Headliner: An integrated headline suggestion system

Shuguang Wang
The Washington Post
1301 K Street NW
Washington, D.C., U.S.A.
shuguang.wang@washpost.com

Eui-Hong (Sam) Han
The Washington Post
1301 K Street NW
Washington, D.C., U.S.A.
sam.han@washpost.com

Alexander M. Rush
Harvard SEAS
33 Oxford Street
Cambridge, M.A., U.S.A.
srush@seas.harvard.edu

ABSTRACT

Headline generation is a short-form variant of document summarization that has been studied in natural language processing. This paper presents a case study examining the application of several different headline generation models at The Washington Post. Currently for individual news articles, multiple different headlines are manually written in order to target different platforms such as the web site and social media networks. To reduce the amount of effort from the newsroom, our strategy is to automatically propose variable-length headlines for news stories during the process of article posting. After exploring several methods, we present promising initial results. We also describe the challenges encountered in this process and future directions needed for full application of the method.

Keywords

headline generation; deep learning; attention based model; hedge trimmer application; multiple sentence compression application

1. INTRODUCTION

Headlines of news articles provide a high level summary of the article, and are used to attract attention of readers. The importance of good headlines has increased greatly in the digital news age. The news audience has very little friction in switching from one news source to another on the web. People decide to read articles by evaluating headlines first. Hence, good headlines are essential for the audience development of digital news media.

The creation of a headline is a non-trivial process even for trained journalists and copy editors in the newsroom. In recent years, headline creation has become even more challenging for journalists and editors. Dramatic growth of social media platforms allows readers to access news from many different channels. Users' expectations and behaviors on each of these platforms are different and this requires newsroom to create multiple platform-specific headlines for each article. This requirement becomes only stronger with the growth of mobile phones and space-limited domains.

Furthermore, even on a single platform, it is beneficial to have multiple headline candidates. News stories usually cover various aspects of events, which makes it difficult for a single headline to be comprehensive and to be best in attracting readers' attention. With a digital audience from all over the world, the newsroom needs to cater to diverse reader backgrounds. For example, one headline with US slang or humor might attract the attention of US audience, but it might not serve international audience well. With multiple headlines for a news article, newsrooms can utilize headline variation testing tools to decide which headline performs well overall or for a specific user segment such as US or international. For example, The Washington Post utilizes a tool called Bandito [4] for the such headline variation testing. The tool helped The Washington Post newsroom to identify and serve best headlines to the audience. However, the availability and success of such testing tools have put more pressure on the newsroom to manually create multiple headlines, and spurred interests in automatic headline generation algorithms and tools.

Quite a few automatic headline generation algorithms have been proposed during last decade with differing assumptions and models. Recently there has also been significant progress on this problem with the use of deep learning methods for headline generation [15, 19]. In this work, we describe the Headliner system which incorporates several such algorithms including rule-based and learning-based systems. This system can be linked to the content management system (CMS) or content testing tools like Bandito. The proposed system is horizontally scalable and can easily incorporate additional algorithms to expand the pool of headline candidates.

To the best of our knowledge, this is one of the first efforts to use a deep learning approach in an automated journalism task. We conduct both quantitative evaluation and manual judgements on the quality of several headline generation approaches. Initial results demonstrate great potential of this system to support the newsroom. Our experiments show that the deep learning, neural machine translation model is very effective for this headline generation task and can be used to produce more accurate outputs.

The work is organized as follows. In Section 2, we discuss a few recent related work on automated journalism and document summarization. Then we present our system in Section 3. Section 4 presents quantitative and manual qualitative evaluations of our system. Section 5 summarizes our future work and finally Section 6 has the conclusion.

2. RELATED WORK

In recent years, significant effort has been put into improving the news generation process. Our headline generation system is closely related to two strands of work, automatic news content generation and document summarization.

Automated Support for Journalism.

