Extending Monolingual Semantic Textual Similarity Task to Multiple Cross-lingual Settings

Yoshihiko Hayashi[†] **and Wentao Luo**[‡]
[†]Faculty of Science and Engineering, Waseda University
2-4-12 Ohukubo, Shinjuku, Tokyo, 169-0072 Japan
yshk.hayashi@aoni.waseda.jp
[‡]Graduate School of Language and Culture, Osaka University
1-8 Machikaneyama, Toyonaka, Osaka, 560-0043 Japan
u172761i@ecs.osaka-u.ac.jp

Abstract

This paper describes our independent effort for extending the monolingual semantic textual similarity (STS) task setting to multiple cross-lingual settings involving English, Japanese, and Chinese. So far, we have adopted a "monolingual similarity after translation" strategy to predict the semantic similarity between a pair of sentences in different languages. With this strategy, a monolingual similarity method is applied after having (one of) the target sentences translated into a pivot language. Therefore, this paper specifically details the required and developed resources to implement this framework, while presenting our current results for English-Japanese-Chinese cross-lingual STS tasks that may exemplify the validity of the framework.

Keywords: semantic textual similarity, STS, cross-lingual semantic similarity, machine translation quality

1. Introduction

A Semantic Textual Similarity (STS) measures the degree of semantic equivalence between two target sentences (Agirre et al., 2012; Agirre et al., 2013; Agirre et al., 2014). Given its potential in several types of NLP applications, the series of STS shared tasks[1] accumulates a range of approaches for predicting semantic textual similarity. However the previous STS tasks have been focusing on only monolingual task in English or Spanish: Just only recently, a cross-lingual task (SemEval 2016 Task 1[2]) has been proposed, but the target languages are still limited to English and Spanish.

This paper describes our independent effort for extending the monolingual STS task setting to multiple cross-lingual settings involving English, Japanese, and Chinese. Our current framework (section 2) adopts a "monolingual similarity after translation" strategy: We first convert the given target sentence pair in different languages into a *pivot* language, and then apply a method to predict the monolingual similarity between in the pivot language. Note however that the pivot language can be one of the languages of target sentence pair, or it could be a third language.

Resources (section 3) developed for enabling this framework, as well as the current results (section 4) are shown to exemplify the fundamental validity of our framework. Future directions are discussed while summarizing the current achievements and the issues (section 5).

2. Proposed framework

Figure 1 overviews our current framework for predicting cross-lingual semantic textual similarities. It implements "monolingual similarity after translation" strategy, in which a pair of target sentences (S_1, S_2) in different languages (L_1 and L_2, respectively) are first translated into a *pivot language* (*PL*), and then a method for predicting monolingual textual similarity is applied. Note however that the *PL* can be either of L_1 or L_2; in that case, translation of the sentence in the *PL* is obviously not necessary.

We have evaluated three approaches for predicting monolingual semantic textual similarity, which are referred to as *ML*, *AL*, and *ML+* respectively.

- The Machine Learning-based approach (*ML*) applies a regression method that employs multiple computed similarity features (Bar et al., 2012; Saric et al., 2012).
- The Alignment-based approach (*AL*) takes the alignment score, obtained from a word-level alignment process, originally proposed by (Sultan et al., 2014), as the similarity score.
- Additionally, we assess a combined approach (*ML+*), which incorporates the alignment score as an additional feature in the machine learning process.

As in the conventional STS tasks, the end-to-end prediction performances are evaluated by comparing the Pearson correlation coefficients that computed between the gold-standard scores and the predicted similarities. In addition, we investigate into the impact of machine translation by associating the resulted performances (in correlation coefficients) with assessed quality measures of translation.

3. Resources and Features

In order to implement the proposed framework in a multiple cross-lingual settings, the following resources have to be made available: (1) **data** for training and testing, (2) machine **translation engines** for the target language pairs; in addition, we are in need of a **translation quality measure** to assess the impact of machine translation, (3) linguistic

