2016 IEEE/WIC/ACM International Conference on Web Intelligence

"Best Dinner Ever!!!": Automatic Generation of Restaurant Reviews with LSTM-RNN

Alberto Bartoli*, Andrea De Lorenzo*, Eric Medvet*, Dennis Morello*, Fabiano Tarlao*
*Department of Engineering and Architecture, University of Trieste, Trieste, Italy

***Abstract*—Consumer reviews are an important information resource for people and a fundamental part of everyday decision-making. Product reviews have an economical relevance which may attract malicious people to commit a review fraud, by writing false reviews. In this work, we investigate the possibility of generating hundreds of false restaurant reviews automatically and very quickly.**

We propose and evaluate a method for automatic generation of restaurant reviews tailored to the desired rating and restaurant category. A key feature of our work is the experimental evaluation which involves human users. We assessed the ability of our method to actually deceive users by presenting to them sets of reviews including a mix of genuine reviews and of machine-generated reviews. Users were not aware of the aim of the evaluation and the existence of machine-generated reviews. As it turns out, it is feasible to automatically generate realistic reviews which can manipulate the opinion of the user.

I. Introduction

Online product reviews play a crucial role in both the electronic and conventional commerce [1]. Many websites and user forums allow online communities to share their experience about products, touristic destinations, cultural offerings, and so on. Such information may be very useful to both users interested in a certain item and sellers interested in increasing their revenue. Since users tend to trust the opinion of other users, online reviews strongly influence decisions.

In this scenario, the opinion of a user can be biased by malicious sellers who try to gain unfair competitive advantages for their products, by disseminating either fake positive reviews for their products, or fake negative reviews for the products of their competitors. This phenomenon, called *opinion spamming*, is well known by web-oriented business companies which forbid or strongly discourage such practice. Despite being forbidden, the economic returns potentially involved in committing review fraud can be so high to motivate users in devoting time and resources for praising or discrediting a specific target. It is clear that a tool capable of automatically generating a large number of false and diverse reviews with the desired bias may be potentially disruptive, as it might allow manipulating the opinions of consumers on a large scale. Although the services hosting product reviews do apply filters and procedures aimed at limiting the proliferation of false reviews, an attacker able to generate thousands of fake reviews quickly and cheaply could be able to generate a sufficient amount of reviews which slip through the sanity checks. Such reviews could suffice to manipulate the opinion of at least a fraction of the interested users and, more broadly, could undermine the confidence in the overall ecosystem of online product reviews. In this work, we aim at investigating the feasibility of a tool of this sort. We focus only on the actual content of a review. Systems which attempt to identify non genuine reviews usually consider also ancillary information such as, e.g., number and temporal distribution of reviews submitted from the same user or IP address. These features are beyond the scope of this work.

The contribution of our work is two-fold: (i) we propose a method for generating a review, given a restaurant category and a rating; (ii) we perform an experimental campaign involving human users in which we evaluate the impact of our automatically generated deceptive reviews when mixed with genuine reviews.

Our method is based on a *Long Short-Term Memory based Recurrent Neural Network (LSTM-RNN)*. We train the network with a set of genuine reviews in order to obtain a tool capable of generating text which looks like a restaurant review. Then, in order to tailor the review to the desired rating and category, we use a set of classifiers (also previously trained with genuine reviews) in order to pick from the text generated by the network only the portions which matches the desired rating and category.

The experimental campaign is performed on a cohort of 39 users, who were not aware of the fact that they were dealing with automatically-generated reviews. We performed an *extrinsic* evaluation aimed at assessing the impact on the decision about whether to go to a specific restaurant, and an *intrinsic* evaluation aimed at assessing the ability of generating a review which looks like as a review generated by an human author.

II. Related work

Methods for *Natural Language Generation* (NLG) are widely used in spoken dialogue systems [2], machine translation [3], and image caption generation [4]. We are not aware of any proposal for automatic generation of product reviews.

