Letting Emotions Flow: Success Prediction by Modeling the Flow of Emotions in Books

Suraj Maharjan⋆ **Sudipta Kar**⋆ **Manuel Montes-y-Gómez**†
Fabio A. González‡ **Thamar Solorio**⋆
⋆Department of Computer Science, University of Houston
†Instituto Nacional de Astrofisica Optica y Electronica, Puebla, Mexico
‡Systems and Computer Engineering Department, Universidad Nacional de Colombia
{smaharjan2, skar3}@uh.edu, solorio@cs.uh.edu
mmontesg@ccc.inoep.mx
fagonzalezo@unal.edu.co

Abstract

Books have the power to make us feel happiness, sadness, pain, surprise, or sorrow. An author's dexterity in the use of these emotions captivates readers and makes it difficult for them to put the book down. In this paper, we model the flow of emotions over a book using recurrent neural networks and quantify its usefulness in predicting success in books. We obtained the best weighted F1-score of 69% for predicting books' success in a multitask setting (simultaneously predicting success and genre of books).

1 Introduction

Books have the power to evoke a multitude of emotions in their readers. They can make readers laugh at a comic scene, cry at a tragic scene and even feel pity or hate for the characters. Specific patterns of emotion flow within books can compel the reader to finish the book, and possibly pursue similar books in the future. Like a musical arrangement, the right emotional rhythm can arouse readers, but even a slight variation in the composition might turn them away.

Vonnegut (1981) discussed the potential of plotting emotions in stories on the "Beginning-End" and the "Ill Fortune-Great Fortune" axes. Reagan et al. (2016) used mathematical tools like Singular Value Decomposition, agglomerative clustering, and Self Organizing Maps (Kohonen et al., 2001) to generate basic shapes of stories. They found that stories are dominated by six different shapes. They even correlated these different shapes to the success of books. Mohammad (2011) visualized emotion densities across books of different genres. He found that the progression of emotions varies with the genre. For example, there is a stronger progression into darkness in horror stories than in comedy. Likewise, Kar et al. (2018) showed that movies having similar flow of emotions across their plot synopses were assigned similar set of tags by the viewers.

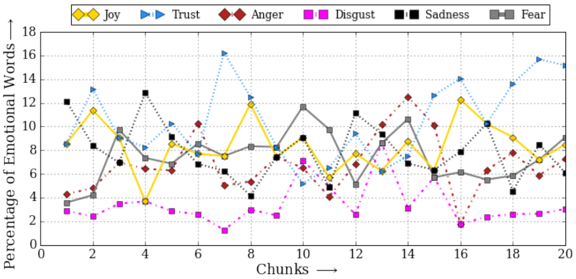


Figure 1: Flow of emotions in *Alice in Wonderland*.

As an example, in Figure 1, we draw the flow of emotions across the book: *Alice in Wonderland*. The plot shows continuous change in *trust*, *fear*, and *sadness*, which relates to the main character's getting into and out of trouble. These patterns present the emotional arcs of the story. Even though they do not reveal the actual plot, they indicate major events happening in the story.

In this paper, we hypothesize that readers enjoy emotional rhythm and thus modeling emotion flows will help predicting a book's potential success. In addition, we show that using the entire content of the book yields better results. Considering only a fragment, as done in earlier work that focuses mainly on style (Maharjan et al., 2017; Ashok et al., 2013), disregards important emotional changes. Similar to Maharjan et al. (2017), we also find that adding genre as an auxiliary task improves success prediction.

2 Methodology

We extract emotion vectors from different chunks of a book and feed them to a recurrent

The source code and data for this paper can be downloaded from https://github.com/sjmaharjan/emotion_flow

Proceedings of NAACL-HLT 2018, pages 259–265
New Orleans, Louisiana, June 1 - 6, 2018. ©2018 Association for Computational Linguistics

neural network (RNN) to model the sequential flow of emotions. We aggregate the encoded sequences into a single book vector using an attention mechanism. Attention models have been successfully used in various Natural Language Processing tasks (Wang et al., 2016; Yang et al., 2016; Hermann et al., 2015; Chen et al., 2016; Rush et al., 2015; Luong et al., 2015). This final vector, which is emotionally aware, is used for success prediction.

