Si-Ta: Machine Translation of Sinhala and Tamil Official Documents

Surangika Ranathunga, Fathima Farhath, Uthayasanker Thayasivam, Sanath Jayasena, Gihan Dias
Department of Computer Science and Engineering, University of Moratuwa
Katubedda 10400, Sri Lanka

{surangika, FathimaFarhath, rtuthaya, sanath, gihan}@cse.mrt.ac.lk

Abstract— **Although Sri Lanka is a multi-ethnic country with Sinhala and Tamil being the official languages, most of the population is familiar with only one of these languages. This results in a lack of Sinhala-Tamil translators, which in turn has an impact on the government agencies that are required to issue official government documents in both languages. Although Machine Translation can be considered as a possible solution, available translation systems for this language pair have a poor performance, mainly because they do not focus on official government documents. This paper presents Si-Ta, the first Machine Translation system for Sinhala and Tamil official government documents. Si-Ta uses a Statistical Machine Translation engine, and provides a user-friendly web interface. Results show that Si-Ta can be used to eliminate the need for manual translators, if the only requirement is to understand the document received in the source language. In other words, current version of Si-Ta is capable of translating without loss of semantics at a level that is enough for any common target language reader to understand the message in source language.**

Keywords— Machine Translation; SMT; Moses; Sinhala; Tamil

I. Introduction

Sri Lanka is a multi-ethnic country where Sinhala and Tamil are the official languages. However, only a small number of the population can communicate in both the languages, especially in written form. Therefore, to smoothly carry out official government communications with the public, government agencies should be able to proficiently handle both languages. Moreover, as per the constitution of Sri Lanka, all official documents such as gazettes and circulars must be issued in Sinhala and Tamil, as well as in English. In most cases, these official documents are first written in Sinhala and manually translated into the other two languages. Translation between Sinhala and Tamil has become a burden to government institutions as they lack translators proficient in Sinhala and Tamil. Although the Sri Lankan government has undertaken some initiatives to produce Sinhala-Tamil translators, this shortage is likely to continue for some time.

An alternative is to use the assistance of computers in translation between Sinhala and Tamil. In particular, a computer system can reduce the burden on the translators by providing an initial translation of a document with a reasonable accuracy, where the human translator only has to carry out necessary proofreading and corrections (post editing).

Machine Translation (MT) between major languages such as English and French can be done with a greater accuracy. However, the same is not observed for languages such as Sinhala and Tamil, simply due to the lack of research efforts and resources. Previous research on Sinhala-Tamil MT [1-5] can be considered as prototype systems, and none has yet specifically focused on translating official government documents. All these systems are based on the popular statistical machine translation (SMT) approach. Despite the fact that SMT requires a large parallel corpus (where for each sentence/phrase of the source language, there is a corresponding sentence/phrase in the target language) to be effective, these systems have employed small parallel corpora, always less than 25,000 sentences.

An alternative is to use a service such as *Google Translate* [6]. However, according to our observations, its performance on official documents is much lower than that on general documents. Furthermore, using such a service for the translation of official, possibly confidential, government documents is a concern. The *Subasa* translation system [7] too does not provide a sufficiently accurate translation. Also none of these systems has considered providing any post editing support. These factors clearly indicate the need for an efficient machine translation system with post editing support for Sinhala and Tamil.

This paper presents Si-Ta, which is a MT system with post editing support for Sinhala and Tamil, focused on official government documents. It currently targets short documents (up to two pages). The translation process starts with the input of the source document, either in Sinhala or Tamil. Si-Ta translates the source document, and highlights words that it could not translate. The human translator can manually translate these words, and fix any other translation errors she sees. Thus, instead of having to translate a document from scratch, Si-Ta allows human translators to be proofreaders, where they simply have to fix the issues they see in output.

Si-Ta employs a simple client-server architecture, with a user-friendly web interface. The back-end translation system is designed in such a way that different MT systems can be plugged in without affecting the translator's interface.

