Generating Clinically Relevant Texts: A Case Study on Life-changing Events

Mayuresh Oak[1], **Anil K. Behera**[3], **Titus P. Thomas**[1], **Cecilia Ovesdotter Alm**[2], **Emily Prud'hommeaux**[2], **Christopher Homan**[3], **Raymond Ptucha**[1]
[1]Kate Gleason College of Engineering
[2]College of Liberal Arts
[3]Golisano College of Computing and Information Sciences
Rochester Institute of Technology
Rochester, NY 14623, USA
{mso4106†|akb2701†|tpt7797†|coagla†|emilypx†|cmh§|rwpeec†
†@rit.edu §@cs.rit.edu

Abstract

The need to protect privacy poses unique challenges to behavioral research. For instance, researchers often can not use examples drawn directly from such data to explain or illustrate key findings. In this research, we use data-driven models to synthesize realistic-looking data, focusing on discourse produced by social-media participants announcing life-changing events. We comparatively explore the performance of distinct techniques for generating synthetic linguistic data across different linguistic units and topics. Our approach offers utility not only for reporting on qualitative behavioral research on such data, where directly quoting a participant's content can unintentionally reveal sensitive information about the participant, but also for clinical computational system developers, for whom access to realistic synthetic data may be sufficient for the software development process. Accordingly, the work also has implications for computational linguistics at large.

1 Introduction

Behavioral research using personal data, such as that from social media or clinical studies, must continually balance insights gained with respect for privacy. Ethical and legal demands also come into play. De-identification involves removing information such as named entities, address-specific information and social security numbers. However, naive approaches are often prone to privacy attacks. Such de-identified data will often still contain information that, when combined with other data from different resources, can point to the individual who generated it. For example, if a de-identified dataset contains detailed demographic information, it could then be possible to extract a small list of people matching this information and to identify a specific person using other, publicly available data.

One approach that strikes a good balance is to synthesize realistic-looking data with the same statistical properties as actual data. Our contribution is to compare different techniques for synthesizing behavioral data. Specifically, we explore this problem in a case study with social media texts that involve social media participants making announcements about life-changing events, which are personal in nature and which also may affect, positively or negatively, a person's well-being.

Two immediate applications to clinical research that motivate this approach are: *qualitative results reporting involving textual data* and *data access issues for software development purposes*. Neither readers of scientific reports nor software developers need access to the original data as long as realistic looking synthetic data is available.

2 Related Work

In the clinical setting, data privacy is important. *Anonymization* aims to ensure that data is untraceable to an original user, whereas *de-identification* may allow the data to be traced back to a user with third-party information.

Szarvas et al. (2007) developed a model for anonymizing personal health information (PHI) from discharge records. The model identifies PHIs

Proceedings of the 3rd Workshop on Computational Linguistics and Clinical Psychology: From Linguistic Signal to Clinical Reality, pages 85–94,
San Diego, California, June 16, 2016.