Representation of Emotions: NRC Emotion Lexicons provide ∼14K words (Version 0.92) and their binary associations with eight types of elementary emotions (*anger*, *anticipation*, *joy*, *trust*, *disgust*, *sadness*, *surprise,* and *fear*) from the Hourglass of emotions model with polarity (*positive* and *negative*) (Mohammad and Turney, 2013, 2010). These lexicons have been shown to be effective in tracking emotions in literary texts (Mohammad, 2011).

Inputs: Let X be a collection of books, where each book $x \in X$ is represented by a sequence of n chunk emotion vectors, $x = (x_1, x_2, ..., x_n)$, where x_i is the aggregated emotion vector for chunk i, as shown in Figure 2. We divide the book into n different chunks based on the number of sentences. We then create an emotion vector for each sentence by counting the presence of words of the sentence in each of the ten different types of emotions of the NRC Emotion Lexicons. Thus, the sentence emotion vector has a dimension of 10. Finally, we aggregate these sentence emotion vectors into a chunk emotion vector by taking the average and standard deviation of sentence vectors in the chunk. Mathematically, the ith chunk emotion vector (x_i) is defined as follows:

$$x_i = \left[\frac{\sum_{j=1}^{N} s_{ij}}{N}; \sqrt{\frac{\sum_{j=1}^{N}\left(s_{ij}-\bar{s}_i\right)^2}{N}}\right] \qquad (1)$$

where, N is the total number of sentences, s_{ij} and $\bar{s}_i$ are the jth sentence emotion vector and the mean of the sentence emotion vectors for the ith chunk, respectively. The chunk vectors have a dimension of 20 each. The motivation behind using the standard deviation as a feature is to capture the dispersion of emotions within a chunk.

Model: We then use bidirectional Gated Recurrent Units (GRUs) (Bahdanau et al., 2014) to summarize the contextual emotion flow information from both directions. The forward and backward GRUs will read the sequence from x_1 to x_n, and from x_n to x_1, respectively. These operations will compute the forward hidden states $(\overrightarrow{h_1}, \ldots, \overrightarrow{h_n})$ and backward hidden states $(\overleftarrow{h_1}, \ldots, \overleftarrow{h_n})$. The annotation for each chunk x_i is obtained by concatenating its forward hidden states $\overrightarrow{h_i}$ and its backward hidden states $\overleftarrow{h_i}$, i.e. h_i=$[\overrightarrow{h_i}; \overleftarrow{h_i}]$. We then learn the relative importance of these hidden states for the classification task. We combine them by taking the weighted sum of all h_i and represent the final book vector r using the following equation:

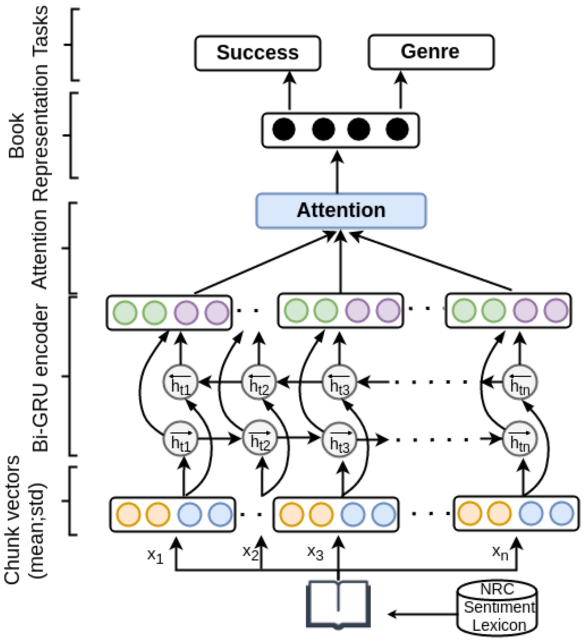


Figure 2: Multitask Emotion Flow Model.

