Finding Spoiler Bias in Tweets by Zero-shot Learning and Knowledge Distilling from Neural Text Simplification

Avi Bleiweiss
BShalem Research
Sunnyvale, CA, USA
avibleiweiss@bshalem.onmicrosoft.com

Abstract

Automatic detection of critical plot information in reviews of media items poses unique challenges to both social computing and computational linguistics. In this paper we propose to cast the problem of discovering spoiler bias in online discourse as a text simplification task. We conjecture that for an item-user pair, the simpler the user review we learn from an item summary the higher its likelihood to present a spoiler. Our neural model incorporates the advanced transformer network to rank the severity of a spoiler in user tweets. We constructed a sustainable high-quality movie dataset scraped from unsolicited review tweets and paired with a title summary and meta-data extracted from a movie specific domain. To a large extent, our quantitative and qualitative results weigh in on the performance impact of named entity presence in plot summaries. Pretrained on a split-and-rephrase corpus with knowledge distilled from English Wikipedia and fine-tuned on our movie dataset, our neural model shows to outperform both a language modeler and monolingual translation baselines.

1 Introduction

People who expose themselves to the process of satisfying curiosity expect to enhance the pleasure derived from obtaining new knowledge (Loewenstein, 1994; Litman, 2005). Conversely, induced revelatory information about a plot of a motion picture, TV program, video game, or book can spoil the viewer sense of surprise and suspense, and thus greatly diminish the enjoyment in consuming the media. As social media has thrived into a medium for self-expression, live tweets, opinion dumps, or even hashtags tend to proliferate within minutes of the media reaching the public, and hence the risk of uncovering a spoiler widens rapidly. Spoilers on review websites may inevitably contain undesired information and disclose critical plot twists that evoke far less interest to users who consult online reviews first, and later engage with the media itself.

Social media platforms have placed elaborate policies to better guard viewers from spoilers. On the producer side, some sites adopted a strict convention of issuing a spoiler alert to be announced in the subject of the post (Boyd-Graber et al., 2013). Recently, Twitter started to offer provisions for the tweet consumer to manually mute references to specific keywords and hashtags and stop displaying tweets containing them (Golbeck, 2012). But despite these intricate mechanisms that aim to preemptively ward off unwanted spoilers, the solutions proposed lack timely attraction of consumers and may not scale, as spoilers remain a first-class problem in online discourse. Rather than solicit spoiler annotations from users, this work motivates automatic detection of spoiler bias in review tweets, of which spoiler annotations are unavailable or scarce.

Surprisingly, for the past decade spoiler detection only drew little notice and remained a relatively understudied subject. Earlier work used machine learning techniques that incorporated human-curated features in a supervised settings, including a Latent Dirichlet Allocation (LDA) based model that combines simple bag-of-words (BOA) with linguistic cues to satisfy spoiler detection in review commentary (Guo and Ramakrishnan, 2010), baseline n-gram features augmented with binary meta-data that was extracted from their review dataset showed dramatically improved performance of spoiler detection in text (Boyd-Graber et al., 2013), and while frequent verbs and named entities play a critical role in identifying spoilers, adding objectivity and main sentence tense improved classification accuracy by about twelve percentage points (Jeon et al., 2013; Iwai et al., 2014).

More recently researchers started to apply deep learning methods to detect spoiler sentences in review corpora. The study by Chang et al. (2018) pro-

Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion, pages 51–60
April 19, 2021.

poses a model architecture that consists of a convolutional neural network (CNN) based genre encoder and a sentence encoder that uses a bidirectional gated recurrent unit (bi-GRU) (Cho et al., 2014). A genre-aware attention layer aids in learning spoiler relations that tend to vary by the class of the item reviewed. Their neural model was shown to outperform spoiler detection of machine-learning baselines that use engineered features. Using a book review dataset with sentence-level spoiler tags, Wan et al. (2019) followed a similar architecture explored by Chang et al. (2018) and introduced SpoilerNet that comprises a word and sentence encoders, each realizing a bi-GRU network. We found their error analysis interesting in motivating the rendition of spoiler detection as a ranking task rather than a conventional binary classification. Although only marginally related, noteworthy is an end-to-end similarity neural-network with a variance attention mechanism (Yang et al., 2019), proposed to address real-time spoiled content in time-sync comments that are issued in live video viewing. In this scenario, suppressing the impact of often occurring noisy-comments remains an outstanding challenge.