Si-Ta system is currently used by institutions that belong to the Ministry of National Integration, Reconciliation and Official Languages. The standard translation accuracy measurement matrix called BLEU (Bilingual Evaluation Understudy) score for the current system is reported at 25.05 and 32.85 for Sinhala-Tamil, and Tamil-Sinhala, respectively. Human translators reported an accuracy of 3.32 on a scale of 1-5, which indicates that the translation output conveys the intended meaning, although some amount of post editing is required. It is also shown that a human translator could save time when using Si-Ta compared to manual translation. This shows the viability of

This research was funded by the Department of Official Languages of Sri Lanka, and Senate Research Committee (SRC) Grants of the University of Moratuwa..

using Si-Ta as a translation system for Sinhala and Tamil official government documents.

The rest of the paper is organized as follows. Section II discusses related, work. Section III contains the implementation details of the Si-Ta system, and Section IV presents the system evaluation. Finally, Section V concludes the paper.

II. Related Work

For translation between local languages in Sri Lanka (Sinhala -Tamil), only a limited amount of research has been conducted. Performance of these systems is considerably very low [1-5]. One reason for this lower performance is the lack of sufficient data and linguistic resources.

A feasibility study of SMT for the Sinhala –Tamil pair was conducted with a corpus of 5,000+ parallel sentences from news articles related to politics and culture in Sri Lanka [1]. Another system with a corpus of 5,000+ parallel sentences of parliament proceedings is demonstrated by Sripirakas et al. [2]. The authors concluded that Tamil-to-Sinhala translation performs better than Sinhala-to-Tamil. Pushpananda et al. [3] investigated the behavior of SMT systems against the size of the parallel corpus. Rajpirathap et al. [4] demonstrated an analysis on the SMT system behavior with and without tuning, with a corpus of 5,000+ parallel sentences from parliament order papers as the source. Pushpananda et al. [5] extended the work of Pushpananda et al. [3], where they used the same data set as in [3] to elaborate a study on incorporating an unsupervised morphological analyzer to the system using the Mofessor algorithm [8].

Despite some of this research made use of parliament proceedings and order papers as the parallel corpus (because, as we believe, this is the readily available source for parallel data), none focused specifically on translating official government documents such as official letters. Moreover, all these systems are based on SMT.

As per functioning MT systems, there are two: *Google Translate* [6], and *Subasa* [7]. As shown later, performance of these two systems with respect to official documents is not satisfactory. Moreover, *Subasa* supports only Sinhala- Tamil translation, not vice-versa. Most importantly, these systems do not provide any post editing support.

III. The Si-Ta System

A. Corpus Creation

As mentioned earlier, the main objective of the current version of Si-Ta is to translate short documents such as official letters. However, collecting a sufficiently large parallel corpus from official letters was quite challenging. As shown in Table I, only 8360 parallel sentences could be collected from letters provided by various government institutions.

Since the amount of parallel data collected from letters was not sufficient, a reasonable amount of additional data was gathered from other government document sources such as annual reports, parliament order papers, circulars, and establishment codes. Though these were from government institutions, the writing style was different from letters described above (e.g. the parliament order papers were more like question and answer form). Thus, with respect to official letters (in-domain data), these can be categorized under pseudo in-domain.

Some source documents of in-domain and pseudo in-domain were hard copies in a single language (i.e., either the Tamil or Sinhala version of the document), while some were soft copies in PDF format. The single-language source documents in printed form were manually translated and typed. Data from PDF documents were extracted using a custom developed tool. Font issues were fixed using another custom developed tool, and the final version of the document was manually verified. Font issues were one of the prominent problems we faced in this data collection process. Parallel data was created by using a sentence alignment tool [9]. This aligned data was again manually verified before adding to the parallel corpus. Depending on the status of the original document, some of the steps of this process may not be needed. For example, if the original source document was received in text format, there is no need for typing or text extraction of the source document.

Other easily accessible data sources were from the web, (such as articles from blogs, news and wiki dumps), and other free sources. This data was collected from some freely available sources [10,11], as well as by web crawling. Yet, their context with respect to official government letters was quite different. Therefore, these were categorized as out-domain data. However, it was possible only to gather monolingual data under this category, and it was found out that the use of this monolingual data negatively affects the performance of the translation system [12].

Tables I, II, III show the statistics of data collected from different sources for parallel data, Sinhala monolingual data, and Tamil monolingual data, respectively.