Many tools have been proposed to generate automated news content in various forms [13]. Some successful tools includes fact checking [20], graphic creation [11], and even article generation [2]. To automatically generate full news articles, news organizations have started with simpler and more structured news contents, such as earning report news articles [22] and sports stories [3]. These technologies are still in early stage and rely heavily on structured data and predefined templates.

Automated journalism allows news organizations to create news fast with a large scale. However there are also many concerns about automated journalism technologies as they can not be held accountable for errors in news articles. At the same time, when these technologies are used to facilitate personalized new content, it still remains challenging for news organizations to ensure ethical integrity of news stories and legal right of manipulated source information [21].

Document Summarization.

Headline generation is a sub-task of document summarization. The problem was first tackled using rule-based systems such as Hedge Trimmer [12]. This algorithm works by iteratively pruning subtrees from the parsed sentence based on human-curated rules. Subsequent following studies started to explore other alternatives. Multi-sentence compression (MSC) [14] constructs a word graph from several related sentences in a document and compresses them into a single sentence summary. The resulting headline from MSC is the shortest path in the word graph. HEADS [10] models headline generation problem as a sequential prediction task. The system described in [15] uses a deep learning model to compress a sentence. These *extractive* approaches assume headlines can be created directly from the text of a news article body.

Recently several studies have approached headline generation as an abstractive problem, i.e. generating a headline without being constrained to reuse words from original article. Recent abstractive approaches such as [6, 9, 23] are inspired by frameworks used for machine translation tasks [7, 8]. HEADY [6] utilizes syntactic co-occurrence patterns to train a Nosiy-OR Bayesian network. A state-of-art attention model [9] adopts a word-based neural machine translation model to predict a headline utilizing text from an article. For Headliner, we have adopted Hedge Trimmer, MSC, and this attention model (NMT) in our headline generation system so far and we will describe more details about them in Section 3

3. HEADLINER SYSTEM

At The Washington Post, we are building a general-purpose system [1] designed to facilitate a more efficient news publishing process. An important component of this system is better support for the newsroom team including a suite of automation tools. Existing automated journalism technologies are still in early stage, thus the purpose of our tools is to collaborate with and assist journalists and editors at The Washington Post.

Headliner is a module in this system designed to automatically suggest news article headlines. The headline generation uses multiple independent algorithms to generate headlines for a given article. The generated headlines are then scored and filtered to select promising candidates. As with most automated algorithms, all headline generation algorithms so far incur errors, so we need to ensure that candidate headline pass a certain quality threshold before we present them to journalists and editors. Figure 1 demonstrates the architecture of the system.

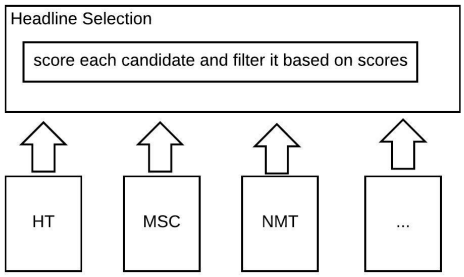

Figure 1: *System architecture of Headliner.*

A important advantage of this two-step setup is extendability. It makes it easy to plug-in additional headline generation algorithms as they are developed. At the same time, since each algorithm is treated independently, we need to configure and tune each algorithm separately with little sharing of the effort. Fortunately, most of the headline generation algorithms make different assumptions and adopt different underline models, and we are not re-engineering the same thing repeatedly.

The system under development is consists of three algorithms, i.e., Hedge Trimmer (HT) [12], multi-sentence compression (MSC) [14], and an attention-based neural machine translation system (NMT) [9].

3.1 Hedge Trimmer

Hedge Trimmer (HT) is a rule-based sentence summarization algorithm that is applied to the syntactic parse tree of a source sentence. It works by iteratively removing sub-trees based on a set of human-curated rules. The Hedge Trimmer approach does not require any training or underlying model to work, which makes it very fast and straightforward to implement. However it requires the development of problem specific rules. We expect performance improvements by enriching the rules manually.

