Co-Feedback Ranking for Query-Focused Summarization

Furu Wei[1,2,3] **Wenjie Li**[1] **and Yanxiang He**[2]

[1] The Hong Kong Polytechnic University, Hong Kong
{csfwei,cswjli}@comp.polyu.edu.hk

[2] Wuhan University, China
{frwei,yxhe}@whu.edu.cn

[3] IBM China Research Laboratory, Beijing, China

Abstract

In this paper, we propose a novel ranking framework – Co-Feedback Ranking (Co-FRank), which allows two base rankers to supervise each other during the ranking process by providing their own ranking results as feedback to the other parties so as to boost the ranking performance. The mutual ranking refinement process continues until the two base rankers cannot learn from each other any more. The overall performance is improved by the enhancement of the base rankers through the mutual learning mechanism. We apply this framework to the sentence ranking problem in query-focused summarization and evaluate its effectiveness on the DUC 2005 data set. The results are promising.

1 Introduction and Background

Sentence ranking is the issue of most concern in extractive summarization. Feature-based approaches rank the sentences based on the features elaborately designed to characterize the different aspects of the sentences. They have been extensively investigated in the past due to their easy implementation and the ability to achieve promising results. The use of feature-based ranking has led to many successful (e.g. top five) systems in DUC 2005-2007 query-focused summarization (Over et al., 2007). A variety of statistical and linguistic features, such as term distribution, sentence length, sentence position, and named entity, etc., can be found in literature. Among them, query relevance, centroid (Radev et al., 2004) and signature term (Lin and Hovy, 2000) are most remarkable.

There are two alternative approaches to integrate the features. One is to combine features into a unified representation first, and then use it to rank the sentences. The other is to utilize rank fusion or rank aggregation techniques to combine the ranking results (orders, ranks or scores) produced by the multiple ranking functions into a unified rank. The most popular implementation of the latter approaches is to linearly combine the features to obtain an overall score which is then used as the ranking criterion. The weights of the features are either experimentally tuned or automatically derived by applying learning-based mechanisms. However, both of the above-mentioned "*combine-then-rank*" and "*rank-then-combine*" approaches have a common drawback. They do not make full use of the information provided by the different ranking functions and neglect the interaction among them before combination. We believe that each individual ranking function (we call it base ranker) is able to provide valuable information to the other base rankers such that they learn from each other by means of mutual ranking refinement, which in turn results in overall improvement in ranking. To the best of our knowledge, this is a research area that has not been well addressed in the past.

The inspiration for the work presented in this paper comes from the idea of Co-Training (Blum and Mitchell, 1998), which is a very successful paradigm in the semi-supervised learning framework for classification. In essence, co-training employs two weak classifiers that help augment each other to boost the performance of the learning algorithms. Two classifiers mutually cooperate with each other by providing their own labeling results to enrich the training data for the other parties during the supervised learning process. Analogously, in the context of ranking, although each base ranker cannot decide the overall ranking well on itself, its ranking results indeed reflect its opinion towards the ranking from its point of view. The two base rankers can then share their own opinions by providing the ranking results to each other as feedback. For each ranker, the feedback from the other ranker contains additional information to guide the refinement of its ranking results if the feedback is defined and used appropriately. This process continues until the two base rankers can not learn from each other any more. We call this ranking paradigm Co-Feedback Ranking (Co-FRank). The way how to use the feedback information

Proceedings of the ACL-IJCNLP 2009 Conference Short Papers, pages 117–120,
Suntec, Singapore, 4 August 2009. ©2009 ACL and AFNLP

varies depending on the nature of a ranking task. In this paper, we particularly consider the task of query-focused summarization. We design a new sentence ranking algorithm which allows a query-dependent ranker and a query-independent ranker mutually learn from each other under the Co-FRank framework.

2 Co-Feedback Ranking for Query-Focused Summarization

2.1 Co-Feedback Ranking Framework

Given a set of objects O, one can define two base ranker f_1 and f_2: $f_1(o) \rightarrow \Re, f_2(o) \rightarrow \Re, \forall o \in O$. The ranking results produced by f_1 and f_2 individually are by no means perfect but the two rankers can provide relatively reasonable ranking information to supervise each other so as to jointly improve themselves. One way to do Co-Feedback ranking is to take the most confident ranking results (e.g. highly ranked instances based on orders, ranks or scores) from one base ranker as feedback to update the other's ranking results, and vice versa. This process continues iteratively until the termination condition is reached, as depicted in Procedure 1. While the standard Co-Training algorithm requires two sufficient and redundant views, we suggest f_1 and f_2 be two independent rankers which emphasize two different aspects of the objects in O.

```
Procedure 1. Co-FRank($f_1$, $f_2$, $O$)
1: Rank $O$ with $f_1$ and obtain the ranking results $r_1$;
2: Rank $O$ with $f_2$ and obtain the ranking results $r_2$;
3: Repeat
4:     Select the top $N$ ranked objects $\tau_1$ from $r_1$ as
       feedback to supervise $f_2$, and re-rank $O$ using $f_2$
       and $\tau_1$; Update $r_2$;
5:     Select the top $N$ ranked objects $\tau_2$ from $r_2$ as
       feedback to supervise $f_1$, and re-rank $O$ using $f_1$
       and $\tau_2$; Update $r_1$;
5: Until I($O$).
```