In our approach we propose to cast the task of spoiler detection in online discourse as ranking the quality of sentence simplification from an item description to a multitude of user reviews. We used the transformer neural architecture that dispenses entirely of recurrence to learn the mapping from a compound item summarization to the simpler tweets. The contributions of this work are summarized as follows:

- A high-quality and sustainable movie review dataset we constructed to study automatic spoiler detection in social media microblogs. The data was scraped from unsolicited Twitter posts and paired with a title caption and meta-data we extracted from the rich Internet Movie Database (IMDb). [1] We envision the dataset to facilitate future related research.
- We propose a highly effective transformer model for ranking spoiler bias in tweets. Our novelty lies in applying a preceding text simplification stage that consults an external paraphrase knowledge-base, to aid our downstream NLP task. Motivated by our zero-shot learning results (Table 1), we conjectured that the more simplified tweet, predicted from the movie caption, is likely to make up a spoiler.
- Through exhaustive experiments that we conducted, we provide both qualitative analysis and quantitative evaluation of our system. The results show that our method accomplished performance that outperforms strong baselines.

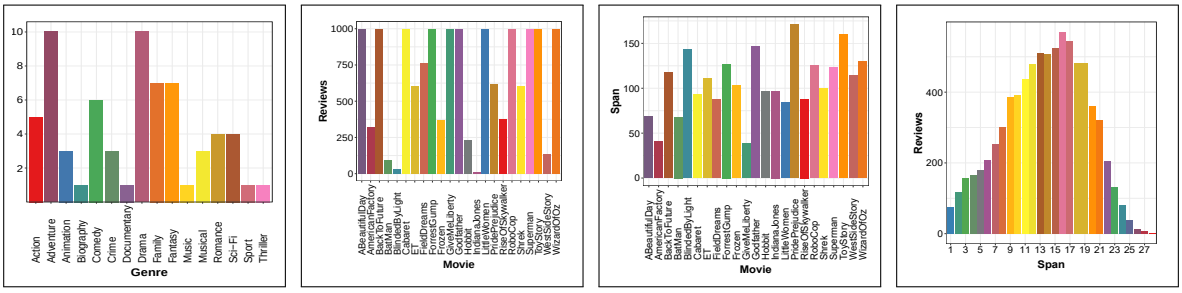
✓	i literally tried everything the force is strong with daisy ridley theriseofskywalker *(p=0.08)*
×	nfcs full podcast breakdown of star wars episode ix the rise of skywalkerstarwars *(p=0.18)*
×	new theriseofskywalker behindthescenes images show the kijimi set see more locations here *(p=0.27)*
✓	weekend box office report for janfebbadboysforlifemoviedolittlegret *(p=0.12)*
×	in my latest for the officialsite i examine some of the themes from theriseofskywalker and how they reinforce some *(p=0.23)*

Table 1: Zero-shot classification of unlabeled tweets about the movie The Rise of the Skywalker (annotating spoiler free ✓ and spoiler biased × posts). Along with text simplification ranking of spoiler bias $p \in [0, 1]$.

2 Related Work

Text simplification is an emerging NLP discipline with the goal to automatically reduce diction complexity of a sentence while preserving its original semantics. Inspired by the success of neural machine translation (NMT) models (Sutskever et al., 2014; Cho et al., 2014), sentence simplification has been the subject of several neural architectural efforts in recent years. Zhang and Lapata (2017) addressed the simplification task with a long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997) based encoder-decoder network, and employed a reinforcement learning framework to inject prior knowledge and reward simpler outputs. In their work, Vu et al. (2018) propose to combine Neural Semantic Encoders (NSE) (Munkhdalai and Yu, 2017), a novel class of memory augmented neural networks which offer a variable sized encoding memory, with a trailing LSTM-based decoder architecture. Their automatic evaluation suggests that by allowing access to the entire input sequence, NSE present an effective solution to simplify highly complex sentences. More recently, Zhao et al. (2018) incorporated a hand-crafted knowledge base of simplification rules into the self-attention transformer network (Vaswani et al., 2017). This model accurately selects simplified words and shows empiri-

[1] www.imdb.com



(a) (b) (c) (d)

Figure 1: Distributions across the entire movie dataset of (a) genre, (b) user reviews (capped per title at 1,000), and span length in words of (c) plot summaries, and (d) user reviews.

cally to outperform previous simplification models.

Title Name	Year	Title Name	Year
A Beautiful Day	2019	The Hobbit	2012
American Factory	2019	Indiana Jones	1981
Back to the Future	1985	Little Women	2019
Batman	1989	Pride and Prejudice	1995
Blinded by the Light	2019	The Rise of Skywalker	2019
Cabaret	1972	RoboCop	1987
ET	1982	Shrek	2001
Field of Dreams	1989	Superman	1978
Forrest Gump	1994	Toy Story	1995
Frozen	2013	West Side Story	1961
Give Me Liberty	2019	The Wizard of Oz	1939
The Godfather	1972		

Table 2: Movie title names and release dates.