We implemented a basic version of the Hedge Trimmer algorithm. In this implementation, the first line of the article is used to produce a sentence summary as the article headline. In preliminary experiments, we found that it was easy to overtrim or undertrim a sentence. Therefore we needed to set the length limit to ensure that irrelevant phrases are excluded in the resulting headlines. As the same time, backoff rules were needed to allow overtrimmed headline to maintain necessary phrases.

3.2 Multi-Sentence Compression

Multi-Sentence Compression (MSC) [14] produces a compression by utilizing multiple related sentences and extracting a single shortened headline. This algorithm first induces a word graph from the related sentences in a summarization set. Since important contexts tend to be repeated multiple times in these sentences, there are more paths to these important contexts in the word graph. After applying proper weights on paths in the word graph, the shortest path in

the word graph will represent a good compressed summary of an news article. Like HT, the MSC approach does not require any training procedure, and in fact does not require syntactic parsing at all. The main challenge is to identify the related sentences within a document.

Unlike HT, which just uses the first sentence in the article, MSC requires a set of related sentences with overlapping important words or phrases as input. We use a centroid-based algorithm to identify related sentences in an article. Formally let $\mathcal{V}$ be the vocabulary, we run the following procedure.

1. Represent each sentence i as a feature vector $\mathbf{s}_i \in \mathbb{R}^{|\mathcal{V}|}$, using either word indicator features or TF-IDF [18].
2. Find the centroid feature vector by averaging the all feature vectors of sentences in an article, $\mathbf{c} = \sum_i \mathbf{s}_i$.
3. Find top k most similar sentences, $(i_0, ..., i_{k-1})$, to this centroid feature vector using cosine similarity,

$$(i_0, i_1, ..., i_{k-1}) = \arg \max_{0..k-1} \frac{\mathbf{s}_i \cdot \mathbf{c}}{||\mathbf{s}_i|| ||\mathbf{c}||}$$

.

In practice we need to tune k to find a best set for MSC to perform reasonably well.

Instead of relying on only the first sentence, MSC is able to utilize all the sentences in the article. This is potentially a benefit, since important information may not be contained in the first sentence. However, the same context can often be phrased in various ways. MSC algorithm will often miss important paraphrased context without appropriate coreference resolution.

3.3 Neural Machine Translation

Our final approach is to utilize a state-of-the-art neural machine translation system to generate headlines. This approach is based on the attention-based neural machine translation baseline used for summarization in [9], known as attention-based neural machine translation (NMT) [7]. This approach uses recently popularized deep-learning technique known as recurrent neural network (RNN) in a sequence-to-sequence framework to learn a transformation from an input sentence to a summarized headline. To implement this approach we used the Harvard seq2seq-attn implementation of NMT from [5] built using the Torch (GPU) framework.

Unlike the previous two systems this system works in a supervised setting, and therefore requires significant training time and a large set of training data consisting of pairs of an article sentence and headline. On the positive side, this setup allows the model to learn abstractive summaries that may not contain words from the input sentence. Additionally the model can be trained to learn different models for specific domains, such as social media headlines or very short mobile headlines. In theory the model can be trained with the headline and any sentence in the article. For simplicity though we used the first sentence of the article at both train and test time.

Additionally following [19], we filtered the training set to find sentence pairs that are similar enough for training purpose. As a proxy, we checked that the pairs contained some overlapping words to eliminate bad matches. We implement a simple scoring algorithm to compute the similarity:

1. Tokenize both source and target sentences.
2. Compute TF-IDF scores of each word in both sentences.
3. Sum up the scores of words that appear in both source and target sentences, and threshold.

Once we have a trained model, headline generation can be done by feeding unseen articles into the model and running a beam search decoding process. The decoding process is relatively efficient and can be tuned to produce headlines of different lengths or even to enforce a percentage of word overlap with the original source.