TABLE I. Statistics of Parallel Data

Source	#of sentences	#of words (Sinhala)	#of words (Tamil)
In-domain	8360	114,912	103,988
Pseudo in-domain	15,946	333,400	284,741

TABLE II. Statistics of Sinhala Monolingual Data

Source	#of sentences	#of words (Sinhala)
In-domain	6,428	73,066
Pseudo in-domain	15,946	333,400
Out-domain	4,735,658	72,531,342

TABLE III. Statistics of Tamil Monolingual Data

Source	#of sentences	# of words (Tamil)
In-domain	6,428	80,849
Pseudo in-domain	76,692	788,544
Out-domain	1,525,966	21,348,15

In addition, bilingual text from three parallel lists was used [13]:

- A glossary of 19,861 terms related to government administration and operations, where the terms come from financial regulations, land administration, public administration, Air Force, Army, Navy and Police (consists of phrases of length 1-3 words, including nouns, verbs, adjectives and adverbs)
- A list of 5,291 names of government organizations and official designations (consists of nominal phrases of length of 2 -5 words, with an average of 3 words)
- A list of 19,250 terms created using a general-purpose bilingual dictionary (most of the entries were single words consisting of nouns, verbs, adjectives and adverbs).

B. System Architecture

Fig. 1 shows the architecture of the Si-Ta system. Simple client-server architecture has been employed in the design. User Interface (UI) has multiple modules to support translation, user management, organization management, etc. To achieve separation of concerns, each of these UI modules is connected to a separate module at the data management back-end. Most importantly, the MT system is well separated from the other components, which allows us to experiment with other MT systems without having to modify the rest of the system. In addition to SMT mentioned above, Neural Machine Translation (NMT) was also experimented with [14,15]. NMT has outperformed SMT in the very recent past for languages such as English and French [16]. However, NMT requires a much larger parallel corpus than SMT. Thus with the current parallel corpus of about 30,000 sentences, SMT far outperforms NMT.

C. Machine Translation Backend

As mentioned earlier, both SMT and NMT were experimented with. However, only SMT is presented here, since under the current corpus size (a low-resourced setup), SMT outperformed NMT by a significant margin as the performance of NMT tends to be inferior when the vocabulary is not closed [17].

Out of the many SMT systems, the popular SMT framework *Moses* [18] was selected as the translation engine. It is widely used, well established, and has strong community support when compared to the other SMT platforms.

The basic input requirement for Moses is Translation Model (TM) and Language Model (LM). The TM is derived through word aligned parallel data. LM is derived through a larger target side monolingual dataset.

As the initial step of this workflow, the gathered data should be tokenized. Since the freely available tokenizers did not work for Sinhala and Tamil, a tailor-made tokenizer was used. Then parallel data was filtered using standard Moses filtrations to remove the mis-aligned sentences and sentence pairs with high length ratio differences, as this could misguide the word alignment process.

Giza++ [19] was used with ‘grow-diag-final-and’ as the symmetrization heuristic (Although Mohamed et. al [20] have shown a combination of multiple alignment tehniques produced a better result, we did not experience the same with our dataset). ‘msd-bidirectional-fe’ was used as the reordering technique for the word alignment for the parallel data. Parallel data with the word alignment information data was used to generate the TM. At the TM generation process, ‘Good Turing’ was used as the smoothing technique for the phrase table score smoothing. Phrase translation score, lexical translation scores, word and phrase penalties, and linear distortion were used as features in the TM, which are commonly used features in TM [21].

Along with the target side of the parallel data, target side monolingual data was used to generate the LM. Models of order 5 (5-gram) were created using *SRILM* [22]. Based on the experiments on the impact of the data usage in creating

