Predicting Personal Opinion on Future Events with Fingerprints

Fan Yang*
University of Houston
Houston, USA
fyangskip@gmail.com

Eduard Dragut
Temple University
Philadelphia, USA
edragut@temple.edu

Arjun Mukherjee
University of Houston
Houston, USA
arjun@cs.uh.edu

Abstract

Predicting users' opinions in their response to social events has important real-world applications, many of which with political and social impacts. Existing approaches derive a population's opinion on a going event from large scores of user generated content. In certain scenarios, we may not be able to acquire such content and thus cannot infer opinion on those emerging events. To address this problem, we propose to explore opinion on unseen articles based on an user's fingerprinting: the prior reading and commenting history. This work presents a focused study on modeling and leveraging fingerprinting techniques to predict a user's future opinion to an unseen event or topic. We introduce a recurrent neural network based model that integrates fingerprinting. We collect a large dataset that consists of event-comment pairs from six news websites. We evaluate the proposed model on this dataset. The results show substantial performance gains demonstrating the effectiveness of our approach.

1 Introduction

Opinion mining and sentiment analysis has important applications with practical socio-political and economical benefits. There are numerous examples of works that show applications of sentiment analysis beyond classification. It can be used for analyzing political preferences of the electorate or for mining sentiments and emotions of people who lived in the past. The goal of these studies is not only to recognize sentiments, but also to understand how they were formed. All these approaches share the same starting point: abundant availability of user-generated content (UGC), such as reviews and Twitter posts (Dave et al., 2003; Hu and Liu, 2004; Pang and Lee, 2008; Pak and Paroubek, 2010; Liu, 2010; Taboada et al., 2011; Liu, 2012; Zhu et al., 2011; Yang et al., 2016; Joulin et al., 2017). They focus on document classification and predict sentiment, stance, subjectivity, or aspect.

The success of these methods relies on the assumption that people have already expressed their opinion on a topic or event. However, we may not acquire those opinion if people are not willing to reveal them or did not have the chance to express their opinions (e.g., about a new piece of legislature). Another instance is the availability of biased content on the web. One cannot possibly read all the content on the web and can only peruse a tiny part of the information, and people often read the content that is consistent with their prior beliefs (Mullainathan and Shleifer, 2005; Xiang and Sarvary, 2007). As a result, existing methods cannot deal with the situation when there is no user-generated content and might fail to collect opinion on certain topics.

We aim to fill this gap. We propose to predict opinion about public events before people leave their comments. We consider news articles as the description of events, because people usually check breaking news about emerging events from news portals, micro-blogs, and social media (Di Crescenzo et al., 2017) and leave comments. We predict the opinion of users on all events regardless of whether they have read those articles. The key idea is to access the historical event-opinion pairs of users and learn about a user's fingerprinting, which we assume to be consistent over a period of time. We expect this fingerprinting

* The work was done prior to joining Amazon.

...
Event: Hillary Clinton reacts to controversial Trump retweet: 'We need a real president'.
Comment: Actually Trump is still your president so don't cry.
...
Event: Pelosi creates new House committee with subpoena power for coronavirus oversight.
Comment: Still your president. Btw Trumps polls are higher than 2016 so that is a win.
...

Table 1: An example of one's historical event-comment pairs

	Article	Comment	User
AchiveIs	2043	812768	25474
DailyMail	21040	9374073	127419
FoxNews	4675	15989665	78678
NewYorkTimes	3647	2328597	52930
TheGuadian	6393	5467755	72904
WSJ	8365	1822599	18600

Table 2: Data Description

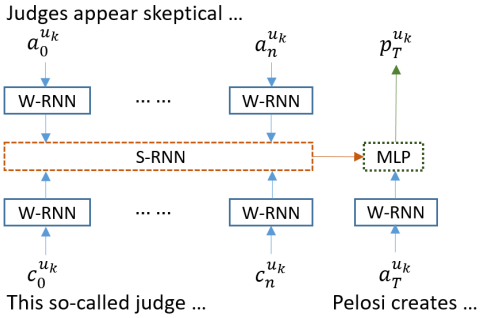

Figure 1: The illustration of our model.

to implicitly capture what a person has said about events and assist opinion mining on unseen articles. For example, a user commented "*still your president*" for events about "*Hillary Clinton*" and "*Pelosi*" as shown in Table 1, and we might expect this user to write similar comments on other events of the Democratic Party, such as the candidacy of Elizabeth Warren to U.S. presidency.