4. EVALUATION

4.1 Experimental Setup

The ideal evaluation of the Headliner system is in an end-to-end setting interacting with journalists and editors. As a proxy for this setup we instead evaluate the headlines generated by each of the the systems using automatic method and with human judgments done by authors. This setup allows us to see how well each algorithm is performing on real news articles published by The Washington Post, and act as a surrogate for the real feedback from journalists and editors.

All experiments are conducted on articles published in The Washington Post over the last few decades. The full data set contains several million published articles. For automatic evaluation, we arbitrarily picked $5,000$ of these articles, and for manual evaluation we selected 100 of these at random. Automatic evaluation is done using BLEU [17] score. BLEU was originally proposed as a machine translation metric and it measures how many words overlap in a given translation when compared to a reference[1].

As the news data set is noisy, we implemented a simple preprocessing step to clean the input sentences. We run the following procedure: (a) lower case all words, (b) remove bylines, (c) remove parenthetical phrases, (d) remove punctuation (e) replace digits with #.

The HT and MSC algorithms do not require training, and so they are run directly on these data sets. HT requires the parse structure of the article, and we use Stanford NLP parser [16] to generate the parse tree. MSC requires multiple related sentences to produce the headline candidates, we use top k=5 sentences using the centroid algorithm.

The NMT model requires a more extensive training process. We cleaned and preprocessed all available articles as described in Section 3.3, then we chose 500k most similar pairs of source sentences from article bodies and respective target headlines as the training data.

4.2 Quantitative Experiments

Our main quantitative experiment is to apply all three systems to the $5,000$ article automatic evaluation set and compare using BLEU. Table 1 summarizes the main results. Generally this problem is quite difficult as real-world headlines are quite different than the article content, therefore the absolute BLEU scores are relatively low. However, we

[1]ROUGE is more commonly used for summarization evaluation, but past work has shown that it is quite sensitive to length issues. We found that BLEU was a reasonable proxy for human eval on this data set.

Methods	BLEU	Manual
Hedge Trimmer (HT)	8.11	31%
Multi-Sentence Compression (MSC)	8.82	35%
Neural Machine Translation (NMT)	**9.62**	**46%**

Table 1: *Comparison of different approaches on the automatic evaluation (5000 sent), and the manual evaluation (100 sent) data sets.*

did find that the neural machine translation system was relatively effective for this task, and statistically better than the other two approaches (even as it produces abstractive summarizations).

The second experiment uses manual judgment to evaluate whether a headline is acceptable. For each article in the smaller evaluation set, we manually reviewed the headlines generated by the three algorithms and assigned a binary score according to the acceptability. Fluent sentences alone are not considered good headlines. To be acceptable headlines must be fluent and capture the same amount of context as the original headlines.

Table 1 shows the percentages of acceptable headlines generated from three algorithms on manual set. The results here roughly correlate with the BLEU results. Again the NMT model is the best and able to generate acceptable headlines for 46% testing articles. This is very impressive results, and some generated headlines do not contain same information as the original headline but they can still be valid candidates given the content of the article. While the absolute value of 46% seems low, we note that for many of the articles the first sentence does not contain enough information to really generate a correct headline. In Section 6 we discuss future work for selecting multiple sentences to be used within the NMT system.

While the neural machine translation system outperforms the other two algorithms on the eval set in term of BLEU in both experiments, it does require a more extensive and time-consuming training step. Considering the ultimate goal of Headliner is to provide multiple good headline candidates for a same news articles, it is still beneficial to have a diverse multi-system approach.

4.3 Qualitative Examples

To give a sense of the headline generation results for the three algorithms, we review some interesting differences in output. Table 2 shows headline candidates for three different articles from each of the systems.

The first is a common sports article. Here both NMT and HT are able to capture all necessary context from the first line. NMT is also able to apply correct paraphrasing, for instance replacing "will not" with "won't" and shortening the team name to "Redskins"'. MSC's candidate has good headline content, but contains a grammatical error. Note that all the systems miss "Atogwe" another player who is not mentioned in the lede.

