Content Selection for Real-time Sports News Construction from Commentary Texts

Jin-ge Yao   Jianmin Zhang   Xiaojun Wan   Jianguo Xiao
Institute of Computer Science and Technology, Peking University, Beijing 100871, China
The MOE Key Laboratory of Computational Linguistics, Peking University, China
{yaojinge, zhangjianmin2015, wanxiaojun, xiaojianguo}@pku.edu.cn

Abstract

We study the task of constructing sports news report automatically from live commentary and focus on content selection. Rather than receiving every piece of text of a sports match before news construction, as in previous related work, we novelly verify the feasibility of a more challenging setting to generate news report on the fly by treating live text input as a stream. We design scoring functions to address different requirements of the task and use stream substitution for sentence selection. Experiments suggest that our proposed framework can already produce comparable results compared with previous work that relies on a supervised learning-to-rank model.

1 Introduction

Live text commentary services are available on the web and are becoming increasingly popular for sports fans who do not have access to live video streams due to copyright reasons. Some people may also prefer live texts on portable devices. The emergence of live texts has produced huge amount of text commentary data. Currently there exists very few studies about utilizing this rich data source.

On the other hand, manually-written sports news for game reporting usually share the same information and vocabulary as live texts for the corresponding sports game. Sports news and commentary texts can be treated as two different sources of descriptions for the same sports events. It is tempting to investigate whether we can utilize the huge amount of live texts to automatically generate sports news for sports game reporting. Building an automatic sports news generation system will largely relax the burden of sports news editors, making them free from repetitive efforts for writing while producing sports news more efficiently and covering more sports games.

As a promising starting point, one recent study (Zhang et al., 2016) successfully demonstrated that it is technically feasible to generate sports news from given live text commentary scripts. They treat the task as a special kind of document summarization and adapt supervised learning-to-rank models to learn preference for which sentences should be extracted for construction.

However, sports news providers demand more on automatic generation, from a practical point of view. Taking this to the extreme, a sports news reporter typically starts writing early following the game proceeding, without even having seen an entire game played to the final minute. Manually written match reports usually get uploaded within a few minutes after the game, which is rather speedy. An automatic writer should likewise avoid long wait times until the game finished before the writing procedure. A more natural way to view the problem is to treat commentary texts as stream data, which come in to the system one by one as input. Unfortunately, previously used strategies cannot fulfill such requirements.

In this work we proposed a simple framework as a response to stream data requirements. By studying the properties of the task, we design scoring schemes to address different aspects of the problem. To extract the subset of commentary texts that maximize the score when the data come in stream, while considering a possible overall length budget constraint, we design an efficient stream substitution algorithm

Proceedings of The 10th International Natural Language Generation conference, pages 31–40,
Santiago de Compostela, Spain, September 4-7 2017. ©2017 Association for Computational Linguistics

that requires only single pass of data, based on a priority queue implementation. The overall framework forms a rather simple, efficient, practical approach that produces results comparable to the supervised learning-to-rank framework used by previous studies that involves rather heavy feature engineering, as shown on a real world dataset containing Chinese commentary texts.

2 Task Formulation

Following Zhang et al. (2016), we can also treat the task of constructing sports news from commentary texts as a special type of extractive summarization: extracting sentences from commentary scripts to form a news report for the described sports game. Formally, given the commentary texts for a sports match, containing a collection of candidate sentences $U = \{s_1, s_2, \dots, s_n\}$, the goal is to extract a subset of sentences $S \in U$ to form a summary report for the match. For experimental comparison, we require the total length of selected extraction not to exceed a pre-specified length budget B measured by the total number of Chinese characters.

Compared with generic document summarization, the candidate sentence processed here has a richer structure. Other than commentary text, it also contains the time when the currently described action or event happens, along with a current scoreline. See Table 1 for an example segment of commentary texts that we used for experiments, consisting of texts crawled from easily available sports live texts.[1]

Time	Scoreline	Commentary Texts
21’	1-0	The flag is up for a foul from Costa.
22’	2-0	2-0! Goal for Everton!
22’	2-0	The substitute Naismith scored twice to establish the two-goal lead for his team.