[1] http://ixa2.si.ehu.es/stswiki/index.php/Main_Page

[2] http://alt.qcri.org/semeval2016/task1/

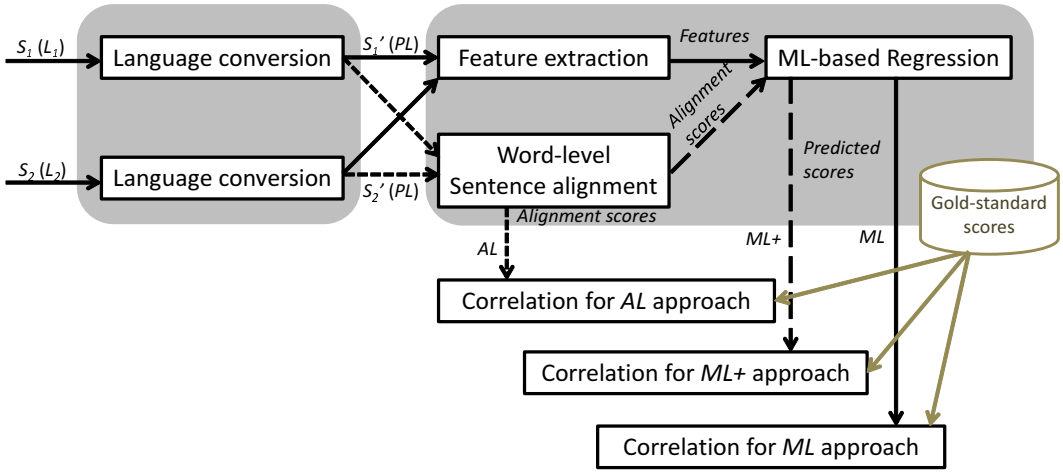

Figure 1: Proposed framework.

Score range	Count in MSvid	Count in MSpar
0	165	0
1	122	45
2	103	95
3	131	328
4	156	212
5	73	70

Table 1: Distributions of similarity scores.

feature extractors for the target languages, and (4) monolingual word-level sentence **aligners** also for the target languages.

3.1. Data

As our goal is to establish a way to extend the monolingual STS tasks in English to cross-lingual settings involving English (`en`) , Japanese (`ja`), and Chinese (`zh`), we employed the English monolingual STS data provided by SemEval 2012 shared task (henceforth STS-12)[3] as a primary resource. Among the STS-12 datasets, two data sets (MSRvid for short sentences and MSRpar for longer sentences), each contains 1,500 sentence pairs[4], were utilized. We extended these datasets for cross-lingual settings by assuming that the gold similarity scores (ranged in [0,5]) are preserved even for the translated sentence pairs. More specifically, we translated one-half of each dataset (750 sentence pairs each) into Japanese and Chinese while keeping the similarity scores. Note that the translation required to create these cross-lingual extension, professional translators as well as foreign students fluent in both languages were employed. Table 1 shows the distribution of similarity scores in the datasets.

Table 2 displays some examples of the prepared sentence pairs. Notice from the entries in this table that we acquired two cross-lingual sentence pairs from a monolingual sentence pair. For example, ("A man with a hard hat is dancing", "一人のヘルメットを被った男がダンスしている") along with ("A man wearing a hard hat is dancing", "一人のヘルメットをした男がダンスしている") are obtained from the following original sentence pair ("A man with a hard hat is dancing", "A man wearing a hard hat is dancing") in English.

Finally, our data sets are populated with: 1,500 sentence pairs for each of the language combinations (`en-ja`, `en-zh`, and `ja-zh`), 1,500 sentence pairs for the English monolingual task, and 750 sentence pairs for the Japanese and Chinese monolingual task.

3.2. Machine translation engines and the translation quality measure

To apply a monolingual similarity computation method in a pivot language in the prediction time, at least one sentence in the target sentence pair has to be translated into the pivot language. To do this, we employed off-the-shelf Web-based machine translation (MT) services[5]. Note that by applying two distinct translation services for each of the language pairs, we acquired 3,000 pivot language sentence pairs for each cross-lingual task.

The qualities of the translated sentences were then measured by applying a metric called RIBES[6] (Isozaki et al., 2010) to investigate into the impact of MT qualities in computing cross-lingual semantic textual similarities. Among other competing metrics for assessing translation quality, we adopted RIBES since it was developed especially for distant language pairs.

More specifically, RIBES relies on rank correlation coefficients to compare the word ranks in the reference with those in the hypothesis, which enables the method to take very care of word order differences that could pose a crucial issue in the comparison of translation between, for example, Japanese and English. The range of a RIBES score is normalized to [0,1].

[3] `https://www.cs.york.ac.uk/semeval-2012/task6/`

[4] We utilized "train" as well as "test-gold" data.