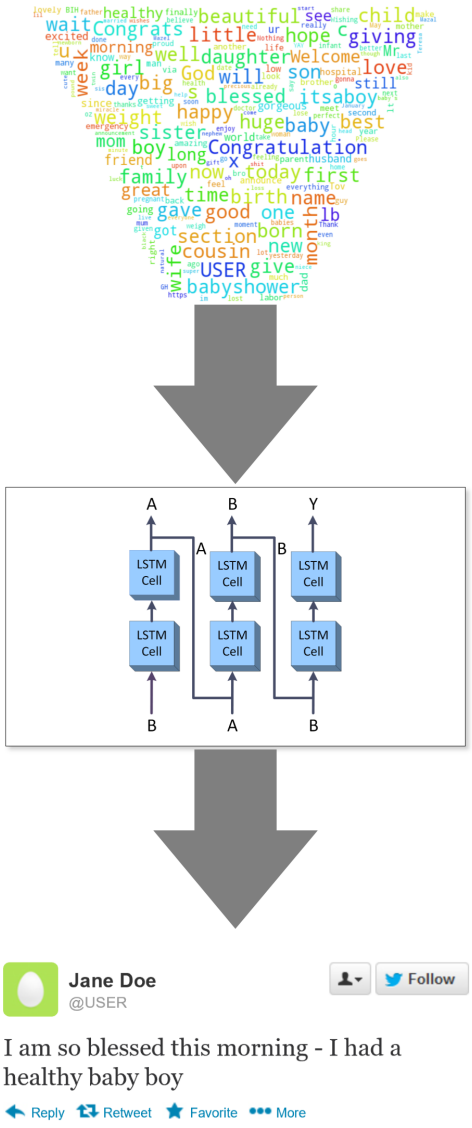

Figure 1: Top level view of the proposed anonymization system. Data is fed to a model which here is a character-based Long-Short Term Memory (LSTM). The LSTM generates new tweets based on the input data.

in several steps and labels all entities which can be tagged from the text structure. It then queries for additional PHI phrases in the text with help from tagged PHI entities.

Bayardo and Agrawal (2005) present improved k-anonymity methods and provide efficient algorithms for data dimensionality reduction. However, even if information such as names of people or providers or quasi-identifiers (QIs) are removed, there are still ways to compare the de-identified data with other records having these identifiers.

In contrast to traditional anonymization and de-identification methods, generation of synthetic data can handle various aspects of hiding individuals, by aggregating and severing data from individual users, yet maintaining the statistical properties of the data used to train generation models. For this paper we explore several forms of data generation, using social media (Twitter) data about life-changing events as a case study. For example, Twitter data has been used for studying important life-changing events (De Choudhry et al., 2013; Li et al., 2014). Other studies present methods for anonymizing Twitter datasets. Terrovitis et al. (2008) model social media as an undirected, unlabeled graph which does retain privacy of social media users. Daubert et al. (2014) discuss the different methods for anonymization of Twitter data. However, there is a lack of work that addresses synthetic data creation using machine generation models.

This paper compares traditional statistical language models and Long Short Term Memory (LSTM) models to learn models from a training set of Twitter data to generate synthetic tweets. LSTMs are recurrent neural networks designed to learn both long and short term temporal sequences. These networks were introduced by Hochreiter and Schmidhuber (1997), with several improvements over the years, the most common of which include individual gating elements (Graves and Schmidhuber, 2005). LSTMs have been shown to perform at state-of-the-art levels for many tasks, including handwriting recognition and generation, language modeling, and machine translation (Greff et al., 2015).

3 Data

Twitter is a microblogging platform used by people to post about their lives. If harnessed properly, tweets can be used for analysis and research of behavioral patterns as well as in studying health information.

We collected tweets using Twitter's streaming API along with customized query strings. These queries targeted the life-changing events of *birth*, *death*, *marriage*, and *divorce*. The tweet collection process suggested that users were more likely to share joyful news about marriage and birth, and

Table 1: Birth patterns

birth of baby/brother/son/daughter/brother/sister
parents of baby/son/daughter/boy/girl/angel
arrival of baby/brother/son/daughter/sister/angel
just gave birth to baby/son/daughter/boy/girl
weigh/weighing #Number lbs/pounds
its a boy/girl
pregnant/c-section

Table 2: Marriage patterns

I'm/we are getting/sister/brother/mother married
friend/uncle/aunt is getting married
I/we/sister/brother/friend/uncle/aunt got married

Table 3: Death patterns

RIP mom/mama/dad/father/grandmother/brother/
RIP grandpa/grandfather/sister/friend
he/mom/mama/dad/father passed away
grandfather/grandpa/grandma passed away
brother/sister/friend passed away

less likely to share difficult news about death and divorce. Tweets on divorce were particularly scarce, so this event was ignored as the study continued.

The pool of tweets came from a collection of tweets from a mid-sized city in the US North East in 2013 as well as streaming tweets irrespective of location from early 2016. Roughly 18 million tweets were collected, including tweets for the three aforementioned categories of birth, death, and marriage. Only the text of the tweets was utilized for this study.