Table 1: Example excerpt of commentary data format

More importantly, in this work we emphasize that our data are assumed to come in stream, provided in a real-time fashion. In other words, we receive commentary sentences one by one during the sports game playing, without seeing the description of future events. The goal here is to perform sentence extraction simultaneously, with the hope that once the game finishes, we immediately get the news report right in the first second to ensure that the automatic writer is faster than a human author.

As an additional comment, this setting directly blocks the possibility to apply standard document summarization methodologies as they often involve global optimization or sentence graph ranking, requiring global information that cannot be well captured by a partial stream of data.[2] The effective approaches based on learning-to-rank models used in (Zhang et al., 2016) cannot be adapted here either. Instead, structurally simpler frameworks should be used, such as element-wise regression or classification, direct function evaluation, etc. Since it is nontrivial to address the length budget as well as some other possible requirements when training a supervised learning model, we opt for an even more simpler way of designing characterizing functions to address different aspect of the task, followed by a properly designed stream algorithm for content selection.

3 Our Proposed Approach

The framework of the approach proposed in this paper has a rather simple nature. Once a piece of commentary data comes, the system immediately performs a scoring function evaluation for it, and store the top scoring pieces of text in memory, while conforming to the total length budget. See Figure 1 for an overview illustration of the framework we leverage in this study.

3.1 Sentence Scoring

The way to describe sports events in commentary is different from that in a written news report. There are multiple aspects that should be taken into consideration if we would like to use commentary sentences to construct sports news.

3.1.1 Importance of described actions

The most straightforward criterion for deciding whether to preserve a sentence for news construc-

[1] We will be using the data collected by previous work which contain Chinese texts only. For succinctness we only show corresponding English translations here in this paper.

[2] Exceptions mainly include methods that are based on singleton predictions by simply ignoring structural dependencies or relative preferences.

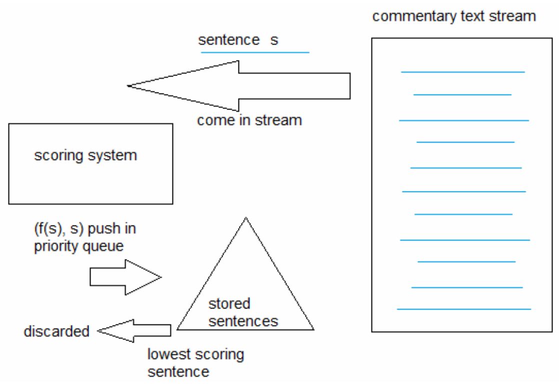

Figure 1: Overview of the framework

tion is to quantify the amount of importance for the described actions or events in the sports game. In sports events, actions are mostly described using verbs, nouns and compound nouns. Therefore, we count the main importance calculations on such types of words only for an efficient approximate processing. Specifically, we define *indicator words* to be all valid nouns and verbs that are not in the stop words list. Such words characterize the main indicative information described in the sentence, typically covering actions, events and locations (e.g. specific area of the pitch or arena). Note that here proper nouns such as team names or player names should be excluded, since almost every piece of commentary sentence contains such proper nouns, making these words not discriminative for our local decisions of sentence extraction.

For a given sentence s, we use $I(s)$ to denote the collection of the *indicator words*, i.e. all valid nouns and verbs contained in sentence s excluding proper nouns. The question is how to find those words that are more important and more indicative, and have a strong tendency to be selected to compose news reports.