Since there is no research directly addressing this problem, we collect event-comment pairs from six websites: ArchiveIs, DailyMail, FoxNews, NYTimes, TheGuardian, and WSJ. There are in total more than 45K articles, 35M comments, and 376K users in this dataset. The overview of the dataset is given in Table 2. Empirically, we quantify the concept of opinion by sentiment and subjectivity and apply four methods for auto labeling. We propose an approach that takes two inputs: (i) a user identifier that enables the system to access historical data; (ii) an event as reported by a news article. We evaluate four baselines: three of them leverage the historical data and one baseline learns user embedding with neural collaborative filtering (He et al., 2017). We further introduce a recurrent neural network model that encodes a user's reading and commenting history. Our experiment indicates that encoding user historical data with recurrent neural network improves the performance of predicting sentiment and subjectivity on unseen articles over these baselines.

2 Method

We describe the proposed model in this section. Assuming that among K users we are interested in knowing the opinion of the user u_k on a newly occurring event a_T, and the user u_k has previously left n article-comment pairs, i.e., $[c_0^{u_k}, \ldots, c_n^{u_k}]$ and $[a_0^{u_k}, \ldots, a_n^{u_k}]$, we aim to model the prediction $p_T^{u_k}$ as the opinion of the user on a future article $a_T^{u_k}$. An overview of the model is illustrated in Figure 1.

Since recurrent neural network architecture has achieved the state-of-the-art results on encoding sequences, we build our model using RNN for both words and the user history. Specifically, we leverage two GRUs (Cho et al., 2014) instead of LSTM (Hochreiter and Schmidhuber, 1997) because the former has fewer parameters. The first GRU, denoted as W-RNN in Figure 1 for modeling **W**ords, encodes events and comments into fixed-length vectors, i.e, $[\mathbf{h}_0^{a,u_k}, \ldots, \mathbf{h}_n^{a,u_k}]$, $[\mathbf{h}_0^{c,u_k}, \ldots, \mathbf{h}_n^{c,u_k}]$, and $\mathbf{h}_T^{a,u_k}$. The second GRU, denoted as S-RNN in Figure 1 for modeling the **S**equence of history, takes as the input the concatenation of prior events and comments, i.e, $[[\mathbf{h}_0^{a,u_k}; \mathbf{h}_0^{c,u_k}], \ldots, [\mathbf{h}_n^{a,u_k}; \mathbf{h}_n^{c,u_k}]]$ where $[;]$ means concatenation, and outputs the user's fingerprint embedding $\mathbf{h}^{f,u_k}$. Finally, we give the concatenation of the fingerprinting embedding and the current event embedding, i.e., $[\mathbf{h}^{f,u_k}; \mathbf{h}_T^{a,u_k}]$, to a one layer feed-forward network, which is denoted as MLP in Figure 1. This MLP outputs the final prediction with a softmax activation function.

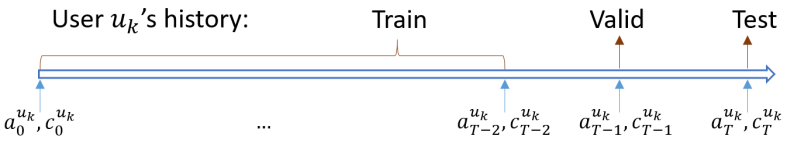

Figure 2: Data split

3 Experiments

We describe the experimental setup in this section. We first explain how we process the dataset. Then we detail the baselines and present the results.

