We know what you are going to post: User Status Generation Based on Personality and Topic

Yilun Wang
Department of Computer Science
Stanford University
yilunw@stanford.edu

Shijie Liu
Department of Computer Science
Stanford University
shijie2@stanford.edu

Abstract

Automatic sentence generation is an easy task for humans but an extremely challenging task for computers. Recent research has shown that Recurrent Neural Networks (RNN) based language model offers a promise to perform this task. However, traditional RNN is not able to generate semantically different sentences for people with different personalities and on different topics. In this project, we propose a model called Recurrent Neural Network based on Personality and Topic (RNNPT) to generate social media user status condition on user personality and status topics. In order to do so, we use a huge set of user personality and status data, and initialize the hidden layer in our models using the personality and topic representations. The personality representation are collected using Five-Factor Model in Psychology [11] while the topic representations are learnt using Latent Dirichlet Allocation (LDA) [3] which separates the status into a certain number of topics. We concatenated the topic representation and the personality representation to be the initialization vector for the hidden layer at time step 0. Two variants of our proposed models, namely RNNPT and RNNPT1 (with one-hot topic representation) are trained and used to output sentences based on different initialization vectors which indicated different user personalities and topics. We analyzed the performance of our proposed models both qualitatively and quantitatively and concluded that our model was able to generate high-quality user status which reflects the input personality and topics at a satisfactory level. Furthermore, our models outperform baseline RNN models in terms of both perplexity and BLEU scores.

1 Introduction

Social media status may be very easily generated by a human being, but seems to be an almost impossible task for computers, considering individuals with different personalities should be able to generate very distinct states on various topics like music, sports or politics. For example the words used by a politically-enthusiastic person and a politically-indifferent person on the topic of a recent election would be very different and we wish to capture this difference. Besides that, for the same person, the sentence generated on different topics like politics and sports should be very different.

Recurrent neural network (RNN) [5, 4] has been proved to be a powerful deep learning tools in generating sequences of data, like sentences, speeches and handwritings. However the model does not have a lot of variability in the sense that there is not distinction between the sentences generated by possibly different people on distinct topics. One of the ideas of improvement is to train several different RNNs on distinct datasets corresponding to different combination of personality and topics so that the same model with different sets of parameters can be trained, and each set of parameter

can be used to generate different sets of sentences based on personality and topics. However, given the enormous time and computing cost spent on training, this task is too expensive to carry out.

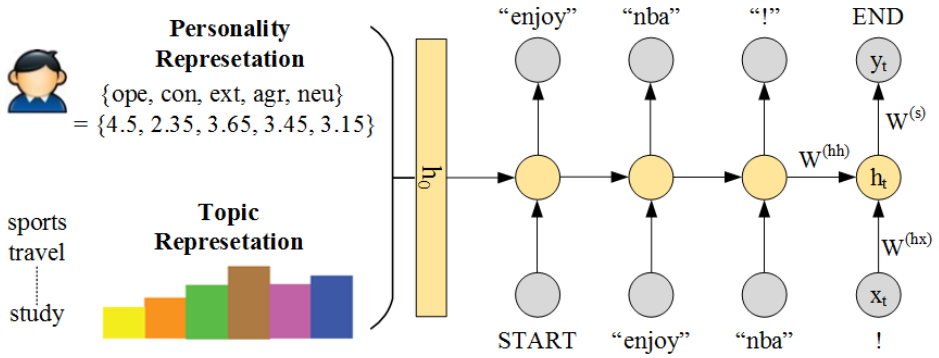

Figure 1: Diagram of our Recurrent Neural Network generative model. The RNN takes a word, the context from previous time steps and defines a distribution over the next word in the sentence. The RNN is conditioned on the topic and personality vector at the first time step. START and END are special tokens.

In this project, we want to exploit the ability of recurrent neural network in sentence generation, specifically Facebook status generation. As in Figure 1, we aims to present a model that automatically generates social media status of users based on different user personality and topic, by presenting a RNN conditioned on the the initialization vector for the hidden layer at time step 0. We will first represent each topic with some encoding, which will be concatenated with users' personality attribute to form the input vector. The input vector will then be passed into a Recurrent Neural Network to train and generate distinct user status for different user and on different topics. The output user status will serve a number of purposes. For example, besides automatically generating social media status for Facebook users, we can use some of the outputs to understand how extrovert or introvert persons talk about sports.