$$r = \sum_i \alpha_i h_i \qquad (2)$$

$$\alpha_i = \frac{\exp(score(h_i))}{\sum_{i'} \exp(score(h_{i'}))} \qquad (3)$$

$$score(h_i) = v^T selu(W_a h_i + b_a) \qquad (4)$$

where, α_i are the weights, W_a is the weight matrix, b_a is the bias, v is the weight vector, and $selu$ (Klambauer et al., 2017) is the nonlinear activation function. Finally, we apply a linear transformation that maps the book vector r to the number of classes. In case of the single task (ST) setting, where we only predict success, we apply sigmoid activation to get the final prediction probabilities and compute errors using binary cross entropy loss. Similarly, in the multitask

(MT) setting, where we predict both success and genre (Maharjan et al., 2017), we apply a softmax activation to get the final prediction probabilities for the genre prediction. Here, we add the losses from both tasks, i.e. L_{total}=L_{suc} + L_{gen} (L_{suc} and L_{gen} are success and genre tasks' losses, respectively), and then train the network using backpropagation.

3 Experiments

3.1 Dataset

We experimented with the dataset introduced by Maharjan et al. (2017). The dataset consists of 1,003 books from eight different genres collected from Project Gutenberg[1]. The authors considered only those books that were at least reviewed by ten reviewers. They categorized these books into two classes, *Successful* (654 books) and *Unsuccessful* (349 books), based on the average rating for the books in Goodreads[2] website. They considered only the first 1K sentences from each book.

3.2 Baselines

We compare our proposed methods with the following baselines:

Majority Class: The majority class in training data is *success*. This baseline obtains a weighted F1-score of 0.506 for all the test instances.

SentiWordNet+SVM: Maharjan et al. (2017) used SentiWordNet (Baccianella et al., 2010) to compute the sentiment features along with counts of different Part of Speech (PoS) tags for every 50 consecutive sentences (20 chunks from 1K sentences) and used an SVM classifier.

NRC+SVM: We concatenate the chunk emotion vectors (x_i) created using the NRC lexicons and feed them to the SVM classifier. We experiment by varying the number of book chunks.

These baseline methods do not incorporate the sequential flow of emotions across the book and treat each feature independently of each other.

3.3 Experimental Setup

We experimented with the same random stratified splits of 70:30 training to test ratio as used by Maharjan et al. (2017). We use the SVM algorithm for the baselines and RNN for our proposed emotion flow method. We tuned the *C* hyperparameter of the SVM classifier by performing grid search on the values (1e{-4,...,4}), using three fold cross validation on the training split. For the experiments with RNNs, we first took a random stratified split of 20% from the training data as validation set. We then tuned the RNN hyperparameters by running 20 different experiments with a random selection of different values for the hyperparameters. We tuned the weight initialization (Glorot Uniform (Glorot and Bengio, 2010), LeCun Uniform (LeCun et al., 1998)), learning rate with Adam (Kingma and Ba, 2015) {1e-4,. . . ,1e-1}, dropout rates {0.2,0.4,0.5}, attention and recurrent units {32, 64}, and batch-size {1, 4, 8} with early stopping criteria.

4 Results

Book Content		**1000 sents**		**All**	
Methods	**Chunks**	**ST**	**MT**	**ST**	**MT**
Majority Class	-	0.506	0.506	0.506	0.506
SentiWordNet + SVM	20	0.582	0.610	-	-
NRC + SVM	10	0.526	0.597	0.541	0.641
NRC + SVM	20	0.537	0.590	0.577	0.604
NRC + SVM	30	0.587	0.576	0.595	0.600
NRC + SVM	50	0.611	0.586	0.597	0.636
Emotion Flow	10	0.632	0.643	0.650	0.660
Emotion Flow	20	0.612	0.639	0.640	0.668
Emotion Flow	30	0.630	0.657	0.662	0.677
Emotion Flow	50	0.656	0.666	0.674	**0.690***

Table 1: Weighted F1-scores for success classification in single task (ST) and multi task (MT) settings with varying chunk sequences when using all the book or only the first 1K sentences. $*p < 0.05$ (McNemar significance test between Emotion Flow (chunks 50, MT, All) and NRC+SVM (chunk 10, MT, All))

Table 1 presents the results. Our proposed method performs better than different baseline methods and obtains the highest weighted F1-score of 0.690. The results highlight the importance of taking into account the sequential flow of emotions across books to predict how much readers will like a book. We obtain better performance when we use an RNN to feed the sequences of emotion chunk vectors. The performance decreases with the SVM classifier, which discards this sequential information by treating each feature independently of each other. Moreover, increasing the granularity of the emotions by increasing the number of chunks seems to be helpful for success prediction. However, we see a slight decrease in performance beyond 50 chunks (weighted F1 score of 0.662 and 0.664 for 60 and 100 chunks, respectively).

