Using Tweets to Help Sentence Compression for News Highlights Generation

Zhongyu Wei[1], **Yang Liu**[1], **Chen Li**[1], **Wei Gao**[2]
[1]Computer Science Department, The University of Texas at Dallas
Richardson, Texas 75080, USA
[2]Qatar Computing Research Institute, Hamad Bin Khalifa University, Doha, Qatar
{zywei,yangl,chenli}@hlt.utdallas.edu[1]
wgao@qf.org.qa[2]

Abstract

We explore using relevant tweets of a given news article to help sentence compression for generating compressive news highlights. We extend an unsupervised dependency-tree based sentence compression approach by incorporating tweet information to weight the tree edge in terms of informativeness and syntactic importance. The experimental results on a public corpus that contains both news articles and relevant tweets show that our proposed tweets guided sentence compression method can improve the summarization performance significantly compared to the baseline generic sentence compression method.

1 Introduction

"Story highlights" of news articles are provided by only a few news websites such as CNN.com. The highlights typically consist of three or four succinct itemized sentences for readers to quickly capture the gist of the document, and can dramatically reduce reader's information load. A highlight sentence is usually much shorter than its original corresponding news sentence; therefore applying extractive summarization methods directly to sentences in a news article is not enough to generate high quality highlights.

Sentence compression aims to retain the most important information of an original sentence in a shorter form while being grammatical at the same time. Previous research has shown the effectiveness of sentence compression for automatic document summarization (Knight and Marcu, 2000; Lin, 2003; Galanis and Androutsopoulos, 2010; Chali and Hasan, 2012; Wang et al., 2013; Li et al., 2013; Qian and Liu, 2013; Li et al., 2014). The compressed summaries can be generated through a pipeline approach that combines a generic sentence compression model with a summary sentence pre-selection or post-selection step. Prior studies have mostly used the generic sentence compression approaches, however, a generic compression system may not be the best fit for the summarization purpose because it does not take into account the summarization task in the compression module. Li et al. (2013) thus proposed a summary guided compression method to address this problem and showed the effectiveness of their method. But this approach relied heavily on the training data, thus has the limitation of domain generalization.

Instead of using a manually generated corpus, we investigate using existing external sources to guide sentence compression for the purpose of compressive news highlights generation. Nowadays it becomes more and more common that users share interesting news content via Twitter together with their comments. The availability of cross-media information provides new opportunities for traditional tasks of Natural Language Processing (Zhao et al., 2011; Subašić and Berendt, 2011; Gao et al., 2012; Kothari et al., 2013; Štajner et al., 2013). In this paper, we propose to use relevant tweets of a news article to guide the sentence compression process in a pipeline framework for generating compressive news highlights. This is a pioneer study for using such parallel data to guide sentence compression for document summarization.

Our work shares some similar ideas with (Wei and Gao, 2014; Wei and Gao, 2015). They also attempted to use tweets to help news highlights generation. Wei and Gao (2014) derived external features based on the relevant tweet collection to assist the ranking of the original sentences for extractive summarization in a fashion of supervised machine learning. Wei and Gao (2015) proposed a graph-based approach to simultaneously rank the

Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers), pages 50–56,
Beijing, China, July 26-31, 2015.

original news sentences and relevant tweets in an unsupervised way. Both of them focused on using tweets to help sentence extraction while we leverage tweet information to guide sentence compression for compressive summary generation.

We extend an unsupervised dependency-tree based sentence compression approach to incorporate tweet information from the aspects of both informativeness and syntactic importance to weight the tree edge. We evaluate our method on a public corpus that contains both news articles and relevant tweets. The result shows that generic compression hurts the performance of highlights generation, while sentence compression guided by relevant tweets of the news article can improve the performance.

2 Framework

We adopt a pipeline approach for compressive news highlights generation. The framework integrates a sentence extraction component and a post-sentence compression component. Each is described below.

