Developing The Pointer Backward-Forward Algorithm In Token Of Test In Text To Know The Level Of Individual Depression

Hasanul Fahmi¹, Muhammad Zarlis², Zakarias Situmorang³, Adli A Nababan⁴, Yulia A Dalimunthe⁵, Yessi F Annisah⁶, Mufida Khaıraini⁷

¹,² Departemnt of Computer Science Doctoral Program, Faculty of Computer Science and Information Technology, Sumatera Utara University, St. Universitas No.9A, Medan-Indonesia,
³ Computer Science Departement, Universitas Katolik Santo Thomas,
⁴ STMIK Pelita Nusantara Sumatera Utara, Indonesia,
⁵,⁶,⁷ Teknik Informatika - Universitas Harapan Medan, Indonesia

*fahmihasanul9@yahoo.co.id

Abstract. The way you move and sleep, interact with people around you, depression change everything. This even looks at the way you talk and express yourself in writing. Sometimes this "depression language" can have a strong effect to others. Natural Language Processing (NLP) is one of the fields of computer science, artificial intelligence, and language (linguistics) related to the interaction between computers and natural human languages, such as Indonesian or English. The main objective of the NLP study is to make a machine that is able to know and understand the meaning of language, lexical analysis has the role of reading input characters and generating output tokens and making spaces, tabs, newlines and connecting rows of programs. However, the problems will occur when lexical analyzers where lexical is not able to determine when there are misspellings on tokens. Therefore, with the development of the backward-forward pointer algorithm, it is expected that in the future it will be easier to analyze both the words in front, in the middle and behind, so that the machine can quickly determine the individual depression level of speech tokens and speech tones.

1. Introduction

Natural language processing (NLP) is a variety of computational techniques that are motivated by theory for automatic analysis and representation of human language. NLP research has evolved from the era of punch and batch processing, where analysis of sentences can take up to 7 minutes, in the era of Google, where millions of web pages can be processed in less than one second. [1] In many scenarios of the generation of natural world real language, it is necessary to introduce lexical boundaries into the resulting sequence, which is referred to as making lexically limited sentences. [2] In terms of producing sentences that must contain certain words or phrases and active research topics in the generation of natural languages. For example, to avoid universal responses in dialogue, contextual keywords can be entered into replies. [3] For machine translation, some specific domain terminology may need to be included in the result translation [4].
In recent years, the artificial neural network (RNN) model has made remarkable progress in several natural language generation tasks, including neural machine translation, product review generation, abstractive summation, and generation of affective text. However, network-based models existing artificial nerves (RNN) usually produce sentences gradually from left to right by file search. It is difficult for these models to directly produce sentences containing lexical boundaries. Replacing arbitrary words in output with the desired word will damage the smoothness of the sentence. Given the words as additional inputs, there is no guarantee that they will appear in the output. In addition, the artificial neural network (RNN) method does not consider what specific words need to be entered early in the generation, but try to impose specific words into the sentence at each step of the sentence generation process. This improper way can affect the quality of the sentence produced. The researchers find that this problem was more serious when applying artificial neural network methods in the generative language model without conditions. Another more natural way to produce lexically bounded sentences is based on the Reverse and Advanced Language Models (B/F-LM) introduced by Lili Mou, in which language models back and forth work together to produce lexically limited sentences. The backward language model takes a lexical constraint as input to produce the first half of the sentence. Then the advanced language model takes this first half as input to produce all sentences. Both of these language models are trained using the maximum likelihood estimation goal (MLE). So from previous research can facilitate us to determine the level of individual depression using the backward and forward algorithms.

2. Methodology

2.1. Back and Forward Language Models

The first proposed three variants of backward and advanced language models: separated, synchronized, and backward and advanced asynchronous language models (called sep-BF, syn-BF, and asyn-BF). Their experiments show that asyn-BF is the most natural way to model forward and backward sequences. To insert the asyn-BF model into the seq2seq framework to generate a reply to the conversation containing the given keyword. The BFGAN method produces a lexical boundary sentence similar to asyn-BF as discussed in the Introduction. However, as discussed above, training the reverse and forward models separately with MLE will provide serious exposure bias, especially for unconditional language models.

2.2. GAN for Text Generation

BFGAN is different from the others, because it uses two generators to produce sentences lexically, and gives a differentiator to guide the combined training of two generators. To solve difficult training problems from the model. Therefore the researchers proposed several training techniques to make the training process more stable and efficient. As far as we know, this is the first time the GAN has been successfully used to produce lexically limited sentences.

2.3. Frameworks

The researchers will explain the generation process shown in Figure 1, including:

a. Discriminator

The function of the discriminator is to distinguish real sentences from lexical sentences produced by the machine. This guides joint training of two generators by assigning them.

b. Backward Generator

Given the lexical constraint 1, the backward generator considers it the starting point sentence, and produces half a sentence backwards.
c. Forward Generator

The sequence generated by the reverse generator is reversed and inserted into front generator. Then learn to produce all sentences in order to trick the differentiators.

