Hybrid sub-word segmentation for handling long tail in morphologically rich low resource languages

Sreeja Manghat[1], *Sreeram Manghat*[1], *Tanja Schultz*[2]

[1]IEEE Graduate Member
[2]Cognitive Systems Lab, University Bremen, Germany
sreejamanghat@ieee.org, sreeram9@ieee.org, tanja.schultz@uni-bremen.de

Abstract

Dealing with Out Of Vocabulary (OOV) words or unseen words is one of the main issues of Machine Translation (MT) as well as automatic speech recognition (ASR) systems. For morphologically rich languages having high type token ratio, the OOV percentage is also quite high. Sub-word segmentation has been found to be one of the major approaches in dealing with OOVs. In this paper we present a hybrid sub-word segmentation algorithm to deal with OOVs. A sub-word segmentation evaluation methodology is also presented. We also present results of our segmentation approach in comparison to some of the popular sub-word segmentation algorithms. Malayalam is a morphological rich low resource Indic language with very high type token ratio. All the experiments are done for conversational code-switched Malayalam-English corpus.

Index Terms: OOV, low resource languages, Malayalam, code-switching, language modelling, sub-word segmentation

1. Introduction

OOV words are the words that are not appearing in the training corpus and would appear only in the test corpus. Presence of OOVs degrades the translation quality of MT systems and affects the accuracy of ASR systems. More the number of OOVs, larger is the degradation in the system performance. This problem gets aggravated predominantly in morphologically rich languages but can also pose challenge for any language in low resource setting. One of the predominant methods in handling OOVs is using sub-words. This is under the notion that any unknown word can be represented by its constituent sub-parts whether it is morpheme, syllable or character.

Malayalam is a morphological rich low resource Indic language. Malayalam is having productive morphology with inflection, derivation, boundary mutation and compounding. These mutations occur at morpheme boundaries during concatenation [1]. As seen in Figure 1, the type token growth rate of Malayalam is higher than English and most of the Indian language [2]. The growth of Malayalam seems even higher than morphologically rich European languages like Finnish, Turkish and Estonian [3]. Dialect variation, loan words and slang word presence in the conversational Malayalam speech also add to the increase in number of unique tokens in Malayalam. This is reflected in the OOV rate for Malayalam to be on the higher side. There is difference in the words used in conversational and non-conversational Malayalam .Hence the OOV rate does not subside much with the use of other domain data as well.

Some of the recent approaches in handling OOVs in morphologically rich languages are segmentation algorithms are Byte Pair Encoding (BPE) [4] and Unigram (UNI)[5] . Although these approaches seem to do well with unsupervised segmentation, the OOV modelling in morphologically rich languages is ended up as an array of characters. Unigram model and word piece (WP) perform slightly better over BPE in a few cases [6]. Furthermore, these algorithms fail in many cases while predicting the segmentation boundary for complex morphemes [7]. These are fixed vocabulary algorithms and hence they cannot be directly used with Malayalam having productive morphology with high number of rare words. To the best of our knowledge no sub-word segmentation algorithms have been presented specifically for Malayalam handling all the special cases like compound boundary and affix prediction. This motivates the need of having specific segmentation strategies for Malayalam.

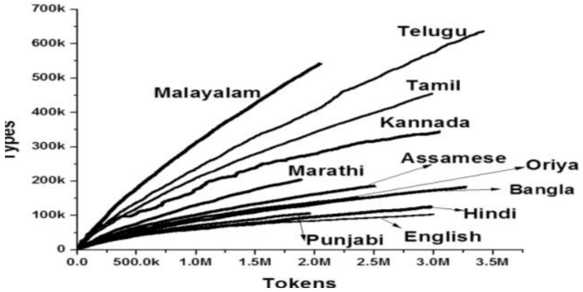

Figure 1. Types vs token ratio [2]

We believe that strategies for one of the most morphologically complex language and low resource language like Malayalam further motivates the adaptation of these strategies to other similar Indic languages. In this paper we propose a hybrid sub-word segmentation algorithm for conversational Malayalam. We also present the evaluation results of the proposed algorithm with other state of the art segmentation algorithms BPE Unigram and WP. We also present the comparison results on intra-word code-switching and sub word ASR. All the evaluations are done on the Malayalam and Malayalam-English intra-word code-switched data of code-switched Malayalam-English speech database [24].