[5] The translation services, including Google Translate, are provided by the Language Grid project. `http://langrid.org/en`

[6] `http://www.kecl.ntt.co.jp/icl/lirg/ribes/index.html`

S_1	S_2	gold similarity
A man with a hard hat is dancing. 一人のヘルメットをした男がダンスしている. 一个頭戴帽子的男人正在跳舞.	A man wearing a hard hat is dancing. 一人のヘルメットを被った男がダンスしている. 一个戴着帽子的男人正在跳舞.	5.00
A woman is playing the guitar. 一人の女がギターを弾いている. 一个頭戴帽子的男人正在跳舞.	A man is playing guitar. 一人の男がギターを弾いている. 一个戴着帽子的男人正在跳舞.	2.40
A woman is slicing big pepper. 一人の女が大きな胡椒を薄切りにしている. 一个女人在切大辣椒.	A dog is moving its mouth. 一匹の大きな犬がその口を動かしている. 一只狗張着它的嘴.	0.00

Table 2: Examples of sentence pairs.

3.3. Monolingual linguistic features

Many of the monolingual STS approaches proposed so far are based on supervised machine learning approaches, where a regression model that employs several semantic features is learned to predict the overall textual similarity between the given pair of sentences. The overall textual similarity could attribute to several aspects, including commonalities in *visible* sequences of a linguistic unit, and latent semantic similarities between the textual components, such as words or phrases.

In the presented work, fifteen features, each dictating a kind of similarity, were combined in the regression process. In the following we explain these features by classifying into four feature groups.

3.3.1. Similarities based on word overlap

This type of overlap is further divided into: word-set overlap and N-gram overlap.

- We compute two types of similarities that measure the degree of word-set overlap: Sim_{f1} that computes Dice coefficient, and Sim_{f2} that binarizes the set inclusion between S_1 and S_2. Neither stop-word deletion nor lemmatization was applied in computing these features.

$$Sim_{f1}(S_1, S_2) = \frac{2 * |S_1 \cap S_2|}{|S_1| + |S_2|} \quad (1)$$

$$Sim_{f2}(S_1, S_2) = \begin{cases} 1 & (S_1 \subseteq S_2) \vee (S_2 \subseteq S_1) \\ 0 & (otherwise) \end{cases} \quad (2)$$

- N-grams could provide more informative evidence of sentence similarity. Thus as in (Saric et al., 2012), we calculate three types of N-gram overlap-based similarities $Sim_{f3\sim f5}$ (for $N = 1, 2, 3$, respectively), which are computed by the following formula. In the feature computation, stop words were deleted, and the remaining words are lemmatized. Notice that even with Uni-grams, the similarity Sim_{f3} does not necessarily yield the same results with Sim_{f1}, due to the difference in the linguistic normalization process.

$$Sim_{f3\sim f5}(S_1, S_2) = 2 * \left(\frac{|S_1|}{|S_1 \cap S_2|} + \frac{|S_2|}{|S_1 \cap S_2|} \right)^{-1} \quad (3)$$

3.3.2. Similarities based on named entity overlap

In addition to the word overlap-based similarities, we further consider the overlap in named entities (NEs) observed in the target sentences in two ways. A binary similarity (Sim_{f6}) only checks whether the same number of NEs are detected, whilst the other binary similarity (Sim_{f7}) rigorously examines that the identical (in word sequences and NE types) NEs are detected. Note that the considered NE types in the presented work are limited to: `People`, `Time`, `Organization`, and `Place`.

3.3.3. Similarities based in word embedding (Word2Vec) vectors

Recently the distributed representation of words has been excessively studied. Among the proposed methods, the method known as Word2Vec (Mikolov et al., 2013) has particularly attracted researchers in related fields, due to its good performances in several types of similarity/analogy tasks, as well as its significant efficiency in training. In the present research, we first assign a semantic vector $v(S)$ to each of the target sentences by simply adding the Word2Vec word vectors[7] $v(w_i)$. The semantic textual similarity (Sim_{f8}) of the pair of target sentences is then computed by the cosine between the semantic sentence vectors as shown in equation 4.

$$Sim_{f8}(S_1, S_2) = cos(\sum\nolimits_{w_{1i} \in S_1} v(w_{1i}), \sum\nolimits_{w_{2j} \in S_2} v(w_{2j})) \quad (4)$$