After inspecting the data, we formulated a set of lexical keywords, phrases and regular expressions to collect tweets by category. These reflected topical patterns, such as announcements of marriage or birth in the family, the weight of the newborn baby or whether it is a girl or a boy, or the passing of a friend or family member. Table 1 shows the patterns used to extract tweets about *birth*. Similarly, Table 2 shows the patterns for *marriage*, and Table 3 for *death*. We attempted to remove tweets about celebrities, TV shows, news stories, and jokes. After filtering, we selected and hand-annotated for each category a set of 2000 tweets. For comparison's sake we also chose randomly 2000 (unlabeled) tweets from the data, and call this the *general* category. Note that any tweet could be present in this category, including those from the first three categories.

We replaced Twitter usernames with the token *@USER*, while URL links, retweets, and emoticons were replaced with the keywords *URL*, *RT*, and *EMOT*, respectively. We removed the pound signs from hashtags to make it look more like general written language and to reduce the dictionary size of the word-based language models.

For the character-based models, we performed the following further steps. We separated each character in the input data by a space and replaced the usual space characters with <*space*>. We considered the tags introduced in the earlier pre-processing phase (e.g. - *@USER*) to be unique characters. On output, we replaced all space characters with the null string and replace the space tag <*space*> with the space character.

Tables 4 through 6 show samples of collected tweets.

Table 4: Birth tweets

She gave birth to the baby aww congrats loulou @USER
birth of Baby Tyler (They picked my baby name suggestion)

Table 5: Death tweets

my grandpa passed away today All I hope is that things get better
@USER my grandma passed away

Table 6: Marriage tweets

me and @USER just got married
we getting married

4 Methods

4.1 Long-Short Term Memory

Recurrent neural networks (RNN) are popular models that have shown great potential in many natural language processing (NLP) tasks. LSTMs (Hochreiter and Schmidhuber, 1997; Graves and Schmidhuber, 2005) are a specific subset of RNNs that have been modified to be especially good at conditioning on both long and short term temporal sequences. LSTMs modify the standard design of neural networks in several ways: they eliminate the strict requirement that neurons only connect to other neurons in succeeding layers (adding recurrence), convert the standard neuron into a more complex *memory cell*, and add non-linear gating units which serve to govern the information flowing out of and recursively flowing back into the cell (Greff et al., 2015). The memory cell differentiates itself from a simple neuron by including the ability to remember its state over time; this coupled with gating units gives the LSTM the ability to recognize important long-term dependencies while simultaneously forgetting unimportant collocations.

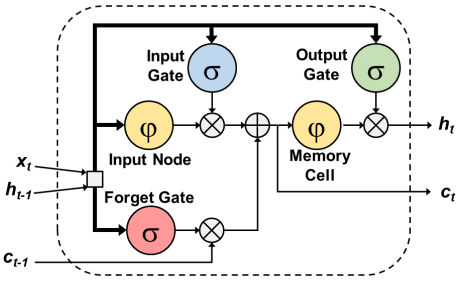

Figure 2: A single LSTM memory block. The three gates govern the input node and memory cell to allow long term memory. The function φ is the *tanh* function and the function σ is the *sigmoid* function.

The LSTM we use here, as implemented by Karpathy (2015) modifies the original architecture by removing *peephole connections*. The intuitive understanding of the components in an LSTM memory block can be summarized as:

1. **Input node**: Also known as input modulation gate or new memory gate, takes the input and the past hidden state to summarize the new input in light of the past context from h_{t-1}.

2. **Input gate**: Also known as write gate, takes the input and the past hidden state to determine the importance of the current input as it effects the cell.

3. **Forget gate**: Also known as reset gate, takes the input and the past hidden state and gives the provision for the hidden layer to discard or forget the historical data.

4. **Output gate**: Takes the input and the past hidden state and determines what parts of the cell output c_t need to be present in the new hidden state h_t for the next timestep.

5. **Memory cell**: Takes advice from the forget gate and governed Input Node to determine the usefulness of the previous memory c_{t-1} to produce the new memory c_t.

The functionality above describes only how a *single* LSTM memory block works, analogous to a single neuron in a regular neural network. To create an LSTM which learns, hundreds of these blocks are combined in a single layer (analogous to hundreds of nodes in a hidden layer), with the hidden output, h_t,c_t of one block feeding into the input of another. Further complexity (and learning power) is added by including multiple layers of LSTM memory blocks. The final output of LSTM memory blocks (or inputs from one layer to the next) are provided by calculating $y_t = W_y f(h_t)$, where W_y is an output weight matrix to learn and $f(\cdot)$ is an activation function which can vary depending on use case.