The second example is an article describing a vehicle accident. Based on the true headline context, Hedge Trimmer generates the best headline. However, from context in the original source sentence, both NMT and MSC are capturing another aspect of the event. This shows one advantage of a multi-system approach for suggesting headlines to editors.

Src (1)	As expected John Beck will not play in the Washington Redskins' preseason opener against the Pittsburgh Steelers
Truth	Beck, Atogwe out for preseason opener
HT	John Beck will not play in the Washington Redskins' preseason opener
MSC	Beck not play in the Washington Redskins' preseason opener
NMT	John Beck won't play in Redskins' preseason opener
Src (2)	A two vehicle accident injured four people Wednesday night and forced authorities to shut down Suitland Parkway DC police said
Truth	Four injured in Suitland Parkway crash
HT	A two vehicle accident injured four people
MSC	Authorities shut down Suitland Parkway
NMT	Accident shuts down Suitland Parkway
Src (3)	Twenty three members and associates of what prosecutors have described as a drug gang known as the P Street crew were indicted yesterday on charges of selling hundreds of kilograms of cocaine and guarding their turf through killings rape and bombings
Truth	Alleged drug gang indicted
HT	Twenty three members were indicted yesterday
MSC	P Street indicted yesterday on cocaine
NMT	P Street crew indicted

Table 2: *Example summaries produced by the three systems. Src and Truth are the first line of the article and the true headline. NMT, HT, MSC are the three headline generation algorithms.*

The third example is a longer lede for a news article. None of the algorithms is able to produce a good enough headline compared to the true article. The HT headline is much too specific compared to the true headline. Both NMT and MSC are able to pick out the name of the gang, and the NMT headline is compact. None of the approaches can abstract to source to generate the general description "Alleged gang".

From these examples, we can see MSC algorithms tends to miss context pieces as it relies on having good centroid sentences. When Hedge Trimmer works, it tends to be on sentences that are close to the headline and it can focus on the most important aspect of the event. There is a good potential for the NMT model. It is able to go beyond the words or phrases in original source sentences, and it can potentially surface alternative aspect of news event. However it has the potential to be very wrong in a way that extractive systems avoid. Moreover, these algorithms performs differently enough that with a proper scoring, we will likely be able to come out with one or more good headline when they are used together.

5. FUTURE WORK

These experiments demonstrate three different approaches for headline generation and how they can be applied in a real newsroom environment. For this work to be effective, we will need several future extensions to the system.

Improved Sentence Selection.

When doing qualitative analysis, one of the major issues we found was that in real articles the first line of the article

did not provide the necessary information to generate a reasonable headline. When this occurs both HT and NMT will produce an incorrect result. MSC uses a different centroid-based approach which can get around this issue, however it also can make mistakes in the process. In future work we will train a *supervised* model to identify the more salient sentences in an article and use these to generate the headline with each of these approaches. Additionally we will extend the NMT system to be able to use multiple sentences as input, so that it can generate broader headlines.

Scoring of Headline Candidates.

The goal of Headliner system is to help the newsroom team and save their time and effort. Therefore we need to ensure all headline candidates from various algorithms are at least fluent sentences. We propose to use a simple scoring system with a n-gram language model trained from historical headlines of articles published by The Washington Post. The NMT system generally produces fluent sentences, but the other two could be improved in this regard, Only headline candidates with sufficient scores will be presented to journalists and editors. We are currently calibrating this system for production use.

Platform Specific Generation.

While generating a single headline is useful for editors, the real benefit of automation comes from being able to generate multiple headlines that can be targeted to the language of different platforms. This has the potential of saving significant time, and allowing for experimentation of different styles for different target audiences. The NMT approach allows us to train a single headline model, and then fine tune it based on supervised data for different platforms. With this approach the integrated system can be used to suggest headlines in different styles. We are currently in the process of building these data sets and experimenting this approach.

Efficient Model Update.

