the basic real-time filtering framework in conjunction to our BWS and SPFS approaches, we submitted four runs (uogTr2A, uogTr4A, uogTr4AC and uogTr4ARas). The first three runs aim to test the effectiveness of BWS and SPFS, while the final run enables a comparison between the 'rank-and-select' style approach deployed last year and our new proposed framework.

- **uogTr2A**: Uses the basic real-time filtering framework with BWS using a window of two sentences for selection.
- **uogTr4A**: Uses the basic real-time filtering framework with BWS using a window of four sentences for selection.
- **uogTr4AC**: The uogTr4A run combined with SPFS, where features are extracted from the closest four sentences.
- **uogTr4ARas**: The uogTr4A run that simulates a 'rank-and-select' configuration with 1 hour worth of latency.

Table 2 reports the performance of our four submitted runs in terms of expected latency gain (ELG) and latency comprehensiveness (LC) and the ELG/LC Mean (the task target metric). From our submitted runs, we observe the following points of interest. First, under all measures, all of our runs outperform the average of the submitted systems, indicating that our proposed summarisation framework is effective. Second, comparing the uogTr2A and the uogTr4A runs that use our BWS approach, we see that using a larger browsing window increases summary comprehensiveness but harms expected latency gain. Third, comparing the uogTr4A run to the uogTr4AC run that incorporates our SPFS approach, we see that SPFS increases the expected latency gain of the updates issued, but at a modest cost to comprehensiveness, resulting in a net gain under the ELG/LC mean. This indicates that our SPFS classifier is able to better identify sentences containing novel content than just using the fixed size browsing window. Finally, comparing the real-time filtering uogTr4A run to the rank-and-select uogTr4ARas run, we see that uogTr4ARas

outperforms uogTr4A, but exhibits the lowest overall comprehensiveness of any of our submitted runs. Overall, we conclude that the basic real-time filtering framework that we proposed when combined with either our BWS or SPFS approaches can be effective for sequential update summarisation, as highlighted by their enhanced performance in comparison to the TREC average. Furthermore, the high performance achieved by our BWS and SPFS approaches under Comprehensiveness indicate that using sentence proximity within documents can tackle the semantic gap when performing summarisation.

4. WEB TRACK

Our participation in the adhoc and risk-sensitive tasks had two overall goals: (1) to evaluate our recently proposed risk-sensitive learning to rank approach that is based on hypothesis testing; and (2) to continue our development of novel selective retrieval techniques that can attain an effective and robust retrieval performance. These aims build upon our existing data-driven learning infrastructure [12] that has proven effective during previous participations on ClueWeb09 [10, 18, 19, 26]. Indeed, our infrastructure encapsulates researching and deploying learning to rank approaches within Terrier using our *fat framework* [13] for the fast computation of document features. Moreover, as there are not yet many training queries available on ClueWeb12, we investigate using ClueWeb09 training data via transfer learning. Finally, we examine two approaches to minimise risk-sensitivity within a learning environment, based on risk-sensitive learning to rank [27] and the predictive selection of retrieval models per-query using estimated risk.

For TREC 2014, we indexed only the category A (∼716M English documents) subset of the ClueWeb12 corpus, without stemming or stopwords. At retrieval time, we apply one of several retrieval models (DPH from the Divergence from Randomness framework [2], DFIC from the Divergence from Independence framework [7] or BM25) to identify the *sample* documents to re-rank using the learned models. Following the recommendations of [13] for ClueWeb09, we select the top 5000 documents for re-ranking using learning to rank, where the weighting model does not consider anchor text. For applying learning to rank, all of our runs use a total of 64 features, as described in Table 3. Note that many different weighting model features are computed, as they can contribute differently to the learned models [13]. We also observe that there is no need to train the hyper-parameters of those weighting models that typically control document length normalisation, as the learning to rank technique will implicitly address any bias towards short or long documents as part of its learning process [13].

