Factored Phrase-Based Statistical Machine Pre-training with Extended Transformers

Vivien L. Beyala[1], Perrin Li Litet[3]
Computer Science
URAIA-University of Dschang
Dschang, Cameroon

Marcellin J. Nkenlifack[2]
Computer Science
URIFIA-University of Dschang
Dschang, Cameroon

***Abstract*—This paper presents the development of a cascaded hybrid multi- lingual automatic translation system, by allowing a tight coupling between the two underlying research approach in machine translation, namely, the neuronal (deterministic approach) and statistical (probabilistic approach), while fully taking advantage of each method in order to improve translation performance. This architecture addresses two major problems frequently occurring when dealing with morphologically richer languages in MT, that is, the significant number unknown tokens generated due to the presence of out of vocabulary (OOV) words, and size of the output vocabulary. Additionally, we incorporated factors (additional word-level linguistic information) in order to alleviate data sparseness problem or potentially reduce language ambiguity, the factors we considered are lemmatization and Part-of-Speech tags (taking into consideration its various compounds). We combined a fully-factored transformer and a factored PB-SMT, where, the training data is pre-translated using the trained fully-factored transformer, and afterwards employed to build an PB-SMT system, parallelly using the pre-translated development set to tune parameters. Finally, in order to produce the desired results, we operated the FPB-SMT system to re-decode the pre-translated test set in a post-processing step. Experiments performed on translations from Japanese to English and English to Japanese reveals that our proposed cascaded hybrid framework outperforms the strong HMT state-of-the-art by over 8.61% *BLEU* and 7.25% *BLEU*, respectively, for validation set, and over 8.70% *BLEU* and 7.70% *BLEU*, respectively, for test set.**

***Keywords*—Machine translation; transformer; statistical machine; morphologically rich; hybrid**

I. Introduction

Machine translation has known an improvement in the state-of-the-art performance by the intervention of Transformers [1] which is a new paradigm in Neural Machine Translation (NMT) [2] [3] powered by frameworks of sequence to sequence learning, thus rivaling since then the factored statistical machine translation paradigm [4] which has achieved the state-of-the-art in SMT frameworks [5] [6]. However, the fundamental design of NMT models which imposes them to make reliable the input representation of a word by observing several instances of that word in multiple examples, and make them to eventually face coverage issues during the computational complexity control by limiting the input and output vocabulary sizes, greatly affects their translation performance when processing rare or OOV (out of vocabulary) words (which are those neither included in the vocabulary nor seen in the training data set, therefore mapped to an UNK token since being considered as unknown words) for languages that are morphologically rich and of low resources (such as Cameroon local languages and some national well known languages namely Arabic, Czech, German, Italian and Turkish). Though having fluent translations in most cases, NMT face challenges in modeling languages syntactic and semantic deeper aspects.

As such, for low-resource (or small corpus) and morphologically rich language conditions, the necessity to incorporate for the surface level words various linguistic annotations was found to resolve semantic ambiguities and data sparseness, thus leading to better translation of rare words or OOVs and greater generalization capacity as illustrated [4] when addressing this issue for the traditional SMT architecture [7] by proposing the factored translation model. This linguistic annotations or factors include features such as lemmas, stems, morphological classes, roots, data-driven clusters, data-driven clusters, part-of-speeches, constituency parsing and compounds. With the vision of alleviating data sparseness and reducing language ambiguity, such extra features may be of enormous benefits when added to both NMT and Phrase-based SMT frameworks.

However, the aim of improving translation performance has inspired much research works through the combination of NMT and SMT paradigms [8] [9] [10] [11] in order to fully take advantage of each system’s strength, and therefore overcoming the deficiencies of meaningless translations (those with meanings totally different compared to source sentences) and limited vocabulary size usually faced by pure NMT models, although its strong language modeling capacities. By contrast, the hard word alignment technic of PBSMT models reflects the source sentences adequacy extremely well, thus helping to some extend to restore the meaning of source sentence whenever wrong translations are produced. The framework proposed by [12] is very close to our work in the global context and overall all architecture but as compared to theirs, ours integrates outperforming paradigms in both the NMT and PBSMT frameworks, that is, Factored Transformers and Factored SMT, respectively. Also, we used linguistic features taking into consideration compounds bot at the NMT (augmenting its embedding layer so as to learn various compositional input representations at different granularity levels) and SMT levels and finally, we proposed a novel UNK replacement algorithm. Our experimental findings reveals that our hybrid model provide consistently and significantly better translation quality for morphologically rich and low resourced languages when coming across rare and unknown words than the state-of-the-art of hybrid translation models.

This paper is organized as follows: A literature review is performed in Section 2. We discuss the factorization process with the integration of compounds in Section 3. In Section 4, we describe the transformer operation with the incorporation of linguistic factors in detail. Section 5 detail our proposed neural hybrid MT framework. In Section 6, the results of two sets of experiments on Japanese to English and English to Japanese tasks are reported measured by their BLEU score. Finally, in Section 7 we summarize our findings and outline future plans.

II. Analogous Researches

By using a combination of different modules, paradigms, resources and approaches, many researchers have explored Hybrid MT systems. In order to produce publishable quality translations, corrections of repetitive errors have to be implemented through the development of various automatic or semi-automatic post-processing techniques, human post-edition usually still have to be operated on the overall resulting MT output [13] [14]. Although human post-editing (PE) is needed over MT outputs, MT output post-edition more often remains cheaper and faster as compared to performing human evaluation from scratch. The authors in [15] [16], and [17] revealed that in some cases productivity can be increased as well as the quality of human translations exceeded by the quality of MT plus PE. More to that, a further optimization of the PE process needs to be done aiming at a time saving and cost-effective use of MT [13].