The input, x_t, to an LSTM memory block differs depending on implementation and use case. When using LSTMs for NLP, the input can be word or character-based. The LSTM used in this research (Karpathy, 2015), takes as input a vector representing an individual data item (character/word) and predicts the most probable data item given the current data item and the LSTM's previous states. Training, therefore, is done by taking an example sequence of data items, predicting the next data item using the current weights, calculating the difference between what was predicted and what should have been predicted, and back propagating this difference to up-

date the weights. All LSTM models were trained for 500 epochs and sequence length of 50, where the sequence length is the length of time the LSTM cell is unrolled per iteration. Two LSTM layers were used to train the model on the input data. Each LSTM layer had 512 hidden nodes. Language generation can be performed after training, in which the LSTM is given either a starting sequence of data items (or it calculates the most probable sequence to start with), and then generates new data items based on its own predictions in previous time steps.

4.2 Standard N-gram Language Models

In order to demonstrate the particular utility of LSTMs for generating realistic tweets, the output of our character- and word-based LSTM methods was compared to that of standard n-gram backoff language models. Such models are widely used to model the probability of word sequences for many NLP applications, including machine translation, automatic speech recognition, and part-of-speech tagging. The SRI Language Modeling Toolkit (SRILM) was used to build 4-gram word- and character-based language models (Stolcke, 2002). Using these models, we then generate synthetic tweets using the OpenGRM Ngram library (Roark et al., 2012).

4.3 Experimental Design

For each event category, we divided the dataset of 2000 tweets into 1800 training and 200 testing instances. We used the machine translation quality metric BLEU (Papineni et al., 2002) to measure the similarity between machine generated tweets and the held out tests sets. For each model, we generated ten sets of 200 tweets. We calculated BLEU scores (without the brevity penalty) using the full 200-tweet test set as the reference for each candidate tweet and report the average of the BLEU scores of all ten sets of tweets generated by a given model.

To gain further insight into the effectiveness of the machine generated data, we asked human annotators to evaluate the generated tweets. We selected 800 tweets by randomly sampling: 400 human generated tweets (100 from each category), and 400 machine generated tweets. The 400 machine generated tweets consisted of 25 tweets for each combination of model (LM-char, LM-word, LSTM-char, LSTM-word) and category (*birth*, *marriage*, *death*, *general*). For each tweet, the annotators indicated if they thought the tweet was generated by a human or machine, and they rated the quality of the tweet on the basis of syntax and semantics. Also, they indicated which topic category they thought the tweet belonged to.

Table 7: Mean BLEU scores and their standard deviation over ten generated test sets of 200 tweets per model, by topic, model, and linguistic unit.

Topic	Model		BLEU
Birth	LSTM	char	**34.61** ±2.53
		word	32.36 ±2.21
	LM	char	12.15 ±0.63
		word	32.01±0.96
Marriage	LSTM	char	31.14±2.30
		word	26.22±0.77
	LM	char	12.54±1.08
		word	**32.26**±0.96
Death	LSTM	char	**20.16**±2.78
		word	17.84±9.45
	LM	char	6.04±0.62
		word	16.93±0.67
General	LSTM	char	40.55±4.27
		word	17.62±2.17
	LM	char	5.46±0.60
		word	**44.74**±1.33

5 Results

BLEU, a measure of n-gram precision widely used to evaluated machine translation output, was used to objectively evaluate the similarity between the human-generated tweets and the synthetic tweets produced by our models. Table 7 shows the BLEU scores for each combination of topic, model, and linguistic unit. The character-based LSTM models and the word-based LM models both perform very strongly, with each reporting the highest BLEU score in two of the four topics. We further note that the character-based LSTM always outperforms the word-based LSTM. Although it might be surprising that a character-based model would produce higher values for a word n-gram precision metric such as

Table 8: The percent of instances where the four human annotators (A1 - A4) were deceived into thinking a synthetic tweet was human generated. The values in bold are the best performing models for each category by annotator. (*B* = *birth*, *D* = *death* , *M* = *marriage* , *B* = *general*).

