PARAPHRASE DETECTION ON SMS MESSAGES IN AUTOMOBILES

Wei Wu[1], Yun-Cheng Ju[2], Xiao Li[2], Ye-Yi Wang[2]

[1]University of Washington, [2]Microsoft Research, Washington, U.S.A.

ABSTRACT

Voice search technology has been successfully applied to help drivers reply SMS messages in automobiles, in which a predefined SMS message template set is searched with ASR hypotheses to form the reply candidate list. In order to efficiently organize the SMS message template set and improve the quality of the reply candidate list, we proposed to apply n-gram translation model and logistic regression to detect paraphrase SMS messages. Both of the proposed algorithms outperform the edit distance based paraphrase detection baseline, brining 40.9% and 50.5% EER reduction (relative), respectively.

Index Terms— SMS message, paraphrase detection, n-gram, translation model, logistic regression

1. INTRODUCTION

A novel application of automatic speech recognition (ASR) is to help drivers reply SMS messages. However, the ASR dictation performance can be significantly degraded by the challenging acoustic environment in automobiles. Furthermore, it is inconvenient and even dangerous for drivers to correct ASR errors while driving [1]. An alternative approach is to use voice search: the ASR hypotheses are used as queries to search through a predefined SMS message template set, and the top *N* retrieved templates are returned as reply candidates. Our previous work showed that the voice search approach consistently outperforms the ASR dictation approach [2].

A potential problem in the current voice search based SMS message reply system is redundancy in the SMS message template set. For instance, in Fig. 1, all templates have essentially the same meaning. During the retrieval process, many paraphrase templates will be included in the top *N* candidates list, which actually gives the user fewer choices. As the template set increases when more templates have been extracted and accumulated from SMS message data, the redundancy will become a more serious problem. It not only increases search cost, but may also lead to accuracy degradation by letting too many paraphrase templates crowd the correct template out of the top *N* candidate list. Therefore, paraphrase detection is required for more efficiently organizing the SMS message template set.

I'll be there in <d> minutes.
Yes, I'll be there in <d> minutes.
Yep, I'll get there in <d> minutes or so.
Be there in like <d> minutes.
I will arrive within <d> minutes.

Fig.1. Example of redundancy in the SMS reply template set

I **can** be there on time.
I **can't** be there on time.

(a) Non-paraphrase template pairs with small edit distance

I'm running a **few** **minutes** late.
I **will** **be** a **little bit** late.

(b) Paraphrase template pairs with large edit distance

Fig.2. Failed examples of edit distance based SMS template paraphrase detection

In recent years, paraphrase detection has been studied for preparing monolingual text-to-text rewritten corpora. In [3], multiple sequence alignment (MSA) [4] is employed to cluster paraphrase sentences. In [5] [6], edit distance has been applied for paraphrase extraction in news corpora. All of the above methods use a heuristic rule by extracting paraphrase from news articles in different sources from the same time period. It assumes that sentences with small edit distance tend to describe similar news events, and therefore are highly probable to be paraphrases. However, in SMS template set, we cannot use such heuristics. In addition, since most of the SMS templates are short, edit distance along is not a reliable measure for paraphrase detection (see examples in Fig. 2). In later literatures, more sophisticated features or models have been applied, such as SVM with morphological and synonymy features [7] and dissimilarity significance [8]. These methods are mainly designed for paraphrase extraction from generic news articles. However, data sparseness associated with a small labeled training set in our task prevents us from applying complicated feature extraction and modeling approaches. To address the constraints of the SMS message template set, we proposed n-gram translation model and logistic regression model for paraphrase detection.

The remainder of this paper is organized as follows. Section 2 and Section 3 introduce the n-gram translation model and logistic regression model for paraphrase detection, respectively. Experiments and results are

presented and analyzed in Section 4, followed by conclusions and future work in Section 5.

2. N-GRAM TRANSLATION MODEL BASED PAPRAPHASE DETECTION

Previous studies [6] [9] showed that machine translation technique can be successfully extended to paraphrase generation. Paraphrase sentences are the translation result from the original sentences. Inspired by these studies, we employ n-gram translation model for paraphrase detection.

