UNIVERSITÉ PARIS DESCARTES

Recurrent Neural Networks for Spelling and Grammatical Error Correction

Sina Ahmadi

Paris Descartes University

sina.ahmadi@etu.parisdescartes.fr

December 21, 2017

Overview

Introduction

Automatic spelling and grammar correction is the task of automatically correcting errors in written text.

- This cake is *basicly* sugar, butter, and flour. [→ basically]
- We went to the store and bought *new stove*. [→ a new stove]
- *i*'m *entirely* awake. [→ {I, wide}]

The ability to correct errors accurately will improve

- the reliability of the underlying applications
- the construction of software to help foreign language learning
- to reduce noise in the entry of NLP tools
- better processing of unedited texts on the Web.

Problem definition

Given a N-character source sentence $S = s_1, s_2, ..., s_N$ with its reference sentence $T = t_1, t_2, ..., t_M$, we define an error correction system as:

Definition

$$\widehat{T} = MC(S) \quad (1)$$

where $\widehat{T}$ is a correction hypothesis.

Question: How can the MC function can be modeled?

Background

Various algorithms propose different approaches:

- **Error detection**: involves determining whether an input word has an equivalence relation with a word in the dictionary.
  - Dictionary lookup
  - n-gram analysis
- **Error correction**: refers to the attempt to endow spell checkers with the ability to correct detected errors.
  - Minimum edit distance technique
  - Similarity key technique
  - Rule-based techniques
  - **Probabilistic techniques**

Probabilistic techniques

We assume the task of error correction as a type of monolingual *machine translation* where the source sentence is potentially erroneous and the target sentence should be the corrected form of the input.

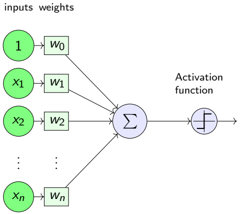
Aim

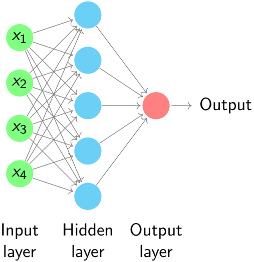
To create a probabilistic model in such a way that:

$$\widehat{T} = \underset{T}{argmax} P(T|S;\theta) \tag{2}$$

where θ is the parameters of the model.
This is called the Fundamental Equation of Machine Translation [Smith, 2012].

Neural networks as a probabilistic model





- Mathematical model of the biological neural networks
- Computes a single output from multiple real-valued inputs:

$$z = b + \sum_{i=1}^{n} w_i x_i \tag{3}$$

- Putting the output into a non-linear function:

$$tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1} \tag{4}$$

- Back-propagates in order to minimize the loss function H:

$$\theta^* = \underset{\theta}{argmin} \mathbf{H}(\widehat{y} - y) \tag{5}$$

NLP challenges in Machine Translation

Large input state spaces → **word embedding**

No upper limit on the number of words.

Long-term dependencies

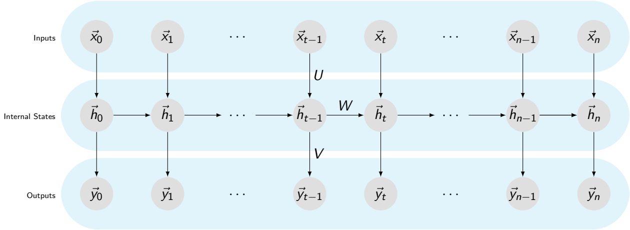
- Constraints: **He** did not even think about **himself**.
- Selectional preferences: I ate salad with **fork** NOT ~~rake~~.

Variable-length output sizes

- This structure have anormality → **30** characters
- This structure has an abnormality. → **34** characters

Recurrent Neural Network (RNN)

Unlike a simple MLP, RNNs can make use of all the previous inputs. Thus, it provides a memory-like functionality.

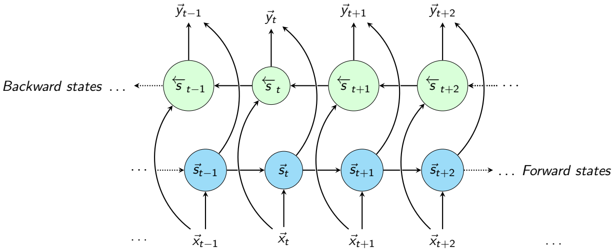


$$h_t = tanh(Wx_t + Uh_{t-1} + b) \tag{6}$$
$$\widehat{y}_t = softmax(Vh_t) \tag{7}$$

W, U an V are the parameters of our network that we want to learn.