We separately characterize two aspects: the *tendency* to be described in news reports, as well as individual *importance*. We rely on simple corpus statistics to address the estimation problem of each. Specifically, for *tendency* estimation we first align descriptions in live commentary texts to the corresponding human written news, utilizing the time stamps as well. We mark the frequency count of indicator words that can be aligned to manual news as $C_{aligned}(w)$, and use $C_{total}(w)$ to denote the total frequency of w appearing in the live texts. For *importance* estimation, we crawled another collection of sports news, without any need to be aligned to any commentary texts. We simply use the logarithm of frequency counts[3] of an indicator word appeared in manually written news, as an importance indicator.

In summary, the importance score for an indicator word w is defined as

$$imp(w) = \frac{C_{aligned}(w)}{C_{total}(w)} \log C_{news}(w), \qquad (1)$$

In Table 2 we list a proportion of the top scoring indicator words (translated from original Chinese data) as calculated by the aforementioned method. We can observe that the words that are assigned to be indicator words are indeed intuitive as they capture the most important events, motions, or key locations in some cases, during a soccer game. In the implementation a few manual modifications of indicators have been made to promote more reasonable selection.

shoot (shè)	shot (shèmén)	substitution
find	score	penalty area
change	goal	red card
threat	one-on-one	top corner

Table 2: Example top scoring indicator words

3.1.2 Description style

Descriptive languages in sports commentary and news report are different in general. However, they also share some commonalities since they are describing the same events. For selecting sentences to form news reports, we may tend to preserve those that are close to the description of news already. With the minimum amount of post-editing they can almost be directly used for news construction.

To find sentences that are close to news descriptions style, we make use of the additionally crawled data used for calculating individual importance of indicator words, as described earlier. Specifically, we use log bigram frequency to conceptually simulate an effect of a n-gram language model. In this step we also exclude proper nouns as usual, to

[3] We do not use raw counts since they are in greater scale and the differences between words are huge and too sensitive to the specific corpus.

exclude pairs that are not generalizable to games played between different teams and different players. A bit more formally, we write $bigram(s)$ for a given sentence s to denote the description quality characterized by news bigrams:

$$bigram(s) = \sum_{b \in bigrams(s)} \log C_{news}(b) \quad (2)$$

In Table 3 we also show the top scoring bigrams to depict what kinds of local wording choices are typically used in sports news. The bigrams are formed with Chinese words, therefore they may not correspond to English bigrams formally. We fill some commonly appearing compositions in parentheses in order to show the use of bigrams with more clarity.

inside the penalty area	the shot was (saved)
was shown (the yellow card)	the shot from
hit the (bar/post)	minutes later
just wide	was over

Table 3: Example top scoring bigrams

Note that such scoring scheme may also partially characterize the preservation of important information from a slightly different angle, leading to possibly overlapping effects with the previous aspect as we described. Bigrams counts have been used to capture concept importance in previous work on summarization as well (Gillick et al., 2008; Gillick et al., 2009). In the implementation a few noisy bigrams have been manually filtered to promote more reasonable selection.

3.1.3 Closeness to key changes

For every type of sport there exist certain types of key events that should definitely be reported in the news, possibly in slightly more details. Take soccer for example, the most important change during a game is the scoreline change, triggered by goal scoring events. It is appropriate to assign related descriptions with higher sentence scores.

We characterize the closeness of a sentence at time t to the latest scoreline change point t' as:

$$sc(t, t') = \exp(-\frac{|t - t'|^2}{2\sigma^2}), \quad (3)$$

where σ is a width parameter for controlling the scale of difference. A larger σ assign less preference for those who are close to the scoreline change point, but not as close enough. For simplicity we directly set $\sigma = 1$ in this work.

3.1.4 Sentence scoring function in sum

Taking all three aspects we just described together, we form the scoring function for a given sentence s using a simple summation as follows:

$$f(s) = bigram(s) + \sum_{w \in I(s)} imp(w) + sc(s.time, t'), \quad (4)$$

where each term has been described in earlier subsections, addressing different aspects for the task respectively.