The termination condition I(O) can be defined according to different application scenarios. For example, I(O) may require the top K ranked objects in r_1 and r_2 to be identical if one is particularly interested in the top ranked objects. It is also very likely that r_1 and r_2 do not change any more after several iterations (or the top K objects do not change). In this case, the two base rankers can not learn from each other any more, and the Co-Feedback ranking process should terminate either. The final ranking results can be easily determined by combining the two base rankers without any parameter, because they have already learnt from each other and can be equally treated.

2.2 Query-Focused Summarization based on Co-FRank

The task of query-focused summarization is to produce a short summary (250 words in length) for a set of related documents D with respect to the query q that reflects a user's information need. We follow the traditional extractive summarization framework in this study, where the two critical processes involved are sentence ranking and sentence selection, yet we focus more on the sentence ranking algorithm based on Co-FRank. As for sentence selection, we incrementally add into the summary the highest ranked sentence if it doesn't significantly repeat[1] the information already included in the summary until the word limitation is reached.

In the context of query-focused summarization, two kinds of features, i.e. query-dependent and query-independent features are necessary and they are supposed to complement each other. We then use these two kinds of features to develop the two base rankers. The query-dependent feature (i.e. the relevance of the sentence s to the query q) is defined as the cosine similarity between s and q.

$$f_1 \Leftrightarrow rel(s,q) = \cos(s,q) = \vec{s} \bullet \vec{q} / \left\|\vec{s}\right\| \cdot \left\|\vec{q}\right\| \qquad (1)$$

The words in s and q vectors are weighted by *tf*isf*. Meanwhile, the query-independent feature (i.e. the sentence significance based on word centroid) is defined as

$$f_2 \Leftrightarrow c(s) = \sum_{w \in s} c(w) / \sqrt{|s|} \qquad (2)$$

where $c(w)$ is the centroid weight of the word w in s and $c(w) = \sum_{s \in D} \left(tf_w^s \cdot isf_w \right) / N_S^D$. N_S^D is the total number of the sentences in D, tf_w^s is the frequency of w in s, and $isf_w = \log\left(N_S^D / sf_w \right)$ is the inverse sentence frequency (ISF) of w, where sf_w is the sentence frequency of w in D. The sentence ranking algorithm based on Co-FRank is detailed in the following Algorithm 1.

```
Algorithm 1. Co-FRank($f_1$, $f_2$, $D$, $q$)
1: Extract sentences $S=\{s_1, \dots s_m\}$ from $D$;
2: Rank $S$ with $f_1$ and obtain the ranking results $r_1$;
3: Rank $S$ with $f_2$ and obtain the ranking results $r_2$;
4: Normalize $r_1$, $r_1(s_i) = (r_1(s_i) - \min(r_1)) / (\max(r_1) - \min(r_1))$;
5: Normalize $r_2$, $r_2(s_i) = (r_2(s_i) - \min(r_2)) / (\max(r_2) - \min(r_2))$;
6: Repeat
```

[1] A sentence is discarded if the cosine similarity of it to any sentence already selected into the summary is greater than 0.9.

```
7: Select the top $N$ ranked sentences at round $n$ $\tau_1^n$
   from $r_1$ as feedback for $f_2$, and re-rank $S$ using $f_2$
   and $\tau_1^n$,
   $\pi_2(s_i) \leftarrow \sum_{k=1}^{n} sim(s_i, \tau_1^k)/n$，$\pi_2 = \frac{\pi_2 - \min(\pi_2)}{\max(\pi_2) - \min(\pi_2)}$
   $r_2(s_i) \leftarrow \eta \cdot f_2(s_i) + (1-\eta) \cdot \pi_2(s_i)$     (3)
8: Select the top $N$ ranked sentences at round $n$ $\tau_2^n$
   from $r_2$ as feedback for $f_1$, and re-rank $S$ using $f_1$
   and $\tau_2^n$;
   $\pi_1(s_i) \leftarrow \sum_{k=1}^{n} sim(s_i, \tau_2^k)/n$，$\pi_1 = \frac{\pi_1 - \min(\pi_1)}{\max(\pi_1) - \min(\pi_1)}$
   $r_1(s_i) \leftarrow \eta \cdot f_1(s_i) + (1-\eta) \cdot \pi_1(s_i)$     (4)
9: Until the top $K$ sentences in $r_1$ and $r_2$ are the same,
   both $r_1$ and $r_2$ do not change any more, or
   maximum iteration round is achieved.
10: Calculate the final ranking results,
   $r(s_i) = (r_1(s_i) + r_2(s_i))/2$ .     (5)
```