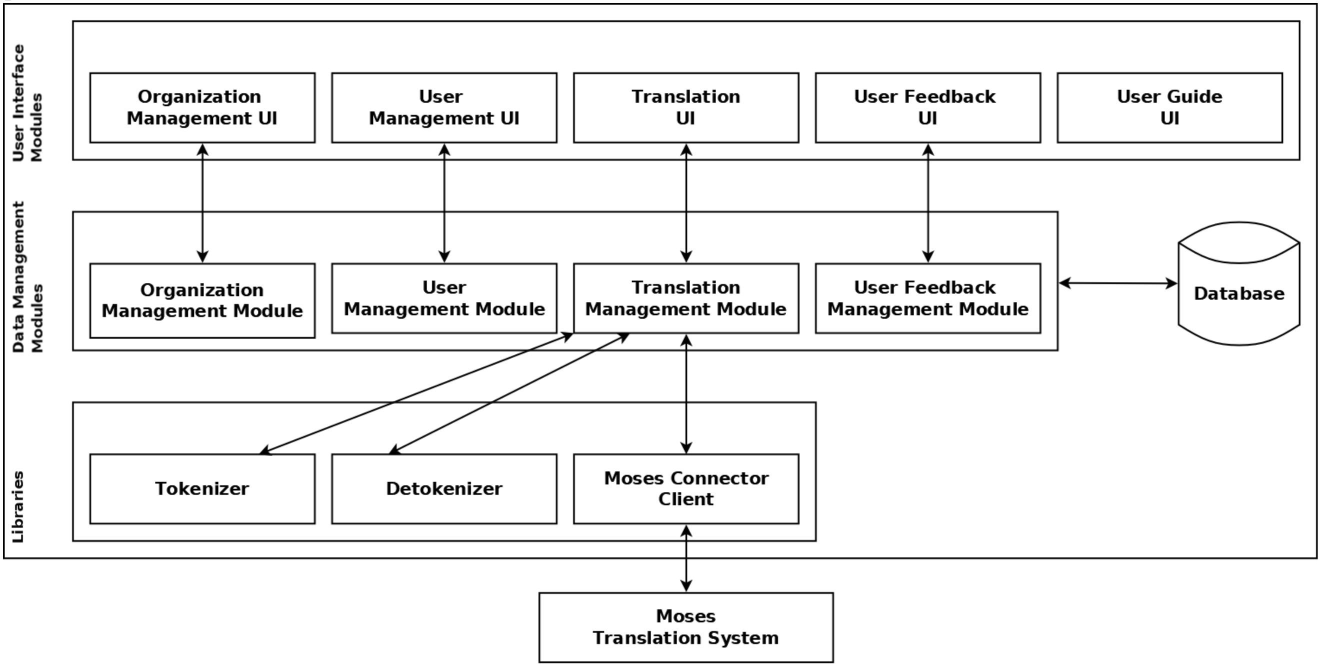


Fig. 1 Si-Ta Architecture.

different data configurations (single model with all data, log-linear interpolation of multiple LM and linear interpolation), log-linear of multiple models was selected as the best configuration. The weights of each model were adjusted at the time of tuning based on their relevance to the tuning set. Feature weight tuning was done using Minimum Error Rate Training (MERT) [23] on 100 best translations for a set of 1,000 randomly selected sentences. For the process of decoding, cube pruning techniques with a stack size of 5,000 and the maximum phrase length of 5 was used.

D. Translation Workflow and User Interface

Translation workflow (see Fig. 2) of the current Si-Ta system is catered to the needs of the organizations that use Si-Ta. However, it is fairly simple to change the same into a different process. The process starts with the head of Translation assigning translation work (sent by an external party) to the individual translators. She can either type in the course text, or import from a file. The translator enters the source text into the input box, and Si-Ta automatically detects the source language and carries out the translation. Any untranslated words are highlighted, and the human translator is able to edit this translated output. Target side has a rich editor, and it is possible to export the target text into an MS Word document with the rich formatting done using the editor. Once done, she submits the translated text for verification.

The verifier can either accept the submitted translation or refer it back to the human translator with any suggestions to improve translation. If the translation is accepted by the verifier, it is sent back to the party that requested the translation (i.e. who sent the source document). Fig. 3 shows the user interface of Si-Ta when being used for translation.

IV. Evaluation and Analysis

Training dataset of the Si-Ta system is comprised of data from the following sources: official government letters (in-domain data), data from circulars, order papers, etc. (pseudo in-domain data), and bi-lingual lists. Results when out-domain data is used are not included because it reduced the translation quality. Testing dataset consisted only of official government letters. Table IV shows the statistics of the training, validation, and testing datasets. In addition to complete sentences, parallel entries from glossaries and other lists mentioned above are included.

Testing dataset shown in Table IV was used to test the Si-Ta system that had the configurations reported in Section 3.B trained with the dataset reported in Table IV (No sample from the testing set was used to train the Si-Ta system). This same test set was used to test *Google Translate* and *Subasa* systems. Tuning data set was used to adjust the weights of the models (TM and LM) at the time of tuning. Translation quality was evaluated using the BLEU score. Results are reported in Table V.

