A REVIEW ON KANGLISH TO ENGLISH TEXT-TO-TEXT TRANSLATOR USING NATURAL LANGUAGE PROCESSING

Manikanta[1], Revanth Reddy[2], Shaik Amaan[3], Shamanth[4], Prof. Yogaraja GSR[5].
[1,2,3,4] Final Year Student, [5]Assistant Professor Department of Information Science and Engineering, SJC Institute of Technology, Chikkaballapura, India

Abstract— Natural language processing is that the core of Machine Translation. In history, its development process is nearly the identical as AI, and also the two complement one another. This text compares the linguistic communication processing of statistical corpora with neural computational linguistics and concludes the tongue processing: Neural AI has the advantage of deep learning, which is extremely suitable for handling the high dimension, label-free and massive data of tongue, therefore, its application is more general and reflects the facility of massive data and massive data thinking. The Kanglish to English project worked assuming to make Kannada characters 'less intimidating'. To inform the phonetic sound of a Kannada character, all that a foreigner needed to try to be to seem at English people letter superimposed thereon. Clear, this appears to be a decent idea, but the more we predict about it, the harder. The fundamental premise of this project is that it's come up with a Kanglish font that superimposes a Kannada character, an English character whose pronunciation matches that of the Kannada one closely. But that's also where the issues begin. First, the direction of transliteration seems to be reversed. The entire point of the Kanglish project is to form foreigners read Kannada, whereas the font they need developed maps 26 English characters to their approximate Kannada counterparts. Code-mixing is that the phenomenon of using over one language in an exceeding sentence. In multilingual communities, it's a really frequently observed pattern of communication on social media platforms. Flexibility to use multiple languages in one text message might help to speak efficiently with the audience. But, the noisy user-generated code-mixed text adds to the challenge of processing and understanding linguistic communication to a far larger extent. Artificial intelligence from a monolingual source to the target language could be a well-studied research problem. Here, demonstrate that widely popular and complex translation systems like Google Translate fail now and then to translate code-mixed text effectively. To address this issue, we propose a parallel dataset of 13,738 code-mixed Kannada-English utterances as well as their English human translations. Additionally, we also propose a translation pipeline built on top of Google Translate.

Keywords: Kannada, English, Machine translation, Code-mix, Code-switch, Kanglish

I. INTRODUCTION

India is a large multilingual country, different states in India have different regional languages; hence for proper communication, there is a need for machine translation. But in India, the earliest efforts started in the mid-80s and early 90s. In India, several Institutes work on Machine Translation. The prominent Institutes are as follows:

• The research and development projects at the Indian Institute of Technology (IIT), Kanpur

• National Centre for Software Technology (NCST) Mumbai (Renamed as Centre for Development of Advanced Computing (CDAC), Mumbai

• Computer and Information Sciences Department, University of Hyderabad.

•Centre for Development of Advanced Computing (CDAC), Pune

• Ministry of Communications and Information Technology

Project Institutes co-operates an indispensable role within the field of machine translation the years ago. Most of the machine translation systems have been developed by these Institutes by using numerous domains. There are many domains which have been identified for the development of domain-specific translation systems, parliamentary questions and answers, pharmaceutical information, government documents and notice. Numerous machine translation systems have been developed in India using various systems for language translation from English to Indian languages.

Machine translation systems for translation from English to Indian languages and from regional languages to regional languages have been created in India. These systems are also used to teach students and researchers about machine translation. Most of these systems are in English to Kannada domain with the exceptions of a Kannada to English and English to Kannada machine translation system. English is an SVO language while Indian regional languages are SOV and are relative to free word order. The translation domains are mostly government documents, health, tourism, news reports and stories.

It is revealed that machine translation software is used in field testing or is readily available for Indian languages and languages among Indian languages.

In this work, we focus on building a machine translation (MT) system that converts a mono-lingual sequence of words into a code-mixed sequence. More specifically, we focus on translating from English to Kannada code-mixed with English. In the literature, work has been done on translating from Kanglish into English

English-Kanglish translation can have several practical applications. For example, it can be used to create engaging conversational agents that mimic the code-mixing norm of a human user who uses code- mixing. Another use of resulting kanglish data would be to create training data for some downstream applications such as token-level language identification.