The update strategies used in Algorithm 1, as formulated in Formulas (3) and (4), are designed based on the intuition that the new ranking of the sentence s from one base ranker (say f_1) consists of two parts. The first part is the initial ranking produced by f_1. The second part is the similarity between s and the top N feedback provided by the other ranker (say f_2), and vice versa. The top K ranked sentences by f_2 are supposed to be highly supported by f_2. As a result, a sentence that is similar to those top ranked sentences should deserve a high rank as well. $sim(s_i, \tau_2^n)$ captures the effect of such feedback at round n and the definition of it may vary with regard to the application background. For example, it can be defined as the maximum, the minimum or the average similarity value between s_i and a set of feedback sentences in τ_2. Through this mutual interaction, the two base rankers supervise each other and are expected as a whole to produce more reliable ranking results.

We assume each base ranker is most confident with its first ranked sentence and set N to 1. Accordingly, $sim(s_i, \tau_2^n)$is defined as the similarity between s_i and the one sentence in τ_2^n. η is a balance factor which can be viewed as the proportion of the dependence of the new ranking results on its initial ranking results. K is set to 10 as 10 sentences are basically sufficient for the summarization task we work on. We carry out at most 5 iterations in the current implementation.

3 Experimental Study

We take the DUC 2005 data set as the evaluation corpus in this preliminary study. ROUGE (Lin and Hovy, 2003), which has been officially adopted in the DUC for years is used as the evaluation criterion. For the purpose of comparison, we implement the following two basic ranking functions and the linear combination of them for reference, i.e. the query relevance based ranker (denoted by QRR, same as f_1) and the word centroid based ranker (denoted by WCR, same as f_2), and the linear combined ranker, $LCR = \lambda\, QRR + (1-\lambda)\, WCR$, where λ is a combination parameter. QRR and WCR are normalized by $(x - \min)/(\max - \min)$, where x, max and min denote the original ranking score, the maximum ranking score and minimum ranking score produced by a ranker, respectively.

Table 1 shows the results of the average recall scores of ROUGE-1, ROUGE-2 and ROUGE-SU4 along with their 95% confidence intervals included within square brackets. Among them, ROUGE-2 is the primary DUC evaluation criterion.

	ROUGE-1	ROUGE-2	ROUGE-SU4
QRR	0.3597 [0.3540, 0.3654]	0.0664 [0.0630, 0.0697]	0.1229 [0.1196, 0.1261]
WCR	0.3504 [0.3436, 0.3565]	0.0644 [0.0614, 0.0675]	0.1171 [0.1138, 0.1202]
LCR*	0.3513 [0.3449, 0.3572]	0.0645 [0.0613, 0.0676]	0.1177 [0.1145, 0.1209]
Co-FRank+	0.3769 [0.3712, 0.3829]	0.0762 [0.0724, 0.0799]	0.1317 [0.1282, 0.1351]
LCR**	0.3753 [0.3692, 0.3813]	0.0757 [0.0719, 0.0796]	0.1302 [0.1265, 0.1340]
Co-FRank++	**0.3783** **[0.3719, 0.3852]**	**0.0775** **[0.0733, 0.0810]**	**0.1323** **[0.1293, 0.1360]**

* The worst results produced by LCR when $\lambda = 0.1$

+ The worst results produced by Co-FRank when $\eta = 0.6$

** The best results produced by LCR when $\lambda = 0.4$

++ The best results produced by Co-FRank when $\eta = 0.8$

Table 1 Compare different ranking strategies

Note that the improvement of LCR over QRR and WCR is rather significant if the combination parameter λ is selected appropriately. Besides, Co-FRank is always superior to LCR regardless of the best or the worst ouput, and the improvement is visible. The reason is that both QRR and WCR are enhanced step by step in Co-FRank, which in turn results in the increased overall performance. The trend of the improvement has been clearly observed in the experiments. This observation validates our motivation and the rationality of the algorithm proposed in this paper and motivates our further investigation on this topic.

We continue to examine the parameter settings in LCR and Co-FRank. Table 2 shows the results of LCR when the value of λ changes from 0.1 to

1.0, and Table 3 shows the results of Co-FRank with η ranging from 0.5 to 0.9. Notice that η is not a combination parameter. We believe that a base ranker should have at least half belief in its initial ranking results and thus the value of the η should be greater than 0.5. We find that LCR heavily depends on λ. LCR produces relatively good and stable results with λ varying from 0.4 to 0.6. However, the ROUGE scores drop apparently when λ heading towards its two end values, i.e. 0.1 and 1.0.