As can be seen, for the domain of official documents, Si-Ta system far outperforms the other two systems. This result can be attributed to the type of data used to train these three systems – while Si-Ta is entirely trained with official documents, other two systems are trained with open domain data. In particular, even with a much larger corpus than ours (as we assume), *Google Translate* performs far below Si-Ta. This result shows the importance of the consideration of domain adaptation when implementing MT systems – for different domains such as medical, legal, and government, a domain-specific corpus has to be used to achieve acceptable results. This is because each domain has a terminology (and may be a language flow) specific to itself. It was also noted that Tamil-Sinhala direction worked better than Sinhala-Tamil. The reason for this is that Tamil being more inflected than Sinhala, makes the Tamil-Sinhala direction less complex than vice-versa.

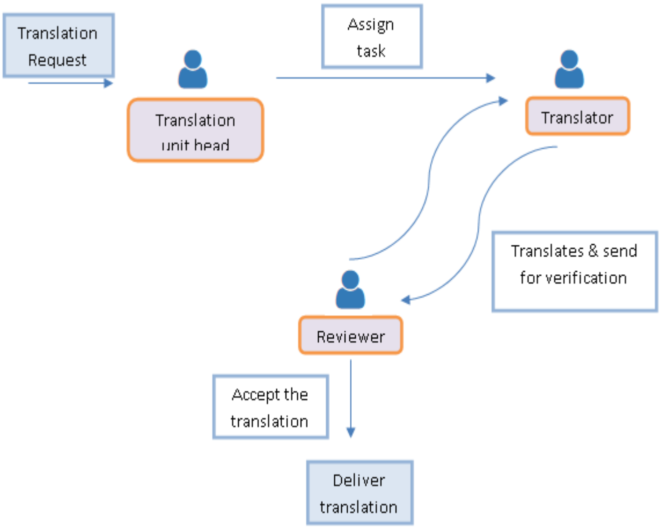


Fig. 2 Translation Workflow.

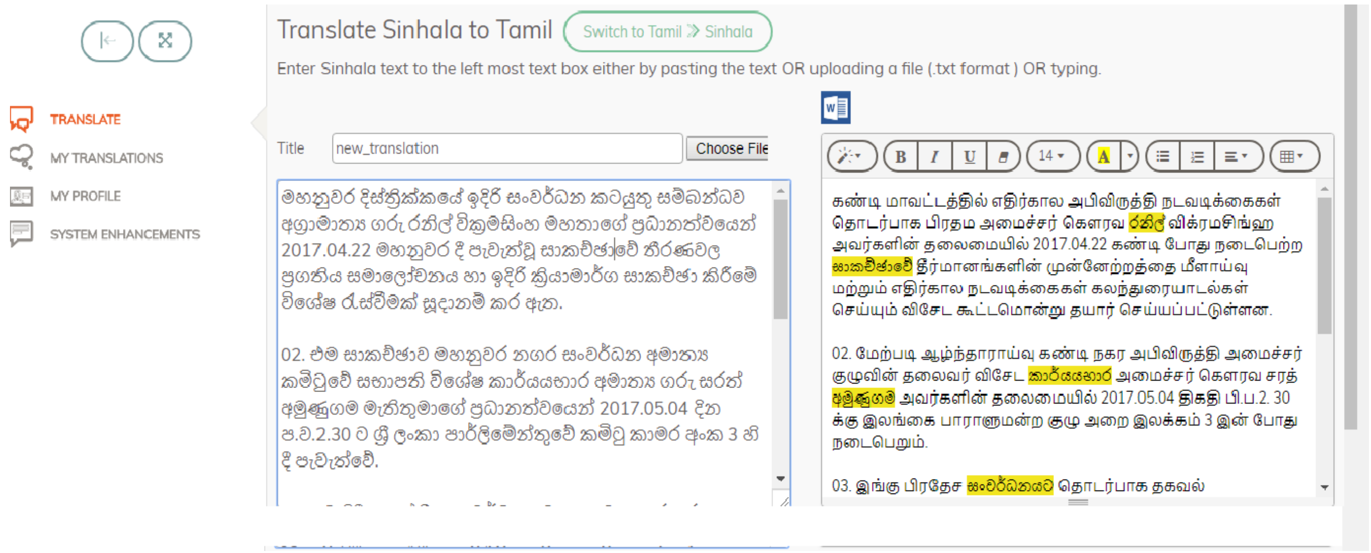
When analyzing the translation output manually, it was noticed that *Google Translate* gave acceptable translations for phrases more than for complete sentences. For longer sentences, the output had the meaning deviating from that of the original sentence. Also, some translations were literally correct, yet the output was not context appropriate. Furthermore, there were instances where the output is left blank when the system failed to translate.