2 Related Work

2.1 Recurrent Neural Network

As discussed by Mikolov et al. [4], Recurrent neural network based language model has been proven to be very effective, as the research results in the paper 'indicate(s) that it is possible to obtain around 50% reduction of perplexity by using mixture of several RNN LMs, compared to a state of the art backoff language model'. Their work also suggests that although the computational cost at trainning time is high, their RNN model is superior than the traditional N-gram language model, as 'Recurrent neural networks outperformed significantly state of the art backoff models in all (their) experiments, most notably even in case when backoff models were trained on much more data than RNN LMs'. Their research can also be applied to speech recognition tasks and the result also shows it's a much better model than traditional n-gram models. Later, Mikolov et al. [5] optimize the computational complexity problem of the RNN language model. By introducing classes for all the words in the corpus, the model leads to more than 15 times speedup for both training and testing phases.

2.2 Topic Modeling

Latent Dirichlet Allocation (LDA) has been used widely used as a method of topic modeling. It is a flexible generative probabilistic model for collections of discrete data, such as the large status corpra we have collected in our task. In [3], Blei, Ng and Jordan describes it as 'a three-level hierarchical Bayesian model, in which each item of a collection is modeled as a finite mixture over an underlying set of topics', and then 'each topic is, in turn, modeled as an infinite mixture over an underlying set of topic probabilities'. Although exact inference of the parameters in LDA is intractable, the paper presents 'a simple convexity-based variational approach for inference', which approximates

the parameters, and evaluated the performance on several tasks namely document modeling, text classification and collaborative filtering. Several extensions of Latent Dirichlet Allocation have been proposed. For example, the Correlated Topic Model [6] follows LDA and introduces a correlation structure between topics by using the logistic normal distribution instead of the Dirichlet. Another extension is the hierarchical LDA (hLDA) [7] where topics are joined together in a hierarchy by using the nested Chinese restaurant process.

2.3 Sentence Generation

Recurrent Neural Networks (RNNs) and Long-Short Term Memory (LSTM) are very powerful sequence models that can be used in sentence generation [10, 8]. There have been some recent works about sentence generation (or caption generation) based on images that are relevant with our paper. For example, Karpathy and Fei-Fei [1] propose to use Region Convolutional Neural Network (RCNN) to extract the 4096-dimensional CNN codes of every region in the image and Recurrent Neural Network (RNN) to generate the sentence descriptions of images. Their word shows that the model 'generated descriptions significantly outperform retrieval baselines on both full images and on a new dataset of region-level annotations'. Also, Vinyals et al. [2] design a similar pipeline but use LSTM as the sentence generator. They evaluated their model qualitatively and quantitavely (by using the BLEU score) and concluded that their model is 'often quite accurate' and can achieve state-of-art performance.

3 Problem Statement

Our formal problem can be described as: given a dataset of user personality and the status that these users posted on Facebook, we want to generate new user status based on different users personality and topic combination. The status generated in our model should be a sentence that is related to the topic and the personality of the user.

The training data we will use is collected from the myPersonality project[1], which contains 3.1 million users' personality data and 22 million statuses of 154 thousand users. In the user personality data, each user is evaluated based on their answers to an anonymous questionnaire and their personality is calculated to be a five dimensional vector which correspond to the user's openness, conscientiousness, extraversion, agreeableness,and neuroticism respectively. In the user status data, each line consists of the user's id, the timestamp that the status was posted on Facebook, and the actual status in text. We split the status data into a training set to train the RNN parameters, a development set to tune the hyperparameters of the RNN, and a test set to evaluate the performance of our model.

There will be three major tasks for our project, namely personality and topic representation, neural network training and status generation. As our personality data has already conveniently represented each user's personality, we can just use the five dimensional vector as personality representation. For topic representation, firstly we will use the collected status to generate a certain number of topics for the user statuses in the training data. Then we will be using these data to train the parameters of a recurrent neural network as well the word vectors, condition on the topic representation and user personality. After that we will use the trained recurrent neural network to generate status for a person with a particular personality and on a particular topic.

We expect our model to output different 'types' of status with different combination of user personalities and topics. For example we expect the status generated by a very extrovert person on the topic sports to contain a lot of exclamation marks and words that are passionate, which is expected to have a clear distinction from the status generated by a cynical person on the topic of a political activity.