Bidirectional Recurrent Neural Network

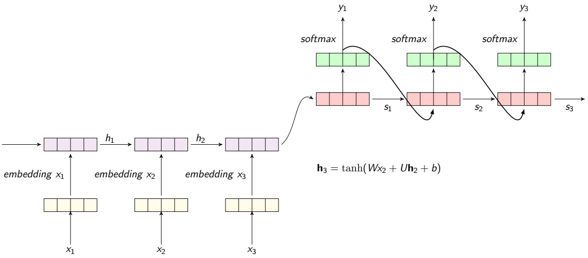
We can use two RNN models; one that reads through the input sequence forwards and the other backwards, both with two different hidden units but connected to the same output.



$$\vec{h}_t = tanh(\vec{W}x_t + \vec{U}\vec{h}_{t-1} + \vec{b}) \tag{8}$$

$$\overleftarrow{h_t} = tanh(\overleftarrow{W}x_t + \overleftarrow{U}\overleftarrow{h}_{t+1} + \overleftarrow{b}) \tag{9}$$

The sequence-to-sequence model is composed of two processes : *encoding* and *decoding*.



$$h_t = RNN(x_t, h_{t-1}) \tag{10}$$

$$c = tanh(h_T) \tag{11}$$

$$p(y_{1:T}|c) = \prod_{t=1}^{T} p(y_t|\{y_1, y_2, ..., y_{t-1}\}, c) \tag{12}$$

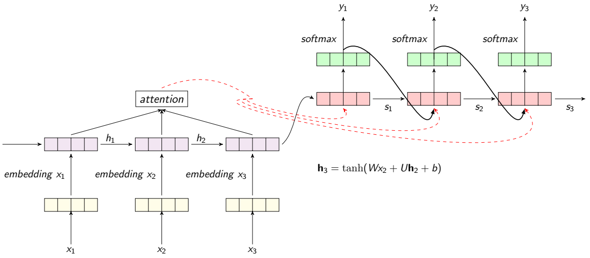
Attention mechanism

The attention mechanism calculates a new vector c_t for the output word y_t at the decoding step t.

$$c_t = \sum_{j=1}^{T} \alpha_{tj} h_j \quad (13)$$

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T} exp(e_{ik})} \quad (14)$$

$$e_{ij} = a(s_{i-1}, h_j) \quad (15)$$



where h_j is the hidden state of the word x_j, and a_{tj} is the weight of h_j for predicting y_t. This vector is also called *attention vector*.

Output generation (RNN and BRNN)

Algorithm 1 Correction of an input sequence using a character-level model(RNN or BRNN)

Input: Input sequence $C \leftarrow c_1, c_2, ..., c_N$
Output: Corrected sequence $O \leftarrow o_1, o_2, ..., o_N$

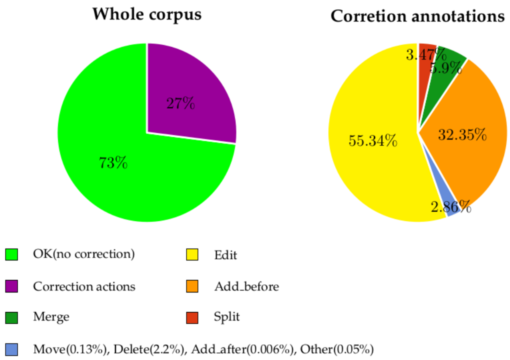
procedure GENERATEOUTPUT(C)
  $embeddedInput \leftarrow$ `embedSequence(C)`
  $modelOutputs \leftarrow$ `runModel`$(embeddedInput)$
  $p_0, p_1, ..., p_N \leftarrow$ `getProbabilities`$(modelOutputs)$
  $index \leftarrow$ `0`
  while $i < N$ **do**
    $predictedIndex \leftarrow$ `argmax`(p_i)
    $o_i \leftarrow$ `indexToCharacter`$(predictedIndex)$
    $i \leftarrow i + 1$
  return O

Output generation (Seq2Seq and Attention)

Algorithm 2 Correction of an input sequence using a character-level model(encoder-decoder or attention-based encoder-decoder)

Input: Input sequence $C \leftarrow c_1, c_2, ..., c_N$
Output: Corrected sequence $O \leftarrow o_1, o_2, ..., o_M$

procedure GENERATEOUTPUT(C)
  $embeddedInput \leftarrow$ `embedSequence(C)`
  $modelOutputs \leftarrow$ `runModel`$(embeddedInput)$
  $p_0, p_1, ..., p_N \leftarrow$ `getProbabilities`$(modelOutputs)$
  $index \leftarrow 0$
  $MaxSize \leftarrow N \times 2$
  while $i < MaxSize$ **do**
    $predictedIndex \leftarrow$ `argmax`(p_i)
    $o_i \leftarrow$ `indexToCharacter`$(predictedIndex)$
    $i \leftarrow i + 1$
    if $o_i =$ " < EOS > " **then**
      break
  return O


Whole corpus
27%
73%
Corretion annotations
3.47%
5.9%
55.34%
32.35%
2.86%
OK(no correction)
Correction actions
Merge
Edit
Add_before
Split
Move(0.13%), Delete(2.2%), Add_after(0.006%), Other(0.05%)

Various metrics are used to evaluate the correction models, including MaxMatch M^2 [Dahlmeier, 2012], I-measure [Felice, 2015], BLEU and GLEU [Napoles, 2015].