2.1 Tweets Involved Sentence Extraction

We use LexRank (Erkan and Radev, 2004) as the baseline to select the salient sentences in a news article. This baseline is an unsupervised extractive summarization approach and has been proved to be effective for the summarization task.

Besides LexRank, we also use Heterogeneous Graph Random Walk (HGRW) (Wei and Gao, 2015) to incorporate relevant tweet information to extract news sentences. In this model, an undirected similarity graph is created, similar to LexRank. However, the graph is heterogeneous, with two types of nodes for the news sentences and tweets respectively.

Suppose we have a sentence set S and a tweet set T. By considering the similarity between the same type of nodes and cross types, the score of a news sentence s is computed as follows:

$$p(s) = \frac{d}{N+M} + (1-d)\left[\epsilon \sum_{m\in T} \frac{sim(s,m)}{\sum_{v\in T} sim(s,v)} p(m)\right] + (1-d)\left[(1-\epsilon) \sum_{n\in S\setminus\{s\}} \frac{sim(s,n)}{\sum_{v\in S\setminus\{s\}} sim(s,v)} p(n)\right] \quad (1)$$

where N and M are the size of S and T, respectively, d is a damping factor, $sim(x,y)$ is the similarity function, and the parameter ϵ is used to control the contribution of relevant tweets. For a tweet node t, its score can be computed similarly. Both d and $sim(x,y)$ are computed following the setup of LexRank, where $sim(x,y)$ is computed as cosine similarity:

$$sim(x,y) = \frac{\sum_{w\in x,y} tf_{w,x} tf_{w,y} (idf_w)^2}{\sqrt{\sum_{w_i\in x} (tf_{w_i,x} idf_{w_i})^2} \times \sqrt{\sum_{w_i\in y} (tf_{w_i,y} idf_{w_i})^2}} \quad (2)$$

where $tf_{w,x}$ is the number of occurrences of word w in instance x, idf_w is the inverse document frequency of word w in the dataset. In our task, each sentence or tweet is treated as a document to compute the IDF value.

Although both types of nodes can be ranked in this framework, we only output the top news sentences as the highlights, and the input to the subsequent compression component.

2.2 Dependency Tree Based Sentence Compression

We use an unsupervised dependency tree based compression framework (Filippova and Strube, 2008) as our baseline. This method achieved a higher F-score (Riezler et al., 2003) than other systems on the Edinburgh corpus (Clarke and Lapata, 2006). We will introduce the baseline in this part and describe our extended model that leverages tweet information in the next subsection.

The sentence compression task can be defined as follows: given a sentence s, consisting of words $w_1, w_2, ..., w_m$, identify a subset of the words of s, such that it is grammatical and preserves essential information of s. In the baseline framework, a dependency graph for an original sentence is first generated and then the compression is done by deleting edges of the dependency graph. The goal is to find a subtree with the highest score:

$$f(X) = \sum_{e\in E} x_e \times w_{info}(e) \times w_{syn}(e) \quad (3)$$

where x_e is a binary variable, indicating whether a directed dependency edge *e* is kept (x_e is 1) or removed (x_e is 0), and *E* is the set of edges in the dependency graph. The weighting of edge e considers both its syntactic importance ($w_{syn}(e)$) as well as the informativeness ($w_{info}(e)$). Suppose edge *e* is pointed from head *h* to node *n* with dependency label *l*, both weights can be computed from a background news corpus as:

$$w_{info}(e) = \frac{P_{summary}(n)}{P_{article}(n)} \quad (4)$$

$$w_{syn}(e) = P(l|h) \quad (5)$$

where $P_{summary}(n)$ and $P_{article}(n)$ are the unigram probabilities of word n in the two language models trained on human generated summaries and the original articles respectively. $P(l|h)$ is the conditional probability of label l given head h. Note that here we use the formula in (Filippova and Altun, 2013) for $w_{info}(e)$, which was shown to be more effective for sentence compression than the original formula in (Filippova and Strube, 2008).