TABLE IV. Statistics of Data Sets Used for Evaluation

Dataset	#Sentences
Training	23,006
Tuning	6000
Testing	100

TABLE V. BLEU Scores of Different Systems

Direction	Si-Ta	Subasa	Google
Sin-Tam	25.05	1.16	8.05
Tam-Sin	32.85	-	8.37



Translate Sinhala to Tamil

Enter Sinhala text to the left most text box either by pasting the text OR uploading a file (.txt format) OR typing.

Title new_translation

මහනුවර දිස්ත්‍රික්කයේ ඉදිරි සංවර්ධන කටයුතු සම්බන්ධව අග්‍රාමාත්‍ය ගරු රනිල් වික්‍රමසිංහ මහතාගේ ප්‍රධානත්වයෙන් 2017.04.22 මහනුවර දී පැවැත්වූ සාකච්ඡාවේ තීරණවල ප්‍රගතිය සමාලෝචනය හා ඉදිරි ක්‍රියාමාර්ග සාකච්ඡා කිරීමේ විශේෂ රැස්වීමක් සූදානම් කර ඇත.

02. එම සාකච්ඡාව මහනුවර නගර සංවර්ධන අමාත්‍ය කමිටුවේ සභාපති විශේෂ කාර්යභාර අමාත්‍ය ගරු සරත් අමුණුගම මැතිතුමාගේ ප්‍රධානත්වයෙන් 2017.05.04 දින ප.ව.2.30 ට ශ්‍රී ලංකා පාර්ලිමේන්තුවේ කමිටු කාමර අංක 3 හි දී පැවැත්වේ.

கண்டி மாவட்டத்தில் எதிர்கால அபிவிருத்தி நடவடிக்கைகள் தொடர்பாக பிரதம அமைச்சர் கௌரவ රනිල් விக்ரமசிங்ஹ அவர்களின் தலைமையில் 2017.04.22 கண்டி போது நடைபெற்ற සාකච්ඡාවේ தீர்மானங்களின் முன்னேற்றத்தை மீளாய்வு மற்றும் எதிர்கால நடவடிக்கைகள் கலந்துரையாடல்கள் செய்யும் விசேட கூட்டமொன்று தயார் செய்யப்பட்டுள்ளன.

02. மேற்படி ஆழ்ந்தாராய்வு கண்டி நகர அபிவிருத்தி அமைச்சர் குழுவின் தலைவர் விசேட කාර්යභාර அமைச்சர் கௌரவ சரத் අමුණුගම அவர்களின் தலைமையில் 2017.05.04 திகதி பி.ப.2. 30 க்கு இலங்கை பாராளுமன்ற குழு அறை இலக்கம் 3 இன் போது நடைபெறும்.

03. இங்கு பிரதேச සංවර්ධනයට தொடர்பாக தகவல்

Fig. 3 Translator's Interface.

Subasa showed the worst performance. Also the output had issues with fonts where the words were split in between (specially with vowels 'pillam'). Therefore in many cases, even if the output was marginally correct, the BLEU scores penalized the scores as the BLEU looked for exact word match. This was the critical reason for the very low BLEU score for *Subasa*. It must be noted that we used the currently available web-based user interface of *Subasa* for translation. The problem with fonts could also be related to the user interface, but not to the MT system. However, we do not have information to verify this.

With respect to Si-Ta, the main challenge was data sparseness, which resulted in a high rate of out-of-vocabulary (OOV) words. Both the languages being highly inflectional added more to this issue.

A human evaluation was also conducted on the translation quality of the three systems. 27 human translators participated in this laboratory experiment. They were divided into 9 groups, each group consisting of 3 members. Each group was given 10 source sentences, along with their translation outputs from the three different systems. They were asked to rate the quality of each translation output. The three translation outputs were given without indicating which system generated the output. The rating is based on a 5-point Likert scale (see Fig. 4). Only Sinhala-Tamil translation was experimented with, since the participants were not fluent on the Tamil-Sinhala direction.