Artificial Neural Networks (ANN) are largely used in the field of NLG. The first ANN-based approach to NLG is the system presented in [5], which implements a stock reporter system where text generation is done at phrase level. A recent work [6] has shown the effectiveness of *Recurrent Neural Networks* (RNN) for NLG at character level. A key aspect of character-level generation with RNN is the ability of these models to autonomously learn grammatical and punctuation rules—e.g., opening and closing parentheses. Furthermore,

978-1-5090-4470-2/16 $31.00 © 2016 IEEE
DOI 10.1109/WI.2016.129

character-level RNN tend to be more efficient than word-level RNN in terms of computational cost, which grows with the size of the input and output dictionaries. The works [7], [8] show that character-level RNN provide slightly worse performance than the equivalent word-based model, but the character-level approach allows to prediction and generation of new words and strings.

Long Short-term Memory (LSTM) networks [9], [10] are a form of RNN which has proven able to effectively generate characters sequences with long-range structures [11]. The work [3] bases on character-level LSTM RRN for machine translation tasks and proves their superiority over other statistical approaches.

An interesting NLG application of RNN is abstractive summarization [12], where the system produces a condensed representation of an input text that maintains its original meaning. The work in [13] employs RNN for question answering, with the NLG system producing correct answers to questions expressed in natural language. The work [14] provides a conversation system—a generator—for more fluent responses as part of a conversation.

A remarkable use of LSTM for NLG has been done in the generation of image descriptions [4], [15], [16] and in the generation of descriptive captions for video sequences [17]. Concerning the text generation for artistic purpose, Zhang and Lapta [18] proposed an RNN-based work for generating Chinese poetry. In [19], the authors show the ability of a LSTM framework to automatically generate rap lyrics tailored to the style of a given rapper.

III. Our approach

A restaurant is associated with a possibly empty set $C \subset \mathcal{C}$ of *categories*, with $\mathcal{C} = \{\text{italian}, \text{pub}, \text{spanish}, \dots\}$ (Table II shows the list of all categories in $\mathcal{C}$). A review for a restaurant is associated with a *rating* $s \in \{1, 2, 3, 4, 5\}$. We address the problem of automatically generating a review with a specified rating s for a restaurant with a specified set C of categories. The generated review should look like a review written by a human author and should be tailored to the rating and categories specified as input.

Our method is based on 3 steps: (i) a generative phase based on a LSTM character-level recurrent neural network, (ii) a category classification phase and (iii) a rating classification phase.

In the step i), we use LSTM RNN for generating text as follows. We first train the network (see next section) to predict the probability of the next *token* for a fixed-length sequence of tokens given as input—a token being a single character. Then, we generate text with the trained network by starting, as input, with an input sentence selected from a dataset of genuine reviews at random. We stochastically sample a token from the output of the network and append to the starting sequence. We then set the next input sequence for the network by shifting the starting sequence of one token, that is, we exclude the first token and we include the newly generated. We repeat this iterative procedure until a predefined number of reviews has been generated. We detect this condition by counting the number of occurrence of a special token t_{end}, which we included at the end of each genuine review in the training text. We remove the first review from each set of generated reviews in order to neutralize any strong dependence from the starting sentence.

Rating	# of reviews
1	111 218
2	111 833
3	245 896
4	671 610
5	1 028 707

TABLE I: Number of reviews grouped by rating.

Category	# reviews
french	346 175
spanish	334 429
italian	310 816
american	306 499
pizza	239 311
mediterranean	238 141
british	224 719
asian	178 640
pub	169 564
european	169 362

TABLE II: Number of reviews grouped by the 10 most frequent categories.

Concerning step ii, given the set of R reviews generated by the network, we use a set of binary classifiers to remove from R those reviews that are not coherent with the categories C specified as input. To this end, we input all reviews in R to a set of a 10 binary Naive Bayes classifiers, one for each category (we chose to consider only the 10 most frequent categories in our dataset). These classifiers were previously trained with genuine reviews (see next section) and use frequencies of 1-grams, 2-grams and 3-grams as features. We discard from R those reviews deemed to belong to less than half of the categories in C.