The optimization problem can be solved under the tree structure and length constraints by integer linear programming[1]. Given that L is the maximum number of words permitted for the compression, the length constraint is simply represented as:

$$\sum_{e \in E} x_e \leq L \quad (6)$$

The surface realizatdion is standard: the words in the compression subtree are put in the same order they are found in the source sentence. Due to space limit, we refer readers to (Filippova and Strube, 2008) for a detailed description of the baseline method.

2.3 Leverage Tweets for Edge Weighting

We then extend the dependency-tree based compression framework by incorporating tweet information for dependency edge weighting. We introduce two new factors, $w^T_{info}(e)$ and $w^T_{syn}(e)$, for informativeness and syntactic importance respectively, computed from relevant tweets of the news. These are combined with the weights obtained from the background news corpus defined in Section 2.2, as shown below:

$$w_{info}(e) = (1-\alpha) \cdot w^N_{info}(e) + \alpha \cdot w^T_{info}(e) \quad (7)$$

$$w_{syn}(e) = (1-\beta) \cdot w^N_{syn}(e) + \beta \cdot w^T_{syn}(e) \quad (8)$$

where α and β are used to balance the contribution of the two sources, and $w^N_{info}(e)$ and $w^N_{syn}(e)$ are based on Equation 4 and 5.

The new informative weight $w^T_{info}(e)$ is calculated as:

$$w^T_{info}(e) = \frac{P_{relevantT}(n)}{P_{backgroundT}(n)} \quad (9)$$

$P_{relevantT}(n)$ and $P_{backgroundT}(n)$ are the unigram probabilities of word n in two language models trained on the relevant tweet dataset and a background tweet dataset respectively.

The new syntactic importance score is:

$$w^T_{syn}(e) = \frac{NT(h,n)}{NT} \quad (10)$$

$NT(h,n)$ is the number of tweets where n and head h appear together within a window frame of K, and NT is the total number of tweets in the relevant tweet collection. Since tweets are always noisy and informal, traditional parsers are not reliable to extract dependency trees. Therefore, we use co-occurrence as pseudo syntactic information here. Note $w^N_{info}(e)$, $w^T_{info}(e)$, $w^N_{syn}(e)$ and $w^T_{syn}(e)$ are normalized before combination.

3 Experiment

3.1 Setup

We evaluate our pipeline news highlights generation framework on a public corpus based on CNN/USAToday news (Wei and Gao, 2014). This corpus was constructed via an event-oriented strategy following four steps: 1) 17 salient news events taking place in 2013 and 2014 were manually identified. 2) For each event, relevant tweets were retrieved via Topsy[2] search API using a set of manually generated core queries. 3) News articles explicitly linked by URLs embedded in the tweets were collected. 4) News articles from CNN/USAToday that have more than 100 explicitly linked tweets were kept. The resulting corpus contains 121 documents, 455 highlights and 78,419 linking tweets.

We used tweets explicitly linked to a news article to help extract salience sentences in ***HGRW*** and to generate the language model for computing $w^T_{info}(e)$. The co-occurrence information computed from the set of explicitly linked tweets is very sparse because the size of the tweet set is small. Therefore, we used all the tweets retrieved for the event related to the target news article to compute the co-occurrence information for $w^T_{syn}(e)$. Tweets retrieved for events were not published in (Wei and Gao, 2014). We make it available here[3]. The statistics of the dataset can be found in Table. 1.

[1] In our implementation we use GNU Linear Programming Kit (GULP) (`https://www.gnu.org/software/glpk/`)