The same features are also computed on the ClueWeb09 corpus for queries from TREC 2009-2012, for the purposes of training models from the older corpus. We thereafter deploy two learning to rank techniques, namely AFS [20] – which creates a linear learned model – and also the state-of-the-art LambdaMART learning to rank technique [9, 28],[5] which creates a learned model based on regression trees.

Next, for the purposes of the risk-sensitive retrieval task, we experimented with two techniques for reducing risk during retrieval, namely (a) our recently proposed Fully-Adaptive Risk-sensitive Optimisation and Semi-Adaptive Risk-sensitive Optimisation variants of LambdaMART (known as FARO and SARO, respectively) [8], and (b) a novel selection technique that aimed to select the most effective/safe retrieval strategies for a given query. FARO and SARO are based on a new risk-sensitive evaluation measure called T_{Risk}, and are integrated into the loss function that LambdaMART deploys, to favour learned models that are less risky when compared to a baseline retrieval effectiveness. In particular, SARO concentrates on down-side risk, while FARO considers both downside and upside risk. For more information on our proposed FARO and SARO techniques, we refer the reader to [8].

Next, we investigated how the application of techniques from transfer learning can reduce risk. In particular, we mixed the transfer of learning to rank models obtained from training on the older ClueWeb09 corpus, which are then 're-trained' on ClueWeb12. Finally, through a thorough statistical analysis of 115 features that are calculated for each query, we trained a novel selection technique that aimed to select the most effective/safe retrieval strategies based upon the user query.

We submitted six runs to the adhoc and risk-sensitive retrieval tasks of the Web track, all using the category A ClueWeb12 corpus, and deploying 64 features for the purposes of learning to rank. The submitted runs were selected through a detailed cross-validation study conducted on the TREC 2013 Web track topics. In particular, for the adhoc task, we submitted three runs:

- **uogTrIwa**: Uses a DFI model and the linear AFS learning to rank technique.
- **uogTrDwl**: Uses the DFR DPH model and the LambdaAMRT learning to rank technique.
- **uogTrDuax**: Deploys the xQuAD diversification framework [25], on top of the DFR DPH model and the AFS learning to rank technique.

Meanwhile, for the risk-sensitive task, three runs were submitted, using the two standard runs as baselines, as well as one of our submitted adhoc runs:

- **uogTrDwsts** Deploys our recently proposed hypothesis testing-based risk-sensitive learning to rank technique as well as leverages transfer learning. This considers the provided standard Terrier run as the baseline during risk-sensitive learning.
- **uogTrq1**: Deploys a selective approach using different learned models on a per-query basis. The corresponding baseline for this run is uogTrDwl (as submitted to the adhoc task).
- **uogTrBwf** Uses our risk-sensitive learning to rank technique when building upon the provided Indri standard baseline.

Table 4 summarises the configuration of each of these six submitted runs, as well as several unsubmitted runs that we evaluate for comparison.

Table 5 reports the effectiveness of all six of our submitted Web track runs, as well as various unsubmitted runs, and the four provided standard baselines. Results are reported in terms of NDCG@20 and ERR@20.

[5] `http://code.google.com/p/jforests/`

Features	Total
Sample: DPH, DFIC or BM25	1
Weighting models on the whole document [13] (DFRee, DPH [2], PL2 [2], BM25, Dirichlet LM, MQT [12], LGD, DFIC [7], DFIZ [7])	8
Weighting models as above on each field, namely: title, URL, body and anchor text; + PL2F	37
Term-dependence proximity models (MRF [21], pBiL [24])	2
URL (e.g. length) link (e.g. PageRank,inlink counts) & content quality (e.g., fraction of stopwords, table text [4], spam classification [5]) features	16
TOTAL	64

Table 3: Document features used in the Web track, for both ClueWeb09 and ClueWeb12.