2. Related Work

There have been various generic and language specific approaches on sub-word segmentation to handle OOV problem for machine translation and ASR tasks. Various sub-word units like phoneme, syllable, character, morpheme and combination have been used in different approaches of sub-word modelling. Also, there have been generic and language

978-1-6654-0540-9/22/$31.00 ©2022 IEEE

specific approaches as well. Below enlists some of the major sub-word segmentation approaches. One of the earlier approaches to ASR was Korean syllable-based segmentation [8]. Some of the language specific earlier approaches were in German LVSR [10] and Polish [11]. There was Morpheme based OOV handling approach for Turkish ASR keyword spotting task [9] and multiple languages [12]. Some of the popular recent approaches in unsupervised segmentation are WordPiece [13], BPE and Unigram. Both Byte Pair Encoding and WordPiece algorithms works on merging adjacent characters. In BPE the merge pair is chosen based on frequency whereas in WordPiece, merge is based on maximizing likelihood. Unigram and BPE dropout [14] are some of the sub-word segmentation regularization techniques. Some of libraries implementing segmentation algorithms are sentencepiece [15], bpeNMT [16], morfessor [17] and Morph a gram [16]. There were approaches using mixed word /character model [18] [19]. As far as Malayalam is concerned, there have not been many specific approaches. Few generic approaches [20] [21] used BPE based algorithm from Wikipedia data. There was syllable based segmentation attempted for Malayalam isolated word recognition [22]. It is seen that frequency-based approaches based on BPE will fall back while handling OOVs by splitting the word in wrong boundaries giving either string of characters or longer suffixes with over merging [6] [7]. For low resource languages, disregarding morpheme boundaries during splitting results in ambiguous sub words. This affects the utilization of common sub-word features from closely related languages. Hence it is very important to have specialized strategies for sub-word segmentation in Malayalam.

3. Dataset Analysis

The Malayalam-English code-switched speech corpus used for this study contains 20 hours of speech data from 42 speakers with 22640 utterances. The utterances were Malayalam only, English only or code-switched. There was inter-sentential, intra-sentential and intra-word code-switching. The statistics of the speech corpus is shown in Table 1 [23].

Table 1. Statistics of speech corpus

Total number of utterances	22640
English words	32%
Malayalam words	66%
Intra-word code-switched words	2%

The Malayalam words from Malayalam only utterances and Malayalam words from other utterances are considered for the evaluation. We took 20% Malayalam words as test for calculating OOV percentage. This was repeated on a cross validation way with different sets of test data. The average OOV percentage was found as 16.1.

Malayalam is one of the most morphologically rich languages available which presents inflection, derivation and compounding. This is reflected in the OOVs as well. Table 2 shows the morphological transformation of the OOV and their original constituent words.

Considering the example of derivation – 'Kazhivullayaal. It means 'the one who is able'. If we want to represent the "of the one who is able" the 'Kazhivullayaal' changes to 'kazhhivullayalude'. This shows the productive nature of derivation in Malayalam morphology. Futher 'Kazhivullayaal' is an example of a compound of kazhiv+ulla+aal with a boundary mutations at ulla+aal=>ullayal .

Table 2: Morphological Nature of OOV

Nature of Morpheme transformation	**Constituent words**	**OOV word**
Inflection + Mutation	Aashayam + Kal	Aashaya**n**gal **(ആശയങ്ങൾ)**
Compounding	Vyakthi + Hathya	Vyakthihathya **(വ്യക്തിഹത്യ)**
Derivation	Kazhivu + Ulla + Aal	Kazhivullayaal **(കഴിവുള്ളയാൾ)**

It is seen from figure 2 that the number of words occurring once and twice is very high for conversational Malayalam. This is further more evidence for the higher rate of rare words in Malayalam.