3 Dataset

Our task requires a dataset that pairs user reviews obtained from discourse on social media with a summarization record collected from an item specific domain. By and large, a user-item model implies distinct review authors for each item. In this study we chose the movie media as the item, or the subject to review, of which a plot description lets users read about the title before watching the film. To the extent of our knowledge, publicly accessible spoiler corpora to date include the dataset used in the work by Guo and Ramakrishnan (2010) that consists of four movies and a total of 2,148 review comments, and more recently, the IMDb Spoiler Dataset, a large-scale corpus provided by Kaggle. [2] However, these datasets were entirely collected from IMDb. Instead, we scraped unlabeled movie reviews from Twitter and matched them with short-text descriptions we assembled from IMDb.

Movies posted on IMDb are each linked with a storyline at different plot detail, including keywords, outlines, summaries, and a single synopsis. Plot summaries are reasonably brief, about a paragraph or two long, and tend to contain small spoilers. Conversely, a plot synopsis contains a detailed description about the entire story of the film, excluding commentary, and may include spoilers that give away important plot points. In casting our spoiler detection task as sentence simplification, we chose to use for our model input to simplify the most comprehensive plot summary of each movie.

We constructed a new dataset for our experiments that comprises meta-data and a total of 15,149 user reviews across twenty three movies. [3] We selected movies released as early in 1939 till 2019 (Table 2), thus spanning an extended period of eighty years. In our title selection we attempted to obtain content coverage and represent a broad range of sixteen genre types. Figure 1a shows genre distribution across movies and highlights adventure and drama motion pictures the most popular, whereas biography, documentary, music, sport, and thriller narratives were anticipated more lower key with each holding a moderate single title presence. We used the Twitter API to search movie hashtags of which we fetched authorized tweets. In scraping user reviews, we placed an upper bound of one thousand asking tweets per movie item. The distribution in Figure 1b shows eleven out of twenty three, close to half the titles, have 1,000 reviews each, however for the remaining films we retrieved less tweets than were requested. On average there were about 658 reviews per title (Table 3).

We viewed our movie corpus as a parallel drawn between a single elaborate plot summary and many simpler review tweets. A text simplification system learns to split the compound passage into several

[2] www.kaggle.com/rmisra/imdb-spoiler-dataset

[3] We made meta-data and tweets publicly available at: https://github.com/bshalem/mst

Property	Min	Max	Mean	StdDev
Reviews	12	1,000	658.7	379.7
Summary Span	38	171	105.8	33.9
Review Span	1	28	13.7	5.4

Table 3: Statistics for distributions across our movie dataset of (a) number of reviews and (b) span length in words for plot summaries and user reviews.

small text sequences further rewritten to preserve meaning (Narayan et al., 2017). To perform lexical and syntactic simplification requires to sustain a reasonable span length ratio of the plot summary to a user review. This predominately motivated our choice of using review tweets that are extremely short-text sequences of 240 characters at most. Distributions of span length in words for both plot summaries and user reviews are shown in Figure 1c and Figure 1d, respectively. Review spans affecting at least 200 tweets are shown to range from 6 to 22 words and apportion around 46 percent of total reviews. Still, the minimum summary span-length has 38 words that is larger than the maximum review span-length at 28 words. As is evidenced in Table 3, at 7.7 on average, the span split ratio of complex to simple text sequences is most plausible.

4 Model

Neural text simplification (NTS) models prove successful to jointly perform compelling lexical simplification and content reduction (Nisioi et al., 2017). Combined with SARI (Xu et al., 2016), a recently introduced metric to measure the goodness of sentence simplification, NTS models are encouraged to apply a wide range of simplification operations, and moreover, their automatic evaluation was shown empirically to correlate well with human judgment of simplicity. Unlike BLEU (Papineni et al., 2002) that scores the output by counting n-gram matches with the reference, i.e. the simplified version of the complex passage, SARI principally compares system output against both the reference and input sentences and returns an arithmetic average of n-gram precisions and recalls of addition, copy, and delete rewrite operations. [4] SARI that correctly rewards models like ours which make changes that simplify input text sequences has inspired us to cast the spoiler detection task as a form of sentence simplification. We hypothesized that the better simplification of a user review from a plot summary, the higher the likelihood of the tweet to contain a spoiler. Moreover, our system is designed to not just confirm the presence or absence of a spoiler, but rather rank the plot revealing impact of a review tweet in a fine-grained scale.

Using a colon notation, we denote our collection of n movie objects $m_{1:n} = \{m_1, \ldots, m_n\}$, each incorporating a plot summary S and k user reviews $r_{1:k} = \{r_1, \ldots, r_k\}$. Thus the input movie m_i we feed our model consists of $k^{(i)}$ pairs $\{S, r_j\}^{(i)}$, where $1 \geq j \leq k^{(i)}$. Given S, a linearly l-split text sequence $s_{1:l} = \{s_1, \ldots, s_l\}$, our model predicts a simplified version $\hat{r}$ of an individual split s.