Finally, concerning step iii, we assign to each review in R a rating s' between 1 and 5 using a previously trained multi-class classifier. This classifier is Naive Bayes and uses the same features as those in the previous step. We remove from R all the reviews for which $|s - s'| > 1$ and pick one element from R at random as output of the procedure. If, at any point, $R = \emptyset$ then the overall procedure is aborted and restarted.

IV. Experimental evaluation

We collected a dataset composed of 2 169 264 reviews distributed over 66 700 restaurants. Table I shows the number of reviews for each rating while Table II reports the number of reviews for the 10 most frequent categories.

We used a LSTM-RNN implementation based on the char-rnn library[1], configured with 3 layers composed of 700 neurons each—as suggested by the library authors. We trained

[1] https://github.com/karpathy/char-rnn

	Useful		Not useful	
	#	%	#	%
Genuine	138	80	35	20
Artificial	51	29	127	71

TABLE III: Number and percentage of reviews considered as useful or not useful by human users (question a of extrinsic evaluation).

the LSTM-RNN with a randomly chosen subset of the full dataset, composed of 500 000 reviews with 100 000 reviews for each rating. The training phase lasted about 1 month on a Intel Xeon E5-2440 (2.40 GHz) CPU equipped with 32 GB of RAM. Once trained, the time spent by the neural network to generate a review is in the order of seconds.

The category and rating classifiers are based on the Naive Base implementation of the Stanford Classifier[2]. We trained each category classifier with 100 000 reviews and the rating classifier with 500 000 elements, all randomly selected from the dataset. Training time of classifiers was negligible with respect to the training time of LSTM-RNN.

In order to assess the effectiveness of our proposal we performed two different evaluations with human users, an *extrinsic* evaluation and an *intrinsic* evaluation, illustrated below. The evaluations were executed by presenting to each user a suite of *forms*, each including a set of reviews and few questions to be answered. This activity was carried out in our laboratory: it is important to remark that users were *not* aware that some revisions were artificially generated.

A. Extrinsic evaluation

In the extrinsic evaluation we assessed the ability of a review generated with our method to influence the decision of a user about whether to go or not to go to the reviewed restaurant. To this end, we constructed a set of forms, each composed of the name of a restaurant, its categories and 3 reviews. Reviews were randomly picked from a set R_g containing *genuine reviews* written by humans for that restaurant and from a set R_a of *artificial reviews*, using our method with inputs given by the categories of the restaurant and a random rating. We make sure that each form included at least one genuine review and one artificial review. In each form, we asked the user (a) for each review, if it was useful for his decision, and (b) if, basing on the 3 reviews, he would have decided to go to that restaurant. We proposed 3 forms to each user and we collected the evaluations of 39 different users.

The results of the extrinsic evaluation concerning question a are shown in Table III which reports the number of reviews marked by users as useful, separately for genuine and artificial. The key, and somewhat surprising, result shown in Table III is that around 30% of the artificial reviews (generated using our method) are indeed considered by a human users as useful for their decision.

Concerning question b, we categorized the forms submitted to users as follows. For each form, we computed the mean genuine rating $\overline{s}_g$ and the mean artificial rating $\overline{s}_a$ of the ratings associated with the genuine and artificial review, respectively. Next, we partitioned the 117 collected forms in 4 partitions, according to whether each of the two mean ratings was (denoted by $\mathcal{P}$) or was not (denoted by $\mathcal{N}$) ≥ 3: two partitions include forms for which genuine and artificial reviews agree (denoted as $\mathcal{PP}$ and $\mathcal{NN}$), two include forms for which genuine and artificial reviews disagree (denoted as $\mathcal{PN}$ and $\mathcal{NP}$)—the two symbols concern $\overline{s}_g$ and $\overline{s}_a$, respectively. We were particularly interested in users answer to question b in the cases in which $\overline{s}_g$ and $\overline{s}_a$ disagree, i.e., in partitions $\mathcal{PN}$ and $\mathcal{NP}$. Table IV shows the results concerning question b for those two partitions. It can be seen that the answer of the user conflicts with the polarity of genuine reviews in 29% of the cases for $\mathcal{PN}$ (genuine reviews are positive) and in 24% of the cases for $\mathcal{NP}$ (artificial reviews are negative).