The proposed machine translation system exploits a multilingual text-to-text Transformer model along with synthetically generated code-mixed data. More specifically, the system utilizes the state-of-the-art pre-trained multilingual generative model, mT5 (a multilingual variant of the “Text-to-Text Transfer Transformer" model as a backbone. The mT5 model is pre trained on large amounts of monolingual text from 107 languages, making it a good starting point for multilingual applications such as question answering and MT. This is the question we explore, empirically, in this paper. We also introduce a simple approach for generating code- mixed data and show that by explicitly fine-tuning the model on this code-mixed data we can acquire sizeable improvements. We use a curriculum learning method for this fine-tuning, in which the model is fine-tuned on synthetically created code-mixed data before being fine-tuned on gold code-mixed data. To synthetically generate code-mixed data, we propose a novel lexical substitution method that exploits bilingual word embedding trained on

shuffled context obtained from English-Kannada bi- text. The method works by substituting Kannada equivalents for select n-grams in English sentences obtained from the bilingual word embedding space. For meaningful comparisons, we experiment with five different methods to create code-mixed training data: (i) Romanization of mono- lingual Kannada from English-Kannada parallel data, (ii)paraphrasing of monolingual English from English-Kanglish parallel data,
(iii) back- translation of output from the mT5 model trained on English-Kanglish parallel data, (iv) adapting social media data containing parallel English-Kanglish sentences by removing emoticons, hashtags, mentions, URLs and (v) code-mixed data generated based on equivalence constraint. The impact of different settings (e.g., size of training data, number of paraphrases per input) applicable for most methods on the translation performance are studied. The mT5 model fine-tuned on the code- mixed data generated by our proposed method based on bilingual word embedding's followed by fine-tuning on gold data achieves a BLEU score of 12.67 and places us first in the overall ranking for the shared task are observed. Our key contributions, in general, are as follows:

1. We propose a simple, yet effective and dependency-free, method to generate English- Kanglish parallel data by leveraging bilingual word embedding's trained on shuffled context obtained through English-Kannada bitext.
2. The effect of several data augmentation methods (based on Romanization, paraphrasing, back-translation, etc.) on translation performance are studied.
3. Exploiting code-mixing generation method in the context of curriculum learning, obtains state-of-the- art performance on the English- kanglish shared task data with a BLEU score of 12.67.

II. LITERATURE REVIEW

Title: Index Maintenance for Time-travel Text Search.

Author: Avishek Anand, Srikanta Bedathur, Klaus Berberich, Ralf Schenkel

Abstract: Users of online archives may quickly get document versions that are judged relevant to a given keyword query and existed during a certain time interval using time-travel text search, which extends ordinary text search with temporal predicates. To efficiently handle time-travel text search, various index structures have been proposed. None of them, on the other hand, are easily updated as the Internet evolves and new document versions are added to the web archive. In this paper, we present a unique index structure for time-travel text searches that may be updated gradually when new document versions are added to the online archive. Our solution uses a sharded index organization, bounds the number of spuriously read index entries per shard, and can be maintained using small in-memory buffers and append-only operations. We experimented on two large-scale real-world datasets demonstrating that maintaining our novel index structure is an order of magnitude more efficient than periodically rebuilding one of the existing index structures, while query- processing performance is not adversely affected.

Limitations: It is not suitable for text translation

Title: A Practical Approach to Fully-automatic Indicative English-Hindi Machine Translation.

Author: Anand, Kavitham, Jjhegde, Shekhar, Ritesh, Sawani and Sasi

Abstract: MaTra is a completely automated system for machine translation (MT) of general-purpose texts from English to Hindi. The strengths of the MaTra method are discussed in this work, with a focus on the system's robust parsing mechanism and intuitive intermediate representation. This method enables for easy development of the translation system's linguistic skills while still allowing us to create acceptable translations as the system evolves.

Limitation: It is not suitable for language translation. It supports converting English text to Hindi only**.**

Title: HindEnCorp - Hindi-English and Hindi- only Corpus for Machine Translation.

Author: Ondřej Bojar, Vojtvech Diatka, Pavel Rychlý,Pavel Straňák

Abstract: In version 0.5, we provide HindEnCorp, a parallel corpus of Hindi and HindMonoCorp, a monolingual corpus of Hindi. Both corpora were collected from web sources and pre-processed primarily for the training of statistical machine translation systems. HindEnCorp consists of 274000 parallel sentences (i.e 3.9 million Hindi and 3.8 million English tokens). HindMonoCorp amounts to 787 million tokens in 44 million sentences. Both corpora are publicly accessible for non-commercial study, and many participants in the WMT 2014 shared translation job have utilised their preliminary release.

