A Monte-Carlo approach to Automatic Metamorphic Testing of Machine Translation Software

Daniel Pesu
4726686
dp604@uowmail.edu.au

Kieran MacRae
4493217
km433@uowmail.edu.au

Daniel Barnes
5053511
db059@uowmail.edu.au

Shixin Wang
4945815
sw173@uowmail.edu.au

Boyang Yan
4329764
by932@uowmail.edu.au

October 23, 2017

I. Introduction

The task of assessing the quality of machine translation software is naturally difficult, due to the lack of test oracle. This is known as the *oracle problem*. To address the inability in addressing translations, metamorphic testing can be applied to generate a suitable test case such that the metamorphic relations of translations are not violated.

In this study, we will focus on the machine translation services:

- Google,
- Bing, and
- Youdao,

and will attempt to use a metamorphic approach to quantify the *consistency* of each of these services as well as rank their general performance on a select number of focus languages. In this study we consider

- English,
- Chinese,
- Japanese,
- Korean,
- French,
- Russian,
- Portuguese,
- Spanish, and
- Swedish.

II. Method

A common approach is to take an initial string S and perform a two-way translation of the string from a target language back to the original language, resulting in S'. Then a comparison is made between the two strings, S and S', to assess their similarity. This process is known as Round Trip Translation (RTT) is pointed out by Somers (2005, p. 130) to be a poor predictor of translation quality.

Other methods involve human-assisted approaches to evaluate machine translation (Nießen, et al n.d.) which compromise efficiency in order to obtain close to human evaluations.

In our approach, we implement a One-Way (or *Uni-Directional*) method for evaluating the quality of each translation service which performs the comparison process at the targeted language domain. By doing so, we avoid the misleading performance of RTT methods, while maintaining an entirely automatic method for performing these evaluations.

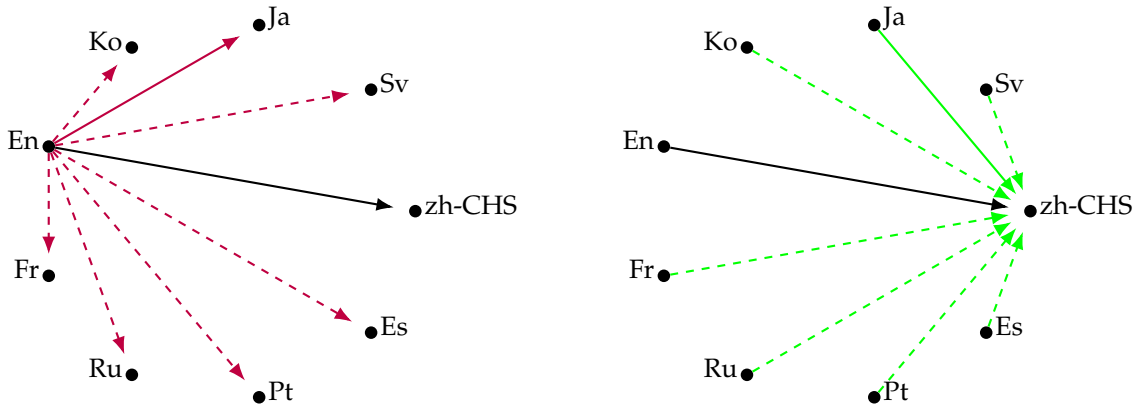

Figure 1: *Illustration of the Uni-Directional method for the case when the target language is Chinese. Initially, a test phrase is translated from English to every other language (left), then a randomly selected language, in this instance, Japanese, is also translated to Chinese (right). The resulting two translations are then compared and scored.*

Metamorphic Procedure

To begin, suppose a phrase P is to be translate from English to some target language L. Then the direct translation using the translation service T is denoted $P \xrightarrow{T} P_L$. Next a Monte-Carlo approach is used to obtain a base-line measure for the quality of this translation. In this, a randomly selected language $\mathcal{M}$ is chosen, such that $\mathcal{M} \neq L$, and the 'side-translation' $P \xrightarrow{T} P_{\mathcal{M}} \xrightarrow{T} P'_L$ is obtained.

From this we can use a comparison function to compare the two translations P_L and P'_L in the domain of the language L rather than the origin language, English. Under a *consistent* translator T we would expect these two results to be exactly equal, thus giving the metamorphic relation:

$$P_L = P'_L \tag{1}$$

This process is illustrated in Figure 1, and the general process is given in Algorithm 1.