		Going		Not going	
Genuine	Artificial	#	%	#	%
$\mathcal{P}$	$\mathcal{P}$	21	47	23	53
$\mathcal{P}$	$\mathcal{N}$	10	71	4	29
$\mathcal{N}$	$\mathcal{P}$	9	24	28	76
$\mathcal{N}$	$\mathcal{N}$	5	23	17	77

TABLE IV: Number and percentage of forms resulting in decision to go or not to go to a restaurant (question b of extrinsic evaluation).

A different point of view about this finding is given by Figure 1, which has one point for each form. The x and y coordinates of each point are the sums s_g^{tot} and s_a^{tot} of the ratings for the genuine and artificial reviews in the form, respectively. A circle represents a negative answer (not going) while a cross represents a positive answer (going). In a scenario in which the user decision conflicts with genuine reviews, there would be a concentration of circles (not going) answers in the bottom-right corner of the scatter plot and a concentration of crosses (going) in the left part of the image. In our case, Figure 1 highlights a concentration of crosses (going) answers in the top-left portion, i.e., positive answers in a region with high ratings of false reviews and low ratings of true reviews.

B. Intrinsic evaluation

With the intrinsic evaluation we aimed at evaluating if a human user is able to discriminate between genuine and generated reviews. In other words, we wanted to evaluate the effectiveness of our method in generating human-like reviews.

To this end, we constructed a set of forms, each composed of the name of a restaurant and 5 reviews. In each form, reviews could be partitioned in 4 classes according to the way we chose them (at least one review in each class): R_{gs} refers to reviews written by a human for the restaurant in the form; R_{gd} refers to reviews written by a human for a different restaurant; R_{as} refers to reviews generated with our method with the categories of the restaurant and a random rating as inputs; and R_{ad} of reviews generated using only the

[2] http://nlp.stanford.edu/software/classifier.shtml

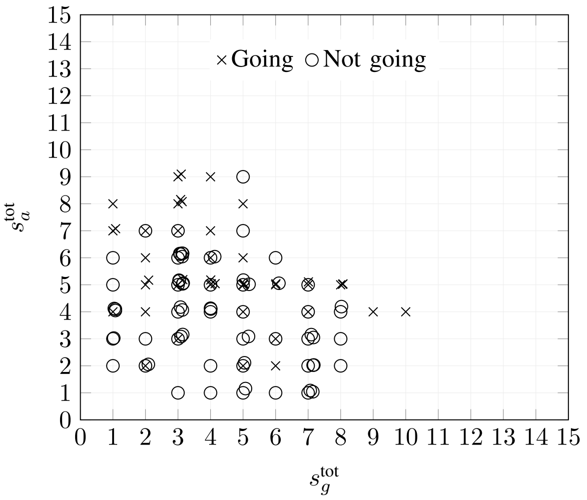

Fig. 1: Answers to question b of extrinsic evaluation, one mark for each form.

	Looks genuine		Looks artificial	
	#	%	#	%
R_{gs}	158	81	37	19
R_{gd}	102	52	93	48
R_{as}	47	24	148	76
R_{ad}	46	24	149	76

TABLE V: Number and percentage of answers for the intrinsic evaluation, grouped by class of reviews.

first step of our method. We asked the user, for each review, if the review was written by a human for the restaurant in the form. Since the nature of the question could suggest that some reviews were not written by humans, the forms of intrinsic evaluation were presented to each user after the forms for extrinsic evaluation. We proposed 4 forms to each user and we collected the evaluations of 39 different users.