In the evaluation part, we want to have quantitative and qualitative experiments to prove the effectiveness of our proposed model. For quantitative experiment, following [1], given the topics and personalities of user-status in test set, we can automatically generate a set of status and compute the BLEU score between computer-generated status and ground truth status. In addition, we can present the BLEU score in 1-gram, 2-gram, and 3-gram. For qualitative experiment, we hope to automatically generate user status based on topics and personalities and analyze them. For example, people

[1] http://mypersonality.org/wiki/doku.php

with different personalities talking about same topic, or people with similar personalities describing different topics.

4 Recurrent Neural Network based on Personality and Topic (RNNPT)

As stated above, our approach mainly consists of three tasks: personality and topic representation, neural network training and status generation.

4.1 Personality and Topic representation

In topic representation, we generate N topics for the user statuses in the training data using topic model following the approach of Blei, Ng and Jordan[3]. For each sentence, the result of LDA is able to quantify it as a linear combination of the N topics.

Based on topic model, we propose to use two ways to represent the topic of the sentences, and we then shows the performance of these two representations in our experiment. First, we use the raw topic distribution of each sentence as the topic representation. Second, we introduce 'one-hot' representation $T \in \mathcal{R}^N$ where the ith topic is largest topic of the sentence with $T_i = 1$ and the rest elements are 0.

For example, in the case N=4, let the results of LDA on a sentence S generated by user U with personality be P be

$$S = 0.5 \times topic1 + 0.3 \times topic2 + 0.1 \times topic3 + 0.1 \times topic4 \tag{1}$$

Using raw topic distribution, the topic represetation of sentence S is,

$$T_A = \begin{bmatrix} 0.5 \\ 0.3 \\ 0.1 \\ 0.1 \end{bmatrix} \tag{2}$$

Using 'one-hot' representation, the topic represetation of sentence S is,

$$T_A = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{3}$$

For each user's status update, we use the corresponding user's personality attributes P collected using International Personality Item Pool (IPIP) Five-Factor Model [11]. P for each user is in 5 dimension and the attributes are representing traits of openness, conscientiousness, extraversion, agreeableness, and neuroticism respectively.

For example if the personality of user U is

$$\{ope = 4.5,\ con = 2.35,\ ext = 3.65,\ agr = 3.45,\ neu = 3.15\} \tag{4}$$

The personality representation will be a five dimensional vector

$$P_U = \begin{bmatrix} 3.5 \\ 2.35 \\ 3.65 \\ 3.45 \\ 3.15 \end{bmatrix} \tag{5}$$

4.2 Neural Network Training

In neural network training, we develop a recurrent neural network model that takes in a vector representation of the user's personality P and the topic T, by concatenating the user personality

vector (as provided in the data we collected from myPersonality) and the topic vector we generated earlier. Our RNN will output a word at a timestamp, by calculating the probabilities of each word at timestamp t and do a sampling according to these probabilities. We preserve a set of most frequently used words for each personality and only sample from them, or increase the probabilities of them so that they will appear more often.

As illustrated in Figure 1, we firstly parse each training status by adding sentence start <s>and sentence end </s>, and remove all punctuations except for common ones including full stop, comma, exclamation mark and question mark. Then we parse the modified sentences and counted the term frequencies for each term that appears in the dataset and retain words that appear more than 600 times. This procedure leaves us with 3142 terms that account for 92% of the total words in our corpus.

For every status $x_1, ...x_{t-1}, x_t, x_{t+1}, ..., x_T$, the precise form of our proposed RNN is as follows:

$$h_t = \sigma(W^{(hh)}h_{t-1} + W^{(hx)}x_{[t]}) \quad (6)$$

$$\hat{y}_t = softmax(W^{(S)}h_t) \quad (7)$$

$$P(x_{t+1} = \hat{v}_j|x_t, .., x_1) = \hat{y}_{t,j} \quad (8)$$

where σ is the sigmoid activation function and $h_0 \in \mathcal{R}^{D_h}$ is some initialization vector for the hidden layer at time step 0, $x_{[t]}$ is the word vector of the word at time step t, $W^{(hh)} \in \mathcal{R}^{D_h \times D_h}$, $W^{(hx)} \in \mathcal{R}^{D_h \times d}$ and $W^{(S)} \in \mathcal{R}^{|V| \times D_h}$ are the weights we need to learn.

By concatenate the vector representation of the user's personality P_U and the topic T_A, the hidden layer at the time step 0 is given by

$$h_0 = \begin{bmatrix} P_U \\ T_A \end{bmatrix} \quad (9)$$

We will use backpropagation to calculate the gradients for the parameters in the RNN, and use stochastic gradient descent to update the parameters, just as we learnt in class and assignments.