3.1 Data Preparation

News articles were randomly collected from AchiveIs, DailyMail, FoxNews, NewYorkTimes, TheGuadian, and WSJ. We remove users if they have less than ten comments and then we remove articles that the remaining users have not commented. We have manually checked a subset of articles and their comments, and find that irrelevant comments are few enough to ignore. We split the training, validation, and test as in Figure 2. Suppose in the past T time user u_k contributed a sequence of article-comment pairs, we use the most recent article $a_T^{u_k}$ as the unseen article for test and $a_{T-1}^{u_k}$ as the unseen article for validation, so that their corresponding comments (opinion) are not viewed during the training. For the training set $[a_1, c_1, \ldots, a_{T-2}, c_{T-2}]$, each article is used as the unseen article to form a training instance. Given the user and an unseen article, we include previous m article-comment pairs as the historical data to model the user fingerprint, and we assume that users have consistent views and stance on the same event within these m pairs . Thus, the numbers of examples in the test set and the validation set are equal to the user number U, and the number of training examples is equal to $U * (T - 2)$.

The opinion expressed in the comments are hard for evaluation, so we quantify the concept of opinion by sentiment polarity and subjectivity. Given the volume of our dataset, it would be inefficient to conduct manual annotation. We apply four methods, including Vader (Hutto and Gilbert, 2014), Flair (Akbik et al., 2019), BlobText sentiment, and BlobText subjectivity (Loria et al., 2014), to automatically label all comments. **Vader** is a rule-based model for general sentiment analysis. It is constructed from a generalizable, valence-based, human-curated gold standard sentiment lexicon. When assessing the sentiment of tweets, *Vader outperforms individual human raters* (Hutto and Gilbert, 2014). **Flair** presents a unified interface for all word embeddings and supports methods for producing vector representation of entire documents. We use the Flair pretrained classification model for sentiment labels. The model is trained on the IMDB dataset and has 90.54 micro F1 score. Flair predicts either *positive* or *negative*. **BlobText** is a simple rule-based API for sentiment analysis. It has both sentiment model and subjectivity model, and we refer them Bsent, Bsubj in Table 3, respectively. We cast the real value prediction to categorical value by comparing with a threshold. For example, Vader predicts the sentiment between -1 and 1, and we take the threshold of zero.

Notably, we do not consider stance prediction because some websites might have a clear political orientation, e.g., the FoxNews favors the Republican Party, and their readers and comments reveal a similar trend. Nevertheless, we do not restrict sentiment and subjectivity in our task. For example, we can also predict one's emotional reaction about unseen articles (Bostan and Klinger, 2018), which we will explore in future shortly. Essentially, we argue that the proposed task requires models to effectively capture the fingerprinting of users based on what they have read and commented, so that we can generalize one's history to predict their opinion on unseen events.

To validate that the automatic annotation gives reasonable labels, we perform standard document classification on these labels with a RNN classifier and report micro F1 in Table 3, which we denoted as **Oracle**. According to the result, labeling with rule-based methods (Vader and BlobText) gives a better performance than pre-trained neural network model Flair. The reason could be that rule-based methods provide less noise: they will label an instance neutral if no sentiment lexicon is found. It is worthwhile to note that the performance of the oracle is consistent across a variety of news sites yielding confidence on the labels that we use to evaluate our model.