The main results of the intrinsic evaluation are reported in Table V. Each row corresponds to one of the classes described above and shows the distribution of number and percentage of answers, i.e., either genuine or artificial. The main finding of this evaluation is that a review generated with our method is considered genuine more frequently than a genuine review is considered artificial (24% for R_{as} vs. 19% for R_{gs}). On the other hand, the percentage of answers for R_{as} and R_{ad} are essentially identical, suggesting that, for this kind of evaluation, the contribution of steps ii and iii of our method is not significant.

V. Conclusions

We have proposed a method for the automatic generation of restaurant reviews based on LSTM-RNN. The method is able to generate reviews tailored to a rating and a set of categories specified as input.

A key contribution of our work is the experimental evaluation involving 39 human users. The results are promising (or should we say worrisome?): about 30% of reviews generated by our method are considered useful by human users; the opinion of a user on a restaurant, when presented with a mix of genuine and automatically-generated reviews, conflicts with the polarity of genuine reviews in $\approx 25\%$ of the times; automatically-generated reviews are considered truthful more frequently than genuine reviews are considered artificial.

Although our approach is certainly to be investigated further, and although we focus only on the textual content of reviews while systems for detection of non-genuine reviews consider also other features describing users activities, we believe that our work provides strong indications that machine-generated reviews may soon become a real threat for the integrity of review-based systems.

References

[1] G. Lackermair, D. Kailer, and K. Kanmaz, "Importance of online product reviews from a consumer's perspective," *Advances in Economics and Business*, vol. 1, no. 1, pp. 1–5, 2013.

[2] T.-H. Wen, M. Gasic, N. Mrkšić, P.-H. Su, D. Vandyke, and S. Young, "Semantically conditioned lstm-based natural language generation for spoken dialogue systems," pp. 1711–1721, September 2015. [Online]. Available: http://aclweb.org/anthology/D15-1199

[3] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[4] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3128–3137.

[5] K. Kukich, "Where do phrases come from: Some preliminary experiments in connectionist phrase generation," in *Natural language generation*. Springer, 1987, pp. 405–421.

[6] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur, "Recurrent neural network based language model." in *Interspeech*, vol. 2, 2010, p. 3.

[7] T. Mikolov, I. Sutskever, A. Deoras, H.-S. Le, S. Kombrink, and J. Cernocky, "Subword language modeling with neural networks," *preprint (http://www. fit. vutbr. cz/imikolov/rnnlm/char. pdf)*, 2012.

[8] I. Sutskever, J. Martens, and G. E. Hinton, "Generating text with recurrent neural networks," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 1017–1024.

[9] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[10] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[11] A. Graves, "Generating sequences with recurrent neural networks," *arXiv preprint arXiv:1308.0850*, 2013.

[12] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," *arXiv preprint arXiv:1509.00685*, 2015.

[13] J. Yin, X. Jiang, Z. Lu, L. Shang, H. Li, and X. Li, "Neural generative question answering," *arXiv preprint arXiv:1512.01337*, 2015.

[14] K. Yao, G. Zweig, and B. Peng, "Attention with intention for a neural network conversation model," *arXiv preprint arXiv:1510.08565*, 2015.

[15] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (m-rnn)," *arXiv preprint arXiv:1412.6632*, 2014.

[16] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3156–3164.

[17] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," *arXiv preprint arXiv:1412.4729*, 2014.

[18] X. Zhang and M. Lapata, "Chinese poetry generation with recurrent neural networks." in *EMNLP*, 2014, pp. 670–680.

[19] P. Potash, A. Romanov, and A. Rumshisky, "Ghostwriter: Using an lstm for automatic rap lyric generation," pp. 1919–1924, 2015.