Two Phase Model for SMS Text Messages Refinement

Jeunghyun Byun, Seung-Wook Lee, Young-In Song, Hae-Chang Rim
Natural Language Processing Lab.,
Korea University, Seoul, Korea
{jhbyun, swlee, song, rim}@nlp.korea.ac.kr

Abstract

In this paper, we propose a new model for refining SMS text messages where two different kinds of grammatical errors frequently occur together. A two-phase approach based on the divide and conquer strategy is presented where HMM-based model is used for correcting spacing errors in the first phase, and rule-based correction model is used for correcting spelling errors in the second phase. Experimental results show that the proposed approach yields better performance than the translation based approach.

Introduction

Over recent decades, SMS messaging services have grown up to one of the most popular way to communicate between people. Specifically, due to the ubiquitous attribute of a mobile phone, SMS messaging services are strongly preferred for notifying information to a person, for example, a notice on change of schedule or a request for a meeting. Because some of such information is very important to a user, he or she might select and store them for later use.

Some advanced NLP technologies, such as spam filtering, information extraction and summarization, can be useful to reduce such human's efforts for managing SMS messages. However, to adopt such a technology to the SMS domain, we first have to find an answer for the question: *How can we effectively refine very noisy and ungrammatical SMS texts into grammatical ones?* Generally, SMS messages contain many grammatical errors, and it can significantly decrease the effectiveness of technology requiring linguistic analysis such as POS tagging or named entity recognition.

Because a SMS message often contains multiple and complex grammatical errors, refining SMS messages is not a trivial problem. Let us consider an English SMS message *'lemme c'*, the erroneous sentence of *'let me see'*. In this example, we can find one spacing error (*'let me'*→*'lemme'*) and two spelling errors (*'let*→*lem'* and *'see*→*c'*). Furthermore, two different types of errors occur together in the first single word *'lemme'* in the message. For such a very noisy text, the traditional approach for grammatical error correction, such as noisy channel model (Brill and Moore 2000; Toutanova and Moore 2001), may not be feasible; most of them try to identify and correct an error of a word based on its surrounding context, but as shown in the example, there may be no valid context for every word in a SMS message.

In addition, combined errors of spacing and spelling in SMS messages make this problem more difficult to solve. Without spelling correction, the existing approach for spacing error correction (Kang 2000; Lee, Rim, and Yook 2007) may suffer from noisy context problem caused by spelling errors because they train the spacing error correction model from the corpus which does not have any spelling errors. However, the spelling correction without valid spacing information may also have a trouble by the same reason.

For the error correction of SMS messages, Aw et al. (2006) recently propose a statistical model which can correct both spacing and spelling errors at once. They regard the error correction problem of SMS messages as a translation problem from SMS language to the normal language and built a phrase-based statistical translation model from aligned corpus consisting of raw messages and the manually revised ones. Although the approach is rational and shows a promising result in the experiments on English, there still remain a room to improve: resolving both spelling and spacing errors in a single phase can significantly increase the size of model parameters, and as a result, it can cause a serious data sparseness problem, specifically in the case of very morphologically productive languages such as Korean and Finnish.

In this paper, we present a new two-phase approach based on the divide and conquer strategy for refining SMS messages. The main idea of our approach is quite simple: if we can remove some of errors which can be accurately corrected with noisy context at the first step, we can refine remaining errors more robustly with less noisy context information.

Methodology

Our proposed method refines a SMS message based on the following processes: in the first phase, spacing errors in an input SMS message are initially corrected by using noisy context. For this purpose, we use a statistical spacing model trained from *partially revised SMS messages* where all spacing errors are manually corrected but spelling errors still remain. Then, in the second phase, we try to fix spelling errors by using correction rules automatically extracted from pairs

of a partially revised SMS message and its corresponding *fully revised reference*. To train a model and extract correction rules, we manually built two different kinds of revised corpus, partially revised corpus and fully revised corpus, for the same SMS message corpus. We will provide further descriptions for each phase in detail.

HMM-based Spacing Model

To correct spacing errors in the first phrase, we select the HMM-based spacing approach proposed by Lee, Rim, and Yook (2007) because it shows very robust performance with limited information. In order to train a model with noisy data where only a spacing error is corrected, such characteristics of this model are very attractive.