Model	**A1**				**A2**				**A3**				**A4**			
	B	***D***	***M***	***G***	***B***	***D***	***M***	***G***	***B***	***D***	***M***	***G***	***B***	***D***	***M***	***G***
LM-char	14	0	0	0	14	0	20	0	40	52	28	16	16	8	8	12
LM-word	18	8	21	10	18	8	50	**40**	**80**	**80**	**88**	**72**	32	32	**48**	**44**
LSTM-char	**45**	25	**44**	0	36	**25**	**56**	22	60	72	68	64	**40**	**44**	40	**44**
LSTM-word	33	**38**	30	**11**	**44**	**25**	40	11	76	72	48	44	36	36	24	28

BLEU, we suspect this is due to the fact that the large feature space of the word-based model in combination with the relatively small number of training tweets (roughly 1800) is not optimal for learning an LSTM model.

5.1 Human evaluation

A randomized set of 800 tweets, both real and synthetic, from all four topic categories was submitted to a panel of annotators (co-authors). Each annotator was asked to decide whether the tweet was real

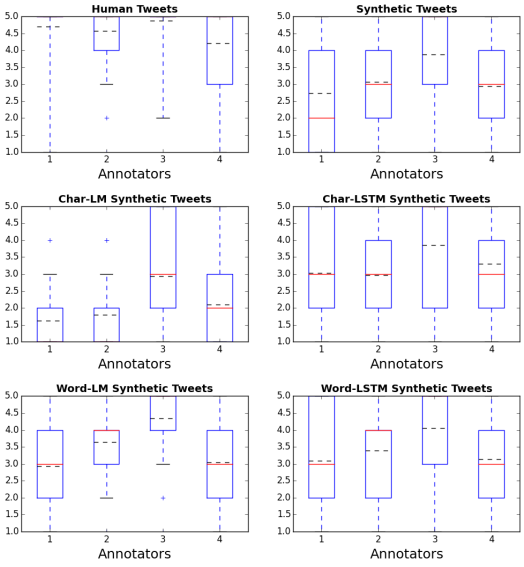

Figure 3: Syntax score by annotator. Higher scores suggest more satisfactory generation of syntactic structures (median = red line, mean = dashed line).

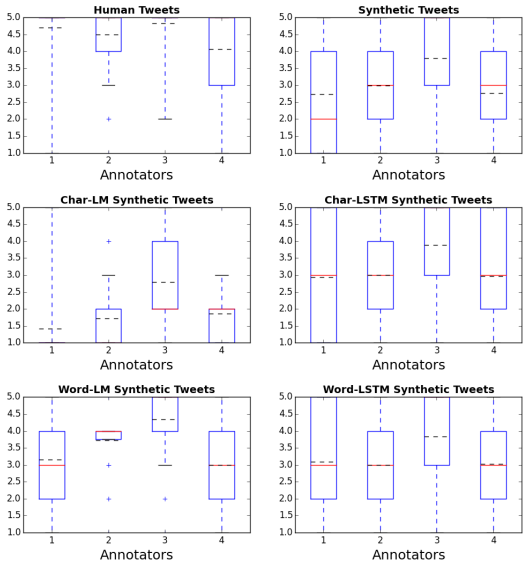

Figure 4: Semantics score by annotator. Higher scores suggest more satisfactory generation of semantic contents (median = red line, mean = dashed line).

(i.e., produced by a human) or synthetic (i.e., generated by one of the LSTM or n-gram language models). Each tweet was also rated in terms of its syntax and semantics on a five point Likert scale. In addition, the annotators were asked to select the intended topic category (*birth*, *death*, *marriage*, or *general*) of the tweet.