The results also show that the MT setting is beneficial over the ST setting, whether we consider

[1] https://www.gutenberg.org/

[2] https://www.goodreads.com/

the first 1K sentences or the entire book. This finding is akin to Maharjan et al. (2017). Similar to them, we suspect the auxiliary task of genre classification is acting as a regularizer.

Considering only the first 1K sentences of books may miss out important details, especially when the only input to the model is the distribution of emotions. It is necessary to include information from later chapters and climax of the story as they gradually reveal the answers to the suspense, the events, and the emotional ups and downs in characters that build up through the course of the book. Accordingly, our results show that it is important to consider emotions from the entire book rather than from just the first 1K sentences.

5 Attention Analysis

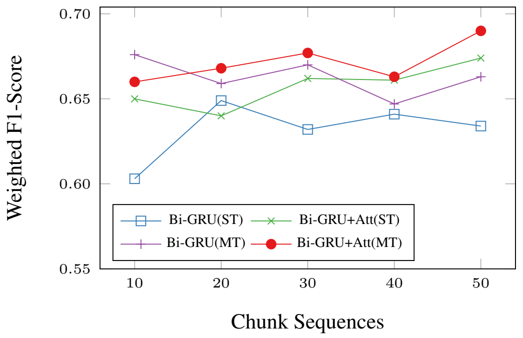


Figure 3: Comparison of the Emotion Flow with and without attention mechanism for different chunk sequences.

From Figure 3, we see that using the attention mechanism to aggregate vectors is better than just concatenating the final forward and backward hidden states to represent the book in both ST and MT settings. We also observe that the multitask approach performs better than the singe task one regardless of the number of chunks and the use of attention.

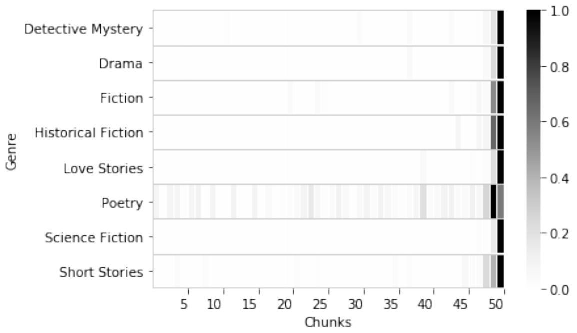


Figure 4: Attention weights visualization per genre.

Figure 4 plots the heatmap of the average attention weights for test books grouped by their genre. The model has learned that the last two to three chunks that represent the climax, are most important for predicting success. Since this is a bidirectional RNN model, hidden representations for each chunk carry information from the whole book. Thus, using only the last chunks will probably result in lower performance. Also, the weights visualization shows an interesting pattern for *Poetry*. In *Poetry*, emotions are distributed across the different regions. This may be due to sudden important events or abrupt change in emotions. For *Short stories*, initial chunks also receive some weights, suggesting the importance of the premise.

6 Emotion Analysis

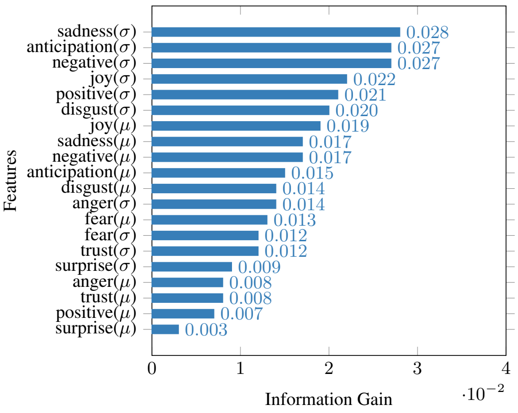


Figure 5: Feature ranking with information gain.