The authors in [18] and [19] brought out the idea of exploiting machine translation systems combined linearly using different paradigms has been successfully operated over SMT and rule-based MT (RBMT). As such, the systematic errors produced by the RBMT system were corrected by this automatic PE (APE) system based on PB-SMT, hence leading to the reduction of post-editing effort. For translation into a morphologically rich language, a rule (20 hand-written rules)-based approach for English-Czech MT outputs APE at the morphological level was applied by [20] and [21], based on the most frequent errors encountered in translation. Words morphosyntactic categories such as case, number, person, and gender as well as dependency labels are efficiently corrected by this approach. Intuitively, one useful way to improve the APE performance is by source-language information integration in APE. The author in [22] proposed a pipeline in order to overcome data sparsity issues, where through task-specific dense features the best pruned phrase table and language model are selected. More to that, they found that consistent improvements in all language pairs can be obtained by including source language information into statistical APE. The author in [23] considered the potential links of individual alignments occurrences and used an arbitrary number of alignments generated by different models (including both a refine model and minimum Bayes risk based models) by constructing over the 1-best alignments from multiple alignments [24] [25] weighted alignment matrices, rather than performing the combination of exactly two bidirectional alignments as proposed [26] and [27]. The works presented by [28] were motivated based on the fact that word alignment quality is constraint by word alignment-based reordering of source words, with the principal objective of producing monotone source and target chunk alignments through the reordering of source chunks. We argue that the problem of long-range reordering can be reduced to only short-range, intra-chunk reordering by obtaining monotone chunk associations from monotone word alignments while some source language syntax is preserved. The assumption is founded on the reflection that translation is performed by human translators much preferably at chunk level rather than at the word level.

Also, translation outputs produced by an SMT were either re-ranked in a post-processing step using NMT [29] [30] [31] [32] [33], or used to produce an NMT system [10]. Another scenario involves re-ranking the translation outputs produced by an NMT in a post-processing step by using an SMT [12], or guiding translation in NMT by integrating an SMT into an NMT, as they revealed significant translation quality improvement over the Chinese-English translation tasks during experiments [9] [34]. In the works of [34], an NMT architecture is trained in an end-to-end manner where at each NMT decoding step, based on decoding information additional recommendations scored by an auxiliary classifier are offered by the SMT in order to generate words, and the SMT recommendations are combined with NMT generations exploiting a gating function while jointly taking part in the training process.

The several aforementioned attempts to improve MT system’s performance did not still properly handled the issues faced by morphologically rich and low resourced languages, and long-term dependency modelling. We argue that, in order to limit the vocabulary size words could equally be split into sub-word units as proposed [35]. Also, lexical probabilities could be integrated into the NMT as successfully investigated [36]. Another latitude to achieve more monotone translation could be to exploit pre-reordering as experimented [37], and finally but not the least, the NMT translation of rare words could be improved in a post-processing step as suggested [38].

III. Word Compounding and Factorization

In order to reduce the rate Out-Of-Vocabulary (OOV) occurrences and the amount of bilingual data when processing morphologically rich languages, factored models are majorly used. Factorization consists of splitting and retrieving from a given word linguistic information/factors such as dependency information, syntactic information, part-of-speech tags and lemma, using Tree-Tagger [39] and integrating it as a vector into a translation system. Machine translation from one morphological rich language to another has been a tedious task especially when not having enough required morphological information on the source side, since to have an exact target language word-form, word compounding is pronounced useful and highly productive [27] [40] [41] since it leads to sparse data problems and increases the vocabulary size. As such, integrating word compounding in the pre-processing phase has proven to be useful to add extra morphological information to the linguistic/morphological factors of the source and target languages.

Compounding is operated at the level of POS Tag, where minimized part-of-speech tag are produced by refining POS-tags from the Tree-Tagger using a dependency parser to add morphological information including gender, number, case, verbs, person for nouns, definiteness, pronouns, determiners and adjectives, provided that both tools agreed on the POS-tag. And in case of disagreement the Tree-Tagger POS-tags were chosen. Morphologically rich language compound are formed by joining words, inserting filler letters (example: -s, -en, -er, -ien) or from the end of all but the last word remove letters (example: -en, -n) of the compound [42].

A. Compound Splitting

The morphologically rich data language model is POS tagged and employed to compute the adverbs, adjectives, negative particles, verbs and nouns frequencies. Then making use of the adjusted version of the corpus-based method proposed by [27], each adjective and noun splits into known words from the corpus also proper names are not split since it would give rise to errors if translated in parts, while permitting filler additions and truncations. Also, due to the fact that compound parts often contain the base form, lemmas are equally used to calculate word frequencies in addition to surface form. As hint, more splits are gotten when using the arithmetic mean of the frequencies of its parts rather than the geometric mean, where the highest arithmetic value is validated. Each compound parts length was limited to 4 characters and the number of parts for adjectives particularly was restricted to ≤ 2 with minimum words length to be split ≥ 7.

All compound parts but the last were marked with the symbol # so as to be handled as separate words.

Special POS-tag are assigned to split words parts based on the compounds last word's POS, with both the full word and the last part receiving the same POS. Finally, words containing hyphens are split based on this same algorithm, and different POS-tags are assigned to their parts, with hyphens left at the end of all but last part. Factorization with compound splitting is integrated in a pre-processing step for training and translation of both the Transformer framework and the Phrase-based statistical machine framework.

B. Compound Merging

For translation into the morphologically rich language, the split compounds are merged based on POS through a post-processing step at the outputs of both the Transformer framework and the Phrase-based statistical machine framework. As such, if a compound-POS is possessed by a word and a matching POS possessed by the following word, they are merged. Alternatively, a hyphen is added to the word in case the next POS does not match, thus allowing for coordinated compounds.

We used the merging algorithm proposed by [41] based on [40], with this algorithm the advantage is that unseen compounds can be merged and coordinated compounds handled.

IV. INCORPORATING LINGUISTIC FACTORS INTO THE TRANSFORMER

Our principal innovation over the standard encoder decoder based Transformer architecture is that we express the encoder input and decoder output as a combination of features such as [43] [44] [45]. Our generalized model supports an arbitrary number of input features.

It is on a number of well-known linguistic features that we focused in this paper, having as empirical question of knowing to which extend does providing linguistic features to both encoder and decoder improves the translation quality more specially in morphologically richer languages when using the transformer paradigm.