Figure 5 shows the ability of human annotators to accurately identify a tweet's topic. In general, the annotators were able to identify the topic of the human tweets, with the weakest performance in the *general* category. Identifying the intended topic of the synthetic tweets was more challenging for the

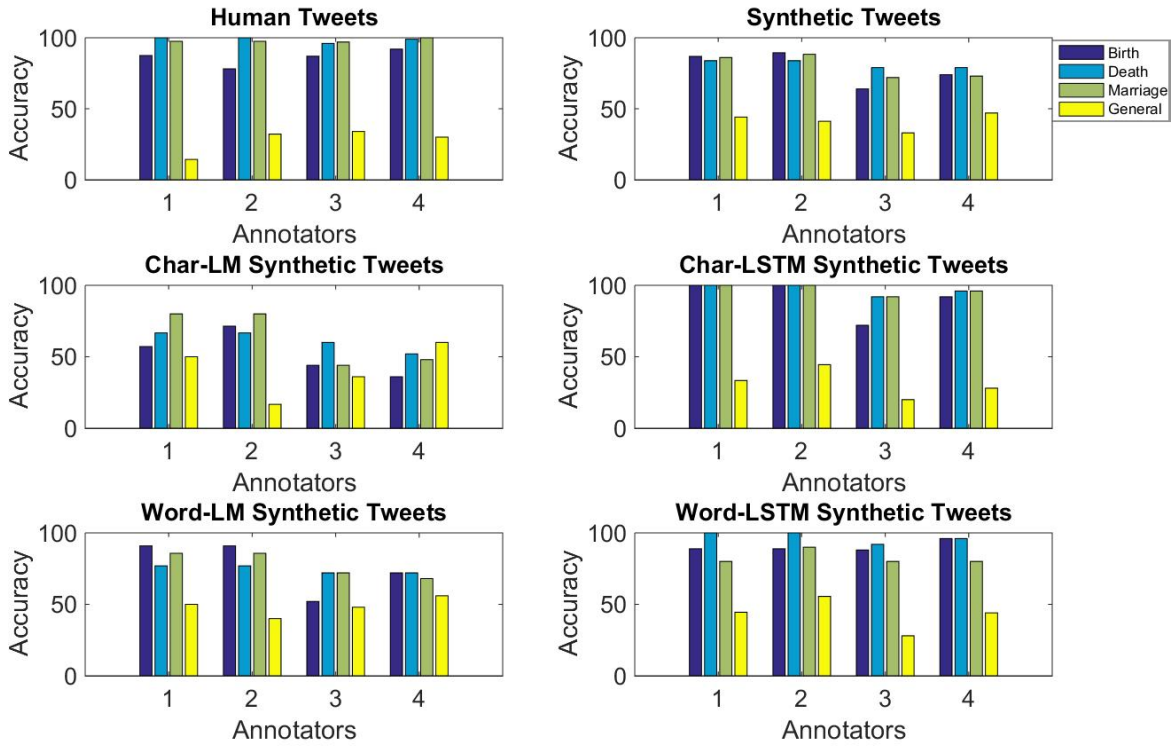

Figure 5: Ability of four human annotators (A1 - A4) to predict the topic category of the data from which a tweet was generated, per model. The top left panel reflects results for human-composed tweets, whereas the top right panel shows results across synthetic tweets, corresponding to the four models in subsequent panels. (Dark blue = *birth* , light blue = *death* , green = *marriage* , yellow = *general*).

annotators, but accuracy was quite high in all topics other than *general*. We note that the *general* category was not filtered to remove tweets that could have belonged to the other topics, which could explain this discrepancy.

Figures 3 and 4 show the distribution of each annotator's syntax and semantics scores for each model. These boxplots show that there was significant variance in the annotators' evaluation of the syntactic and semantic quality of the tweets. We note, however, that the models yielding the highest BLEU scores, char-LSTM and word-LM, tended to receive more favorable scores for syntactic and semantic quality. The character-based LM model, whose BLEU scores were significantly lower than other models, consistently received the most unsatisfactory evaluation of syntactic and semantic quality by all four annotators. It also seems that the LSTM models produce output that is more consistent in its semantic and syntactic quality, with smaller annotator to annotator variance than the LM models.

With regard to Figure 5, Annotators 1 and 2 rated 283 (selected randomly) tweets, while Annotators 3 and 4 rated all 800 tweets; and with regard to Figures 3, 4, and Table 8, all annotators rated 283 tweets. Annotators 1 and 2 have an academic background in linguistics, while the other two annotators do not have prior linguistic training, perhaps explaining why annotators 1 and 2 generally were better able to identify the topic category. Annotators 1 and 2 tended to have similar distributions of semantic and syntactic quality scores across models, which again is likely related to their previous training in linguistics and linguistic annotation. Annotator 4 may have been less forgiving about non-standard language use in the human-composed tweets, while annotator 3 was more tolerant of the syntax and semantics of machine-generated tweets.