ID	Submitted	Stemming	Sample	LTR	Other
Terrier baseline	✖	Weak	DPH	-	-
uogTrDwl	Adhoc	Weak	DPH	LambdaMART	-
uogTrIwa	Adhoc	Weak	DFIC	AFS	-
uogTrDua	✖	None	DPH	AFS	-
uogTrDuax	Adhoc	None	DPH	AFS	xQuAD
uogTrIua	✖	No	DFIC	AFS	-
uogTrDwsts	Risk	Weak	DPH	SARO/SARO	(transfer)
uogTrq1	Risk	-	-	-	Selective (uogTrIua/uogTrDwl)
uogTrBwf	Risk	-	Indri	FARO	(transfer)

Table 4: Summary of submitted and unsubmitted runs to the adhoc and risk-sensitive tasks of the Web track.

On analysis of Table 5, we observe that all of our runs are markedly above the TREC median. Indeed, in terms of NDCG@20, the uogTrDwl run, which deploys the state-of-the-art LambdaMART learning to rank technique, was comparably the most effective, attaining 0.3243. For ERR@20, the unsubmitted uogTrIua run was our most effective, closely followed by uogTrDwl.

The xQuAD diversification technique helped to improve all measures (except ERR@20), but particularly benefited the diversity measures, namely α-NDCG@20 and ERR-IA@20. Indeed, the performance of run uogTrDuax is comparable to our best run uogTrDwl, despite using much simpler learning and less aggressive stemming.

Next, we analyse the runs submitted to the risk-sensitive task. Table 6 reports the $\mathcal{U}_{RISK}$ values for $\alpha = 0$ and $\alpha = 5$ based on ERR@20. While each risk-sensitive run (row) uses a different baseline, to permit cross comparison, we evaluate each risk-sensitive run with respect to a different evaluation baseline (column), and provide a mean column to permit an overall conclusion.

On analysis of Table 6, we first note that the runs submitted to the risk-sensitive task are less effective on average than the adhoc runs (see Table 5). Next, we observe that the selective approach used in the uogTrq1 run is overall less risky than the other runs we submitted to the risk-sensitive task (c.f. last column of the table), across all evaluation baselines, since it balances the risk using two different retrieval approaches (namely uogTrIua & uogTrDwl).

Overall, from our submitted runs, we conclude that our deployments of learning to rank and xQuAD diversification have once again been shown to be effective on ClueWeb12. Moreover, our selective approach (as deployed in run uogTrq1) provides real promise for improving the robustness of a retrieval approach. We leave for future work an analysis of FARO and SARO in the context of ClueWeb12, as well as the benefit of transfer learning.

5. CONCLUSIONS

In TREC 2014, we participated in the Web adhoc and risk-sensitive tasks, the Contextual Suggestion track "entertain me" task and the Temporal Summarisation sequential update summarisation task, using the Terrier IR platform. In particular, for the Web track, we built upon the data-driven learning infrastructure we released with Terrier 4.0, using our state-of-the-art xQuAD and Fat frameworks. We showed that state-of-the-art learning-to-rank techniques augmented with xQuAD are highly effective under both traditional and diversification metrics. Furthermore, our results for the risk-sensitive task indicate that learning how to automatically predict and select the least risky retrieval strategy shows real promise for improving search robustness. For the Contextual Suggestion track, we proposed a novel bundled venue retrieval approach that aims to diversify venue suggestion and examined how to build more effective venue descriptions using user-reviews, which in combination resulted in a marked increase in venue suggestion performance. Finally, for the Temporal Summarisation track, we proposed a new real-time summarisation framework that aims to find good quality candidate sentences to include in a temporal summary of an event, in a low latency manner. Further, using this framework as a base, we investigated two new approaches to increase the comprehensiveness of event summaries using semi-structured nature of news articles, both of which were shown to be highly effective at finding novel on-topic content relating to a given event.

6. REFERENCES

[1] M.-D. Albakour, R. Deveaud, C. Macdonald, and I. Ounis. Diversifying Contextual Suggestions from Location-based Social Networks. In *Proc. of IIiX*, 2014.