In this experiment, out of a maximum score of 5, Si-Ta received 3.2, *Google Translate* received 2.4, and *Subasa* received 1.7. Thus, this result is in-line with the results reported in Table V. Moreover, this result brings out a very important observation of Si-Ta – at its current level, Si-Ta is capable of giving a meaningful translation, meaning that one can use the Si-Ta system to understand the content of a document, although she may not be able to use that translation output as it is. In other words, assume a situation where a letter written in Sinhala was received by someone who only understands Tamil. She can use Si-Ta system and translate the letter into Tamil to understand the letter. Then she herself can write a response letter in Tamil.

Another experiment was carried out to see if there is any performance improvement of the translators when they use the Si-Ta system as opposed to manual translation. 4 translators who are working in a government institution participated in the experiment. They have been using Si-Ta system for about 6 months, and were familiar with the system. Here also only Sinhala-Tamil translation was considered, since some participants were not familiar with Sinhala typing. Each translator was given 4 Sinhala letters of near equal sizes (around 100 words). They were asked to translate two using Si-Ta and the other two manually. Time taken for each translation was recorded. They were given training on how to participate in the experiment. Actual work was done in the computers they were familiar with.

Table VI shows the evaluation results. *S<n>* and *M<n>* refer to translation with Si-Ta, and manual translation, respectively, with *n* being the document number. *Avg* indicates the average time (in minutes) taken for each type of translation. *Avg Diff = average time taken for manual translation – average time taken for translation with Si-Ta*. Different translators have different competency levels, thus they took different times to translate (both manually and using Si-Ta) the same number of words. However, it is evident that they can complete a translation task quicker when using Si-Ta than when they manually translate from scratch. Again, this result is in-line with the result of the

1. Can use the translation without manual alteration
2. Need very little manual alteration, but correct meaning is conveyed
3. Need reasonable amount of manual alteration, but still correct meaning is conveyed
4. Flaws in the meaning. Still can manually alter to get a meaningful output
5. Worse (better translate manually rather than editing)

Fig.4 5-point Likert Scale

human evaluation of the three systems – when compared with manual translation, translators took less but considerable amount of time to edit the Si-Ta output, because although the output conveyed the correct meaning, reasonable amount of editing had to be done on the output.

Conclusion

This paper presented Si-Ta, the first ever machine translation system dedicated for Sinhala and Tamil official documents. According to the evaluation, the system shows very promising results over the other available Sinhala-Tamil translation systems. The output of Si-Ta has to be still checked and edited by a human. However, time taken for this alteration is less than the time taken for complete manual translation. Thus, Si-Ta can be used by human translators to improve their productivity. Moreover, with a translation accuracy of this level, Si-Ta can be used to eliminate the need for manual translators, if the only requirement is to understand the official document received in the source language.

We plan to introduce more features to Si-Ta. These include: integrating spell checkers into Si-Ta editor, integrating an Optical Character Recognition (OCR) tool so that source documents received in hard-copy format can be loaded into the system without having to type in, and post-editing features such as find-and-replacement of manually corrected target words. In addition to these, we keep on increasing the parallel corpus size, and also research on how to improve SMT performance through different data pre-processing techniques.

TABLE VI. HUMAN EVALUATION OF DIFFERENT SYSTEMS (TIME MEASURED IN MINUTES)

Translator	S1	S2	Avg	M1	M2	Avg	Avg Diff
T1	5	11	8	11	8	9.5	1.5
T2	8	9	8.5	12	9	10.5	2
T3	4	3	3.5	8	5	6.5	3
T4	12	14	13	20	18	19	6

References

[1] R. Weerasinghe, "A statistical machine translation approach to sinhala-tamil language translation," in *Proceedings of Towards an ICT enabled Society*, 2003, pp. 136.

[2] S. Sripirakas, A. Weerasinghe and D. L. Herath, "Statistical machine translation of systems for Sinhala-Tamil," in *Proceedings of International Conference on Advances in ICT for Emerging Regions (ICTer)*, 2010, pp. 62-68.

[3] R. Pushpananda, R. Weerasinghe and M. Niranjan, "Sinhala-Tamil Machine Translation: Towards better Translation Quality," in *Proceedings of Australasian Language Technology Association Workshop*, 2014, pp. 123-133.