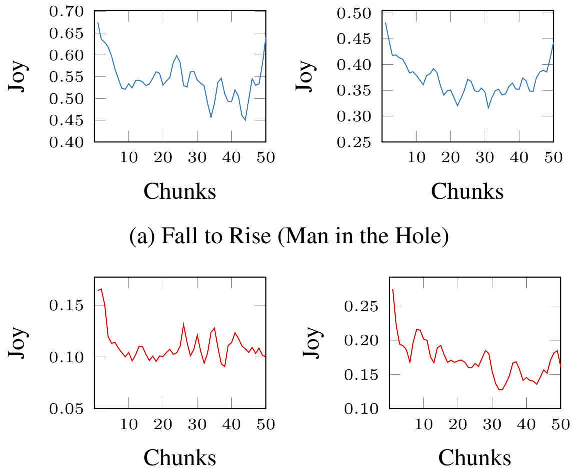
Climax Emotions: Since the last chunk is assigned more weights than other chunks, we used information gain to rank features of that chunk. From Figure 5, we see that features capturing the variation of different emotions are ranked higher than features capturing the average scores. This suggests that readers tend to enjoy emotional ups and downs portrayed in books, making the standard deviation features more important than the average features for the same emotions.

Table 2 shows the mean (μ) and standard deviation (σ) for different emotions extracted for all the data, and further categorized by *Successful* and *Unsuccessful* label from the last chunk. We see that authors generally end books with higher rates of positive words ($\mu = 0.888$) than negative words ($\mu = 0.599$) and the difference is significant ($p < 0.001$). Similarly the means for *anticipation*, *joy*, *trust*, and *fear* are higher than for *sadness*, *surprise*, *anger*, and *disgust*. This further vali-

	Anger		Anticipation		Disgust		Fear		Joy		Sadness		Surprise		Trust	
Dataset	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ
Corpus	0.248	0.249	0.414	0.372	0.179	0.200	0.340	0.327	0.399	0.431	0.323	0.309	0.225	0.191	0.492	0.441
Successful	0.270	0.263	0.447	0.374	0.194	0.207	0.377	0.353	0.435	0.416	0.358	0.334	0.236	0.207	0.517	0.427
Unsuccessful	0.207	0.214	0.351	0.359	0.153	0.183	0.270	0.258	0.331	0.451	0.258	0.243	0.205	0.155	0.445	0.463

Table 2: Mean (μ) and standard deviation (σ) for eight type of emotions for the last chunk.

dates that authors prefer happy ending. Moving on to *Successful* and *Unsuccessful* categories, we see that the means for *Successful* books are higher than *Unsuccessful* books for *anger*, *anticipation*, *disgust*, *fear*, *joy*, and *sadness* (highly significant, $p < 0.001$). We observe the same pattern for *trust*, and *surprise*, although the p value is only $p < 0.02$ in this case. Moreover, the standard deviations for all emotions are significantly different across the two categories ($p < 0.001$). Thus, emotion concentration (μ) and variation (σ) for *Successful* books are higher than for *Unsuccessful* books for all emotions in the NRC lexicon.



(a) Fall to Rise (Man in the Hole)

(b) Riches to Rags (Tragedy)

Figure 6: Emotion flow for four cluster centroids in the dataset. The two curves on top match the "Fall to Rise" shape and the two at the bottom match the "Tragedy" one defined in Reagan et al. (2016).

Emotion Shapes: We visualize the prominent emotion flow shapes in the dataset using K-means clustering algorithm. We took the average *joy* across 50 chunks for all books and clustered them into 100 different clusters. We then plotted the smoothed centroid of clusters having ≥ 20 books. We found two distinct shapes ("Man in the hole" (fall to rise) and "Tragedy" or "Riches to rags" (fall)). Figure 6 shows such centroid plots. The plot also shows that the "Tragedy" shapes have an overall lower value of joy than the "Man in the hole" shapes. Upon analyzing the distribution of *Successful* and *Unsuccessful* books within these shapes, we found that the "Man in the hole" shapes have a higher number of successful books whereas, the "Tragedy" shapes have the opposite.