Besides this conventional method to compose a semantic sentence vector, we employ the following min/max method proposed in (Clarke, 2012) to acquire two alternative types of semantic sentence vectors as shown in equations 5 and 6. Here we denote the m-dimensional word vector $v(w_i)$ as $(\omega_{i1}, ..., \omega_{im})$. The resulting also m-dimensional sentence vectors are then fed into the cosine to compute the minimum vector-based similarity Sim_{f9} and the maximum-based vector similarity Sim_{f10} respectively.

$$v(S) = (\min(\omega_{11}, ..., \omega_{n1}), ..., \min(\omega_{1m}, ..., \omega_{nm})) \quad (5)$$

$$v(S) = (\max(\omega_{11}, ..., \omega_{n1}), ..., \max(\omega_{1m}, ..., \omega_{nm})) \quad (6)$$

[7] For English, we utilized the pre-trained embedding vectors available at `https://code.google.com/p/word2vec/`; For Japanese and Chinese, we have created word embedding vectors by employing Wikipedia dumps in these languages.

The Word2Vec vector-based similarities $S_{f8} \sim S_{f10}$ can be slightly modified by altering the sentence vectors. More precisely, we weight each word in a sentence by introducing *information content* $ic(w)$ as formulated in equation 7 and 8. This gives us the similarities $S_{f11} \sim S_{f13}$ as the weighted version of $S_{f8} \sim S_{f10}$.

$$v(S) = \sum_{w \in S} ic(w)v(w) \tag{7}$$

$$ic(w) = \ln \frac{\sum_{w' \in C} freq(w')}{freq(w)} \tag{8}$$

Furthermore, another semantic textual similarity Sim_{f14} can be formulated by computing a harmonic mean of $wwc(S_1, S_2)$ and $wwc(S_2, S_1)$, which are stated in equation 9. As the formulation shows $wwc(S_1, S_2)$, introduced in (Saric et al., 2012), basically measures an asymmetric coverage of the common words in the target sentences. Here, $ic(w)$ represents the information content of word w computed from a background corpus[8].

$$wwc(S_1, S_2) = \frac{\sum_{w \in S_1 \cap S_2} ic(w)}{\sum_{w' \in S_2} ic(w')} \tag{9}$$

3.3.4. Similarity based on WordNet path length similarity

Saric et al. (2012) proposed the utilization of Princeton WordNet (Miller and Fellbaum, 2007) as a source of semantic textual similarity in English. More specifically, they defined an asymmetric sentence similarity $PWN(S_1, S_2)$ as displayed in equation 10 and 11. Here, $wsim(w, w')$ measures the similarity between word w and w', which converts the length of the shortest path connecting the associated synsets to a similarity score.

$$PWN(S_1, S_2) = \frac{1}{|S_2|} \sum_{w_1 \in S_1} score(w_1, S_2) \tag{10}$$

$$score(w, S) = \begin{cases} 1 & (w \in S) \\ \max_{w' \in S} wsim(w, w') & (otherwise) \end{cases} \tag{11}$$

We utilize the harmonic mean of $PWN(S_1, S_2)$ and $PWN(S_2, S_1)$ as the final symmetric similarity feature Sim_{f15}. For Japanese and Chinese, we utilized Japanese WordNet[9] and Chinese WordNet[10], respectively.

3.4. Word-level sentence alignment

A pair of target sentences in the pivot language can be fed into a word-level sentence alignment process. We have adopted a recently proposed aligner[11] for English sentence pairs (Sultan et al., 2014), which performed well in a monolingual STS task. The excellent performance originates from the algorithm for admitting word-to-word alignments. More specifically, the algorithm links a word w_i in S_1 with the word w_j in S_2, if the similarity given by equation 12 exceeds a pre-defined threshold. If multiple word pairs satisfy this condition, it links the word w_i with the word w_j that gives the maximum similarity. Here, $lsim(w_i, w_j)$ represents a lexical similarity between w_i and w_j; $csim(w_i, w_j)$ dictates a contextual similarity between the context of w_i and that of w_j; α $(0 \leq \alpha \leq 1)$ balances these two factors. In the experiments, we set $\alpha = 0.9$ that means we heavily preferred the lexical similarity.

$$sim(w_i, w_j) = \alpha \times lsim(w_i, w_j) + (1 - \alpha) \times csim(w_i, w_j) \tag{12}$$