In order to better integrate linguistic factors in our NMT framework, we extended the Transformer architecture propose by [1] which employs multiple stacked layers of an encoder-decoder structure. Two sub-layers constitute the encoder layer, which are a self-attention sub-layer succeeded by a position-wise feed-forward sub-layer. Similarly to the encoder, the decoder has an additional sub-layer which serves at preventing information about future output positions to be incorporated by a given output position during training through masking in its self-attention. For all positions in a sequence, the transformer model computes attention scores using as query each position's input representation. The input representations weighted average are computed then using the previously obtained attention scores. More generally, the attention is identified as query and key/value vector pairs mapping to an output. As such, our work is an extension of [1] by the integration of additional linguistic factors. Considering that we have L layers of annotations for linguistic factors, and N training parallel sentences from the training data $\left\{\left\{x^{(n,l)}, y^{(n,l)}\right\}_{l=0}^{L}\right\}_{n=1}^{N}$ where the n-th sentence pair word sequence is denoted in layer zero as $x^{(n,0)}$ and its length denoted as $|x^{n}|$, the annotations of its L layers are denoted by $\left\{x^{(n,l)}\right\}_{l=1}^{L}$, with the target sentence denoted as $y^{(n,l)}$. In other words, for each feature we look up separate embedding vectors, and concatenate them. The total embedding size is matched by the concatenated vectors length, and the internal structure of the transformer's encoder and decoder is maintained. According to this setting we extended our standard encoder-decoder based Transformer architecture, operating as follows:

Given the input sequence $x = (x_1, \dots, x_n)$ of n elements where $x_i \in \mathbb{R}^{d_x}$ on which each attention head operates, and from which a new representation $z = (z_1, \dots, z_n)$ of same length is computed where $z_i \in \mathbb{R}^{d_z}$. The weighted sum of a linearly transformed input elements will be computed from each output representations as [46]:

$$z_i = \sum_{j=1}^{n} \alpha_{ij} (x_j W^V) \qquad (1)$$

Equally, a softmax function is used to compute each weight coefficient, α_{ij} as:

$$\alpha_{ij} = \frac{exp\ e_{ij}}{\sum_{k=1}^{n} exp\ e_{ik}} \qquad (2)$$

And compatibility function which compares two input elements is used computed from e_{ij}:

$$e_{ij} = \frac{(x_i W^Q)(x_j W^K)^T}{\sqrt{d_z}} \tag{3}$$

To enable efficient computation, a scaled dot product was chosen for compatibility function. Where we have as unique parameter matrices per layer $W^Q, W^K, W^V \in \mathbb{R}^{d_x \times d_z}$.

Input representations in multi-headed self-attention are linearly mapped to lower-dimensional spaces firstly, and one multi-headed self-attention layer's output is formed by the concatenation of several attention mechanisms output vectors (provided that each attention mechanism is identified as a head). Thus in the first self-attention layer the vector for position i for a single attention head $\vec{h}$ is computed as:

$$\vec{h}_i = \text{Attention}\left(QW_i^Q, KW_i^K, VW_i^V\right) \tag{4}$$

And multi-head attention computed as:

$$Multi\vec{h}(Q,K,V) = Concat\left(\vec{h}_1, \ldots, \vec{h}_h\right)W^O \tag{5}$$

given the function computing the resulting vector as:

$$Attention\,(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{6}$$

Adding sinusoids of various wave-lengths enables the self-attention paradigm to encode positional information.[1]

A. Beam Search Integration with Factors

We extended our beam search procedure in order to find the best sequences by dealing with multiple word features (outputs), for simplicity reasons we have one beam responsible for generating lemmas and another beam responsible for generating the concatenation of the different factors. With the help of a toolkit such as MACAON [47] or even the more specialized KyTea [48], we performed the grammatical and morphological analysis. While taking into consideration the context, the lemma and factors for each word is output using the MACAON/KyTea POS-tagger [49]. In the various outputs, the generation of the lemmas and factors are made in a synchronous stream thus leading to sequences with different length sizes, ending each after the generation of the $< eos >$ (end-of-sequence) symbol, and creating by such, multiple representations of the $< eos >$ symbol in an output word. Due to the fact that lemmas carry most of the meaning and are closer to the final objective, we constricted the length size of the factors sequence to be equal to that of the lemma sequence. This implies that when the lemma sequence generation has ended we stop the generation of factors while ignoring their $< eos >$ symbol, therefore avoiding both longer and shorter factors sequences.

In order to generate the next word in the sequence, the feedback (previous word) is employed taking into consideration its various features (outputs), in this case, in order to obtain full benefit of both feedback outputs we performed the tanh (non-linear) transformation of both embedding concatenation, thus having more information and learning better by their combination. Given as:

$$Prev_w(y_{t-1}) = tanh\left([y_{t-1}^L; y_{t-1}^F] \cdot W_{Prev_w}\right) \tag{7}$$

Where, the previous output y_{t-1} feedback is computed by $Prev_w$, W_{Prev_w} are trained weights, with y_{t-1}^L and y_{t-1}^F the embedding of the lemma and factors generated at previous time step, respectively.

Finally, for each partial hypothesis we did the cross product of the output spaces of the best generated lemma and factors hypotheses, thus associating each factor hypothesis to each lemma hypothesis. Also, having k as beam size, the $k-$best combinations was kept for each sample. Equally, in order to get the word candidate when having the lemma and some factors, the MACAON toolkit was used. In situations where name entities are processed therefore having no factors found, the lemma was outputted by the system.

V. Hybrid Machine Translation Succession

Although the translations produced by NMT are more fluent than those of SMT, it still does not fully and explicitly exploit the source information as compared to SMT. Thus, sometimes generating translations that are quite different from the source sentence original meaning [50] and some other times may mistakenly ignore some words during source sentence translations causing other words to be repeatedly translated [51].

If we consider as "intermediate language (another language)" the translation produced by the output of the NMT, to some extend we may amend the duplicated and meaningless translations, by building a translation model and operating a word alignment using an SMT.