We start with labeled paraphrase template pairs in the training set. The template pair *(S, T)* is aligned monotonically to obtain a sequence of word pairs as follows,

$$(S,T)=((s_1,t_1),(s_2,t_2),...,(s_L,t_L))$$

where *null* word is added if necessary. Each word pair (s_i, t_i) is treated as a single semantic unit and is used to train a standard n-gram language model. The probability of an aligned paraphrase template pair is

$$P((S,T))=\prod_i P((s_i,t_i)\,|\,(s_{i-n+1},t_{i-n+1}),...,(s_{i-1},t_{i-1})) \quad (1)$$

The initial alignment is obtained by minimizing edit distance. Then the alignment and n-gram model is iteratively updated with maximizing likelihood. The n-gram translation model exploits word-level paraphrases in unigrams with higher order n-gram context.

To exploit discriminative knowledge from non-paraphrase template pairs in the training set, we also trained an n-gram translation model with labeled non-paraphrase template pairs as an anti-model. During the paraphrase detection, the detection score is computed as

$$s((S,T))=\log P((S,T))-w\log P_{anti}((S,T)) \quad (2)$$

where w is the anti-model weight. It is tuned on the developing set by minimizing detection error rate.

3. LOGISTIC REGRESSION BASED PARAPHRASE DETECTION

As shown in Fig. 2 (a), there are many template pairs which share formal similarity but are non-paraphrases due to discrepancy on a few key word pairs. To deal with such cases in the paraphrase detection, we need a model with more discriminative power. We propose to use logistic regression to train a discriminative paraphrase detection model. A logistic regression model is defined as

$$P(Y=1\,|\,f)=\frac{1}{1+\exp\left(w_0+\sum_{i=1}^{n}w_i f_i\right)} \quad (3)$$

$$P(Y=0\,|\,f)=\frac{\exp\left(w_0+\sum_{i=1}^{n}w_i f_i\right)}{1+\exp\left(w_0+\sum_{i=1}^{n}w_i f_i\right)} \quad (4)$$

where

$$Y=\begin{cases}1, \text{paraphrase}\\ 0, \text{non}-\text{paraphrase}\end{cases}$$

f_i is the i-th feature, and w_i is its correspondent model parameter. The logistic regression model is trained with gradient ascent with *L2* normalization.

3.1 "Part-of-message" enhanced edit distance alignment

There are two steps to extract features for logistic regression based paraphrase model: 1) aligning template pairs 2) extracting word pair related features from the template pair alignment. Traditional edit distance alignment doesn't take into account of semantic knowledge, thus cannot always produce proper alignment results. To address this problem, we define "part-of-message" to analyze the semantic structure of SMS messages. "Part-of-message" classifies words in SMS messages into eight semantic types, as shown in Table 1.

Table 1 Definition of "part-of-message" tags

Tag	Description	Examples
P	common prefix, usually appears at the beginning of an SMS message	*yes, no, sure*
S	subjective words	*I, you, I'll*
V	verb and status words	*get there, on my way*
T	time words	*in <d> minutes, soon*
L	location words	*in front of your building, Starbucks*
Q	question words	*when, can you, what's up*
C	condition words	*if, when, as soon as*
O	others, consists of words other than the above types	*cool, hmm, darling*

We trained a linear conditional random field (CRF) model [10] to label "part-of-message" tags for SMS message templates. The word level labeling accuracy is 87.6%. After we obtained the "part-of-message" labeling, template pairs were aligned by minimizing "part-of-message" enhanced edit distance, in which the word pair alignment cost is defined as

$$C_{POM-Enhanced}(s_i,t_i)=C(s_i,t_i)+C_{POM}(POM(s_i),POM(t_i)) \quad (5)$$

where $C(s_i, t_i)$ is the traditional edit distance alignment cost between word s_i and t_i; $POM(s_i)$ and $POM(t_i)$ are "part-of-

message" tags of word s_i and t_i; $C_{POM}(POM(s_i), POM(t_i))$ is the alignment cost between two "part-of-message" tags, which is defined heuristically. Since words corresponding to ***V***, ***T***, ***L***, ***Q*** and ***C*** contain key information in SMS messages, aligning one of them to a word with a different "part-of-message" tag is assigned a higher alignment cost. Table 2 shows the heuristically defined "part-of-message" alignment cost.