The overall target is then to select a subset S of sentences from the total commentary set U, under a length budget. The score of a subset S is simply $f(S) = \sum_{s \in S} f(s)$.

3.2 Stream Data Selection

Our ultimate goal is to select commentary texts that maximize the total score as defined in the previous section, aiming at keeping the most important, representative pieces of information. It is intuitive and easy to verify that when ignoring the third closeness term (3), the remaining bigram scores and indicator importance scores actually satisfy the *submodularity* property,[4] i.e. for $\forall S \subseteq T \subseteq U \setminus u$, we have:

$$f(S \cup \{u\}) - f(S) \geq f(T \cup \{u\}) - f(T). \quad (5)$$

As a result, our objective function consists of a submodular proportion along with a bounded term, as $sc() \leq 1$, which contributes a small, controllable proportion to the total score. Therefore we can treat our objective function as near-submodular.

There exist some studies exploring the strategies to approximately optimize a submodular function in stream data settings, also without seeing the entirety while only a small, constant proportion of memory usage is allowed. For example, the sieving approach proposed in (Badanidiyuru et al., 2014) provides an efficient streaming algorithm that has a constant factor of $1/2 - \epsilon$ of approximation guarantee to the optimal solution, while only requiring a

[4] They are in fact *modular*, i.e. both f and $-f$ are submodular.

data-independent size of memory and a single pass through the data stream.

Most off-the-shelf algorithms for submodular maximization in stream data settings are designed for cases where cardinality constraints are involved, i.e. restricting the total selected number by requiring $|S| \leq k$ where k is a predefined constant integer. This setting is different with what we care about in this work: a knapsack constraint $\sum_{s \in S} length(s) \leq B$ restricting the total length.

As a result, we develop a new stream algorithm called *heap substitution* that fits our target well and finds a good approximate solution. The algorithm can be treated as an adapted version of an earlier work (Krause and Gomes, 2010), which may not be optimal in a theoretical sense, but can achieve very good approximate solution. The nature of our algorithm is simple: keep a priority queue (implemented using a heap) for the currently selected sentences to be preserved. Once the budget constraint could be violated by introducing the current commentary sentence, we push it to the priority queue and pop out the sentence that is evaluated with the least amount of score in the queue. The algorithm is listed in pseudocode in Algorithm 1.

Algorithm 1 The heap substitution algorithm

```
Input:
    Sentences $\{s_i\}$ coming in stream; predefined budget $B$
Output:
    The sentence set $S$ in the sports news;
Initialize $S = \emptyset$ stored in a minimal heap
if $f(s_i) > 0$ then
    if $|S \cup \{s_i\}| \leq B$ then
        $S = S \cup \{s_i\}$
    else
        Push $s_i$ onto the heap $S$
        Pop the top (minimum score) element of $S$
    end if
end if
return $S$
```

As far as we know, currently there exist no study for stream submodular maximization under such knapsack constraints. As theoretical analysis is not the major focus of this work, we leave it as a future work to generalize the algorithm to more generic scenarios beyond sports news construction. Meanwhile, there exist additional stream algorithms with better theoretical or practical properties for cardinality constrained submodular maximization. The modified (multi-)sieve streaming algorithm described in (Badanidiyuru et al., 2014) can be served as an example. These algorithms may also be adapted, but perhaps technically more demanding for this study. We leave such further variants and comparisons to future work study.

4 Experiments

4.1 Data

There are not many datasets available for the particular stream data setting studied in this paper. However, generic datasets for sports news construction actually suffice for our purpose, as long as we treat the texts as stream data and simply ignore future observations during calculation and prediction, while consuming a tiny proportion of memory usage. To form direct comparison with previous work, we simply use the same dataset as constructed in (Zhang et al., 2016). The authors find Chinese commentary text rather easy to acquire and crawled 150 football matches on Sina Sports Live, each assigned with two manually written news reports for the purpose of training or evaluation.