Therefore we propose a factored multi-engine hybrid MT system consisting of an NMT and SMT framework, illustrated in Fig. 1.

Firstly, a preprocessing phase in this pipeline is performed by the transformer, which consist of training the transformer system using the initial factored training data, translating the training data, development set and test set into factored pre-translations; secondly, a target-target SMT system is built using the factored pre-translated training data, with parameters tuned using the pre-translated development set; and finally, the desired output is produced by decoding the pre-translated test set using the tuned SMT system.

When using the transformer to perform the pre-translations, if there is an occurrence of OOV in the source sentence, an 'UNK' token is generated by the transformer when translating the training data, development set and test set. We therefore propose a simpler and efficient technique to replace in the translation sentence, the "UNK" token by the corresponding source word. This method is known as the "labeled UNK replacement algorithm", which alleviates the weaknesses faced by the UNK replacement algorithm inspired from [52] proposed by [12]. The technic is presented in Algorithm 1.

[1] We exploited relative positional encoding as emphasized [46] [60] so as to improve performance with respect to machine translation and relation classification, respectively.

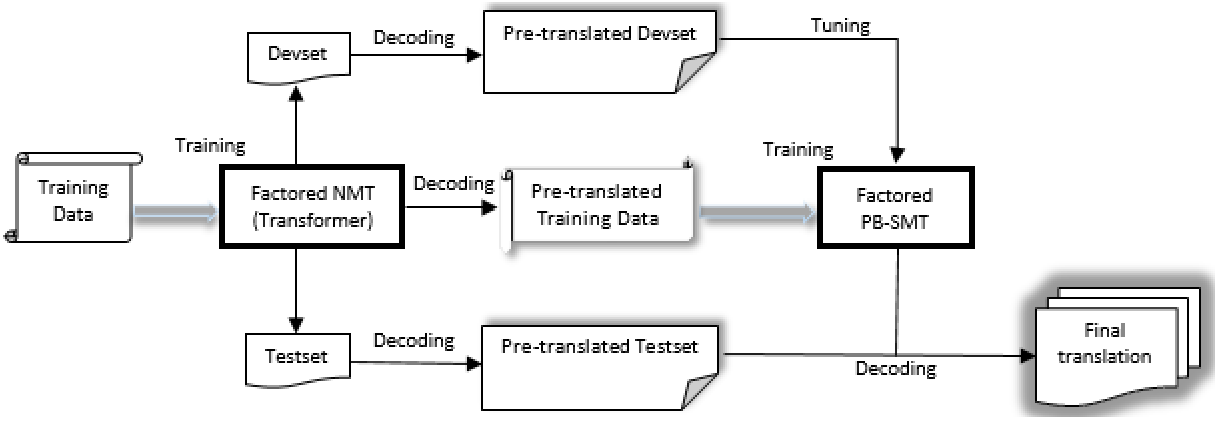

Fig. 1. The Factored Hybrid Transformer-PBSMT Framework.

As such, our algorithm will simply traverse the translation and replace the UNK token they encounter with their corresponding source word (the key at that position), if in the vocabulary there is no existence of the source word. Reference [12] proposed a naïve algorithm to do the UNK replacement, facing the weakness of eventually having between the source sentence and the target sentence different word order, thus creating wrong replacements.

Algorithm 1 Labelled UNK replacement by source words

Require: The translation e_1^m with UNK tokens from the Transformer.

With e as an array of key and value pairs where each key is a source word, and the value the corresponding translation or UNK token.

e.g. for the French sentence: “un chat noir” to translate in English, we may have the corresponding *e* below.

{*un*:*a*}	{*noir*:*black*}	{*chat*:*UNK*}	{*eos*:*eos*}

Considering that we have the following sentence, the replacement will thus be done as:

```
procedure LABELLED UNK REPLACEMENT
    for $i$ = 1 to m do
        if $e_i$.value == UNK then
            $e_i$.value = $e_i$.key
        end if
    end for
end procedure
```

Finally, to post-process these unknown words, instead of using a back-off dictionary [53], we engage by considering more context a factored phrase based SMT system to perform the desired translation. In the factored PB-SMT, for decoding and training we applied the Moses toolkit [7], for sequence models we used SRILM [46] to train a 5-gram language model, and for word alignments creation we employed Giza++ [54], using for feature weights tuning the MERT (Minimum Error Rate Training) [26].

VI. Experiments

In order to verify our proposed framework, we selected translations between Japanese and English languages, noting than Japanese is drastically different in terms of word order and has a far richer grammatical structure as compared to English language.

For fair comparison we re-implemented the hybrid frameworks proposed by [10], training our models using a machine with 8 NVIDIA P100 GPUs.

A. Datasets and Setup

We used as training data Part-1 of the JP–EN Scientific Paper Abstract Corpus (ASPEC-JE) for JP-EN translation task which contains 1M sentence pairs, with the 1,790 sentence pairs contained in the development/validation set, and the 1,812 sentence pairs contained in the test set [55], provided that for the validation and test sets each sentence at the source side has only one reference.

For factorization at both the Transformer and the PBSMT level, we used Lemma and POS tags with compounds (as explained in Section 2 above) as input and output features, which can be produced either by using the MACAON toolkit [47] or the more specialized KyTea [48] especially for the Japanese data.

Due to the fact that unknown words cannot be generated when using Byte-Pair Encoding (BPE) [56] since they are all encoded as BPE units, we thus keep words as translation units. Besides this, incorrect words are sometimes produced by BPE units generation during the final word level processing, thus does not lead to any noticeable improvement in terms of %BLEU [57]. We used the PB-SMT system described in section 4 above. Also, we used as NMT system the transformer [1] default settings with some variants, setting mini batches of size 80, and having as 60 the maximum length of a sentence, with a size of 600 for word embeddings. Parallelly, we have as input and output vocabulary size set to 45K. We reshuffled the training corpus between epochs, and trained the models with the AMSGrad optimizer [58], while at every 5,000 mini batches on the validation set, we validated the model through BLEU (BiLingual Evaluation Understudy) scores, and at every 30,000 performed model safeguard.