Model	M^2 scorer		
	P	R	$F_{0.5}$
RNN	0.5397	0.2487	0.4373
BiRNN	0.5544	0.2943	0.4711
Encoder-decoder	**0.5835**	**0.3249**	**0.5034**
Attention	0.5132	0.2132	0.4155

Table: Evaluation results of the models using MaxMatch M^2 metric. Bold numbers indicate the scores of the best model.

Qualitative comparison

Source sentence	306 characters
لا زال كبير الشبيحه يظن ان ارواح وآلام الناس اقل كلفه من تخليه عن منصبه ، فلذلك اذا كان السوريون لا يرتضون بهذه المعادله المهينه ، فعليهم ان يهبوا هبه قويه واحده وياخذو حقوقهم من هذه العصابه عنوه ، اننا يا أحبائي ندفع ثمن اكثر من اربعين عام ومن الخنوع والذل والثمن سيكون غاليا ولكنه يستأهل هذه التضحيات	
Gold standard reference	**309 characters**
لازال كبير الشبيحة يظن أن أرواح وآلام الناس أقل كلفة من تخليه عن منصبه ، فلذلك إذا كان السوريون لا يرتضون بهذه المعادلة المهينة ، فعليهم أن يهبوا هبة قوية واحدة ويأخذوا حقوقهم من هذه العصابة عنوة . إننا يا أحبائي ندفع ثمن أكثر من أربعين عاما ، ومن الخنوع ، والذل والثمن سيكون غاليا ولكنه يستأهل هذه التضحيات .	
Recurrent neural network model prediction	**307 characters**
لا زال كبير الشبيحة يظن ان ارواح وآلام الناس اقل كلفه من تخليه عن منصبه ، فلذلك اذا كان السوريون لا يرتضون بهذه المعادلة المهينه ، فعليهم ان يهبوا هبه قويه واحده ويااخذذاو حقوهم نن ههه العالبه عنوه أننا يا أحبائي ندفع ثمن أكثر من أربعين عام ، ومن الخنوع ،، واللا والمن سيكنن ااااا ولنن يستتهههههه الضحي	
Bidirectional recurrent neural network prediction	**307 characters**
لا زال كبير الشبيحة يظن أن أرواح والأم الناس أقل كلفه من تخليه عن منصبه ، فلذلك إذا كان السوريون لا يرتضون بهذه المعادلة المهينة ، فعليهم أن يهبوا هبه قويه واحده وياخذو قققهمممم ههه االعصبب ععوهه ننا ياأأحائئي ندف من ككثرممن رربعن امم ، وومن النووع ،،والذل والمثن سيكوننغغايا وككنههستههل ههه التضحيت	
Sequence-to-sequence model prediction	**306 characters**
لا زال كبير الشبيحة يظن أن أرواح وللام الناس أقل كلفه من تخليه عن منصبه ، فلذلك إذا كان السوريون لا يرتضون بهذه المعادله المهينة ، فعليهم أن يهبوا هبه قويه واحده وياخذو وحقققهمممننههه اللصابب عنوه أننا يا حبائييندفع ثمم أككر ن رربعن ع من الخنوو واللل واللمنن ييون االيي ووكنههههستهألههذهاااتتضا	
Attention-based sequence-to-sequence model prediction	**309 characters**
لا زال كبير الشبيحه يظن ان ارواح وللام الناس اقل كلفه من تخليه عن منصبه ، فلذلك اذا كان السوريون لا يرتضون بهذه المعادلة المهينة ، فعليهم ان يهبوا هبه قويه واحده وياخذ واحققققهم من هذه اعصابة عنوه إنا يا حببائي ندف من أكثر من أربيين عام ، ومن الخخوع ، والل والثمن سيكون اليا وككنه يتتهههذه التضضي	

Conclusion and future work

Conclusion

- Modeling correction error for any language.
- Reducing precision in correction of long sentences.

Future studies

- Models to be explored in more levels, e.g., word-level, phrase-level.
- Limiting the length of the sequences in training models.
- Using deeper networks with larger embedding size.
- Preventing over-learning of models by not training them over correct input tokens (action ="OK").

References

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., & Mercer, R. L. (1993)
The mathematics of statistical machine translation: Parameter estimation.
Computational linguistics 19(2), 263-311.

Daniel Dahlmeier and Hwee Tou Ng (2012).
Better evaluation for grammatical error correction.
Association for Computational Linguistics 568-572.

Mariano Felice and Ted Briscoe (2015).
Towards a standard evaluation method for grammatical error detection and correction.
HLT-NAACL 578-587.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault (2015).
Ground truth for grammatical error correction metrics.
Association for Computational Linguistics 588-593.

Questions?