Following previous work, we perform cross-validation during evaluation to utilize the dataset more sufficiently and to draw more reliable conclusions. Specifically, we randomly divide the dataset into three parts with equal sizes, each contains 50 pairs of live texts and gold-standard news. Each time we set one of them as the test set and use the remaining two parts for training, or specific types of corpus statistics as used in our method. We will mainly report the averaged results from all three folds.

4.2 Evaluation Metrics

Similar to the evaluation for traditional summarization tasks, we use the ROUGE metrics (Lin and Hovy, 2003) to automatically evaluate the quality of produced summaries given the gold-standard reference news. The ROUGE metrics measure summary quality by counting the precision, recall and F-score of overlapping units, such as n-grams and skip grams, between a candidate summary and the reference summaries.

Specifically, we report the F-scores of the following metrics in the experimental results: ROUGE-1 (unigram-based), ROUGE-2 (bigram-based) and ROUGE-SU4 (based on skip bigrams with a maximum skip distance of 4). Note that the ROUGE scores are computed for each document set, and then the scores are averaged. We use the ROUGE-1.5.5 toolkit to perform the evaluation.

Note that the results are slightly different with those reported in (Zhang et al., 2016). As we understand, in that work the ROUGE overlaps are calculated based on a rather weak word segmentation tool that breaks many named entities into separated characters or subwords, which boosts the ROUGE quantities slightly larger than expected and may incorrectly reflect the preference between each other. The ROUGE distance between system outputs and gold standard manually written news, which should be treated as an upper bound, is somewhat close. In this work the evaluation is based on another popular Chinese word segmentation toolkit called Jieba,[5] that performs word segmentation results with satisfactory level of accuracy, when provided external sports dictionary.

We also conduct manual evaluation in this study. Specifically, we use the pyramid method (Nenkova and Passonneau, 2004) and modified pyramid scores as described in (Passonneau et al., 2005) to manually evaluate the summaries generated by different methods. We randomly sample 20 games from the data set and manually annotate facts on the gold-standard news. The annotated facts are mostly describing specific events happened during the game. Each fact is treated as a Summarization Content Unit (SCU) (Nenkova and Passonneau, 2004). The number of occurrences for each SCU in the gold-standard news is regarded as the weight of this SCU.

Two types of scores for peers were computed from the peer annotations. Both scores are a ratio of the sum of the weights of the SCUs found in the generated summary (OBServed) to the sum for an ideal gold-standard news (MAXimum). If the number of SCUs of a given weight i that occur in a summary is O_i, the sum of the weights of all the SCUs in a summary is:

$$OBS = \sum_{i=1}^{n} i \times O_i$$

In the original pyramid scoring, the number of SCUs used in computing MAX is the same as the number used to compute OBS. The score is defined as the ratio OBS/MAX. In a more commonly used modified score, MAX_M is computed instead of MAX using the average number of SCUs found in all gold-standard news and the score is defined as the ratio OBS/MAX_M. This modified version avoids assigning high scores to summaries that have retrieved very few SCUs. We conform to the modified version during pyramid evaluation.

4.3 Baselines

The most straightforward baseline is directly performing singleton regression or classification. Specifically, the support vector machine (SVM) and support vector regression (SVR) model serve as strong supervised baselines. We utilize the LIBSVM implementation[6] (Chang and Lin, 2011) with the RBF kernel for classification/regression. We reimplemented the features described in (Zhang et al., 2016) which turn out to be effective for this task. As in stream data settings, features that depends on future observations are not used.

We also generate results from batch processing systems for reference. Specifically, we implemented graph-based document summarization approach including centroid-based summarization (Radev et al., 2000) and the well-known LexRank (Erkan and Radev, 2004). We also rebuilt the learning-to-rank system followed with a probabilistic greedy selection procedure, as used by (Zhang et al., 2016) based on random forests of LambdaMART rankers, and observed similar results as the authors reported. The produced results have been verified to be similar to those reported in their paper, if using the same word segmentation procedure.