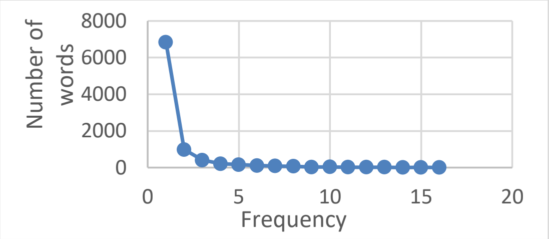

Figure 2. Number of words vs frequency of occurrence

To the best of our knowledge there exist no statistical system which can give the correct segmentation that can be used as reference for conversational Malayalam. The data for our experiments is taken from our annotated Malayalam-English code-switched corpora [23]. Sub-word level boundary marking was done by the transcribers for Malayalam and code-switched Malayalam-English words. The words from this set are taken as the gold standard baseline for our evaluations.

4. Sub-word segmentation algorithm

BPE algorithm operates greedily by combining frequent character pairs in bottom-up manner. Unigram is a regularization top-down approach taking most likely option rather than the best option. Both these algorithms use fixed vocabulary and tend to have a cut-off value which is used to regulate the vocabulary size. This puts the rare sub-words to be ignored by both these models and represented using only characters [6].The frequency-based approach of BPE, will tend to merge with many adjacent units due to their frequency, leading to longer wrong affixes. This was further reconfirmed during our experiments on segmentation as shown in Table 3. The special case like morpheme boundary mutation for Malayalam is not considered by either of these two algorithms. Hence, we have come up with a hybrid approach which will try to get the better morphological segmentation of OOVs. The proposed hybrid algorithm is explained in the below section.

4.1 Hybrid Algorithm

We would like to call hybrid algorithm as 'second chance'. It is inspired by the fact that high type token ratio of Malayalam gives higher rate of rare words. Existing approaches like BPE and Unigram tend to remove a very high percentage of rare Sub-words without even considering them to reach fixed vocabulary size. WP work on likelihood maximization [13], but the pair with the least likelihood is never merged. This results in over segmenting the large number of rare low frequent words in the case of Malayalam. The algorithm tries to address this rare word issue of current approaches hence itself forms as hybrid approach. The algorithm works in bottom-up approach.

Table 3. *Segmentation Comparison*

Words	Reference	BPE	Uni	WP	Hybrid
Maayaajaalam (മായാജാലം)	Maayaa,jaalaam (‘മായാ’,’ജാലം’)	Maa,yaa,ja,a,lam ('മാ', 'യാ', 'ജ', '◌ാ', 'ലം')	Maa,yaa,ja,al,am ('മാ', 'യാ', 'ജ', '◌ാല', '◌ം')	Maa, yaa,ja lam ('മാ', 'യാ', 'ജ', '◌ാല', '◌ം')	Maayaa,jaalaam (‘മായാ’,’ജാലം’)
jalakanyaka (ജലകന്യക)	Jala,kanyaka (‘ജല’,’കന്യക’)	Ja,la,ka,n,ya,ka ('ജ', 'ല', 'ക', 'ന്', 'യ', 'ക')	Ja,la,ka,n,ya,ka ('ജ', 'ല', 'ക', 'ന്', 'യ', 'ക')	Ja,la,ka,n,ya,ka ('ജ', 'ല', 'ക', 'ന്', 'യ', 'ക'	Jala,kanyaka (‘ജല’,’കന്യക’)
jaavithaanubhavangalil (ജീവിതാനുഭവങ്ങളിൽ)	Jaavitha,anubhava,ngal,il ('ജീവിത', '◌ാനുഭവ', ‘ങ്ങള', '◌ിൽ')	Jaavi,thaa,nu,bhava, ngalil ('ജീവി', 'താ', 'നു', 'ഭവ', 'ങ്ങളിൽ')	Jaavitha,anu,bha,va,ngal, il ('ജീവിത', '◌ാനു', 'ഭ', 'വ', 'ങ്ങള', '◌ിൽ')	Jaavitha,anu,bha,va, ngal, il ('ജീവിത', '◌ാനു', 'ഭ', 'വ', 'ങ്ങള', '◌ിൽ')	Jaavitha,anubhava, ngal,il ('ജീവിത', '◌ാനുഭവ', 'ങ്ങള', '◌ിൽ')