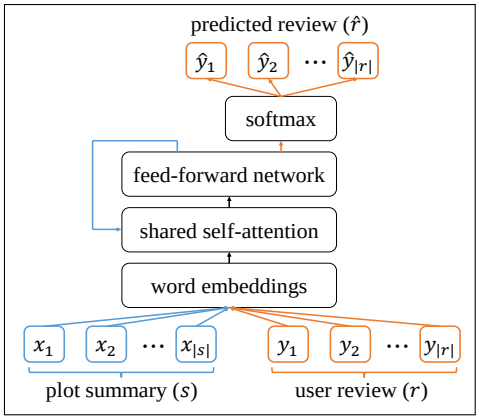


Figure 2: Architecture overview of the transformer (encoder path shown in blue, decoder in brown).

Neural sentence simplification is a form of text-to-text generation closely resembling the task of machine translation in the confines of a single language. In our study we used the self-attention transformer architecture that requires less computation to train and outperforms both recurrent and convolutional system configurations on many language translation tasks (Vaswani et al., 2017). In Figure 2, we present an overview of the transformer architecture. Stacked with several network layers, the transformer encoder and decoder modules largely operate in parallel. In the text simplification framework, the input sequences to the transformer consist of the compound passage words x_i, obtained from a summary split s, and the simple reference words y_i from the current user review r_j. To represent the lexical information of words x_i and y_i, we use pre-trained embedding vectors that are shared across our entire movie corpus. The transformer network facilitates complex-to-simple attention communication and a softmax layer operates on the decoder hidden-state outputs to produce the predicted simplified output $\hat{y}_i$. Our training objective is to minimize the cross-entropy loss of the reference-output review pairs.

[4] https://github.com/cocoxu/simplification

5 Experimental Setup

In this section, we provide preprocessing steps we applied to our movie corpus and details of our training methodology.

5.1 Corpus

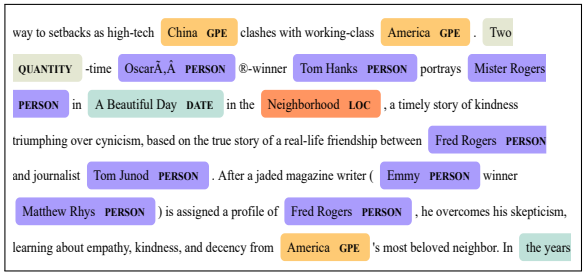


Figure 3: Highlighted named entities in the plot summary of the movie A Beautiful Day in the Neighborhood.

In building our movie dataset, we limited the language of tweets to English. Made of unstructured text, tweets tend to be noisy and often presented with incomplete sentences, irregular expressions, and out of vocabulary words. This involved scraped reviews to undergo several cleanup steps, including the removal of duplicate tweets and any hashtags, replies and mentions, retweets, and links. In addition, we pulled out punctuation symbols and numbers, and lastly converted the text to lowercase. [5] We used the Microsoft Translator Text API for language detection and retained tweets distinguished as English with certainty greater than 0.1. At this stage, we excluded empty and single-word tweets from our review collection, and all in all this reduced the total number of posts from our initial scraping of 15,149 down to 7,928 tweets. In contrast, plot summaries are well-defined and only required the conversion to lowercase. Respectively, vocabularies of plot summaries and user reviews consisted of 863 and 24,915 tokens.

In our evaluation, we compared the quality of text simplification between compound-simple pairs provided with named entities, and pairs of which named entities has been removed altogether. We used the spaCy library that excels at large-scale information extraction tasks, [6] and implements a fast statistical named entity recognition (NER) system. spaCy NER strongly depends nonetheless on the examples its model was trained on. Hence, to address unidentified or misinterpreted named entities in our corpus, we have extended the default training set of spaCy with our specific examples. For instance, we made *Sith* an organization class, *Jo March* a person rather than a date, and *IQ* a quantity. In Figure 3, we highlight the representation of the spaCy named entity visualizer for the plot summary of the movie A Beautiful Day in the Neighborhood. Labels shown include person, geopolitical (GPE) and non-GPE locations, date, and quantity.