Limitation: It is not suitable for other languages except Hindi**.**

4. Title: Interlingua-based English–Hindi Machine Translation and Language Divergence.

Author: Shachi Dave, Jignashu Parikh and Pushpak Bhattacharyya

Abstract: Machine translation systems based on Interlingua and transfer have long been used in competing and complementary ways. The former is more cost-effective in scenarios involving multilingual translation and can be utilized as a knowledge representation technique. But given a particular Interlingua, its adoption depends on its ability (a) to capture the knowledge in texts precisely and accurately and

(b) to handle cross-language divergences. The linguistic divergence between English and Hindi is investigated in this research, as well as the implications for machine translation between both languages using the Universal Networking Language (UNL). The United Nations University (UNU), Tokyo, has launched UNL to enable the transfer and exchange of knowledge through the internet. The representation operates at the level of single phrases, establishing a semantic network- like structure with nodes representing word ideas and arcs representing semantic relationships between them. The divergences between the SOV and SVO classes of languages can be represented by the linguistic divergences between Hindi, an Indo-European language, and English. To our knowledge, the approach described here is the only one that uses computational linguistics to describe language divergence phenomena.

Limitation: It supports converting English text to Hindi only.

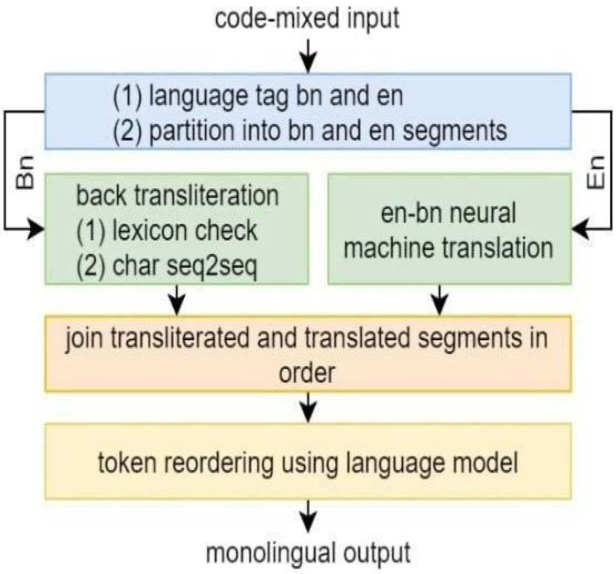
5. Title: The IIT Bombay Hindi-English Translation System

Author: Piyush Dungarwal, Rajen Chatterjee, Abhijit Mishra, Anoop Kunchukuttan, Ritesh Shah, Pushpak Bhattacharyya

Abstract: This article discusses the statistical systems submitted to the WMT14 shared task in English Hindi and Hindi-English. Phrase-based (Hindi-English) and factored (English-Hindi) SMT algorithms are the foundations of our translation systems. It is shown that the use of the number, case and Tree Adjoining Grammar information as factors helps to improve English-Hindi translation, primarily by generating morphological inflexions correctly. We show improvements to the translation systems using pre-processing and post-processing components. Pre- order the source side sentence to comply with the target language word order to overcome the structural difference between English and Hindi. Many words are not translated due to the restricted parallel corpus. The

translate out-of-vocabulary words and transliterate named entities in a post-processing stage. We also investigate the ranking of translations from multiple systems to select the best translation. Limitation: It is not suitable for other languages except Hindi.

II. SYSTEM ARCHITECTURE



The approach to the English-kanglish MT task is simple. We first identify the best text-to-text Transformer model on the validation set and follow a curriculum learning procedure to finetunethe model for the downstream task. The curriculum learning procedure works such that we first finetune the model using synthetic code-mixed data from our generation method, then further finetune on the gold code-mixed data. We now present our proposed method to generate synthetic code-mixed text for a given language pair.