![Diagram of Forward and Backward Generators](image)

**Figure 1:** The process of generating a generator backwards and a generator forward

### 2.4. Backward and Forward Generators

Let us give the parable of the \( wc \) as a reference to the lexical constraints given. The researchers give a limit to the number of sentences lexically by \( m = w_1 \ldots w_c \) ... \( W_m \) the half of the back sentence \( s < c = w_c-1 \ldots w_1 \) generated by a reverse generator \( G_{O}^{(bw)} \) and half the front sentence \( s > c = w_c+1 \ldots w_m \) produced by the front generator \( G_{O}^{(fw)} \). Then the combination of the reverse generator and the front generator can be written as follows:

\[
G(s|w_c; \Theta, \Theta') = P_{O}^{(bw)}(s < c|w_c)P_{O}^{(fw)}(s > c|s_1:c),
\]

Where \( s_1:c = w_1 \ldots wc \)

Reverse Generator Model \( G_{O}(bw) \) with the possibility of half of the back sentence:

\[
P_{O}^{(bw)}(s < c|w_c) = \prod_{i=1}^{c-1} P_{O}^{(bw)}(w_c-i|w_c \ldots w_c-i+1)
\]

Front Generator Model \( G_{O}(fw) \) with the possibility of half of the front sentence:

\[
P_{O}^{(fw)}(s > c|s_1:c) = \prod_{i=1}^{m-c} P_{O}^{(fw)}(w_c+i|w_1 \ldots w_c+i-1)
\]

### 3. Result And Discuss

#### 3.1. Technical Training

To establish lexical boundaries during training, we randomly sample the real sentences at each step. For each real sentence \( s \) with the length \( m \), we slice it from the middle to the sentence fragment \( Sc \) in length \( m \Delta x \). This sentence fragment is taken as a lexical constraint and inserted into a generator to produce a false example.

#### 3.2. Data set

The researchers will provide examples of data from words that are commonly used by someone in a state of depression:
Table 1.1 Depression and Sentence Depression Tokens

| Depression Token (TD) | Q1 | Q2 |
|-----------------------|----|----|
| I, myself, them, he, always, nothing, total, lonely, sad, miserable | Alone, alone, as if he never needed anyone. Feel afraid to fall in love, tired of being called a loyal friend, and had begun to be difficult according to parents. They say I'm selfish, yeah I'm selfish indeed. But do you understand if what they do has a huge impact on me? Not the cuss I want to hear, or my explosive anger only they hear my voice. Is there something wrong with me? How bad is I am? | Someone please hold me, I am exhausted from this world. Someone please wipe me, I am drenched with tears. Someone please notice me, My struggle first Please acknowledge the poor me Please help me |

Based on the table 1.1 above where the researchers display negative tokens issued by someone who is depressed, and Q1, Q2 is a sentence issued by someone who has high depression and ends in suicide. Here the researcher will provide a specific process of training algorithms BFGAN 1

Algorithm 1 Experiment BFGAN algorithm

Data needs: A backward generator $G_{\theta_{bw}}$ with parameters $\Theta$
A forward generator $G_{\theta_{fw}}$ with parameter $\Theta$

1: Pre-train the $G_{\theta_{fw}}$, $G_{\theta_{bw}}$ and $D_{\theta}$ for certain epochs
2: for $K = T$, 1, - $\Delta$ do
3: for number of training iterations do
4: Sample $s$ from the dataset. Get the lexical constraints, with length $K$
5: Given $s_c$, let $G_{\theta_{fw}}$ and $G_{\theta_{bw}}$ generate the whole sentence $\hat{s}$ using beam search
6: Compute the reward $r$ of $\hat{s}^s$ using $D_{\theta}$
7: Update $G_{\theta_{fw}}$ on $\hat{s} < c$ with reward $r$
8: Update $G_{\theta_{bw}}$ on $\hat{s} > c$ with reward $r$
9: Update $G_{\theta_{bw}}$ and $G_{\theta_{fw}}$ on $s_c$ using cross entropy
10: Teacher-Forcing: Update $G_{\theta_{bw}}$ and $G_{\theta_{fw}}$ on $s$
11: for $i = 1$, D-steps do
12: Sample $s$ from real data, construct the negativesamples $\hat{s}$
13: Update $D_{\theta}$ using $s$ as as positive samples and $\hat{s}$ as negative samples
14: end for
15: end for
16: end for

In this study the researchers will use a fairly large dataset, where the researchers will compose vocabulary into 10 thousand and have 35 sentences in length. Where the data sets are in the form of text, and raised with BFGAN to be implemented using the algorithm forward and backward.
4. Conclusion

1) In the Forward-backward algorithm research is limited to the greeting tokens in the text.
2) By using the application of the forward and backward generator models, we can find out easily both behind the sentence and in front of the sentence can make it easier for us to know the level of depression in individuals.
3) By using the BFGAN application, we can make the process more stable and efficient to produce lexically limited sentences.

Reference

[1]. E. Cambria and B. White, “Jumping NLP curves: A review of natural language processing research,” IEEE Computational Intelligence Magazine, vol. 9, no. 2, pp. 48–57, 2014.
[2]. Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In ACL.
[3]. LiliMou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016a. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. In COLING.
[4]. Matt Post and David Vilar. 2018. Fast lexically constrained decoding with dynamic beam allocation for neural machine translation. In NAACL-HLT.
[5]. Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google’s neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144.
[6]. Franz Josef Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. Computational linguistics.
[7]. Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2017. Guided open vocabulary image captioning with constrained beam search. In EMNLP.
[8]. LiliMou, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. 2016b. Backward and forward language modeling for constrained sentence generation. Computer Science