5.2 Training

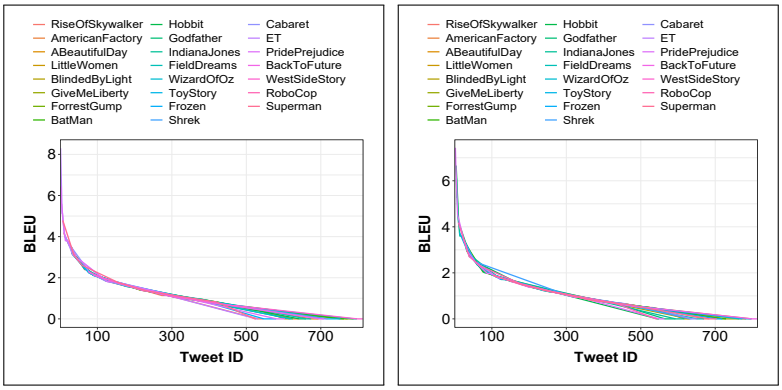
After replicating plot summaries to match the number of user reviews for each movie, we apportion the data by movie into train, validation, and test sets with an 80-10-10 split that amounts to 6,334, 781, and 813 summary-tweet pairs, respectively. To initialize shared vector representations for words that make up both plot summaries and user reviews, we chose GloVe 200-dimensional embeddings pretrained on the current largest accessible Twitter corpus that contains 27B tokens from uncased 2B tweets, with a vocabulary of 1.2M words. [7] Given the fairly small vocabulary size of plot summaries, we expected the embedding matrix generated from the large tweet corpus to perform vector lookup with a limited number of instances that flag unknown tokens.

Hyperparameter settings of the transformer included the number of encoder and decoder layers $N = 6$, the parallel attention head-count $h = 8$, and the model size $d_{model} = 512$. Then, the dimensionality of the query, key, and value vectors were set uniformly to $d_{model}/h = 64$, and the inner feed-forward network size $d_{ff} = 2,048$. Using a cross-entropy loss, we chose the Adam optimizer (Kingma and Ba, 2014) and varied the learning rate, and for network regularization we applied both a fixed model dropout of 0.1 and label smoothing.

6 Experiments

Our evaluation comprises both a spoiler classification and ranking components. First, we applied zero-shot learning (Lampert et al., 2014) to our unlabeled tweets and identified the presence or absence of a spoiler bias. Then, our proposed NTS model produced probability for each summary-post pair to rank the bias of spoilers in tweets. Performance of our NTS approach is further compared to a language modeling and monolingual translation baselines. Moreover, our study analyzes the

[5] All our tweets were rendered on February 2nd 2020.
[6] https://spacy.io
[7] https://nlp.stanford.edu/projects/glove/



(a) Named Entity Removal (b) Named Entity Inclusion

Figure 4: Monolingual Translation Baseline: Spoiler rank in BLEU scores for unsplit plot summaries with (a) named entities removed and (b) named entities included.

quality impact of named entity presence in the plot summary and offers an explanation to model behavior once named entities were removed from the text. We have conducted both automatic and human evaluation to demonstrate the effectiveness of the SARI metric, and suggest possible computation paths forward to improve performance.

	Precision	Recall	F1-Score	Support
Negative	0.48	0.41	0.44	388
Positive	0.51	0.59	0.55	425

Table 4: Zero-shot classification of our tweets. The support column identifies the number of user posts for a given category— spoiler free (negative) or biased (positive).

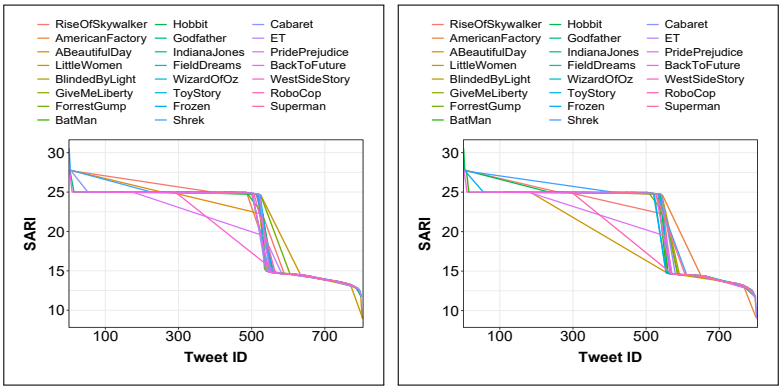
Zero-shot Spoiler Classification Baseline. We used a pretrained model on natural language inference (NLI) tasks to perform zero-shot classification of our unlabeled tweets. [8] To classify the presence or absence of a spoiler in tweets, we provided the NLI pretrained model premise-hypothesis pairs to predict entailment or contradiction. Table 4 summarizes our zero-shot classification results for spoilers in tweets with 388 and 425 user posts categorized negative and positive, respectively. In contrast with human judgment that had a fairly even distribution of spoiler free and spoiler biased user posts at 410 at 403, respectively. We achieved 0.51 accuracy compared to 0.74 on SpoilerNet (Wan et al., 2019) that uses a fully labeled, tenfold larger dataset.