We only utilize the baseline transformer system pre-translated training data and devset as input to the SMT engine for its training and tuning. For tuning, the optimized configuration file settings for our translation model is found using Batch MIRA (equally known as k-best MIRA) [59] [60], which is a version of MIRA (a margin-based classification algorithm) working within a batch tuning framework when we have sparse features OR using Minimum error rate training (MERT), but the use of more than about 20-30 features cannot be supported. After which the pre-translated test set is re-decoded utilizing the tuned SMT system.

B. Evaluation and Results

Through bootstrap re-sampling significance test we calculated the statistical significance [61], and also, case-insensitive BLEU scores were used to report all results.

Table I shows the BLEU score based translation results for $JP \leftrightarrow EN$ with non-reordered data, considering as baseline systems a standard PB-SMT [62] for statistical based translations and a NMT proposed by [3] for neuronal based translations. Thus, we observe that:

- The hybrid translation system where the SMT system is used to pre-translate data which serves as input to the NMT, performs significantly gets worse than both the baseline NMT systems and the FNMT system, when operating on $\boldsymbol{JP \rightarrow EN}$ and $\boldsymbol{EN \rightarrow JP}$ languages. The baseline SMT systems has been outperformed in $\%\boldsymbol{BLEU}$ points by all the SMT ⇒ NMT systems on $\boldsymbol{JP \rightarrow EN}$ and $\boldsymbol{EN \rightarrow JP}$, except for the $\boldsymbol{JP \rightarrow EN}$ validation set which reports a decrease in result of $-\,\boldsymbol{0.18}\,BLEU$ points.

- The hybrid NMT⇒SMT model results indicates that the translations produced by the baseline NMT system are re-decoded by the NMT⇒SMT pipeline, leading to a significant improvement of $+\,\boldsymbol{1.25}\,BLEU$ points and $+\,\boldsymbol{1.13}\,BLEU$ points on the $\boldsymbol{JP \rightarrow EN}$ validation and test sets translation performance, respectively, and also, a significant improvement of $+\,\boldsymbol{1.41}\,BLEU$ points and $+\,\boldsymbol{1.96}\,BLEU$ points on the $\boldsymbol{EN \rightarrow JP}$ validation and test sets translation performance, respectively, compared to the baseline NMT system. As compared to the factored NMT system, the hybrid Factored NMT ⇒ SMT model results indicates a slight but noticeable improvement of $+\,\boldsymbol{0.41}\,BLEU$ points and $+\,\boldsymbol{0.42}\,BLEU$ points on the $\boldsymbol{JP \rightarrow EN}$ validation and test sets translation performance, respectively, and also, a significant improvement of $+\,\boldsymbol{1.25}\,BLEU$ points and $+\,\boldsymbol{1.65}\,BLEU$ points on the $\boldsymbol{EN \rightarrow JP}$ validation and test sets translation performance, respectively.

- Finally, we observe that the hybrid model where translations produced by the factored transformer at both its input and output (fully-factored transformer), and which are further re-decoded by the factored SMT, outperforms the translations on the $\boldsymbol{JP \rightarrow EN}$ validation set generated by the fully-factored transformer, and the transformer, by $+\,\boldsymbol{0.86}\,BLEU$ points and $+\,\boldsymbol{2.74}\,BLEU$ points, respectively, and also, translations on the $\boldsymbol{JP \rightarrow EN}$ test set generated by the fully-factored transformer, and the transformer, by $+\,\boldsymbol{1.04}\,BLEU$ points and $+\,\boldsymbol{2.86}\,BLEU$ points, respectively. Similarly, both the translations on the $\boldsymbol{EN \rightarrow JP}$ validation set generated by the fully-factored transformer, and the transformer, by $+\,\boldsymbol{1.06}\,BLEU$ points and $+\,\boldsymbol{2.66}\,BLEU$ points, respectively, and those on the $\boldsymbol{EN \rightarrow JP}$ test set generated by the fully-factored transformer, and the transformer, by $+\,\boldsymbol{1.27}\,BLEU$ points and $+\,\boldsymbol{3.25}\,BLEU$ points, respectively, are as such outperformed by our proposed hybrid system.

C. Discussion

From the above results with reference to the state of the art, we analyze that:

As compared exceptionally to [10] framework consisting of an SMT⇒NMT pipeline which has a higher computational complexity due to the integration of the source information into both the SMT and NMT (concatenating at this level the pre-translated and source sentences as input), and other state of the art hybrid frameworks particularly [12] consisting of an NMT⇒SMT pipeline, our hybrid MT pipeline is more simpler, viable and efficient, by employing source-side information only during the transformer training and exceptionally during OOVs processing, thus favoring its faster computation. Analytical studies for rare/OOV word impact on the translation quality were operated over the Scientific Paper Abstract Corpus (ASPEC-JE) for Japanese-to-English, sorted by the words average inverse frequency and validation sentences were split into groups with comparable numbers of rare words independently evaluated. All target words which occur in the training data for each number of sentence occurrence less than N times were replaced by the UNK token, for all analyzed systems. Given $N \in \{0K, 0.5K, 1K, 1.5K, 2K, 2.5K, 3K\}$. Thus, a higher occurrence of rare words is obtained for large N, hence in the reference only the most frequent words are exploited. Meanwhile a lesser occurrence of rare words is obtained for lower N, using hence more words. We observed that our best performing model (FF-Transformer ⇒ FSMT) considerably outperforms the state of the art both stand-alone and hybrid MT systems on sentences with many OOV words, as a greater occurrence of OOV words implies an increased amount of data size. This boost in performance can be justified by the fact that attention mechanisms which makes up the Transformer operates better on lager data sizes.