Tags <S>, <E> and @ are added to start of word, end of word and sub word segment respectively so that all sub-words and words will have tags on both sides.

The training phase consists of two stages - vocabulary set for segmentation (V1) creation and rare word vocabulary set (V2) creation.

Stage 1, V1 creation:-

1. Start the vocabulary V1 as set of all symbols. These are vowels, consonants, mathras, chillaksharam and other symbols [1]. Chillu or Chillaksharam is a special independent form of certain consonant letters that is not Followed by chandrakkala symbol (◌്). For example, ണ (na) is the Malayalam consonant and ൺ (n) is its chillu.
2. Create the n-gram language model of training data with V1.
3. Find the pair among the higher frequencies which has the largest increase in likelihood (maximum likelihood) once merged. Here in our case the direction is taken from left to right for merge.
4. Add the new merged symbol to V1
5. Repeat steps 2 to 4 until the limit factor is reached. We have taken the sub-word frequency minimum F_{min} as limit factor. We use F_{min} value 2 in our case.

After creating vocabulary there will be words having sub-words which occur only once. This is by re-running the segmentation of training data with V1. Also, the tag symbol is updated at each merge step.

The rare sub-word vocabulary V2 is created in stage 2.

The steps are as below.

6. Calculate the bigram probability of all the sub-words in the vocabulary V1.
7. Segment the words in the training data with the sub-words in the vocabulary V1. Choose the segmentation sequence which gives maximum combined probability.
8. Inspect each segmented word and check the presence of token with length 1 and merge them till the adjacent token length is more than 1. It is very important to note that in our case for Malayalam we are considering the C, CV and CVV as single character. Where C is consonant and V is vowel. The words are checked from left to right.
9. Add the merged token with tags to second set of vocabulary V2 of the rare words after confirming the merged token frequency is 1 with the original vocabulary V1.

The training stage is over after V1 and V2 creation.

Next is the segmentation of the words in the test data by taking one word at a time.

Segmentation process is as mentioned below:

a) Segment each word wx from the test set W by finding the optimal segment sequence using vocabulary V1 as mentioned in step 7.
b) After tokenization let t1,t2,t3..etc be the tokens of wx. wx=[ti,t2,t3…]. Among these tokens, adjacent tokens with character length 1 is taken for merging together into a new token ‘tn’ as mentioned in step 9.
c) Compare the ‘tn’ with the words in V2 for sub-word overlap.
d) If a longest sub-word overlaps between tn and any word in V2 is vn, segment tn at the sub word boundary with reference to vn. This is explained with example in 4.2.

The main features of this algorithm are:

- Maximum generalized approach.
- Rare sub-words occurring once are not discarded.
- Positional probability of each sub-word is considered with the presence of tags.
- Works on rare intra-word code-switches.

Though we have not considered adding the linguistic information in the algorithm at this stage, we believe the algorithm can be improved further by incorporating some of the linguistic characteristics. One of them is considering phonetic sub-words. This will give better information for correlation between mathras and corresponding vowels to predict vowel mutation. Mathras are the symbolic representation of vowels in a C-V combination. കുട (kuda) + ഉടെ (ude) is കുടയുടെ (kudayude) in which there is transformation at the point of addition. യ is the consonant and ഉ is a vowel. In the combination of യ + ഉ = യു , ◌ു is the mathra. Another one is the using phonemes instead of characters. Further the phonetic transformation rules can be leveraged for better boundary mutation prediction.