Language Modeling Baseline. We built a language modeling baseline and trained it on a flattened dataset we produced by concatenating all our user reviews into one long text sequence and thus leaving out movie labels, title names, and IDs altogether. Respectively, dataset splits were of 99,575, 12,078, and 12,794 word long for the train, validation, and test sets. The neural network we trained for the language modeling task consisted of a transformer encoder module with its output sent to a linear transform layer that follows a log-softmax activation function. The function assigns a probability for the likelihood of a given word or a sequence of words to follow a sequence of text.

In this experiment, we extracted non-overlapping sub-phrases from each of the plot summaries, and performed on each sub-phrase sentence completion using our language modeling baseline. Uniformly we divided a plot text-sequence into sub-phrases of ten words each, thus leading to a range of four to seventeen sub-phrases per plot summary (Table 3), and a total of 243 phrases to complete for all the 23 titles. We configured our neural model to generate twenty completion words for each sub-phrase and expected one of three scenarios: (i) completion text is contained in one of the review tweets for the title described by the current plot summary, (ii) words partially matching a non-related movie review, and (iii) no completion words were generated. Respectively, these outcomes represent true positives, false positives, and false negatives of which we drew a 0.11 F1-score for sentence completion of plot summaries.

Monolingual Translation Baseline. The baseline for text simplification by monolingual machine translation uses a transformer based neural model that employs both an encoder and decoder modules (Figure 2). To train this model, we used the publicly

[8] https://huggingface.co/transformers/



(a) Named Entity Removal (b) Named Entity Inclusion

Figure 5: Text Simplification Mainline: Spoiler rank in SARI scores for linearly-split plot summaries with (a) named entities removed and (b) named entities included. Scores render the mean over all splits of a plot summary.

available dataset introduced by Surya et al. (2019). The data comprises 10,000 sentence pairs extracted from the English Normal-Simple Wikipedia corpus (Hwang et al., 2015) and the Split-Rephrase dataset (Narayan et al., 2017). [9] The Wikipedia dataset, with a large proportion of biased simplifications, was further filtered down to 4,000 randomly picked samples from good and partial matches with similarity scores larger than a 0.45 threshold (Nisioi et al., 2017). Whereas each of the remaining 6,000 Split-Rephrase records comprised one compound source sentence and two simple target sentences.

Using domain adaptation, our model was trained on distilled text simplification data and directly validated and inferred on our movie-review development and test sets, respectively. Noting that for this baseline we avoided a hyperparameter fine-tuning step on our title-review train set. Following NMT practices, we used the BLEU metric for automatic evaluation of our model, and in Figure 4, we show baseline spoiler ranking for unsplit plot summaries without (4a) and with (4b) named entities. BLEU scores ranged from zero to 8.3 with a mean of 0.9. As evidenced, the absence or presence of named entities has a relatively subtle performance impact.

Text Simplification Mainline. As the first stage for evaluating our mainline model, we conducted qualitative analysis to assess the simpleness of user reviews across our entire movie dataset. We classified tweets based on readability scores and used the Flesch Readability Ease (FRE) metric (Flesch, 1948). In Table 5, we provide statistics for user reviews categorized into seven readability grade-levels, and showing for each the number of tweets, FRE range, and average FRE rate. A standard FRE classification renders reviews into simple and complex groups of 990 and 6,930 tweets, respectively. While FRE has its shortcomings to fully qualify sentence simpleness, we found the bias in user reviews toward complex sentences extremely useful.

Grade Level	Tweets	Min	Max	Mean
Very Easy	40	90.7	100	97.4
Easy	126	80.1	89.9	84.1
Fairly Easy	274	70.3	79.1	74.5
Standard	550	60.0	69.9	64.4
Fairly Difficult	642	50.2	59.4	54.6
Difficult	1,503	30.3	49.8	40.1
Very Confusing	4,793	0	29.8	6.3

Table 5: Statistics of Flesch Readability Ease (FRE) scores for user reviews. Showing for each grade the number of tweets, FRE range, and average FRE rate.

Our mainline model starts off with the checkpoint of hyperparameter state from training the monolingual translation baseline on the text simplification dataset from Surya et al. (2019). We then followed with a fine-tuning step on the train and validation sets from our review dataset. In Figure 5 we show mainline spoiler ranking for plot summaries without (5a) and with (5b) named entities. We ran inference on the movie-review test set and report SARI scores for automatic evaluation. SARI scores range from 8.9 to 30.5 with an average rate of 21.3. Although the metrics we used for each baseline and mainline models differ, our results show that the mainline system outperforms the baselines by a considerable margin, owing to linearly-split plot summaries, the extra review-specific training step, and the use of the SARI metric that is effective for tuning and evaluating simplification systems.