	Vader	Flair	Bsent	Bsubj	Vader	Flair	Bsent	Bsubj	Vader	Flair	Bsent	Bsubj
		ArchiveIs				DailyMail				FoxNews		
Oracle	84.04	69.38	87.12	86.28	92.39	75.11	95.17	95.34	90.50	72.60	93.53	93.64
UF	49.87	50.17	45.69	50.53	44.61	51.85	37.21	51.20	46.48	52.51	39.00	50.24
A-tfidf	48.78	50.96	44.69	49.64	42.08	51.55	36.38	49.14	44.05	52.14	38.38	49.59
A-BERT	47.95	50.72	44.68	49.85	41.91	51.12	36.22	49.28	43.90	52.11	37.98	49.54
CF	52.22	52.35	48.93	39.19	46.63	51.89	37.75	41.78	45.15	52.54	39.10	35.98
FPE	52.72	52.28	49.91	52.21	46.93	54.70	41.38	52.07	46.82	53.92	42.09	52.64
		TheGuardian				WSJ				NewYorkTimes		
Oracle	87.18	73.21	90.64	89.86	83.47	68.84	86.07	85.84	85.78	70.73	89.16	89.16
UF	46.88	51.91	43.76	49.48	47.99	50.88	44.22	50.19	49.47	51.05	45.57	50.52
A-tfidf	44.92	52.19	43.82	47.86	46.50	51.66	43.87	49.29	47.93	50.96	45.00	49.63
A-BERT	44.52	51.58	43.22	47.80	46.35	50.69	43.34	49.25	47.34	51.34	44.53	49.53
CF	48.88	53.22	42.83	38.57	49.74	53.03	43.40	42.51	50.36	51.73	44.46	35.17
FPE	49.51	54.85	46.17	50.84	51.09	53.74	45.11	51.67	54.12	53.22	47.68	52.99

Table 3: The results of our experiment. Bsent: BlobText Sentiment; Bsubj: BlobText Subjectivity

3.2 Baselines and Implementation

We evaluate four baselines to compare. **UF**: we retrieve the most frequent opinion from one's history and use it as the opinion of an unseen article. **A-tfidf**: we compare the cosine distance between the unseen article and one's reading history and extract the opinion of the most similar article as the opinion of the unseen article. We use TF-IDF to represent each article after we remove stop words. **A-BERT**: it also compares the similarity between the unseen article and one's reading history, but adopts pre-trained BERT (Devlin et al., 2018) for representation. **CF**: this baseline uses neural collaborative filtering (He et al., 2017), which has been well applied for recommendation system. In our task, we still use GRU to encode the unseen article but replace the fingerprint embedding with a learnable user embedding.

We denote the proposed method **FPE**, short for **F**inger**P**rint **E**mbedding. We implement the model with the Pytorch package. We arbitrarily set each instance to access at most 14 previous article-comment pairs. In all cases, the size of hidden dimension is 256. We use the Adam optimizer (Kingma and Ba, 2015) with a fixed learning rate of 0.001. We apply dropouts with a fixed probability of 0.2. We further apply BPEmb to process documents and utilize the pretrained 300 dimension sub-word embeddings (Heinzerling and Strube, 2018). We train **CF** and **FPE** for 16 epochs and save the model based on the micro F1 score of the validation set. The best model is usually achieved around five epochs. The code is released at: https://github.com/fYYw/fingerprinting.

3.3 Results and Discussion

We report the micro F1 score in Table 3. All the results are acquired by using the best model on the test set. From Table 3, we first observe that all the evaluated methods suffer a large performance drop compared to the oracle setting. This means that the task of predicting the opinion on unseen events is challenging. Note that the "oracle" directly uses responses, so it is not predicting future opinion from past history. We view the oracle as a quality check to ensure labels are correct. How to effectively model the fingerprint of a user and generalize one's history to future event will require more investigation.

Nevertheless, even on such a hard task, the **FPE** model outperforms **UF**, **A-tfidf**, and **A-BERT** on all labels. This is important because automatic labeling may bring noise on individual labels while considering four labeling schemas together makes better sense. The improvement indicates that using RNN can better leverage one's reading and commenting history, and therefore creates the fingerprint of a user more generalized on unseen events than other methods. Because previous works also suggest that the recurrent neural network can effectively track one's sequential actions (Tan et al., 2016; Pei et al., 2017; Zhu et al., 2017; Beutel et al., 2018), we conclude that the recurrent architecture is the reason for improvements. Comparing **FPE** with **CF**, we see that **FPE** consistently outperforms **CF** except a small drop with the Flair score on ArchiveIs. Since the only difference between **FPE** and **CF** is whether or not we encode the user historical data, we conclude that historical event-comment pairs carry valuable information to build one's fingerprint embedding and therefore benefit prediction on unseen articles. We also observe that A-BERT does not perform well, possibly because there is no fine-tuning step on

the pretrained BERT model. Another choice is to replace BERT with Sentence-BERT (Reimers and Gurevych, 2019) for semantic textual similarity, which could improve A-BERT due to a better sentence embeddings.