We point out that, attention mechanisms are used by neural networks to encode each position while relating two distant words of both the inputs and outputs with respect to itself, by which the training can be accelerated through parallelization. An attention mechanism is a technique created for paying attention to specific words, which have proven to be useful to address the bottleneck issues that arise when handling long sentences with complicated dependencies between words, as it is harder for the context vector to capture all the information contained in the sentence due to the sequential order of word processing. More precisely, the Attention technique focuses on part of a subset of the information it is given, provided that for each input word one hidden state vector is produced. These vectors can then be concatenated, averaged or (even better!) weighted in order to give higher importance to words from the input sentence, most relevant to decode the next word of the output sentence.

TABLE I. RESULTS OF VARIOUS HYBRID (NMT-SMT) MACHINE TRANSLATION EXPERIMENTS PERFORMED ON JP→EN AND EN→JP WHERE, "♠" INDICATES THE BEST TRANSLATION PERFORMANCE

SYSTEM	JP-EN		EN-JP	
	Validation	Test	Validation	Test
SMT [62]	18.46	17.79	27.71	26.54
NMT [3]	24.66♠	24.94♠	35.72♠	35.48♠
FACTORED SMT	18.87	17.91	27.84	26.80
FACTORED NMT	25.70♠	25.83♠	36.57♠	36.31♠
SMT⇒NMT [10]	18.28	17.92	27.82	27.98
T⇒SMT	25.91	26.07	37.13	37.44
FACTORED NMT⇒SMT [12]	26.11♠	26.25♠	37.82♠	37.96♠
TRANSFORMER [1]	31.98	32.09	42.16	42.41
FULLY-FACTORED TRANSFORMER	33.86	33.91	44.01	44.39
FULLY-FACTORED TRANSFORMER ⇒ FSMT	34.72♠	34.95♠	45.07♠	45.66♠

Also, due to the larger vocabulary of the test set by the integration of factors during the PB-SMT post-processing translation, we experienced in our proposed framework a significant decrease in rate of OOVs as compared to the NMT system, of 1.06% and 5.37%, respectively.

We emphasize that, the results on the ASPEC Japanese-to-English corpus should be interpreted with caution. It is the expectation that the attention based HMT when used on longer sentences will show their true potential. In order to investigate on the effect of translating long sentences, sentences of similar lengths having unknown words to the models included were grouped together and the BLEU score was computed per group. The results are delineated in Fig. 2, analyzed over the full validation set.

We observe on Fig. 2 that the buckets of longer sentences are more effectively handled by our Transformer based HMT (purple curve) due to its integrated Attention mechanism at both the encoder and decoder levels as compared to the winning entry recurrent based HMT (green curve) in which the Attention mechanism is integrated only at the level of the decoder, hence as sentences become longer the quality does not degrade. While at shorter sentence lengths, it is observed that our outperforming model performs worse, indicating that although the attention mechanism speeds up training, it is likely not very important and may potentially be redundant. More to that, higher perplexities are produced when operating Attention mechanisms over short sentences, as the model becomes less certain about its predictions than without it.

And we believe that, translations performance will be improved if phrases corrected and reordered are considered. We shall dive deeper by considering this fact in future work.

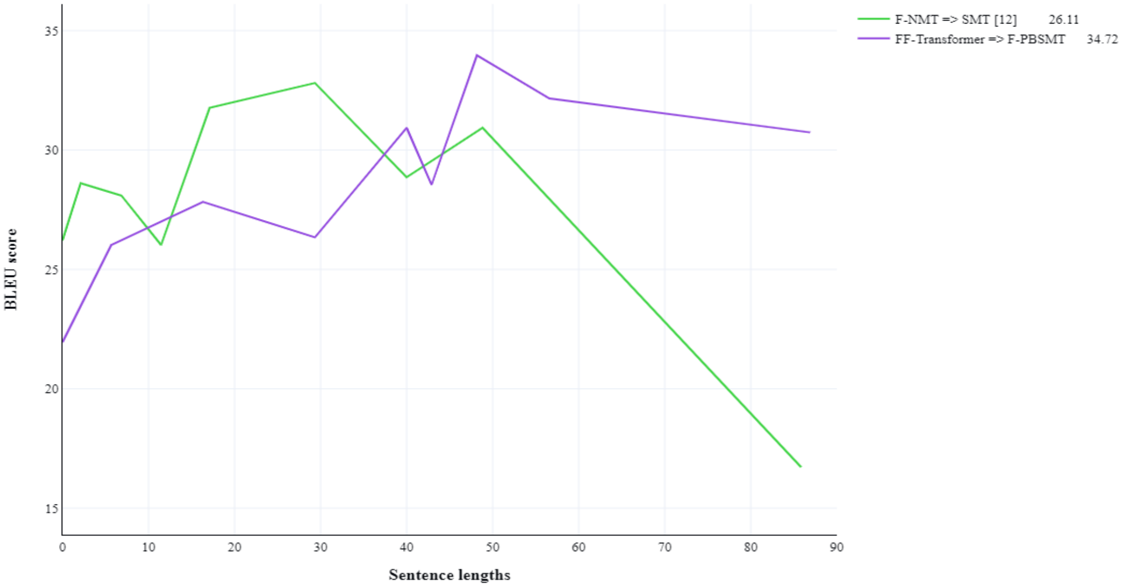

Fig. 2. Length Analysis – Impact of Attention Mechanism on Translation Qualities as Sentences become Longer Performed on ASPEC-JE Data.

VII. CONCLUSION

We have proposed a novel HMT framework cascaded as a Fully-Factored Transformer ⇒ Factored SMT pipeline consisting of integrated linguistic factors at both the source language and target language of the transformer model, and linguistic factors at source language (pre-translated language) of the SMT model. The considered linguistic factors where lemmatization, part-of-speech tagging (taking into consideration its various compounds). Our experimental results on $JP \leftrightarrow EN$ language pairs clearly revealed that our proposed HMT framework with integrated linguistic factors outperforms the state-of-the-art HMT frameworks, in terms of both perplexity and BLEU points. More to that, we observed an OOV rate reduction, due to the generation of new word forms derived from the integrated additional linguistic resources.