Algorithm 1 Uni-Directional method

function EVALUATE(*Translator*, *language*, *phrase*)
  Translate *phrase* to *language* as *direct*.
  Select a random language, $l \neq$ *language*.
  Translate *phrase* to l as *intermediate*.
  Translate *intermediate* to l as *alternate*.
  score $\leftarrow$ **Compare** *direct* with *alternate*.
  return *score*.
end function

Comparison Procedure

In order to make meaningful comparisons between the two translations P_L and P'_L, a set of metrics are used to measure the similarity of these results. In total, 3 metrics were used

- **Levenshtein Distance**, is the number of character-wise insertions, edits, or deletes needed to make one translation identical to another.
- **BLEU**, which uses n-grams to determine the matching number of sub-sequences in each translation.
- **Cosine Similarity**, which vectorises the two phrases and calculates the angle between them.

Each metric produces a score between 0 and 1, where 1 represents a perfect match, and 0 represents two entirely different translations. These metrics are calculated and recorded for each pair of translations.

Test Case Generation

To generate the large number of tests required, a list of sentences are randomly selected from Wikipedia and pre-scanned before being stored into a file. This process was repeated to obtain 1,000 test phrases.

To apply the metamorphic test procedure to all of these phrases would result in

$$1,000 \text{ (phrases)} \times 3 \text{ (translators)} \times 8 \text{ (languages)} = 24,000 \text{ observations.}$$

Rather than performing the test on all of these phrases, a small subset of phrases was randomly selected and the test procedure was carried out until either the trial period for a service ran out, or some quota was reached. In the end, the final number of generated observations was 5,642.

Had we been able to use perform each of these tests without any quota restrictions, then we would have performed all 24,000 observations.

III. Model

The statistical model for the score of each translation is the following

$$S_{ij} = \mu + L_i + T_j + (LT)_{ij} + \varepsilon_{ij}. \qquad (2)$$

where,

- S_{ij} is the average score from the three metrics for the ith language from the jth translator,
- μ is the mean for all observations,
- L_i is the *main effect* from the ith language,
- T_j is the *main effect* from the jth translator,
- $(LT)_{ij}$ is an *interaction* term for the effect of the ith language on the jth translator, and
- ε_{ij} is the *random error* for each observation, assumed iid. $N(0, \sigma^2)$.

Analysis was performed using the statistical package `SAS` to find the values for each of these effects.

Table 1: *Least Squares Estimates for Translator Effects*

Translator	Estimate	Standard Error
T_{Google}	0.6321	0.03089
T_{Bing}	0.6355	0.03088
T_{Youdao}	0.5800	0.03169

From Table 1 we can see that Bing has the highest average score (0.6355) across all test cases, followed by Google (0.6355) and Youdao (0.5800). This is graphed in Figure 2.

To identify if any of these average scores are significantly different, the following hypothesis test was conducted

$$\begin{cases} H_0 : T_1 = T_2 = T_3, & \text{against} \\ H_1 : T_i \neq T_j & \text{for some } i \neq j. \end{cases}$$

The test found a significant difference between at least one pair of main effects ($F(2, 5608) = 8.66$, $p = 0.0002$) at the 5% significance level.
To identify which of the translators differ, multiple comparisons analysis was conducted using a Tukey-adjust family-wise error rate of 0.05. The post-hoc analysis found Youdao to have a significantly lower mean score than both Google ($t(5608) = 3.71$, Adj. $p = 0.0006$) and Bing ($t(5608) = 3.80$, Adj. $p = 0.0004$), but no significant difference between Google and Bing ($t(5608) = 0.12$, Adj. $p = 0.9920$).
The results of this analysis are summarised in Table 2.

Table 2: *Homogeneous subsets for translation score*

Translator	A	B
Bing	0.6355	
Google	0.6321	
Youdao		0.5800

Groups not appearing in the same column are significantly different from one another.