4.4 Results

Table 4 lists the results for different output systems. The results of this work is significantly different ($p < 0.01$) with all baseline systems but LTR,

[5] https://github.com/fxsjy/jieba/

[6] http://www.csie.ntu.edu.tw/ cjlin/libsvm/

with the difference between LTR and the proposed method only at the significant level of 0.1. Bonferroni adjustment (Bonferroni, 1936; Bland and Altman, 1995) has been considered when calculating p-values for multiple comparisons.

From the table we can observe that our proposed method for stream settings clearly outperform graph-based baseline approaches as well as the singleton-prediction baselines of SVR and SVM, while producing comparable or better results compared with the state-of-the-art learning-to-rank system which involves heavy feature engineering.

Since SVR and SVM baselines are trained on labels derived from ROUGE values, they may not learn the discriminative behaviors between those features that lead to preservation or those that are not suitable to be preserved for news construction. Our proposed scoring function is able to address this issue by *direct word-level control*, therefore yielding better results. Note that conceptually the same issue exists in the learning-to-rank system as well and as a result one may observe that there exist a significant improve from the proposed method, in terms of the pyramid score.

Graph-based summarization approaches have been shown not suitable for the task of commentary based sports news construction. The results in this work also corroborate such observations.

Table 5 shows an example output[7] of extracted sentences by our method for the Everton vs Manchester City game played in the English Premier League at season 2015-2016. We can observe that our system is able to capture most of the possible key moments during the game, with a tiny proportion of less important descriptions. There also exist some more difficult problems which we do not focus in this study. For example, the second sentence clearly has a problem of zero anaphora, without clearly stating who is performing the long shot.

4.5 Ablation Analysis

To test the contribution of each component in the scoring scheme, we form combinations by removing each group of scoring respectively. Table 6 shows the results, with "-" denotes experiments without the corresponding group.

(Preview) Man City won only one out of
the last six away games at Goodison Park.
(2') A long range effort is blocked.
(3') Corner for Man City. The ball is crossed
to the middle and headed out by the defender.
...
(60') Sterling dribbles to the penalty area,
throughs the ball to Kolarov,
and Kolarov finds the net from a tight angle.
(63') Kone's shot in the box is blocked.
...
(89') A clever pass from Yaya Toure to the box,
Nasri moves forward, lobs the keeper to score.
(FT) The game finishes at 0-2.

Table 5: Example output for the Everton vs Man City game

System	R-1	R-2	R-SU4	Pyramid
All	0.33247	0.10223	0.13478	0.80759
-bigram	0.31262	0.09412	0.11628	0.77310
-import.	0.30759	0.08660	0.11744	0.65241
-closen.	0.31148	0.09064	0.11749	0.79034

Table 6: Score ablation results

We can observe that removing any of the three components degrades the overall performance. The results also suggest that the indicator word scores contribute the most. This is natural since the indicator part of score has some form of supervision from calculating statistics on aligned training data.

5 Related Work

5.1 Sports News Generation

To the best of our knowledge, generation of sports news by utilizing commentary texts is not a well-studied task in related fields. Very few related work can be backtracked other than the study of (Zhang et al., 2016) which treats the task as single document summarization and develop a supervised learning-to-rank framework to show the feasibility of this task. A few earlier studies attempted to generate sports report from structured data such as event tables (Lareau et al., 2011) and ontology-based knowledge base (Bouayad-Agha et al., 2011; Bouayad-Agha et al., 2012), based on predefined templates. There exist some related studies that focused on generating textual summaries for sports events from status up-

[7] We omit part of the extracted descriptions in the middle due to space limit.