4.3 Status Generation

In status generation, we concatenate the representations of topics and users' personality attributes as input vector, and will use the trained RNN to output the words at each timestamp. Given the input vector (topics and personalities), we are able to generate status through the RNN model we trained. As we mentioned above, we will evaluate the sentences generated quantitatively and qualitatively.

5 Experiments and Results

We evaluate our algorithm on Facebook user status and personality dataset from myPersonality project. First, We need to preprocess the data from original dataset. Specifically, we intersect the user personality table and user status table from the data in myPersonality project. Then, we remove users with less than 200 status update. Since the user personality vector is in 5 dimensions, the statistics of our final dataset is in Table 1.

Num. of Users	14155
Num. of Personality Attributes	14155× 5
Num. of Statuses	1005017

Table 1: Statistics of our dataset

We evaluate three different status generation models in our experiments. Original RNN is applied as a baseline and two variants of our proposed model, namely RNNPT and RNNPT1, are trained to prove the effectiveness of personality and topic vectors. The hidden dimension h in these models are all set as 40 dimension vectors for fair comparison.

- Recurrent Neural Network **(RNN)**. We trained our RNN without conditioning on initialization vector h_0 (i.e. used a randomized vector as h_0). This model serves a baseline which will be compared with other models.
- Recurrent Neural Network based on Personality and Topic **(RNNPT)**. This model uses raw topic distribution as the topic representation vector, and h_0 is the concatenation of topic vector and personality vector.
- Recurrent Neural Network based on One-hot Personality and Topic **(RNNPT1)**. This model represents the topic vector as the 'one-hot' vector where the ith topic is largest topic of the sentence with $T_i = 1$ and the rest elements are 0. h_0 is the concatenation of topic vector and personality vector.

We split our entire dataset into training set, dev set, and test set where the test set accounts for 30% of all the statuses. In following experiments, we tune our hyperparameters on the dev set and evaluate the models on test set.

5.1 Qualitative Experiments

We have clustered the user statuses into 35 topics using LDA, and, as an example, here are the top words for five topics that are randomly chosen (Table 2)

Topic 1	Topic 2	Topic 3	Topic 4	Topic 5
flight	gun	dayi	doin	christ
celebrating	balls	mei	lookin	praise
event	bitches	todayi	watchin	bible
nyc	smoke	oz	feelin	grace
conference	fucked	hubby	thinkin	prayer
arrived	dick	timei	playin	glory
international	punch	lbs	tryin	worship
celebrated	guitar	babies	nothin	mercy
annual	shoot	daddy	havin	spirit
airport	jack	mommy	comin	christian

Table 2: Top words for five randomly chosen topics

We can see intuitively that words with similar meanings or functions have been clustered into the same topics, signaling the success of our topic generation process.

We conditioned our initialization vector on different combination of personality and topics and generated a few sample sentences using RNNPT1

1. Precondition on Topic 1 (Business travel) and random personality: *Off to nyc for meeting, please win a food frustrated me you.*
2. Precondition on Topic 5 (Religion) and random personality: *Imagine this god, the darn ourselves, the lord of the ordered.*
3. Precondition on personality extroversion and Topic 19: *What a happy hot girl! You are thinking off in all the time is a night. oh yay!*
4. Precondition on personality introversion and Topic 19: *This is the heart day. Girls just hear me and keep you would.*

We can see that although some of the sentences generated do not perfectly make sense, they are mostly grammatically correct. Moreover, we have successfully generated distinct sentences based on different personality and topics. Sentence 1 and 2 are based on random personality but different topics, and thus the generated sentence 1 pertains closely to business meeting while sentence 2 is about God and religion. Sentence 3 and 4 are based on the same topic but opposite personality, and thus the two sentences generated also reflects opposing personality based on the words used and the sentiment they express.

5.2 Quantitative Experiments

We evaluated our models on the test dataset using the following two quantitative metrics:

- **Perplexity**. Perplexity is a measurement of how well a probability distribution or probability model predicts a sample. In our case, we calculated Perlexity as e^J, where J is the average cross entropy loss our model's prediction over the test dataset.
- **BLEU Scores**. BLEU (Bilingual Evaluation Understudy) is an algorithm for evaluating the quality of text which has been machine-translated from one natural language to another [9]. Here, we measure the BLEU score between the statuses the models generate and the statuses the users actually post on 1-gram, 2-gram and 3-gram. Here we conducted two sentence generation experiments. In the first experiment sentences are not given any starting words except for sentence start, whereas in the second experiment besides sentence start we also give the 3 starting words and measure the score of the sentence generated. We compute the BLEU scores in both experiments for 1-gram, 2-gram and 3-gram.