4.1.1. Analysis

Table 3 shows five particular examples from the OOV list and their segmentation using manual (reference), BPE, WP, Unigram and hybrid segmentation approaches. Segmentation removing tags is presented for better comparison. Consider the example Maayaajaalam (മായാജാലം) having reference tokenization Maayaa, Jaalam. The vocabularies of UNI and BPE does not have any words that are “Maaya” or “Jaalam”. But there is word “Maayaavi” with one occurrence in the corpus which has sub word “Maayaa”. According to our algorithm, “Maayaajaalam” will be initially split as “Maa”, “yaa”, “Jaa”, “l”, “am” by taking V1 for segmentation. Then in the second chance it is merged together as “Maayaajaalam” and checked for sub-word match in V2. Once the longest sub-word “Maayaa” is found in V2 entry “Maayaavi”, the word “MaayaaJaalam” is split as “Maayaa”, “jaalam”. It is to be noted that here the compound “jaalam” was retrieved correctly as a side effect of the merge. Similarly, for the second example jalakanyaka, the subword jala identified by the subword algorithm with V2 dictionary entry word ‘daahajalam’, which was ignored by both BPE and UNI.

4.2 Intra-word segmentation

The Intra-word segmentation occurs as English root and Malayalam suffix (Eg:- Teacher-inte) Malayalam root with English suffix. (Eg:- Kuttikal s). The hybrid algorithm applied for intra-word segmentation showed promising results with respect to reference. Below are the typical results on the English Malayalam segmentation.

5. Experiments

The experiments are conducted using the data from the annotated corpus as mentioned in the section 3. It is to be noted that the vocabulary sizes of BPE and UNI segmentations are set to match with that of hybrid segmentation. We are using Precision, Recall and F1-Score as a measure of comparing our hybrid approach to BPE and Unigram.

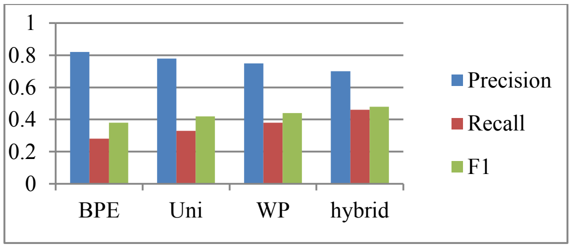

Figure 3.Comparison of algorithms

Reference is the gold standard from our dataset. The Boundary Precision and Recall (BPR) [26] metric calculate similar precision, recall, and F-measure scores. Instead of looking at the morphemes alone, BPR looks at boundary points between these morphemes. BPR helps as a measure in comparing segmentation boundary prediction between the algorithms. 300 OOVs from the test data is used for this test. We run the tests on OOV words for finding Precision, Recall and F1score.The results are shown in figure 3.

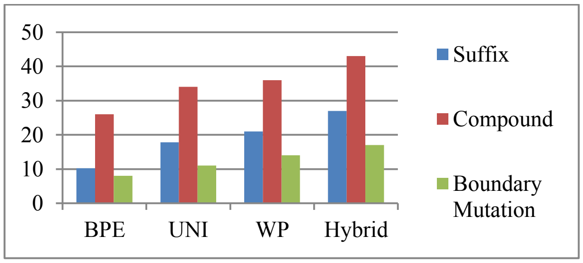

Figure 4. Class accuracy prediction

Another analysis was done to check the class accuracy predictions. The three classes' compounds, suffix and boundary mutation are defined. The manual classification of the 300 OOV words from the test data is done into 3 classes. These words are then segmented using BPE, UNI and Hybrid algorithms in above experiment. The result of class accuracy percentage is shown Figure 4. We did the language modelling test. We created segmentation model using training data (80% of the data). The same training data is segmented using the three segmentation BPE, UNI and hybrid respectively. Test data (20% of data) was also segmented using all 3 models. Different segmentation algorithms produce different number of sub-words after segmentation for the same data. Hence the size of segmented training data and test data size from all the three models has been normalized to have equal size. Language model is created with this normalized segmented training data. Perplexity is calculated with the normalized segmented test data.