[4] S. Rajpirathap, S. Sheeyam and K. C. A. Umasuthan, "Real-time direct translation system for Sinhala and Tamil languages," in *Proceedings of Computer Science and Information Systems (FedCSIS)*, 2015, pp. 1437-1443.

[5] R. Pushpananda, R. Weerasinghe and M. Niranjan, "Statistical Machine Translation from and into Morphologically Rich and Low Resourced Languages," in *Proceedings of International Conference on Intelligent Text Processing and Computational Linguistics*, 2015, pp. 545-556.

[6] "Google Translate." Internet: https://translate.google.com/ [May. 12, 2018]

[7] "Subasa: Sinhala – Tamil translator," Internet: http://translate.subasa.lk/si2ta.php [May. 14, 2018]

[8] M. Creutz and K. Lagus, "Unsupervised models for morpheme segmentation and morphology learning," *ACM Transactions on Speech and Language Processing (TSLP)*, vol. 4, no. 1, p. 3, 2007.

[9] R. A. Hameed, N. Pathirennehelage, A. Ihalapathirana, M. Z. Mohamed, S. Ranathunga, S. Jayasena, G. Dias, and S. Fernando, "Automatic Creation of a Sentence Aligned Sinhala- Tamil Parallel Corpus," in *Proceedings of the 6th Workshop on South and Southeast Asian Natural Language Processing (WSSANLP)*, 2016, pp. 124-132.

[10] R. Loganathan, B. Ondrej and Z. Žabokrtský, "Morphological Processing for English-Tamil Statistical Machine Translation," in *Proceedings of the Workshop on Machine Translation and Parsing in Indian Languages,* 2012, pp. 113-122.

[11] S. Thenmalar, J. Balaji and T. V. Geetha, "Semi-supervised Bootstrapping approach for Named Entity Recognition," *arXiv preprint arXiv:1511.06833*, 2015.

[12] F. Farhath, T. Pranavan, S. Ranathunga, S. Jayasena and G. Dias "Improving Domain-specific SMT for Low-resourced Languages using Data from Different Domains," presented at 11th International Conference on Language Resources and Evaluation (LREC), Miyazaki, Japan 2018.

[13] F. Farhath, S. Ranathunga, S. Jayasena and G. Dias "Integration of Bilingual Lists for Domain-Specific Machine Translation for Sinhala-Tamil", in *Proceedings of the 2018 Moratuwa Engineering Research Conference (MERCon)*, 2018 pp. 538-543. IEEE.

[14] P. Tennage, A. Herath, M. Thilakarathne, P. Sandaruwan, and S. Ranathunga, "Transliteration and Byte Pair Encoding to Improve Tamil to Sinhala Neural Machine Translation," in *Proceedings of the 2018 Moratuwa Engineering Research Conference (MERCon)*, 2018 pp. 390-395. IEEE.

[15] P. Tennage, A. Herath, M. Thilakarathne, P. Sandaruwan, S. Ranathunga, S. Jayasena, and G. Dias. " Neural machine translation for Sinhala and Tamil languages." in *Proceedings of the International Conference on Asian Language Processing (IALP)*, 2017, pp. 189-192. IEEE.

[16] D. Bahdanau, K. Cho and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[17] R. Sennrich, B. Haddow and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909,* 2015.

[18] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens and et. al, "Moses: Open source toolkit for statistical machine translation," in *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*, 2007, pp. 177-180.

[19] F. J. Och and H. Ney, "A Systematic Comparison of Various Statistical Alignment Models," *Computational Linguistics*, vol. 29, no. 1, pp. 19-51, 2003.

[20] M. Z. Mohamed, A. Ihalapathirana, R.A. Hameed, N. Pathirennehelage, S. Ranathunga, S. Jayasena, and G. Dias. "Automatic creation of a word aligned Sinhala-Tamil parallel corpus." in *Proceedings of the 2017 Moratuwa Engineering Research Conference (MERCon)*, 2017 pp. 425-430. IEEE.

[21] P. Koehn, Statistical machine translation, Cambridge University Press, 2009.

[22] A. Stolcke, "SRILM - an extensible language modeling toolkit," in *Proceedings of the International Conference on Spoken Language Processing*, 2002, pp. 901-904.

[23] F. J. Och, "Minimum Error Rate Training in Statistical Machine Translation," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, 2003, pp. 160-167