4 Conclusion

In this work, we introduce a new task that predicts opinion on unseen events based on one's reading and commenting history. We design a recurrent neural network based model to encode the historical data and use the fingerprint embedding to obtain opinion on new articles. Experiments on our newly collected dataset show that leveraging recurrent neural network and one's historical data gives better performance that four used baselines. We believe that the proposed novel problem setting lays the foundation for a variety of more rigorous works to fully explore how to learn and generalize user fingerprints. In the future, we plan to quantify one's comment with more dimensions, such as emotion, and predict them on unseen events. We argue that a successful model would effectively leverage one's fingerprinting, and it is worthwhile to investigate different architectures for this task.

5 Acknowledgement

Research was supported in part by grants NSF 1838147, NSF 1838145, ARO W911NF-20-1-0254. The views and conclusions contained in this document are those of the authors and not of the sponsors. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Alex Beutel, Paul Covington, Sagar Jain, Can Xu, Jia Li, Vince Gatto, and Ed H Chi. 2018. Latent cross: Making use of context in recurrent recommender systems. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 46–54.

Laura-Ana-Maria Bostan and Roman Klinger. 2018. An analysis of annotated corpora for emotion classification in text. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2104–2119, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

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 *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734.

Kushal Dave, Steve Lawrence, and David M Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Cristiano Di Crescenzo, Giulia Gavazzi, Giacomo Legnaro, Elena Troccoli, Ilaria Bordino, and Francesco Gullo. 2017. Hermevent: a news collection for emerging-event detection. In *Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics*, pages 1–10.

Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web*, pages 173–182.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages. In Nicoletta Calzolari (Conference chair), Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12, 2018. European Language Resources Association (ELRA).

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Minqing Hu and Bing Liu. 2004. Mining opinion features in customer reviews. In *AAAI*, volume 4, pages 755–760.

Clayton J Hutto and Eric Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Eighth international AAAI conference on weblogs and social media*.

Armand Joulin, Édouard Grave, Piotr Bojanowski, and Tomáš Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, San Diego, CA, USA.

Bing Liu. 2010. Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2(2010):627–666.

Bing Liu. 2012. Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1):1–167.

Steven Loria, P Keen, M Honnibal, R Yankovsky, D Karesh, E Dempsey, et al. 2014. Textblob: simplified text processing. *Secondary TextBlob: simplified text processing*, 3.

Sendhil Mullainathan and Andrei Shleifer. 2005. The market for news. *American Economic Review*, 95(4):1031–1053.

Alexander Pak and Patrick Paroubek. 2010. Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10, pages 1320–1326.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135.

Wenjie Pei, Jie Yang, Zhu Sun, Jie Zhang, Alessandro Bozzon, and David MJ Tax. 2017. Interacting attention-gated recurrent networks for recommendation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1459–1468.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November. Association for Computational Linguistics.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307.

Yong Kiam Tan, Xinxing Xu, and Yong Liu. 2016. Improved recurrent neural networks for session-based recommendations. In *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems*, pages 17–22.

Yi Xiang and Miklos Sarvary. 2007. News consumption and media bias. *Marketing Science*, 26(5):611–628.

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*, pages 1480–1489.

Jingbo Zhu, Huizhen Wang, Muhua Zhu, Benjamin K Tsou, and Matthew Ma. 2011. Aspect-based opinion polling from customer reviews. *IEEE Transactions on affective computing*, 2(1):37–49.

Yu Zhu, Hao Li, Yikang Liao, Beidou Wang, Ziyu Guan, Haifeng Liu, and Deng Cai. 2017. What to do next: modeling user behaviors by time-lstm. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 3602–3608.