For the lexical similarity $lsim(w_i, w_j)$, we utilize cosine of the Word2Vec-based word vectors (Mikolov et al., 2013), while the original Sultan et al.'s aligner instead relies on the Paraphrase Database PPDB (Ganitkevitch et al., 2013). The contextual similarity $csim(w_i, w_j)$, on the other hand, is computed by using the following formula, where $C(w)$ denote a set of *context* words of word w. Note that we normalize the $csim(w_i, w_j)$ to the range [0,1], which eases the thresholding process.

$$csim(w_i, w_j) = \frac{\sum_{w_k \in C(w_i), w_l \in C(w_j)} lsim(w_k, w_l)}{|C(w_i)| \times |C(w_j)|} \tag{13}$$

Sultan et al. (2014) proposed two kinds of sources of context: syntactic dependencies and n-words context windows. The former type of contexts are effectively exploited to admit a word-to-word alignment that is not explicitly encoded in the result of dependency parsing. They proposed a set of over twenty rules for uncovering latent semantic dependency relations that specializes to the Stanford dependency parser[12]. One of the representative relations is the case relation between the verb of an embedded clause and the modified head noun: for example in "He read the book that she wrote", an object case relation is latent between "wrote" (the verb of the embedded clause) and "book" (the modified head noun).

In order to asses the cases in which the adopted pivot language is other than English, we have developed a Japanese and Chinese version of the Sultan et al.'s aligner. To do this, we have created a dedicated set of rules for identifying latent dependency relations in each language. We employed Japanese dependency analyzer KNP[13], and Chinese version of the Stanford parser respectively for this purpose.

Figure 2 illustrates two representative situations in Japanese, where relations with dotted arrows are completed: (a) an embedded noun phrase: the modified noun (hon:*book*) is the direct-object of the verb (kaita:*wrote*) of the embedded clause, and (b) a subordinate clause: the noun (watashi:*I*) serves as the subject of the main verb (yomu:*read*), also serves as the subject of the verb (itte:*go*) of the subordinate clause. Similar situations appear also in Chinese.

[8] Google Books N-grams (http://storage.googleapis.com/books/ngrams/books/datasetsv2.html) for English and Chinese; Mainichi Shinbun corpus (http://www.nichigai.co.jp/sales/mainichi/mainichi-data.html) for Japanese.

[9] http://compling.hss.ntu.edu.sg/wnja/

[10] http://cse.seu.edu.cn/people/zqgao/index.htm

[11] https://github.com/ma-sultan/monolingual-word-aligner

[12] http://nlp.stanford.edu/software/corenlp.shtml

[13] http://nlp.ist.i.kyoto-u.ac.jp/?KNP

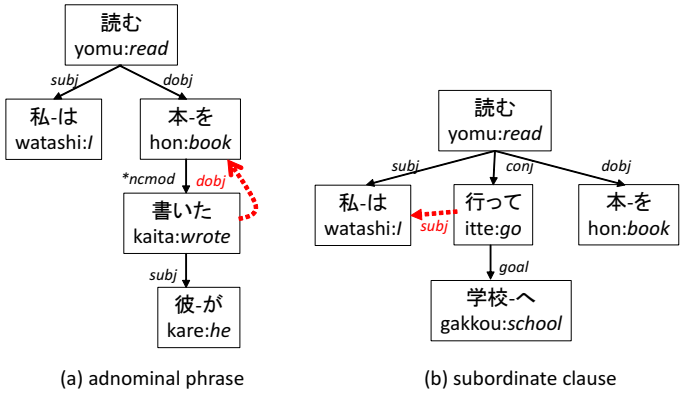

Figure 2: Examples of equivalent dependency relations in Japanese.

	en		ja		zh	
	MSRvid	MSRpar	MSRvid	MSRpar	MSRvid	MSRpar
ML	0.8456	0.6945	0.8078	0.6722	0.8117	0.6802
AL	**0.8725**	0.7239	0.8655	0.7190	0.8269	0.7028
ML+	0.8712	**0.7303**	**0.8672**	**0.7206**	**0.8390**	**0.7244**

Table 3: Results of the monolingual tasks.

4. Experimental results

We adopted the support vector regressor (SVR) as the machine learning algorithm in the experiments. More specifically we employed the module provided by scikit-learn[14], which is a Python-based machine learning tool suit. We performed a grid search for seeking the optimal set of hyper parameters, and applied a standard 5-fold cross validation to obtain the correlation results.