In a similar manner, we investigate the

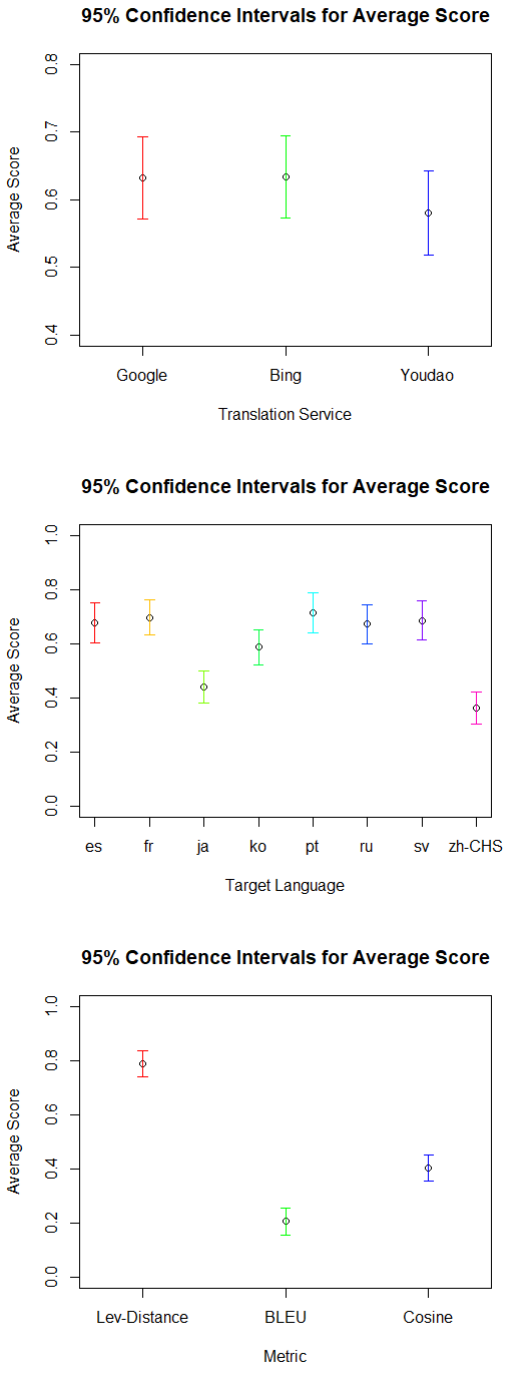

Figure 2: *Confidence intervals for the average score of each translator across all observations (above), the average score of each language across all translators (middle), and the average score for each metric for the languages ja, ko, and zh-CHS (below).*

difference in mean translation scores across each language to determine which, if any, are significantly different from one another. The estimated values for each of these effects are given in Table 3

Table 3: *Least Squares Estimates for Language Effects*

Translator	Estimate	Standard Error
L_{es}	0.6776	0.03704
L_{fr}	0.6973	0.03295
L_{ja}	0.4406	0.03007
L_{ko}	0.5878	0.03295
L_{pt}	0.7149	0.03704
L_{ru}	0.6724	0.03705
L_{sv}	0.6856	0.03703
$L_{\text{zh-CHS}}$	0.3625	0.03007

Now, to identify if any of these average scores are significantly different, the following hypothesis test was conducted

$$\begin{cases} H_0 : L_1 = L_2 = \ldots = L_8, & \text{against} \\ H_1 : L_i \neq L_j & \text{for some } i \neq j. \end{cases}$$

The test found a significant difference between at least one pair of main effects ($F(7, 5608) = 396.84$, $p < 0.0001$) at the 5% significance level.

To identify which of these languages significantly differ, the same post-hoc analysis procedure was conducted, with the results summarised in Table 4. From this we find that Chinese is the worst performing language for the translation services. In particular, there appears to be a trend of Asian languages performing worse off compared to European languages. Of these European languages, Portuguese had the highest average score (0.7419).

To investigate this further, the analysis was again performed but considering only Chinese, Japanese, and Korean. The average score for each metric was then calculated and is plotted in Figure 2. From this we can see that both the

Table 4: *Homogeneous subsets for translation score*

Language	A	B	C	D
pt	0.7419			
fr	0.6973			
sv	0.6856			
es	0.6776			
ru	0.6724			
ko		0.5878		
ja			0.4406	
zh-CHS				0.3625

Groups not appearing in the same column are significantly different from one another.

BLEU and Cosine metrics perform noticeably poorly whereas the Levenshtein Distance metric is largely unaffected. This is likely due to the fact that the Chinese translations tend to be long strings with no spaces, whereas, both BLEU and Cosine are word-wise metrics. As such, any minute errors in the translator being used would result in majority of the words being scored as incorrect, which is even worse when the entire translation is one connected string.
The Levenshtein Distance on the other hand, being a character-wise metric, is unaffected by this.

IV. Discussion

From this proposed method and statistical model, we managed to find various patterns in the test translation services: Google, Bing and Youdao.

We were able to rank these services in terms of consistency, as well as identify which languages performed better or worse across all three of the services.

In addition to this, our study found that BLEU and Cosine metrics perform much worse when comparing Asian languages, compared to European ones, which is an unexpected, yet interesting finding.

It is recommended that further study is taken to investigate the reliability of translations between these two differing

References

[1] Somers, H 2005, 'Round-Trip Translation: What's It Good For?', in *Proceedings of the Australasian Language Technology Workshop*, Sydney, Australia.