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

[3] `http://www.hlt.utdallas.edu/~zywei/data/CNNUSATodayEvent.zip`

Event	Doc #	HLight #	Linked Tweet #	Retrieved Tweet #	Event	Doc #	HLight #	Linked Tweet #	Retrieved Tweet #
Aurora shooting	14	54	12,463	588,140	African runner murder	8	29	9,461	303,535
Boston bombing	38	147	21,683	1,650,650	Syria chemical weapons use	1	4	331	11,850
Connecticut shooting	13	47	3,021	213,864	US military in Syria	2	7	719	619,22
Edward Snowden	5	17	1,955	379,349	DPRK Nuclear Test	2	8	3,329	103,964
Egypt balloon crash	3	12	836	36,261	Asiana Airlines Flight 214	11	42	8,353	351,412
Hurricane Sandy	4	15	607	189,082	Moore Tornado	5	19	1,259	1,154,656
Russian meteor	3	11	6,841	239,281	Chinese Computer Attacks	2	8	507	28,988
US Flu Season	7	23	6,304	1,042,169	Williams Olefins Explosion	1	4	268	14,196
Super Bowl blackout	2	8	482	214,775	Total	121	455	78,419	6,890,987

Table 1: Distribution of documents, highlights and tweets with respect to different events

Method	ROUGE-1			Compr.
	F(%)	P(%)	R(%)	Rate(%)
LexRank	26.1	19.9	**39.1**	100
LexRank + SC	25.2	22.4	29.6	63.0
LexRank + SC+w^T_{info}	25.7	22.8	30.1	62.0
LexRank + SC+w^T_{syn}	26.2	23.5	30.4	63.7
LexRank + SC+$both$	**27.5**	**25.0**	31.4	61.5
HGRW	28.1	22.6	**39.5**	100
HGRW + SC	26.4	24.9	29.5	66.1
HGRW + SC+w^T_{info}	27.5	25.7	30.8	65.4
HGRW + SC+w^T_{syn}	27.0	25.3	30.2	66.7
HGRW + SC+$both$	**28.4**	**26.9**	31.2	64.8

Table 2: Overall Performance. **Bold**: the best value in each group in terms of different metrics.

Following (Wei and Gao, 2014), we output 4 sentences for each news article as the highlights and report the ROUGE-1 scores (Lin, 2004) using human-generated highlights as the reference.

The sentence compression rates are set to 0.8 for short sentences containing fewer than 9 words, and 0.5 for long sentences with more than 9 words, following (Filippova and Strube, 2008). We empirically use 0.8 for α, β and ϵ such that tweets have more impact for both sentence selection and compression. We leveraged The New York Times Annotated Corpus (LDC Catalog No: LDC2008T19) as the background news corpus. It has both the original news articles and human generated summaries. The Stanford Parser[4] is used to obtain dependency trees. The background tweet corpus is collected from Twitter public timeline via Twitter API, and contains more than 50 million tweets.

3.2 Results

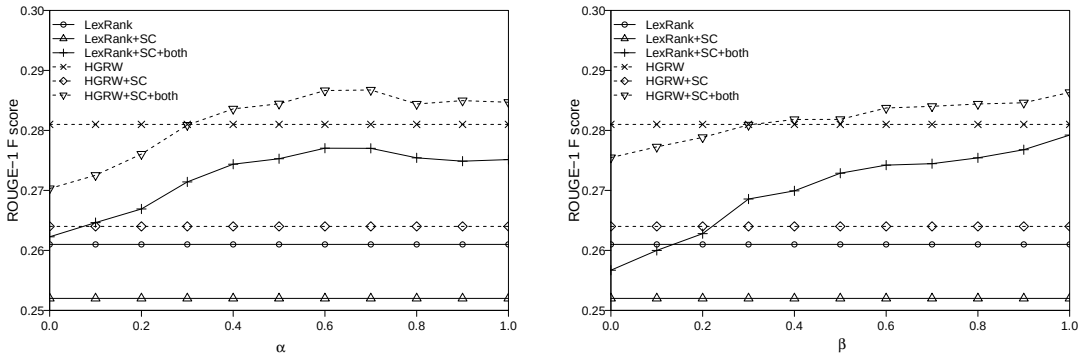
Table 2 shows the overall performance[5]. For summaries generated by both *LexRank* and *HGRW*, "+*SC*" means generic sentence compression baseline (Section. 2.2) is used, "+w^T_{info}" and "+w^T_{syn}" indicate tweets are used to help edge weighting for sentence compression in terms of informativeness and syntactic importance respectively, and "+*both*" means both factors are used. We have several findings.