Table 4. Perplexity.

Segmentation Type	BPE	UNI	WP	Hybrid
Perplexity	212	208	196	166

Table 4 shows perplexity results. Table 5 shows the intra-word segmentation accuracy where the intra-words were taken from the code-switched data.

Table 5. Intra-word segmentation accuracy

Segmentation Type	BPE	UNI	WP	Hybrid
Accuracy	18	25	29	41

The primary hybrid HMM-DNN approach based ASR results on part of the data for sub word error rate is as below in Table 6.

Table 6. ASR results

Segmentation Type	BPE	UNI	WP	Hybrid
Error rate %	63	52	48	39

After evaluations, we found that, in the precision, recall and F1score tests, relatively lower precision, higher recall and a better F1 score for the hybrid approach. The F1 score is used as a measure for over segmentation. The over segmentation can been seen lower for hybrid approach with overall better F1 score compared to UNI, WP and BPE approaches. The class accuracy percentage of suffix, in compound shows significant improvement of around 10% for hybrid algorithm over WP, BPE and Unigram. Boundary mutation class accuracy though seems to have only 4% improvement with hybrid approach. The language model perplexity test shows lower perplexity for hybrid segmentation compared to UNI, WP and BPE. The initial segmentation results on intra-word also gave promising result with a gain of 12% in suffix identification which directly helps in unsupervised code-switch point detection in intra-word code-switching. Further the experiments done in sub word ASR shows that hybrid segmentation is suitable for handling downstream ASR tasks with long tail with 8% improvement in the next better WP .This gives motivation to utilize hybrid segmentation-based language modelling in future ASR tasks.

6. Conclusion

In this paper we presented a sub-word segmentation strategy for the morphologically rich low resource language Malayalam. We used the conversational Malayalam-English speech data set for our experiments. The F1, Recall, precision test and class accuracy tests are used to compare the hybrid algorithm with BPE, WP and unigram with hybrid showing better results over others. Further the lower perplexity is shown by the hybrid segmentation-based language model. The highest success was seen in the intra-word code-switching accuracy with over 15%. We believe the addition of linguistic rules and phonetic transformation would give further improvement performance in the segmentation. To the best of our knowledge no such implementation specifically dealing with rare words impact in OOVs has been done before. The primary experiments done on sub-word on ASR gave improvement over other available approaches. Hence we believe that this hybrid approach for Malayalam can further motivate the adaptation of these strategies to other low resource Indic languages having similar morphology as Malayalam.

7. References

[1] H. Jiang, "Malayalam: a Grammatical Sketch and a Text", Department of Linguistics, Rice University, 2010

[2] G. Bharadwajakumar, K. Arayanam Urthy, and B. B. Chaudhuri, "Statistical Analyses of Telugu Text Corpora", *Available at: "https://www.academia.edu/14719497/Statistical_Analyses_of_Telugu_Text_Corpora,* [Accessed: March 02, 2021]

[3] M. Creutz et al., "Morph-based speech recognition and modeling of out-of-vocabulary words across languages," ACM Transactions on Speech and Language Processing, vol. 5, no. 1, pp. 1–29, Dec. 2007.

[4] R. Sennrich, B. Haddow, and A. Birch., "Neural machine translation of rare words with sub-word units", *In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Volume 1: Long Papers, pages 1715–1725, Berlin, Germany, 2016.

[5] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates", *in Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics* (Volume 1: Long Papers), pages 66–75, Melbourne, Australia, 2018.

[6] A. Richburg, R. Eskander, S. Muresan, and M. Carpuat, "An Evaluation of Subword Segmentation Strategies for Neural Machine Translation of Morphologically Rich Languages", *in Proceedings of the The Fourth Widening Natural Language Processing Workshop*, Seattle, USA, 2020.