As future work, we aim to explore whether the integration of a grammatical error detection and correction (GEC) process [34] will further help in reducing the rate of OOVs. Also, use compositional learned word representations from smaller orthographic symbols inside the words such as character n-grams, which can easily fit in the model vocabulary.

VIII. CONFLICTS OF INTEREST STATEMENT

The authors whose names are listed above certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

REFERENCES

[1] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser Ł, and Polosukhin I. Attention is all you need. In Proceedings of NIPS, 2017.

[2] Devlin J, Zbib R, Huang Z, Lamar T, Schwartz R, Makhoul J. Fast and robust neural network joint models for statistical machine translation. In: Proceedings of the 52st Annual Meeting of the Association for Computational Linguistics (ACL 2014), vol 1, pages 1370–1380, 2014.

[3] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. In Proceedings of the International Conference on Learning Representations (ICLR), 2015.

[4] Koehn Philipp and Hoang Hieu. Factored Translation Models. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL), Association for Computational Linguistics, pages 868–876, 2007.

[5] Kalchbrenner Nal and Blunsom Phil. Recurrent continuous translation models. EMNLP. Pages 1700–1709, 2013.

[6] Stymne S, Holmqvist M, and Ahrenberg L. Effects of morphological analysis in translation between German and English. In: Proceedings of the Third ACL Workshop on Statistical Machine Translation, 2008.

[7] Koehn P, Hoang H, Birch A, Callison-Burch C, Federico M, Bertoldi N, Cowan B, Shen W, Moran C, Zens R, Dyer C, Bojar O, Constantin A, Herbst E. Moses: Open source toolkit for statistical machine translation. In: Proceedings of the 45th Annual Meeting of the ACL, demonstration session, pages 177–180, 2007.

[8] Cho Kyunghyun, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In: Proceedings of the Conference on Empirical Methods on Natural Language Processing, pages 1724–1734, 2014.

[9] He W, He Z, Wu H, Wang H. Improved neural machine translation with SMT features. In: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, pages 151–157, 2016.

[10] Niehues J, Cho E, Ha TL and Waibel A. Pre-translation for neural machine translation. In: Proceedings of the COLING, pages 1828–1836, 2016.

[11] Vaswani Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In: Proceedings of NIPS, 2017.

[12] Du Jinhua and Way Andy. Neural pre-translation for hybrid machine translation. In: Proceedings of MT Summit XVI, MT Summit XVI, vol 1, pages 27–40, 2017.

[13] Report T. Maschine translation post-editing guidelines published (Technical report). TAUS, 2010. Retrieved from http://www.cngl.ie/tauscngl-machine-translation-post-editing-guidelines-published.

[14] Roturier J. Deploying novel MT technology to raise the bar for quality: a review of key advantages and challenges. In Proceedings of the twelfth machine translation summit, 2009.

[15] Fiederer R., and OBrien S. Quality and machine translation: a realistic objective. Journal of Specialised Translation, 11, pages 52–74, 2009.

[16] Koehn P. A process study of computer-aided translation. Machine Translation, 23(4), pages 241–263, 2009.

[17] Palma D. A., and Kelly N. Project management for crowd sourced translation: How user-translated content projects work in real life. In Translation and localization project management: The art of the possible, pages 379–408, 2009.

[18] Dugast L., Senellart J., and Koehn, P. Statistical post-editing on systran's rule-based translation system. In Proceedings of the second workshop on statistical machine translation. Prague, Czech Republic: Association for Computational Linguistics, pages 220–223, 2007.

[19] Simard M., Goutte C., and Isabelle P. Statistical phrase-based post-editing. In Proceedings of naacl-hlt, pages 508–515, 2007.

[20] Rosa R., Mareček D., and Dušek, O. Depfix: A system for automatic correction of czech mt outputs. In Proceedings of the seventh workshop on statistical machine translation, pages 362–368, 2012.

[21] Mareček D., Rosa R., Galuščáková P., and Bojar O. Two-step translation with grammatical post-processing. In Proceedings of the sixth workshop on statistical machine translation, pages 426–432, 2011.

[22] Chatterjee R., Turchi M., and Negri M. The fbk participation in the wmt15 automatic postediting shared task. In Proceedings of the tenth workshop on statistical machine translation, Lisbon, Portugal, pages 210–215, 2015. Retrieved from http://aclweb.org/anthology/W15-3025. (Association for Computational Linguistics. URL).

[23] Tu Z., Liu Y., He Y., Genabith J., Liu Q., and Lin S. Combining multiple alignments to improve machine translation. In The 24th international conference of computational linguistics, pages 1249–1260, 2012.

[24] Liu Y., Xia T., Xiao X., and Liu, Q. Weighted alignment matrices for statistical machine translation. In Proceedings of the 2009 conference on empirical methods in natural language processing, Singapore: Association for Computational Linguistics, pages 1017–1026, 2009.

[25] Tu Z., Liu Y., Liu Q., and Lin S. Extracting hierarchical rules from a weighted alignment matrix. In Proceedings of 5th international joint conference on natural language processing, pages 1294–1303, 2011.

[26] Och FJ. Minimum error rate training in statistical machine translation. In: Proceedings of the 41st Annual Meeting of ACL, pages 160–167, 2003.

[27] Koehn Philipp and Knight K. Empirical methods for compound splitting. In: Proceedings of the tenth conference of EACL, pages 187–193, 2003.

[28] Pal Santanu. A Hybrid Machine Translation Framework for an Improved Translation Workflow. Saarland University, Saarbrücken, Germany, 2018.

[29] Ding S., Duh K., Khayrallah H., Koehn P., and Post M. The jhu machine translation systems for wmt 2016. In Proceedings of the conference on statistical machine translation, Berlin, Germany, 2016.

[30] Farajian M., Chatterjee R., Conforti C., Jalalvand S., Balaraman V., Gangi M., and Federico M. In Fbk's neural machine translation systems for iwslt 2016. In Proceedings of the 13^{th} workshop on spoken language translation, Seattle, USA, pages 8–15, 2016.