7 Conclusions and Future Work

In this paper, we showed that modeling emotions as a flow, by capturing the emotional content at different stages, improves prediction accuracy. We learned that most of the attention weights are given to the last fragment in all genres, except for *Poetry* where other fragments seem to be relevant as well. We also showed empirically that adding an attention mechanism is better than just considering the last forward and backward hidden states from the RNN. We found two distinct emotion flow shapes and found that the clusters with "Tragedy" shape had more unsuccessful books than successful ones. In future work, we will be exploring how we can use these flows of emotions to detect important events that result in suspenseful scenes. Also, we will be applying hierarchical methods that take in the logical grouping of books (sequence of paragraphs to form a chapter and sequence of chapters to form a book) to build books' emotional representations.

Acknowledgments

We would like to thank the National Science Foundation for funding this work under award 1462141. We are also grateful to the anonymous reviewers for reviewing the paper and providing helpful comments and suggestions.

References

Vikas Ashok, Song Feng, and Yejin Choi. 2013. Success with style: Using writing style to predict the success of novels. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Seattle, Washington, USA, pages 1753–1764. http://www.aclweb.org/anthology/D13-1181.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *LREC*. volume 10, pages 2200–2204.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR* abs/1409.0473. `http://arxiv.org/abs/1409.0473`.

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Berlin, Germany, pages 2358–2367. `http://www.aclweb.org/anthology/P16-1223`.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. PMLR, Chia Laguna Resort, Sardinia, Italy, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256. `http://proceedings.mlr.press/v9/glorot10a.html`.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, Curran Associates, Inc., pages 1693–1701. `http://papers.nips.cc/paper/5945-teaching-machines-to-read-and-comprehend.pdf`.

Sudipta Kar, Suraj Maharjan, A. Pastor López-Monroy, and Thamar Solorio. 2018. MPST: A corpus of movie plot synopses with tags. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. European Language Resources Association (ELRA), Paris, France.

Diederik Kingma and Jimmy Ba. 2015. Adam: a method for stochastic optimization. In *International Conference on Learning Representations*.

Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. 2017. Self-normalizing neural networks. *CoRR* abs/1706.02515. `http://arxiv.org/abs/1706.02515`.

T. Kohonen, M. R. Schroeder, and T. S. Huang, editors. 2001. *Self-Organizing Maps*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 3rd edition.

Yann LeCun, Léon Bottou, Genevieve B. Orr, and Klaus-Robert Müller. 1998. Efficient backprop. In *Neural Networks: Tricks of the Trade, This Book is an Outgrowth of a 1996 NIPS Workshop*. Springer-Verlag, London, UK, UK, pages 9–50. `http://dl.acm.org/citation.cfm?id=645754.668382`.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 1412–1421. `http://aclweb.org/anthology/D15-1166`.

Suraj Maharjan, John Arevalo, Manuel Montes, Fabio A. González, and Thamar Solorio. 2017. A multi-task approach to predict likability of books. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Association for Computational Linguistics, Valencia, Spain, pages 1217–1227. `http://www.aclweb.org/anthology/E17-1114`.

Saif Mohammad. 2011. From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In *Proceedings of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*. Association for Computational Linguistics, Stroudsburg, PA, USA, LaTeCH '11, pages 105–114. `http://dl.acm.org/citation.cfm?id=2107636.2107650`.

Saif Mohammad and Peter Turney. 2010. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*. Association for Computational Linguistics, Los Angeles, CA, pages 26–34. `http://www.aclweb.org/anthology/W10-0204`.

Saif M. Mohammad and Peter D. Turney. 2013. Crowdsourcing a word–emotion association lexicon. *Computational Intelligence* 29(3):436–465. `https://doi.org/10.1111/j.1467-8640.2012.00460.x`.

Andrew J. Reagan, Lewis Mitchell, Dilan Kiley, Christopher M. Danforth, and Peter Sheridan Dodds. 2016. The emotional arcs of stories are dominated by six basic shapes. *CoRR* abs/1606.07772. `http://arxiv.org/abs/1606.07772`.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, pages 379–389. `http://aclweb.org/anthology/D15-1044`.

Kurt Vonnegut. 1981. Palm sunday: An autobiographical collage.

Yequan Wang, Minlie Huang, xiaoyan zhu, and Li Zhao. 2016. Attention-based lstm for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, pages 606–615. https://aclweb.org/anthology/D16-1058.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, San Diego, California, pages 1480–1489. http://www.aclweb.org/anthology/N16-1174.