The following result is achieved:

Model	Perplexity
RNN	121.121
RNNPT	125.246
RNNPT1	**117.908**

Table 3: Evaluation using Perplexity

The Table 3 indicates that while RNNPT1 is able to outperform RNN with a lower Perplexity score, RNNPT is doing worse than RNN. This phenomenon might be explained by the fact that in RNNPT, we precondition the initialization vector on the topic distribution, which may contain a lot of noise as it might be the case that more than one topics can have a substantial share of the probability. While in RNNPT1, the vector is conditioned on only one topic, and thus is able to achieve a better result than RNNPT and RNN.

	No Word Given			Three Words Given		
Method	1-gram	2-gram	3-gram	1-gram	2-gram	3-gram
RNN	21.54	11.42	7.93	33.70	22.72	17.58
RNNPT	21.37	11.42	7.99	33.57	22.76	17.70
RNNPT1	**22.06**	**11.91**	**8.42**	**34.21**	**23.32**	**18.21**

Table 4: Evaluation using BLEU Scores

The BLEU score results (Table 4) show that while while RNNPT has similar BLEU scores compared to the RNN baseline, RNNPT1 is able to do better than the other two models by a significant margin as it has higher BLEU score in both experiments and for 1-gram, 2-gram and 3-gram respectively.

6 Conclusion

We introduced a Recurrent Neural Network model that generates social media user status based on different personality and topic, by training with a large set of user personality and status data. Our approach features a novel idea of conditioning the initialization vector as the concatenation of topic and personality vector representation. We showed that our model is able to generate sentences with distinct semantics based on different combination of topic and personality input. Moreover, while RNNPT can achieve similar performance compared to RNN in terms perplexity and BLEU scores, RNNPT1 with the 'one-hot' topic representation vector is able to achieve a better performance than the other two models.

7 Future Work

Firstly we can explore more methods of topic representation. We may also try some deep learning methods to generate the topics and learn the encoding of each topic, so as to get a distinct representation of different topics. We can also try to train a word vector corresponding to different topics and use the average word embedding as the topic vector.

Besides, we may add a hidden layer between the vector of personality and topic representation and h_0 to increase the model size and have arbitrary number of dimension in hidden layer.

Finally we can upgrade the RNN model in our model into Long Short Term Memory (LSTM) [8] to further improve the performance of our proposed model.

References

[1] Karpathy, A., & Fei-Fei, L. (2014). Deep visual-semantic alignments for generating image descriptions. arXiv preprint arXiv:1412.2306.

[2] Vinyals, O., Toshev, A., Bengio, S., & Erhan, D. (2014). Show and tell: A neural image caption generator. arXiv preprint arXiv:1411.4555.

[3] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. the Journal of machine Learning research, 3, 993-1022.

[4] Mikolov, T., Karafit, M., Burget, L., Cernock, J., & Khudanpur, S. (2010, January). Recurrent neural network based language model. In INTERSPEECH 2010, 11th Annual Conference of the International Speech Communication Association, Makuhari, Chiba, Japan, September 26-30, 2010 (pp. 1045-1048).

[5] Mikolov, T., Kombrink, S., Burget, L., Cernocky, J. H., & Khudanpur, S. (2011, May). Extensions of recurrent neural network language model. In Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on (pp. 5528-5531). IEEE.

[6] Blei, D., & Lafferty, J. (2006). Correlated topic models. Advances in neural information processing systems, 18, 147.

[7] Griffiths, D. M. B. T. L., & Tenenbaum, M. I. J. J. B. (2004). Hierarchical topic models and the nested Chinese restaurant process. Advances in neural information processing systems, 16, 17.

[8] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[9] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002, July). BLEU: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting on association for computational linguistics (pp. 311-318). Association for Computational Linguistics.

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

[11] Goldberg, L. R., Johnson, J. A., Eber, H. W., Hogan, R., Ashton, M. C., Cloninger, C. R., & Gough, H. G. (2006). The international personality item pool and the future of public-domain personality measures. Journal of Research in personality, 40(1), 84-96.