[9] https://github.com/shashiongithub

The Rise of the Skywalker	Back to the Future	A Beautiful Day in the Neighborhood
did you know the scene in therise-ofskywalker when rey hears the voices of jedi past during her battle with palpatine was	what do marty mcfly dorothy gale and sadaharu oh have in common a propensity to exceed the established boundarie	tomhanks turns in a fine performance to bring the us childrens tv legend fred rogers to the big screen as matthewrhy

Table 6: A sample of test review tweets for distinct titles with highlighted named entities.

In Figure 6, we analyze the impact of FRE scores on our model performance. Test tweets are shown predominately to fall in two distinct clusters centered around SARI scores of about 14 and 25. Inexplicable at first observation, this rendition is presumably posed by the broadcast nature of online discourse that lets tweets to easily share peer content. On the other hand, FRE grades are shown scattered fairly evenly with no indication to affect performance adversely. We also inspected the reward of automatically extracting named entities from plot summaries and tweets for conducting spoiler identification effectively. In Table 6, we list a sample of user reviews with their named entities highlighted. The correlation between SARI scores obtained from removing and including named entities in plot summaries is presented in Figure 7. About half of the SARI scores turned identical, but the inclusion of named entities produced a marked eighteen percentage points with higher SARI scores.

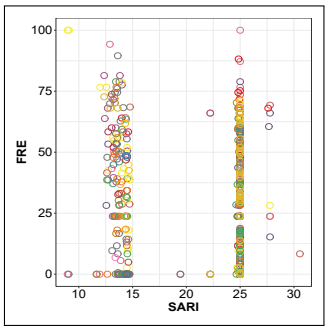


Figure 6: Correlation between FRE and SARI scores.

Data	Min	Max	Mean	SD
Plot Summaries	4	39	16.4	7.4
Reviews	1	12	5.2	2.2

Table 7: Statistics of named entity presence in unsplit plot summaries and reviews.

We also conducted human evaluation by manually rating the spoiledness of individual user reviews in our test set. To this extent, we built a vocabulary V_{ne} of the distinct named entities that are

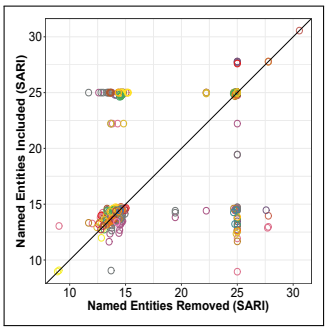
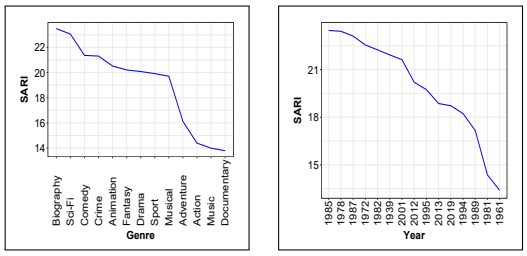


Figure 7: Correlation between SARI scores for plot summaries with named entities removed and included.

present in the entire collection of plot summaries. Our metric for human judgment then follows to rank spoiledness by computing the cosine similarity between two named entity vectors of dimensionality $|V_{ne}| = 274$ that represent a plot summary and a tweet, respectively. Human mediation was essential to inspect the integrity of named entities in tweets that were often misspelled, as evidenced in Table 6. Named entity distribution in unsplit plot summaries and test tweets were at an average ratio of 3 to 1 (Table 7) that correlates well with our SARI automatic scores.



(a) (b)

Figure 8: SARI spoiledness average scores as a function of the movie (a) major genre and (b) year.

In the absence of work with similar goals, we are unable to provide a fair performance comparison.

7 Discussion

Given its proven record of state-of-the-art performance for various NLP tasks, we considered the use of the BERT architecture (Devlin et al., 2019)—

a transformer based language model that was pretrained on a large-scale corpus of English tweets (BERTweet) (Nguyen et al., 2020). However, the smaller number of 850M tweets BERTweet employs compared to 2B posts in our corpus, along with the dearth of support for NTS at the time of publication precluded further use of BERTweet in our study.

Of great practical importance is the answer to how the genre of a movie or the year in which it was released weigh in on the amount of spoiler bias. Average SARI scores of spoiledness as a function of the primary movie genre (family, romance, and thriller considered secondary) are shown in Figure 8a. Somewhat surprisingly, biography and science fiction types came most vulnerable, as adventure and action titles scored lower by about 8 SARI points, and documentary category appears the most immune to spoilers. In Figure 8b, we reviewed bias impact of the year the movie was launched. Most biased are movies of the seventies and eighties, as the more recent twenty first century movies are clustered in the middle, and West Side Story (1961) ranked the least biased.