[2] 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.

[3] J. Aslam, M. Ekstrand-Abueg, V. Pavlu, F. Diaz, and T. Sakai. Trec 2013 temporal summarization. In *Proc. of TREC'13*, 2013.

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

Run	Submitted Task	Adhoc Measures		Diversity Measures	
		NDCG@20	ERR@20	α-NDCG@20	ERR-IA@20
TREC median	-	0.2549	0.1668	0.6592	0.5747
Indri baseline	-	0.2429	0.1530	0.5846	0.5130
Terrier baseline	-	0.2560	0.1887	0.6271	0.5423
uogTrIwa	Adhoc	0.2928	0.1915	0.6391	0.5387
uogTrDwl	Adhoc	**0.3243**	0.1953	**0.6823**	**0.5945**
uogTrDua	✖	0.2973	0.1976	0.6382	0.5454
uogTrDuax	Adhoc	0.3025	0.1881	0.6763	0.5878
uogTrIua	✖	0.2973	**0.1976**	0.6242	0.5454
uogTrDwsts	Risk (Terrier)	0.2749	0.1747	0.6298	0.5354
uogTrq1	Risk (uogTrDwl)	0.2888	0.1863	0.6205	0.5333
uogTrBwf	Risk (Indri)	0.2836	0.1876	0.6241	0.5317

Table 5: Results of our submitted and unsubmitted runs for the Web track under the normal adhoc measures, namely NDCG@20 and ERR@20, as well as their diversity counterparts, α-NDCG@20 and ERR-IA@20. For risk-sensitive runs, the corresponding baseline is denoted in parenthesis.

Run	Submitted Task	$\mathcal{U}_{RISK}$ (Terrier)		$\mathcal{U}_{RISK}$ (Indri)		$\mathcal{U}_{RISK}$ (uogTrDwl)		Mean $\mathcal{U}_{RISK}$	
		$\alpha = 0$	$\alpha = 5$	$\alpha = 0$	$\alpha = 5$	$\alpha = 0$	$\alpha = 5$	$\alpha = 0$	$\alpha = 5$
uogTrDwsts	Risk (Terrier)	-0.01398	-0.26885	0.02178	-0.12092	-0.02059	-0.27401	-0.01152	-0.18095
uogTrBwf	Risk (Indri)	-0.00113	-0.22992	0.03463	-0.13225	-0.00773	-0.26402	-0.00295	-0.16465
uogTrq1	Risk (uogTrDwl)	-0.00242	-0.22741	0.03334	-0.12489	-0.00902	-0.22614	0.0073	-0.1170

Table 6: $\mathcal{U}_{RISK}$ ERR@20 results of our submitted runs for the Web track risk-sensitive task. For risk-sensitive runs, the corresponding baseline for a given is denoted in parenthesis. For each $\mathcal{U}_{RISK}$ column, the corresponding evaluation baseline is also denoted in parenthesis.

[5] G. V. Cormack, M. D. Smucker, and C. L. A. Clarke. Efficient and effective spam filtering and re-ranking for large Web datasets. *Inf. Retr.*, 14(5):441–465, 2011.

[6] R. Deveaud, M.-D. Albakour, C. Macdonald, and I. Ounis. On the Importance of Venue-Dependent Features for Learning to Rank Contextual Suggestions. In *Proc. of CIKM*, 2014.

[7] B. T. Dinçer, I. Kocabas, and B. Karaoglan. Irra at trec 2010: Index term weighting by divergence from independence model. In *TREC*. National Institute of Standards and Technology (NIST), 2010.

[8] B. T. Dinçer, C. Macdonald, and I. Ounis. Hypothesis testing for the risk-sensitive evaluation of retrieval systems. In *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '14, pages 23–32, New York, NY, USA, 2014. ACM.