Table 9: Synthetic tweets marked as human generated by all four annotators.

Congrats to @USER and her husband on the birth of their son Welcome to the Cyclone family, Eally Kinglan URL URL (***Char LSTM Generated***)
@USER congratulations on birth of your son,20 days,ago,URL (***Word LM Generated***)
@USER @USER @USER,looks like we're getting hitched in June URL (***Word LM Generated***)
Im getting married in 17 days death (***Char LSTM Generated***)
RT @USER rip grandma 2 8 16 (***Word LM Generated***)

Table 10: Synthetic tweets marked as synthetic by all four annotators.

RT @USER The new part prigials give birth to bely son Junt and I'm delined a hape proud (***Char LSTM Generated***)
I'm so sorry for your loss and world harry gotting to my funeral it was without URL (***Word LM Generated***)

Table 8 shows the percent of instances a human annotator marked a synthetic tweet as human generated. Table 9 shows some of the tweets that were generated by language models but were identified by all four annotators as human generated. A few example tweets that were correctly identified by all four annotators as synthetic tweets are displayed in Table 10.

6 Conclusion

We have discussed generating synthetic data in the context of readers of scientific reports or software developers. In addition, one potential clinical application might be to apply this to patient transcripts so that they could be shown to other patients suffering from similar problems, e.g., for anonymized virtual group therapy. Such an approach might be especially useful in rural and developing regions, where clinical resources are sparse. Anonymization of data in research is often necessary to protect patient or user identity. This research explores data-driven models to generate realistic-looking discourse with the same statistical properties as a training corpus. Specifically, this research explores the synthetic generation of tweets, contrasting LM and LSTM models, character-based and word-based linguistic units, and the topic categories of birth, death, and marriage. Based on the results from objective BLEU scores and subjective human evaluation, the word-based LM and char-based LSTM models performed well, deceiving annotators 41 and 43 percent of the time on average into thinking a synthetic tweet was human generated. This research shows promising evidence that the synthetic generation of user data may be preferred to existing techniques of naive anonymization which can potentially lead to user identification through combination of demographic data mining and ancillary metadata.

References

Puneet Agarwal, Rajgopal Vaithiyanathan, Saurabh Sharma, and Gautam Shroff. 2012. {Catching the Long-Tail: Extracting Local News Events from Twitter}. *Proceedings of the Sixth International AAAI Conference on Weblogs and Social Media*.

Roberto J Bayardo and Rakesh Agrawal. 2005. Data privacy through optimal k-anonymization. In *Data Engineering, 2005. ICDE 2005. Proceedings. 21st International Conference on*, pages 217–228. IEEE.

Edward Benson, Aria Haghighi, and Regina Barzilay. 2011. Event discovery in social media feeds. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 389–398. Association for Computational Linguistics.

Deepayan Chakrabarti and Kunal Punera. 2011. Event Summarization Using Tweets. *ICWSM*, 11:66–73.

J. Daubert, L. Bock, P. Kikirasy, M. Muhlhauser, and M. Fischer. 2014. Twitterize: Anonymous Microblogging. In *Computer Systems and Applications (AICCSA), 2014 IEEE/ACS 11th International Conference on*, pages 817–823, Nov.

Munmun De Choudhry, Scott Counts, and Eric Horvitz. 2013. Major life changes and behavioral markers in social media: Case of childbirth. In *Proceedings of the 16th ACM Conference on Computer Supported Cooperative Work (San Antonio, TX, USA, Feb 23-27, 2013). CSCW 2013.* ACM.

Munmun De Choudhury, Scott Counts, and Eric Horvitz. 2013. Major life changes and behavioral markers in social media: case of childbirth. In *Proceedings of the*

2013 conference on Computer supported cooperative work, pages 1431–1442. ACM.

Barbara Di Eugenio, Nick Green, and Rajen Subba. 2013. Detecting life events in feeds from twitter. In *2013 IEEE Seventh International Conference on Semantic Computing*, pages 274–277. Ieee.

Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634.