The following equation shows the definition of HMM-based spacing model that we use:

$$p(u_{i,n}|c_{i,n}) \approx \prod_{i=1}^{n} \{p(c_i|c_{i-2,i-1}, u_{i-2,i-1}) \times p(u_i|c_{i-1,i}, u_{i-2,i-1})\} \quad (1)$$

where c denotes a character, u means a word-spacing tag indicating whether the current and next character should have a space between them ($u = 1$) or not ($u = 0$). n indicates length of an input message.

Rule-based Spelling error Correction Model

In the second phase, we simply use error correction rules defined as follows:

$$R =< C_l, S_i, C_r, S_c > \quad (2)$$

where S_i indicates an incorrect syllable found in a message, S_c denotes the correct syllable for S_i, and C_l and C_r means the left and right context of S_i respectively. C_l and C_r can be a NULL or space as well as a syllable. For example, the rule <'le','m','␣me','t'>can correct '*lem me*' to '*let me*'.

We automatically extract the correction rules from pairs of a partially revised SMS message and its spelling revised reference by using Candidate-Elimination Algorithm (Byun, Park, and Rim 2007). Different from the first phase, an error-free corpus not including any spelling or spacing error is used for rule extraction. Table 1 shows our rule extraction method. In the table, P_i indicates the ith partially revised SMS message, F_i indicates the ith fully revised reference for P_i.

We apply the correction rules with the longest matching strategy when refining a SMS message.

Experiments

To evaluate our proposed approach, we have conducted several experiments on Korean SMS message corpus, which consists of about 100,000 raw SMS messages and their revised messages. The SMS messages used in our experiments were gathered from real users. In order to compare the previous approaches, we have reimplemented the translation-based approach proposed by Aw et al. (2006) and used it as a baseline system.

Table 1: Candidate-Elimination Algorithm to Extract Correction Rules

1. Generate every possible correction rule candidate with various context size by comparing every P_i and corresponding F_i.
2. Apply every correction rule candidate to the partially revised corpus, and then estimate the precision of each rule.
3. Eliminate rule candidates whose accuracy is lower than the threshold.
4. Select the rule candidate with the shortest context among the candidates which can correct same spelling errors.

Table 2: Performance comparison between the proposed method and Aw et al. (2006). In the table, Acc(W) and Acc(M) denote word-unit accuracy and message-unit accuracy respectively.

	Acc(W)	Acc(M)
Input text	16.15%	1.16%
After only spacing error correction	65.55%	27.97%
After only spelling error correction	24.29%	6.61%
After spacing and spelling error correction(Proposed method)	86.44%	72.02%
Aw et al. (2006)	32.00%	12.78%

Table 2 shows the experimental results. Input text is very noisy as shown in the table. Compared to the baseline method, our method performs better in both word-unit accuracy and message-unit accuracy. While the proposed model is much simpler than the translation-based model, the experimental results show that our two phase approach utilizing both partially-revised and fully-revised corpus is quite effective for refining very noisy SMS-messages.

References

Brill, E., and Moore, R. C. 2000. An improved error model for noisy channel spelling correction. In *ACL '00: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics*, 286–293. Morristown, NJ, USA: Association for Computational Linguistics.

Byun, J.; Park, S.-Y.; and Rim, H.-C. 2007. Automatic spelling correction rule extraction and application for spoken-style korean text. In *Proceedings of the 6th International Conference on Advanced Language Processing and Web Information Technology*, 195–199.

Kang, S.-S. 2000. Eojeol-block bidirectional algorithm for automatic word spacing of hangul sentences. *Journal of Korea Information Science Society* 27(4):441–447.

Lee, D.-G.; Rim, H.-C.; and Yook, D. 2007. Automatic word spacing using probabilistic models based on character n-grams. *IEEE Intelligent Systems* 22(1):28–35.

Toutanova, K., and Moore, R. C. 2001. Pronunciation modeling for improved spelling correction. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 144–151. Morristown, NJ, USA: Association for Computational Linguistics.