[9] Y. Ganjisaffar, R. Caruana, and C. Lopes. Bagging gradient-boosted trees for high precision, low variance ranking models. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information*, SIGIR '11, pages 85–94, New York, NY, USA, 2011. ACM.

[10] N. Limsopatham, R. McCreadie, M.-D. Albakour, C. Macdonald, R. L. T. Santos, and I. Ounis. University of Glasgow at TREC 2012: Experiments with Terrier in Medical Records, Microblog, and Web Tracks. In *Proc. of TREC*, 2012.

[11] C. Macdonald, R. McCreadie, R. L. Santos, and I. Ounis. From puppy to maturity: Experiences in developing terrier. *Proc. of OSIR at SIGIR*, pages 60–63, 2012.

[12] C. Macdonald, R. Santos, and I. Ounis. The whens and hows of learning to rank for web search. *Information Retrieval*, pages 1–45, 2012. 10.1007/s10791-012-9209-9.

[13] C. Macdonald, R. L. Santos, I. Ounis, and B. He. About learning models with multiple query-dependent features. *ACM Trans. Inf. Syst.*, 31(3):11:1–11:39, Aug. 2013.

[14] S. Mackie, R. McCreadie, C. Macdonald, and I. Ounis. Comparing algorithms for microblog summarisation. In *Proc. of CLEF'14*, 2014.

[15] S. Mackie, R. McCreadie, C. Macdonald, and I. Ounis. On choosing an effective automatic evaluation metric for microblog summarisation. In *Proc. of IIIX'14*, 2014.

[16] R. McCreadie, M.-D. Albakour, S. Mackie, N. Limsopatham, C. Macdonald, I. Ounis, and B. T. Dinçer. University of Glasgow at TREC 2013: Experiments with Terrier in Contextual Suggestion, Temporal Summarisation and Web Tracks. In *Proc. of TREC*, 2013.

[17] R. McCreadie, C. Macdonald, and I. Ounis. Incremental update summarization: Adaptive sentence selection based on prevalence and novelty. In *Proc. of CIKM'14*, 2014.

[18] R. McCreadie, C. Macdonald, I. Ounis, J. Peng, and R. L. T. Santos. University of Glasgow at TREC 2009: Experiments with Terrier—Blog, Entity, Million Query, Relevance Feedback, and Web tracks. In *Proc. of TREC*, 2009.

[19] R. McCreadie, C. Macdonald, R. L. T. Santos, and I. Ounis. University of glasgow at trec 2011: Experiments with terrier in crowdsourcing, microblog, and web tracks. In *Proc. of TREC*, 2011.

[20] D. Metzler. Automatic feature selection in the Markov random field model for information retrieval. In *Proc. of CIKM*, pages 253–262, 2007.

[21] D. Metzler and W. B. Croft. A markov random field

model for term dependencies. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479, New York, NY, USA, 2005. ACM Press.

[22] R. Mihalcea and P. Tarau. TextRank: Bringing Order into Text. In *Proc. of EMNLP*, 2004.

[23] 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 at SIGIR*, 2006.

[24] J. Peng, C. Macdonald, B. He, V. Plachouras, and I. Ounis. Incorporating term dependency in the DFR framework. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, 2007. ACM Press.

[25] R. L. T. Santos, C. Macdonald, and I. Ounis. Exploiting query reformulations for Web search result diversification. In *Proc. of WWW*, pages 881–890, 2010.

[26] R. L. T. Santos, R. McCreadie, C. Macdonald, and I. Ounis. University of Glasgow at TREC 2010: Experiments with Terrier in Blog and Web tracks. In *Proc. of TREC*, 2010.

[27] L. Wang, P. N. Bennett, and K. Collins-Thompson. Robust ranking models via risk-sensitive optimization. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 761–770, New York, NY, USA, 2012. ACM.

[28] Q. Wu, C. J. C. Burges, K. M. Svore, and J. Gao. Ranking, boosting, and model adaptation. Technical Report MSR-TR-2008-109, Microsoft, 2008.