For this method, we assume having access to large amounts of bitext from a given pair of languages (LG1 and LG2) for which we need to generate code-mixed data. Let Bi= {xi ,yi} denote the bitext data, where xi and yi correspond to sentences in LG1 and LG2, respectively. Let n-grams(n, xi , yi
) denote the set of unique n-grams in xi and yi . Let cumulative-n-grams(n, xi , yi) = ∪ j=nngrams(j, xi , yi) denote the cumulative set j=1 of unique ngrams in the set of pairs xi and yi . We shuffle the n-grams in the cumulative set and create a "shuffled" code-mixed sentence by concatenating the shuffled set with n-grams separated by a space. For example, let LG1 denote English and LG2 denote Kannada (assuming Roman script for illustration). We create one shuffled code-mixed sentence per bitext instance, thereby creating a shuffled code-mixed corpus. We train a word2vec model on this shuffled code- mixed corpus to learn embeddings for n-grams in both languages. The resulting word embeddings seem cross-lingually aligned (based on manual inspection), thereby allowing us to do n-gram translation from one language to another language. Once the word embeddings are learned, we can create a code-mixed sentence for the given languages: LG1 and LG2. We first find the ngrams in xi∈ LG1 and then sort all the n-grams by co- sine similarity of the n-gram with its most similar n-gram in LG2. Let num- substitutions denote the number of substitutions performed to convert xi to a code-mixed sentence. We pick one n-gram at a time from the sorted list and replace all occurrences of that n-gram with its top n-gram belonging to language LG2 based on word embeddings. We continue this substitution process until we exhaust the num-substitutions. For thismachine translation task, we assume LG1 and LG2 to be English and Kannada (native) respectively. We feed the OPUS corpus containing 17.2M English-Kannada bitexts (Kannada in the native script) as input to the algorithm that outputs English-Kanglish code-mixed parallel data.

CONCLUSION

The proposed an MT pipeline for translating between English and Kanglish. Testing the utility of existing pre-trained language models on the task and propose a simple, dependency-free, method for generating synthetic code-mixed text from bilingual distributed representations of words and phrases. Comparing the proposed method to five baseline methods, show that our method achieves competitively. The method results in the best translation performance on the shared task blind test data, placing us first in the

official competition. In the future, we plan to (i) scale up the size of code-mixed data, (ii) experiment with different domains of English- Kannada bitexts such as Twitter,
(iii) experiment with recent extensions of mBART, and
(iv) assess the generalizability of our proposed code- mixing method to other NLP tasks such as question answering and dialogue modelling.
In the future, the is to plan to explore other code-mixed languages, especially those that are low-resource and endangered and also plan to extend the corpus for various other code-mixing tasks such as word-embedding, language identification, named-entity recognition, etc. In addition, we can extend the dataset with more annotation using semi-supervised techniques. As the dataset size is significantly small to train a traditional supervised neural machine translation system, we can build the translation systems using few- shots learning techniques.

REFERENCES

1. Muhammad Abdul-Majeed, Chiyu Zhang, Abdel Rahim Elmadany, and Lyle Ungar. 2020. Micro-dialect identification in diagnostic and code-switched environments. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5855–5876
2. Monojit Choudhury, Kalika Bali, Sunayana Sitaram, and Ashutosh Baheti. 2017. Curriculum design for Code-switching: Experiments with language identification and language modelling with deep neural networks. In Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017), pages 65–74, Kolkata, India. NLP Association of India.
3. Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettle- moyer, and VeselinStoyanov. 2019. Unsupervised cross-lingual representation learning at scale. arXiv preprint arXiv:1911.02116.
4. Mathias Creutz. 2018. Open subtitles paraphrase corpus for six languages. arXipreprint arXiv:1809.06142.
5. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186.
6. Mrinal Dhar, Vaibhav Kumar, and Manish Shrivastava. 2018. Enabling code-mixed translation: Parallel corpus creation and MT augmentation approach. In Proceedings of the First Workshop on Linguistic Resources for Natural Language Processing, pages 131–140, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
7. Saurabh Garg, Tanmay Parekh, and Preethi Jyothi. 2018. Code-switched language models using dual RNNs and same-source pretraining. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 3078–3083
8. Brussels, Belgium. Association for Computational Linguistics.
9. Alex Graves, Marc G. Bellemare, Jacob Menick, Remi Munos, and KorayKavukcuoglu. 2017. Automated curriculum learning for neural networks.
10. John J. Gumperz. 1982. Discourse Strategies. Studies in Interactional Sociolinguistics. Cambridge University Press.
11. Deepak Gupta, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A semi-supervised approach to generate the code-mixed text using a pre-trained encoder and transfer learning. In Findings of the Association for Computational Linguistics:
12. EMNLP 2020, pages 2267– 2280, Online.
13. Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The Curious Case of
Neural Text Degeneration. In International Conference onLearning Representations.