Lilian Edwards and Andrea M Matwyshyn. 2013. Twitter (R) evolution: privacy, free speech and disclosure. In *Proceedings of the 22nd international conference on World Wide Web companion*, pages 745–750. International World Wide Web Conferences Steering Committee.

Rumi Ghosh, Tawan Surachawala, and Kristina Lerman. 2011. Entropy-based classification of'retweeting'activity on twitter. *arXiv preprint arXiv:1106.0346*.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, 18(5):602–610.

Klaus Greff, Rupesh Kumar Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. 2015. LSTM: A Search Space Odyssey. *arXiv preprint arXiv:1503.04069*.

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

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137.

Andrej Karpathy. 2015. Char-RNN: Multi-layer recurrent neural networks (LSTM, GRU, RNN) for character-level language models in torch. `https://github.com/karpathy/char-rnn`. Accessed: 2015-07-17.

Ninghui Li, Tiancheng Li, and Suresh Venkatasubramanian. 2007. t-closeness: Privacy beyond k-anonymity and l-diversity. In *Data Engineering, 2007. ICDE 2007. IEEE 23rd International Conference on*, pages 106–115. IEEE.

Jiwei Li, Alan Ritter, Claire Cardie, and Eduard H Hovy. 2014. Major Life Event Extraction from Twitter based on Congratulations/Condolences Speech Acts. In *EMNLP*, pages 1997–2007.

Shirin Nilizadeh, Apu Kapadia, and Yong-Yeol Ahn. 2014. Community-enhanced de-anonymization of online social networks. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*, pages 537–548. ACM.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Rene Pickhardt, Thomas Gottron, Martin Körner, Paul Georg Wagner, Till Speicher, and Steffen Staab. 2014. A generalized language model as the combination of skipped n-grams and modified kneser-ney smoothing. *arXiv preprint arXiv:1404.3377*.

Jacob Ratkiewicz, Michael Conover, Mark Meiss, Bruno Gonçalves, Alessandro Flammini, and Filippo Menczer. 2011. Detecting and Tracking Political Abuse in Social Media. In *ICWSM*.

Brian Roark, Richard Sproat, Cyril Allauzen, Michael Riley, Jeffrey Sorensen, and Terry Tai. 2012. The OpenGrm open-source finite-state grammar software libraries. In *Proceedings of the ACL 2012 System Demonstrations*.

Takeshi Sakaki, Makoto Okazaki, and Yutaka Matsuo. 2010. Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 851–860, New York, NY, USA. ACM.

Pierangela Samarati and Latanya Sweeney. 1998. Generalizing data to provide anonymity when disclosing information. In *PODS*, volume 98, page 188.

Priya Sidhaye and Jackie Chi Kit Cheung. 2015. Indicative Tweet Generation: An Extractive Summarization Problem? *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 138–147.

Amardeep Singh, Divya Bansal, and Sanjeev Sofat. 2014. An approach of privacy preserving based publishing in twitter. In *Proceedings of the 7th International Conference on Security of Information and Networks*, page 39. ACM.

Richard Socher, Milad Mohammadi, and Rohit Mundra. Spring 2015. Cs 224d: Deep learning for NLP. http://cs224d.stanford.edu/lecture_notes/LectureNotes4.pdf.

Andreas Stolcke. 2002. SRILM – An Extensible Language Modeling Toolkit. In *Proceedings of the International Conference on Spoken Language Processing*, volume 2, pages 901–904.

György Szarvas, Richárd Farkas, and Róbert Busa-Fekete. 2007. State-of-the-art anonymization of med-

ical records using an iterative machine learning framework. *Journal of the American Medical Informatics Association*, 14(5):574–580.

Manolis Terrovitis, Nikos Mamoulis, and Panos Kalnis. 2008. Privacy-preserving anonymization of set-valued data. *Proceedings of the VLDB Endowment*, 1(1):115–125.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrksic, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. *arXiv preprint arXiv:1508.01745*.

Jian Xu, Wei Wang, Jian Pei, Xiaoyuan Wang, Baile Shi, and Ada Wai-Chee Fu. 2006. Utility-based Anonymization Using Local Recoding. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, pages 785–790, New York, NY, USA. ACM.