Table 2 "Part-of-message" alignment cost

	V	***T***	***L***	***Q***	***C***	***P***	***S***	***O***	***NULL***
V	0	2	2	2	2	2	2	2	1
T	2	0	2	2	2	2	2	2	1
L	2	2	0	2	2	2	2	2	1
Q	2	2	2	0	2	2	2	2	1
C	2	2	2	2	0	2	2	2	1
P	2	2	2	2	2	0	1	0	0
S	2	2	2	2	2	1	0	1	0
O	2	2	2	2	2	0	1	0	0
NULL	1	1	1	1	1	0	0	0	0

3.2 Features

After obtaining the template pair alignment, the following three groups of features are extracted for logistic regression based paraphrase detection model,

1. Word pair n-gram

Word pair n-grams $((s_{i-n+1}, t_{i-n+1}), ..., (s_i, t_i))$ in aligned template pairs are extracted as features. For a specific template pair alignment, the word pair n-gram feature is defined as

$$f\left(\left(s_{i-n+1}, t_{i-n+1}\right), ..., \left(s_i, t_i\right)\right) = \begin{cases} 1, \text{if } \left(\left(s_{i-n+1}, t_{i-n+1}\right), ..., \left(s_i, t_i\right)\right) \text{exists in the alignment} \\ 0, \text{if} \left(\left(s_{i-n+1}, t_{i-n+1}\right), ..., \left(s_i, t_i\right)\right) \text{ doesn't exist in the alignment} \end{cases}$$

To minimizing the damping effect of the overwhelming number of identical word pairs, we only use non-identical word pair n-grams. For example, word pair n-grams as ((be, be)), ((there, there)) and ((be, be), (there, there)) are discarded, while ((be, get)) and ((be, get), (there, there)) are kept.

2. Identical word pair ratio

The identical word pair ratio is defined as the number of identical word pairs in the template pair alignment divided by the alignment length. It evaluates the formal similarity between the two templates.

3. "Part-of-message" discrepancy

For each of the eight "part-of-message" tag P, we define its "part-of-message" discrepancy feature as

$$f(P) = \begin{cases} 1, P \text{ exists or doesn't exist in both templates} \\ 0, P \text{ exists in only one of the two templates} \end{cases}$$

4. EXPERIMENTS AND RESULTS

4.1. Data description

We built the SMS message paraphrase detection corpus by randomly selecting a group of frequently used SMS message templates from the template set (containing 7,412 SMS templates), and used each of them as a query to retrieve a subset of templates. The retrieval was performed with vector space model (VSM) using TF-IDF representation. The retrieved templates were hand labeled as "paraphrase" or "non-paraphrase" to the query template. We divided the corpus into training, developing, and testing set, as shown in Table 3.

Table 3 SMS message template paraphrase detection corpus

Size	Paraphrase pairs	Non-paraphrase pairs
Training set	1,734	5,188
Developing set	7,66	2,246
Testing set	1,149	3,368

4.2. Experiments

We use the equal error rate (EER) of false accept rate (FAR) and false reject rate (FRR) of paraphrase detection to evaluate the detection performance. The FAR and FRR are defined as

$$FAR = \frac{\text{number of false accepted non - paraphrase pairs}}{\text{number of non - paraphrase pairs}}$$

$$FRR = \frac{\text{number of false rejected paraphrase pairs}}{\text{number of paraphrase pairs}}$$

EER is the rate when FAR and FRR equate.

We first evaluated n-gram translation model for paraphrase detection. We tested its performance with different template pair alignment schemes, including the traditional edit distance alignment [1], iteratively n-gram model updated alignment (TM iterative alignment) and "part-of-message" enhanced edit distance alignment (POM alignment). Under each alignment scheme, we compared the EER obtained with and without n-gram translation anti-model. Experimental results are presented in Table 4. It is shown that the n-gram translation anti-model, which introduces a discriminative property to the detection algorithm, can significantly reduce the EER under all alignment schemes. With the n-gram translation anti-model, the POM enhanced alignment obtains the best performance among the tested settings, with an EER of 17.8%.