System	ROUGE-1	ROUGE-2	ROUGE-SU4	Pyramid
Centroid	0.26201	0.05150	0.08146	0.32483
LexRank	0.24456	0.03533	0.06609	0.29034
SVR	0.30502	0.07371	0.10532	0.42828
SVM	0.30934	0.07482	0.10681	0.46276
LTR	0.32489	0.09464	0.12319	0.56621
This work	0.33247	0.10223	0.13478	0.80759
Gold-standard	0.40802	0.12924	0.16407	0.88219

Table 4: Evaluation results of different approaches

dates in Twitter (Nichols et al., 2012; Kubo et al., 2013; Tagawa and Shimada, 2016). There also exists earlier work from study groups that do not focus on text analysis or language processing, studying generation of sports highlight frames from sports videos, focusing on a very different type of data (Tjondronegoro et al., 2004).

5.2 Submodular Maximization

As we mentioned earlier, the designed scoring function is near submodular. Maximization of submodular functions is a well-studied topic in machine learning and algorithmic analysis, It has been applied to many tasks such as document summarization (Lin and Bilmes, 2010), sensor placement (Krause et al., 2006) network inference (Gomez Rodriguez et al., 2010) and many more applications, with the aim of balancing the coverage or quality measures of selected items while encouraging diversity in selection.

5.3 Stream Data Processing

Stream data settings are becoming popular due to the fact that it is natural in many tasks where enormous amount of data are coming one by one (Gaber et al., 2005). For maximizing submodular functions, there already exist a number of stream algorithms (Krause and Gomes, 2010; Badanidiyuru et al., 2014; Kumar et al., 2015). The heap substitution algorithm we designed in this work resembles the algorithm developed in (Krause and Gomes, 2010) that addresses cardinality constraints. In the task settings for this study the budget is limited in sentences lengths measured by total number of characters, which leads to a knapsack constraint rather than cardinality constraints that are easier to deal with.

5.4 Document Summarization

The approach of selecting sentences to construct news reports can be treated as a special kind of document summarization (Nenkova et al., 2011). Among the large number of papers in summarization literature, some of them are based on simple definitions of sentence scoring with different components addressing different requirements in specific task settings (Yih et al., 2007; Christensen et al., 2013; Macdonald and Siddharthan, 2016, for instance), which is similar to this paper. The main difference between document summarization and the task in this study is the way to characterize importance.

6 Conclusion

We study the task of constructing sports news in real time, treating live commentary texts as stream input. The nature of this setting blocks the use of preference learning or global optimization. As a result, we proposed a more straightforward procedure to perform online evaluation and prediction. We develop a simple heap substitution algorithm to decide which texts should be preserved, subject to a predefined length constraint. Experiments show that our proposed method works well on real world datasets and yields comparable results to the state-of-the-art learning-to-rank framework.

Acknowledgments

This work was supported by 863 Program of China (2015AA015403), NSFC (61331011), and Key Laboratory of Science, Technology and Standard in Press Industry (Key Laboratory of Intelligent Press Media Technology). We thank the anonymous reviewers for helpful comments. Xiaojun Wan is the corresponding author.

References

Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. 2014. Streaming submodular maximization: Massive data summarization on the fly. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 671–680. ACM.

J Martin Bland and Douglas G Altman. 1995. Multiple significance tests: the bonferroni method. *Bmj*, 310(6973):170.

Carlo E Bonferroni. 1936. *Teoria statistica delle classi e calcolo delle probabilita*. Libreria internazionale Seeber.

Nadjet Bouayad-Agha, Gerard Casamayor, and Leo Wanner. 2011. Content selection from an ontology-based knowledge base for the generation of football summaries. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 72–81. Association for Computational Linguistics.

Nadjet Bouayad-Agha, Gerard Casamayor, Simon Mille, and Leo Wanner. 2012. Perspective-oriented generation of football match summaries: Old tasks, new challenges. *ACM Transactions on Speech and Language Processing (TSLP)*, 9(2):3.

Chih-Chung Chang and Chih-Jen Lin. 2011. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2013. Towards coherent multi-document summarization. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1163–1173, Atlanta, Georgia, June. Association for Computational Linguistics.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, pages 457–479.