[31] Lee H. G., Lee J., Kim, J. S., and Lee, C. K. NAVER machine translation system for wat 2015. In Proceedings of the 2nd Workshop on Asian Translation (WAT2015), Kyoto, Japan, pages 69–73, 2015.

[32] Neubig G., Morishita M., and Nakamura S. Neural re-ranking improves subjective quality of machine translation: NAIST at WAT2015. In Proceedings of the 2nd Workshop on Asian Translation (WAT2015), Kyoto, Japan, pages 35–41, 2015.

[33] Zhao Y., Huang S., Chen H., and Chen J. Investigation on statistical machine translation with neural language models. In Proceedings of Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data, pages 175–186, 2014.

[34] Wang X, Lu Z, Tu Z, Li H, Xiong D, Zhang M. Neural machine translation advised by statistical machine translation. In: Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, California, USA, 2017.

[35] Sennrich Rico, Haddow Barry, and Birch Alexandra. Edinburgh Neural Machine Translation Systems for WMT16. In Proceedings of the 1st conference on machine translation, Berlin, Germany, pages 368–373, 2016a.

[36] Philip Arthur, Graham Neubig, and Satoshi Nakamura. Incorporating discrete translation lexicons into neural machine translation. In Conference on Empirical Methods in Natural Language Processing (EMNLP), Austin, Texas, USA, November 2016.

[37] Kay Rottmann and Stephan Vogel. Word Reordering in Statistical Machine Translation with a POS-Based Distortion Model. In Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2007), Skövde, Sweden, 2007.

[38] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal. Association for Computational Linguistics, pages 1412–1421, 2015.

[39] Schmid H. Probabilistic part-of-speech tagging using decision trees. In: Proceedings of the International Conference on New Methods in Language Processing, pages 44–49, 1994.

[40] Popović M, Stein D and Ney H. Statistical machine translation of German compound words. In: Proceedings of FinTAL - 5th International Conference on Natural Language Processing, pages 616–624, 2006.

[41] Stolcke A. SRILM - an extensible language modeling toolkit. In: Proceedings of the International Conference on Spoken Language Processing (ICSLP), pages 901–904, 2002.

[42] Langer S. Zur Morphologie und Semantik von Nominalkomposita. Tagungsband der 4 Konferenz zur Verarbeitung natürlicher Sprache (KON- VENS) pages 83–97, 1998.

[43] Alexandrescu A, Kirchhoff K. Factored Neural Language Models. In: Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers, Association for Computational Linguistics, pages 1–4, 2006.

[44] Wang Y, Wang L, Wong DF, Chao LS, Zeng X, Lu Y. Factored Statistical Machine Translation for Grammatical Error Correction. In: Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task, pages 83–90, 2014.

[45] Youzheng Wu, Xugang Lu, Hitoshi Yamamoto, Shigeki Matsuda, Chiori Hori, Hideki Kashioka. Factored Language Model based on Recurrent Neural Network. In Proceedings of COLING 2012: Technical Papers, pages 2835–2850, 2012.

[46] Shaw P, Uszkoreit J, Vaswani A. Self-Attention with Relative Position Representations. arXiv preprint, 2018.

[47] Nasr A, Béchet F, Rey JF, Favre B and Roux JL. Macaon, an nlp tool suite for processing word lattices. In Proceedings of the ACL-HLT, pages 86–91, 2011.

[48] Graham Neubig and Shinsuke Mori. Word-based partial annotation for efficient corpus construction. In Proceedings of the 7th International Conference on Language Resources and Evaluation, 2010.

[49] Neubi G, Nakata Y and Mori S. Pointwise prediction for robust, adaptable japanese morphological analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 529–533, 2011.

[50] Sutskever Ilya, Vinyals Oriol and Le Quoc V. Sequence to sequence learning with neural networks. Neural Information Processing Systems (NIPS) Montréal pages 3104–3112, 2014.

[51] Toral A, Sánchez-Cartagena V. M. A multifaceted evaluation of neural versus phrase-based machine translation for 9 language directions. In: Proceedings of the Conference of the European Chapter, Association for Computational Linguistics, 2017.

[52] Jean S, Cho K, Memisevic R, Bengio Y. On using very large target vocabulary for neural machine translation. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, pages 1–10, 2015.

[53] Luong T, Sutskever I, Le Q, Vinyals O, Zaremba W. Addressing the rare word problem in neural machine translation. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, pages 11–19, 2015b.

[54] Och FJ, Ney H. A systematic comparison of various statistical alignment models. Computational Linguistics 29(1): pages 19–51, 2003.

[55] Nakazawa T, Yaguchi M, Uchimoto K, Utiyama M, Sumita E, Kurohashi S and Isahara H. ASPEC: Asian Scientific Paper Excerpt Corpus. In: Proceedings of the Tenth International Conference on Language Resources and Evaluation, Portorož, Slovenia, 2016.

[56] Sennrich R, Haddow B, and Birch A. Neural machine translation of rare words with subword units. 2015.

[57] Papineni K, Roukos S, Ward T and Zhu WJ. BLEU: a method for automatic evaluation of machine translation (PDF). ACL-2002: 40th Annual meeting of the Association for Computational Linguistics 9416:311– 318, CiteSeerX 10.1.1.19, 2002.

[58] Sashank J, Reddi Satyen Kale and Sanjiv Kumar. On the convergence of Adam and beyond. Published as a conference paper at ICLR, 2018.

[59] Cherry Collin and Foster George. Batch Tuning Strategies for Statistical Machine Translation. National Research Council Canada, 2012.

[60] Bilan I, Roth B. Position-aware Self-attention with Relative Positional Encodings for Slot Filling. arXiv preprint, 2018.

[61] Koehn Philipp. Statistical significance tests for machine translation evaluation. In: Proceedings of the Conference on Empirical Methods on Natural Language Processing, pages 388–395, 2004.

[62] Koehn Philipp, Franz Josef Och, and Daniel Marcu. Statistical Phrase-Based Translation. HLT/NAACL, 2003.