[7] M. Huck, S. Riess, and A. Fraser, "Target-side Word Segmentation Strategies for Neural Machine Translation in Proceedings of the Conference on Machine Translation (WMT), Volume 1: Research Papers, pages 56–67 Copenhagen, Denmark, 2017.

[8] D. Kiecza, T. Schultz and A. Waibel, "Data-Driven Determination of Appropriate Dictionary Units for Korean LVCSR", in proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1999.

[9] Y. He, B. Hutchinson, P. Baumann, M. Ostendorf, E. Fosler-Lussier, and J. Pierrehumbert, "Subword-Based Modeling For Handling OOV Words In Keyword Spotting", in proceedings of the 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Italy, 2014.

[10] A. El-Desoky, M. Mousa, B. Ali, R. Shaik, H. Schlüter, and Ney, "Sub-Lexical Language Models For German LVCSR", *in proceedings of the 2010 IEEE Spoken Language Technology Workshop* (SLT), 2010.

[11] M.A.B. Shaik, A.E.-D. Mousa, R. Schluter, and H. Ney, "Using morpheme and syllable based sub-words for Polish LVCSR", *in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* pp. 4680–4683, 2011.

[12] M. Creutz, T. Hirsimäki, M. Kurimo, A. Puurula, "Morph-based speech recognition and modeling of out-of-vocabulary words across languages" *in ACM Transactions on Speech and Language Processing (TSLP).* 5(1):3, 2007

[13] M. Schuster and K. Nakajima, "Japanese and Korean voice search," *in proceedings of the 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP),* 2012.

[14] I. Provilkov, D. Emelianenko and E. Voita, "BPE-Dropout: Simple and Effective Subword Regularization", in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1882–1892, 2020.*

[15] R. Eskander, F. Callejas, E. Nichols, J. Klavans, and S. Muresan, "MorphAGram: Evaluation and Framework for Unsupervised MorphologicalSegmentation", in Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 7112–7122, 2020.

[16] "Subword-nmt", *Available at: https://github.com/rsennrich/subword-nmt* [Accessed : 10 January, 2021]

[17] "Morfessor", Available at: https://github.com/aalto-speech/morfessor [Accessed : 10 January, 2021].

[18] M. T. Luong and C. Manning, "Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models", *in Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Volume 1: Long Papers), pages 1054–1063, 2016.

[19] Y. Wu et al., "Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation", *Available at: https://arxiv.org/pdf/1609.08144.pdf* [Accessed : 12 February 2021]

[20] A. Kunchukuttan and P. Bhattacharyya, "Learning variable length units for SMT between related languages via Byte Pair Encoding*", In Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 14-24, 2017.

[21] H. Choudhary, S. Rao, and R. Rohilla, "Neural Machine Translation for Low-Resourced Indian Languages", *in Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, pages 3610–3615.

[22] M. Moneykumar, E. Sherly, and W. Varghese, "Isolated Word Recognition System for Malayalam using Machine Learning", *in Proceedings of the 12th International Conference on Natural Language Processing*, pages 158–165, 2015.

[23] S. Manghat, S. Manghat and T. Schultz, "Malayalam-English Code-Switched: Speech Corpus Development and Analysis", *In proceedings of the First Workshop on Speech Technologies for Code-Switching in Multilingual Communities (WSTCSMC 2020),* 2020.

[24] S. Manghat, S. Manghat and T. Schultz, "Malayalam-English Code-Switched: Grapheme to Phoneme System", *in proceedings of Interspeech 2020*, 2020.

[25] S. Virpioja, T. Turunen, S. Spiegler, O. Kohonen, and M. Kurimo, "Empirical Comparison of Evaluation Methods for Unsupervised Learning of Morphology", TAL Traitement Automatique des Langues 52(2):45-90, 2011.

[26] M. Kurimo, S. Virpioja, V. T. Turunen, G. W. Blackwood, and W. Byrne. "Overview and results of morpho challenge 2009", *In proceedings of the workshop of the Cross-Language Evaluation Forum for European Languages*, pages 578–597. 2009