- The tweets involved sentence extraction model *HGRW* can improve *LexRank* by 8.8% relatively in terms of ROUGE-1 F score, showing the effectiveness of relevant tweets for sentence selection.
- With generic sentence compression, the ROUGE-1 F scores for both *LexRank* and *HGRW* drop, mainly because of a much lower recall score. This indicates that generic sentence compression without certain guidance removes salient content of the original sentence that may be important for summarization and thus hurts the performance. This is consistent with the finding of (Chali and Hasan, 2012).
- By adding either w^T_{info} or w^T_{syn}, the performance of summarization increases, showing that relevant tweets can be used to help the scores of both informativeness and syntactic importance.
- *+SC+both* improves the summarization performance significantly[6] compared to the corresponding compressive summarization baseline *+SC*, and outperforms the corresponding original baseline, *LexRank* and *HGRW*.
- The improvement obtained by *LexRank+SC+both* compared to *LexRank* is more promising than that obtained by *HGRW+SC+both* compared to *HGRW*. This may be because *HGRW* has used tweet information already, and leaves limited room for improvement for the sentence compression model when using the same source of information.

[4] http://nlp.stanford.edu/software/lex-parser.shtml

[5] The performance of HGRW reported here is different from (Wei and Gao, 2015) because the setup is different. We use all the explicitly linked tweets in the ranking process here without considering redundancy while a redundancy filtering process was applied in (Wei and Gao, 2015) .

[6] Significance throughout the paper is computed by two tailed t-test and reported when $p < 0.05$.



(a) Impact of α (b) Impact of β

Figure 1: The influence of α and β. Solid lines are used for approaches based on LexRank; Dotted lines are used for HGRW based approaches.

Method	Example 1	Example 2
LexRank	Boston bombing suspect Tamerlan Tsarnaev, killed in a shootout with police days after the blast, has been buried at an undisclosed location, police in Worcester, Mass., said.	Three people were hospitalized in critical condition, according to information provided by hospitals who reported receiving patients from the blast.
LexRank+SC	suspect Tamerlan Tsarnaev, killed in a shootout after the blast, has been buried at an location, police in Worcester Mass. said.	Three people were hospitalized, according to information provided by hospitals who reported receiving from the blast.
LexRank+SC+both	**Boston bombing** suspect Tamerlan Tsarnaev, killed in a shootout after the blast, has been buried at an location police said.	Three people were hospitalized in **critical condition**, according to information provided by hospitals.
Ground Truth	**Boston bombing** suspect Tamerlan Tsarnaev has been buried at an undisclosed location	Hospitals report three people in **critical condition**

Table 3: Example highlight sentences from different systems

- By incorporating tweet information for both sentence selection and compression, the performance of *HGRW+SC+both* outperforms *LexRank* significantly.

Table 3 shows some examples. As we can see in Example 1, with the help of tweet information, our compression model keeps the valuable part "Boston bombing" for summarization while the generic one abandons it.

We also investigate the influence of α and β. To study the impact of α, we fix β to 0.8, and vice versa. As shown in Figure 1, it is clear that larger α or β, i.e., giving higher weights to tweets related information, is generally helpful.

4 Conclusion and Future Work

In this paper, we showed that the relevant tweet collection of a news article can guide the process of sentence compression to generate better story highlights. We extended a dependency-tree based sentence compression model to incorporate tweet information. The experiment results on a public corpus that contains both news articles and relevant tweets showed the effectiveness of our approach. With the popularity of Twitter and increasing interaction between social media and news media, such parallel data containing news and related tweets is easily available, making our approach feasible to be used in a real system.

There are some interesting future directions. For example, we can explore more effective ways to incorporate tweets for sentence compression; we can study joint models to combine both sentence extraction and compression with the help of relevant tweets; it will also be interesting to use the parallel dataset of the news articles and the tweets for timeline generation for a specific event.