Every day The Washington Post publishes around a thousand articles. Therefore it is important to regularly re-train the model with the latest data. Even using fast machines training the model can take days to finish. Alternative approaches do exist though. For instance we can avoid re-training the full model by updating the word embeddings used by the model to incorporate new vocabulary. This is a fast process that can capture benefit of new terms without requiring full training.

6. CONCLUSION

We have presented Headliner, a headline generation system for newsroom at The Washington Post. Three headline generation algorithms implemented in Headliner show reasonable performance in the evaluation. NMT is the most effective among these algorithms, and demonstrates the potential for a deep learning approach to be used for automated journalism. It also has the potential to be trained to provide different headlines for different news distribution channels. We are currently exploring options to improve the quality of the three algorithms in the system, introduce other headline generation techniques in the system, and integrate the system with existing content management systems and content variation testing tools.

7. REFERENCES

[1] Arc publishing. https://www.arcpublishing.com/.

[2] Case study: Forbes. http://resources.narrativescience.com/h/i/83535927-case-study-forbes.

[3] Ncaa to grow college sports coverage with automated game stories. http://www.ap.org/Content/Press-Release/2015/AP-NCAA-to-grow-college-sports-coverage-with-automated-game-stories.

[4] Real time content testing framework. https://developer.washingtonpost.com/pb/blog/post/2016/02/08/bandito-a-multi-armed-bandit-tool-for-content-testing/.

[5] Sequence-to-sequence learning with attentional neural networks. https://github.com/harvardnlp/seq2seq-attn.

[6] E. Alfonseca, D. Pighin, and G. Garrido. Heady: News headline abstraction through event pattern clustering. In *ACL*, 2013.

[7] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, 2014.

[8] M. Banko, V. O. Mittal, and M. J. Witbrock. Headline generation based on statistical translation. *ACL*, 2000.

[9] S. Chopra, M. Auli, and A. M. Rush. Abstractive sentence summarization with attentive recurrent neural networks. *NAACL*, 2016.

[10] C. A. Colmenares, M. Litvak, A. Mantrach, and F. Silvestri. Heads: Headline generation as sequence prediction using an abstract feature-rich space. In *NAACL*, pages 133–142, 2015.

[11] J. D. N. Hullman and E. Adar. Contextifier: Automated generation of annotated stock visualizations. *Computational Journalism*, 2013.

[12] B. Dorr, D. Zajic, and R. Schwartz. Hedge trimmer: A parse-and-trim approach to headline generation. *NAACL*, 2003.

[13] K. N. Dörr. Mapping the field of algorithmic journalism. *Digital Journalism*, 2015.

[14] K. Filippova. Multi-sentence compression: Finding shortest paths in word graphs. *Coling*, 2010.

[15] K. Filippova, E. Alfonseca, C. A. Colmenares, L. Kaiser, and O. Vinyals. Sentence compression by deletion with lstms. *EMNLP*, 2015.

[16] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. In *ALC*, 2002.

[17] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ALC*, 2002.

[18] A. Rajaraman and J. D. Ullman. Mining of massive datasets. *Data Mining*, 2011.

[19] A. M. Rush, S. Chopra, and J. Weston. A neural attention model for abstractive sentence summarization. In *EMNLP*, 2015.

[20] B. Walenz, Y. W. Wu, S. A. Song, E. Sonmez, E. Wu, K. Wu, P. K. Agarwal, J. Yang, N. Hassan, A. Sultana, G. Zhang, C. Li, and C. Yu. Finding, monitoring, and checking claims computationally based on structured data. *Computational Journalism*, 2014.

[21] L. Weeks. Media law and copyright implications of automated journalism. *Journal of Intellectual Property and Entertainment Law*, 4(1):67–94, 2014.

[22] E. M. White. Automated earnings stories multiply. https://blog.ap.org/announcements/automated-earnings-stories-multiply.

[23] K. Woodsend, Y. Feng, and M. Lapata. Generation with quasi-synchronous grammar. *EMNLP*, 2010.