λ	ROUGE-1	ROUGE-2	ROUGE-SU4
0.1	0.3513 [0.3449, 0.3572]	0.0645 [0.0613, 0.0676]	0.1177 [0.1145, 0.1209]
0.2	0.3623 [0.3559, 0.3685]	0.0699 [0.0662, 0.0736]	0.1235 [0.1197, 0.1271]
0.3	0.3721 [0.3660, 0.3778]	0.0741 [0.0706, 0.0778]	0.1281 [0.1246, 0.1318]
0.4	0.3753 [0.3692, 0.3813]	0.0757 [0.0719, 0.0796]	0.1302 [0.1265, 0.1340]
0.5	0.3756 [0.3698, 0.3814]	0.0755 [0.0717, 0.0793]	0.1307 [0.1272, 0.1342]
0.6	0.3770 [0.3710, 0.3826]	0.0754 [0.0716, 0.0791]	0.1323 [0.1286, 0.1357]
0.7	0.3698 [0.3636, 0.3759]	0.0718 [0.0680, 0.0756]	0.1284 [0.1246, 0.1318]
0.8	0.3672 [0.3613, 0.3730]	0.0706 [0.0669, 0.0743]	0.1271 [0.1234, 0.1305]
0.9	0.3651 [0.3591, 0.3708]	0.0689 [0.0652, 0.0726]	0.1258 [0.1220, 0.1293]

Table 2 LCR with different λ values

As shown in Table 3, the Co-FRank can always produce stable and promising results regardless of the change of η. More important, even the worst result produced by Co-FRank still outperforms the best result produced by LCR.

η	ROUGE-1	ROUGE-2	ROUGE-SU4
0.5	0.3750 [0.3687, 0.3810]	0.0766 [0.0727, 0.0804]	0.1308 [0.1270, 0.1344]
0.6	0.3769 [0.3712, 0.3829]	0.0762 [0.0724, 0.0799]	0.1317 [0.1282, 0.1351]
0.7	0.3775 [0.3713, 0.3835]	0.0763 [0.0724, 0.0801]	0.1319 [0.1282, 0.1354]
0.8	**0.3783** **[0.3719, 0.3852]**	**0.0775** **[0.0733, 0.0810]**	**0.1323** **[0.1293, 0.1360]**
0.9	0.3779 [0.3722, 0.3835]	0.0765 [0.0728, 0.0803]	0.1319 [0.1285, 0.1354

Table 3 Co-FRank with different η values

We then compare our results to the DUC participating systems. We present the following representative ROUGE results of (1) the top three DUC participating systems according to ROUGE-2 scores (S15, S17 and S10); and (2) the NIST baseline which simply selects the first sentences from the documents.

	ROUGE-1	ROUGE-2	ROUGE-SU4
Co-FRank	**0.3783**	**0.0775**	**0.1323**
S15	-	0.0725	0.1316
S17	-	0.0717	0.1297
S10	-	0.0698	0.1253
Baseline		0.0403	0.0872

Table 4 Compare with DUC participating systems

It is clearly shown in Table 4 that Co-FRank can produce a very competitive result, which significantly outperforms the NIST baseline and meanwhile it is superior to the best participating system in the DUC 2005.

4 Conclusion and Future Work

In this paper, we propose a novel ranking framework, namely Co-Feedback Ranking (Co-FRank), and examine its effectiveness in query-focused summarization. There is still a lot of work to be done on this topic. Although we show the promising achievements of Co-Frank from the perspective of experimental studies, we expect a more theoretical analysis on Co-FRank. Meanwhile, we would like to investigate more appropriate techniques to use feedback, and we are interested in applying Co-FRank to the other applications, such as opinion summarization where the integration of opinion-biased and document-biased ranking is necessary.

Acknowledgments

The work described in this paper was supported by the Hong Kong Polytechnic University internal the grants (G-YG80 and G-YH53) and the China NSF grant (60703008).

References

Avrim Blum and Tom Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-Training. *In Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pp92-100.

Chin-Yew Lin and Eduard Hovy. 2000. The Automated Acquisition of Topic Signature for Text Summarization. *In Proceedings of COLING*, pp495-501.

Chin-Yew Lin and Eduard Hovy. 2003. Automatic Evaluation of Summaries Using N-gram Co-occurrence Statistics. *In Proceedings of HLT-NAACL*, pp71-78.

Dragomir R. Radev, Hongyan Jing, Malgorzata Stys, and Daniel Tam. 2004. Centroid-based Summarization of Multiple Documents. *Information Processing and Management*, 40:919-938.

Paul Over, Hoa Dang and Donna Harman. 2007. DUC in Context. *Information Processing and Management*, 43(6):1506-1520.