Acknowledgments

We thank the anonymous reviewers for their detailed and insightful comments on earlier drafts of this paper. The work is partially supported by NSF award IIS-0845484 and DARPA Contract No. FA8750-13-2-0041. Any opinions, findings, and conclusions or recommendations expressed are those of the authors and do not necessarily reflect the views of the funding agencies.

References

YLlias Chali and Sadid A Hasan. 2012. On the effectiveness of using sentence compression models for query-focused multi-document summarization. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 457–474.

James Clarke and Mirella Lapata. 2006. Models for sentence compression: A comparison across domains, training requirements and evaluation measures. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 377–384. 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*, 22:457–479.

Katja Filippova and Yasemin Altun. 2013. Overcoming the lack of parallel data in sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1481–1491. Association for Computational Linguistics.

Katja Filippova and Michael Strube. 2008. Dependency tree based sentence compression. In *Proceedings of the Fifth International Natural Language Generation Conference*, pages 25–32. Association for Computational Linguistics.

Dimitrios Galanis and Ion Androutsopoulos. 2010. An extractive supervised two-stage method for sentence compression. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 885–893. Association for Computational Linguistics.

Wei Gao, Peng Li, and Kareem Darwish. 2012. Joint topic modeling for event summarization across news and social media streams. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 1173–1182.

Kevin Knight and Daniel Marcu. 2000. Statistics-based summarization-step one: Sentence compression. In *Proceedings of The 7th National Conference on Artificial Intelligence*, pages 703–710.

Alok Kothari, Walid Magdy, Ahmed Mourad Kareem Darwish, and Ahmed Taei. 2013. Detecting comments on news articles in microblogs. In *Proceedings of The 7th International AAAI Conference on Weblogs and Social Media*, pages 293–302.

Chen Li, Fei Liu, Fuliang Weng, and Yang Liu. 2013. Document summarization via guided sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 490–500. Association for Computational Linguistics.

Chen Li, Yang Liu, Fei Liu, Lin Zhao, and Fuliang Weng. 2014. Improving multi-documents summarization by sentence compression based on expanded constituent parse trees. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 691–701. Association for Computational Linguistics.

Chin-Yew Lin. 2003. Improving summarization performance by sentence compression: a pilot study. In *Proceedings of the sixth international workshop on Information retrieval with Asian languages-Volume 11*, pages 1–8. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81.

Xian Qian and Yang Liu. 2013. Fast joint compression and summarization via graph cuts. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1492–1502. Association for Computational Linguistics.

Stefan Riezler, Tracy H King, Richard Crouch, and Annie Zaenen. 2003. Statistical sentence condensation using ambiguity packing and stochastic disambiguation methods for lexical-functional grammar. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 118–125. Association for Computational Linguistics.

Tadej Štajner, Bart Thomee, Ana-Maria Popescu, Marco Pennacchiotti, and Alejandro Jaimes. 2013. Automatic selection of social media responses to news. In *Proceedings of the 19th ACM International Conference on Knowledge Discovery and Data Mining*, pages 50–58. ACM.

Ilija Subašić and Bettina Berendt. 2011. Peddling or creating? investigating the role of twitter in news reporting. In *Advances in Information Retrieval*, pages 207–213. Springer.

Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. 2013. A sentence compression based framework to query-focused multi-document summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1384–1394. Association for Computational Linguistics.

Zhongyu Wei and Wei Gao. 2014. Utilizing microblog for automatic news highlights extraction. In *Proceedings of the 25th International Conference on Computational Linguistics*, pages 872–883.

Zhongyu Wei and Wei Gao. 2015. Gibberish, assistant, or master? using tweets linking to news for extractive single-document summarization. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Wayne Xin Zhao, Jing Jiang, Jianshu Weng, Jing He, Ee-Peng Lim, Hongfei Yan, and Xiaoming Li. 2011. Comparing twitter and traditional media using topic models. In *Advances in Information Retrieval*, pages 338–349. Springer.