Mohamed Medhat Gaber, Arkady Zaslavsky, and Shonali Krishnaswamy. 2005. Mining data streams: a review. *ACM Sigmod Record*, 34(2):18–26.

Dan Gillick, Benoit Favre, and Dilek Hakkani-Tur. 2008. The icsi summarization system at tac 2008. In *Proceedings of the Text Understanding Conference*.

Dan Gillick, Benoit Favre, Dilek Hakkani-Tur, Berndt Bohnet, Yang Liu, and Shasha Xie. 2009. The icsi/utd summarization system at tac 2009. In *Proceedings of the Second Text Analysis Conference, Gaithersburg, Maryland, USA. National Institute of Standards and Technology*.

Manuel Gomez Rodriguez, Jure Leskovec, and Andreas Krause. 2010. Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1019–1028. ACM.

Andreas Krause and Ryan G Gomes. 2010. Budgeted nonparametric learning from data streams. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 391–398.

Andreas Krause, Carlos Guestrin, Anupam Gupta, and Jon Kleinberg. 2006. Near-optimal sensor placements: Maximizing information while minimizing communication cost. In *Proceedings of the 5th international conference on Information processing in sensor networks*, pages 2–10. ACM.

Mitsumasa Kubo, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2013. Generating live sports updates from twitter by finding good reporters. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013 IEEE/WIC/ACM International Joint Conferences on*, volume 1, pages 527–534. IEEE.

Ravi Kumar, Benjamin Moseley, Sergei Vassilvitskii, and Andrea Vattani. 2015. Fast greedy algorithms in mapreduce and streaming. *ACM Transactions on Parallel Computing*, 2(3):14.

François Lareau, Mark Dras, and Robert Dale. 2011. Detecting interesting event sequences for sports reporting. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 200–205, Nancy, France, September. Association for Computational Linguistics.

Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 912–920. Association for Computational Linguistics.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic e-valuation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 71–78. Association for Computational Linguistics.

Iain Macdonald and Advaith Siddharthan. 2016. Summarising news stories for children. In *Proceedings of the 9th International Natural Language Generation conference*, pages 1–10, Edinburgh, UK, September 5-8. Association for Computational Linguistics.

Ani Nenkova and Rebecca Passonneau. 2004. Evaluating content selection in summarization: The pyramid method.

Ani Nenkova, Kathleen McKeown, et al. 2011. Automatic summarization. *Foundations and Trends® in Information Retrieval*, 5(2–3):103–233.

Jeffrey Nichols, Jalal Mahmud, and Clemens Drews. 2012. Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, pages 189–198. ACM.

Rebecca J Passonneau, Ani Nenkova, Kathleen McKeown, and Sergey Sigelman. 2005. Applying the pyramid method in duc 2005. In *Proceedings of the Document Understanding Conference (DUC 05), Vancouver, BC, Canada*.

Dragomir R Radev, Hongyan Jing, and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic summarization*, pages 21–30. Association for Computational Linguistics.

Yuuki Tagawa and Kazutaka Shimada. 2016. Generating abstractive summaries of sports games from japanese tweets. In *Advanced Applied Informatics (IIAI-AAI), 2016 5th IIAI International Congress on*, pages 82–87. IEEE.

Dian Tjondronegoro, Yi-Ping Phoebe Chen, and Binh Pham. 2004. Integrating highlights for more complete sports video summarization. *IEEE multimedia*, 11(4):22–37.

Wen-tau Yih, Joshua Goodman, Lucy Vanderwende, and Hisami Suzuki. 2007. Multi-document summarization by maximizing informative content-words. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1776–1782, Hyderabad, India, January 6-12.

Jianmin Zhang, Jin-ge Yao, and Xiaojun Wan. 2016. Towards constructing sports news from live text commentary. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1361–1371, Berlin, Germany, August. Association for Computational Linguistics.