8 Conclusions

We proposed a new dataset along with applying advanced language technology to discover emerging spoiler bias in tweets that represent a diverse population with different cultural values. Performing text simplification, our model principally drew on named entity learning in an item-user paradigm for media review and showed through correlational studies plausible performance gains.

The results we presented in this study carve several avenues of future research such as minimizing spoiler bias before user comments reach their audience, and incorporating novel pretrained language models and training schemes to improve bias ranking quality. When supplied with the proper dataset, we foresee our generic method aid researchers to address a broader bias term in text.

Acknowledgments

We would like to thank the anonymous reviewers for their insightful suggestions and feedback.

References

Jordan Boyd-Graber, Kimberly Glasgow, and Jackie Sauter Zajac. 2013. Spoiler alert: Machine learning approaches to detect social media posts with revelatory information. *American Society for Information Science and Technology (ASIS&T)*, 50(1):1–9.

Buru Chang, Hyunjae Kim, Raehyun Kim, Deahan Kim, and Jaewoo Kang. 2018. A deep neural spoiler detection model using a genre-aware attention mechanism. In *Advances in Knowledge Discovery and Data Mining (PAKDD)*, pages 183–195. Springer Verlag.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, pages 4171–4186, Minneapolis, Minnesota.

Rudolf Franz Flesch. 1948. A new readability yardstick. *The Journal of applied psychology*, 32 3:221–233.

Jennifer Golbeck. 2012. The twitter mute button: A web filtering challenge. In *Conference on Human Factors in Computing Systems (SIGCHI)*, page 2755–2758, Austin, Texas.

Sheng Guo and Naren Ramakrishnan. 2010. Finding the storyteller: Automatic spoiler tagging using linguistic cues. In *Conference on Computational Linguistics (COLING)*, pages 412–420, Beijing, China.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

William Hwang, Hannaneh Hajishirzi, Mari Ostendorf, and Wei Wu. 2015. Aligning sentences from standard Wikipedia to simple Wikipedia. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 211–217, Denver, Colorado.

Hidenari Iwai, Yoshinori Hijikata, Kaori Ikeda, and Shogo Nishida. 2014. Sentence-based plot classification for online review comments. In *Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, page 245–253, Warsaw, Poland.

Sungho Jeon, Sungchul Kim, and Hwanjo Yu. 2013. Don't be spoiled by your friends: Spoiler detection in tv program tweets. In *Conference on Weblogs and Social Media (ICWSM)*, pages 681–684.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980. http://arxiv.org/abs/1412.6980.

C. H. Lampert, H. Nickisch, and S. Harmeling. 2014. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465.

Jordan Litman. 2005. Curiosity and the pleasures of learning: Wanting and liking new information. *Cognition and Emotion*, 19(6):793–814.

George Loewenstein. 1994. The psychology of curiosity: A review and reinterpretation. *Psychological Bulletin*, 116(1):75–98.

Tsendsuren Munkhdalai and Hong Yu. 2017. Neural semantic encoders. In *European Chapter of the Association for Computational Linguistics (EACL)*, pages 397–407, Valencia, Spain.

Shashi Narayan, Claire Gardent, Shay B. Cohen, and Anastasia Shimorina. 2017. Split and rephrase. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 606–616, Copenhagen, Denmark.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online.

Sergiu Nisioi, Sanja Štajner, Simone Paolo Ponzetto, and Liviu P. Dinu. 2017. Exploring neural text simplification models. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 85–91, Vancouver, Canada.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318, Philadelphia, Pennsylvania.

Sai Surya, Abhijit Mishra, Anirban Laha, Parag Jain, and Karthik Sankaranarayanan. 2019. Unsupervised neural text simplification. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2058–2068, Florence, Italy.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Neural Information Processing Systems*, page 3104–3112, Montreal, Canada.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)*, pages 5998–6008. Curran Associates, Inc., Red Hook, NY.

Tu Vu, Baotian Hu, Tsendsuren Munkhdalai, and Hong Yu. 2018. Sentence simplification with memory-augmented neural networks. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, New Orleans, Louisiana.

Mengting Wan, Rishabh Misra, Ndapa Nakashole, and Julian McAuley. 2019. Fine-grained spoiler detection from large-scale review corpora. In *Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 2605–2610, Florence, Italy.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Wenmian Yang, Weijia Jia, Wenyuan Gao, Xiaojie Zhou, and Yutao Luo. 2019. Interactive variance attention based online spoiler detection for time-sync comments. In *Conference on Information and Knowledge Management (CIKM)*, pages 1241–1250, Beijing, China. ACM.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 584–594, Copenhagen, Denmark.

Sanqiang Zhao, Rui Meng, Daqing He, Andi Saptono, and Bambang Parmanto. 2018. Integrating transformer and paraphrase rules for sentence simplification. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 3164–3173, Brussels, Belgium.