Table 5 presents the paraphrase detection results of logistic regression. We compared the performance obtained with traditional edit distance alignment and "part-of-message" enhanced edit distance alignment. The traditional method uses word pair 1-gram, 2-gram and identical word

[1] The insertion, substitution and deletion cost are set to be 1

Table 4 Paraphrase detection results of n-gram translation model

EER (%)	2-gram TM	2-gram TM + Anti-Model
Edit distance alignment	24.5	19.1
TM iterative alignment	24.1	18.5
POM alignment	26.5	**17.8**

Table 5 Paraphrase detection results for logistic regression

Algorithm	Number of features	EER(%)
Logistic regression + Edit distance alignment	17,345	16.7
Logistic regression + POM alignment	13,660	14.9

Table 6 Comparison of paraphrase detection performance among edit distance, n-gram translation model and logistic regression

Algorithm	EER (%)
Edit distance	30.1
2-gram TM + Anti-model + POM alignment	17.8
Logistic regression + POM alignment	**14.9**

pair ratio as features; while our enhanced method uses word pair 1-gram, 2-gram, identical word pair ratio and "part-of-message" discrepancy as features. It is shown that with "part-of-message" enhanced edit distance alignment, the number of features (mostly word pair n-grams) extracted from the training set is reduced by 21.2%. This huge feature size reduction is due to the semantic knowledge introduced by "part-of-message" in the alignment, which reduces the alignment confusability and produces fewer "bad" word pair n-grams caused by incorrect alignment. Reducing the feature size not only saves training time, but also reduces the number of parameters to be estimated in the model, alleviating the over-training problem caused by data sparseness. It is shown that logistic regression with "part-of-message" enhanced edit distance alignment obtains a 10.8% relative EER reduction over traditional edit distance alignment.

Finally, we compared the paraphrase detection performance of n-gram translation model and logistic regression. We used edit distance based paraphrase detection (using edit distance between two templates as the paraphrase detection score) as baseline. The results are presented in Table 6. It is shown that both the n-gram translation model and logistic regression greatly outperform the edit distance based paraphrase detection, by a 40.9% and a 50.5% relative EER reduction, respectively. Logistic regression with "part-of-message" enhanced edit distance alignment produces the best result against the testing set, with a paraphrase detection EER of 14.9%.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed n-gram translation model and logistic regression for paraphrase detection in SMS message templates. Both algorithms greatly outperform the edit distance based paraphrase detection baseline. We defined "part-of-message" tags for analyzing the semantic structure of SMS message templates and introduced the "part-of-message" enhanced edit distance alignment. Experimental results show that the "part-of-message" enhanced edit distance alignment can significantly improve the alignment accuracy, producing better paraphrase detection performance.

Currently, data sparseness due to the small training set prevents us from using more sophisticated feature extraction and modeling schemes. In future work, more SMS message data should be collected and labeled to alleviate the data sparseness problem. Feature selection and phrase based machine translation model can be tried with sufficient training data. Paraphrase detection should also be incorporated into the voice search based SMS message reply system to improve its performance.

11. REFERENCES

[1] A. Kun, T. Paek & Z. Medenica, "The effect of speech interface accuracy on driving performance", in *Proc. of Interspeech'07*, 2007.

[2] Y.-C. Ju and T. Paek, "A voice search approach to replying to SMS messages in automobiles", in *Proc. of Interspeech'09*, 2009.

[3] R. Barzilay and L. Lee, "Learning to paraphrase: an unsupervised approach using multiple-sequence alignment", in *Proc. HLT-NAACL'03*, 2003.

[4] R. Durbin, S. R. Eddy, A. Krogh and G. Mitchison, "Biological sequence analysis: Probabilistic models of proteins and nucleic acids ", Cambridge University Press, 1998.

[5] B. Dolan, C. Quirk and C. Brockett, "Unsupervised construction of large paraphrase corpora: exploit massively parallel news sources", in *Proc. COLING'04*, 2004.

[6] C. Quirk, C. Brockett and B. Dolan, "Monolingual machine translation for paraphrase generation", in *Proc. EMNLP'04*, 2004.

[7] C. Brockett and B. Dolan, "Support vector machines for paraphrase identification and corpus construction", in *Proc. IWP'05*, 2005.

[8] L. Qiu, and M.-Y. Kan, and T.-S. Chua, "Paraphrase recognition via dissimilarity significance classification", in *Proc. of EMNLP'06*, 2006.

[9] X. Li, and Y.-C. Ju, and G. Zweig, and A. Acero, "Language modeling for voice search: a machine translation approach", in *Proc. ICASSP'08*, 2008.

[10] J. Lafferty, and A. Mccallum, and F. Pereira, "Conditional random fields: probabilistic models for segmenting and labeling sequence data", in *Proc. ICML'08*, 2001.