4.1. Monolingual tasks

A monolingual task simulates a situation where the translation into the pivot language is perfectly performed, and hence provides us comparative data to assess the impact of a cross-lingual task setting.

Table 3 organizes the Pearson's correlation coefficients obtained from the monolingual tasks, where each row designates an adopted approach, and each column corresponds to a combination of task type (`en`, `ja`, or `zh`) and dataset (MSRvid or MSRpar). As shown in the table, the *ML+* approach outperformed other approaches in most cases. Also remind that the obtained best results for the task `en` (around 0.87 for MSRvid and 0.73 for MSRpar) are comparative to that of the STS-12 (0.8803 for MSRvid and 0.7343 for MSRpar). This implies that our monolingual similarity computation could almost replicate the best methods in STS-12.

4.2. Cross-lingual tasks

Tables 4 through 6 summarize the results for the cross-lingual tasks. A cross-lingual STS task is represented as, for example, `en/ja*-zh*`, which denotes a task for comparing a Japanese (`ja`) sentence with a Chinese (`zh`) sentence, while employing English (`en`) as the pivot language. `zh`-to-`en` translation processes are necessary. The asterisk mark attached to a language code indicates that the sentence in the language has to be translated into the PL. That is, the task `en/ja*-zh*` requires both of the target sentences (in `ja` and `zh` respectively) to be translated into the PL `en`. Remind that RIBES metrics are shown in the final rows, which signal the qualities of translation in the associated task. Also notice that two RIBES scores are especially shown for the task setting that requires both-side translations: the upper line shows the lower value of the two RIBES scores, while the lower line presents the product of them. The former value is shown in order to provide a rough estimation of the total translation quality, while the latter value is presented to designates a kind of lower bound of the total translation quality.

In the tables:

- It is clearly indicated that employing more similarity features can lead to better results (*ML+* approach). That is, the alignment score, that alone could achieved comparable performances in monolingual tasks, should also be incorporated as one of the similarity features into the machine learning process.

- It is confirmed that the quality of machine translation matters. Thus, machine translation engines with better translation qualities should be utilized; the pivot language should be one of the language of the target sentences (avoid more than one translation processes) where possible.

As a more precise analysis, it can be said that the differences in correlation coefficient are statistically significant (around $p = 0.001$) for MSRpar tasks (with longer sentences), whereas those for MSRvid tasks (with shorter descriptions) are not (around $p > 0.15$). These contrasts cor-

[14] `http://scikit-learn.org/`

	en/en-ja*		ja/en*-ja		zh/en*-ja*	
	MSRvid	MSRpar	MSRvid	MSRpar	MSRvid	MSRpar
ML	0.7956	0.6256	0.8022	0.6618	0.7033	0.5503
AL	0.7725	0.5677	0.7942	0.6557	0.6617	0.5176
ML+	**0.8142**	**0.6363**	**0.8259**	**0.7090**	**0.7108**	**0.5949**
RIBES	0.7762	0.5492	0.7787	0.6971	0.7417	0.6348
					0.5579	0.4600

Table 4: Results of the en-ja task.

	en/en-zh*		ja/en*-zh*		zh/en*-zh	
	MSRvid	MSRpar	MSRvid	MSRpar	MSRvid	MSRpar
ML	0.8114	0.6902	0.6927	0.5424	0.7912	0.7011
AL	0.8042	0.6685	0.6574	0.5089	0.7856	0.6738
ML+	**0.8425**	**0.7087**	**0.7035**	**0.5733**	**0.8295**	**0.7219**
RIBES	0.7845	0.6514	0.7114	0.6438	0.7525	0.7223
			0.5540	0.4488		

Table 5: Results of the en-zh task.

	en/ja*-zh*		ja/ja-zh*		zh/ja*-zh	
	MSRvid	MSRpar	MSRvid	MSRpar	MSRvid	MSRpar
ML	0.7181	0.5853	0.7674	0.6356	0.7808	0.6311
AL	0.6953	0.5370	0.7608	0.6066	0.7769	0.5984
ML+	**0.7295**	**0.6143**	**0.7789**	**0.6537**	**0.7982**	**0.6456**
RIBES	0.7762	0.5492	0.7114	0.6438	0.7414	0.6368
	0.6089	0.3577				

Table 6: Result of the ja-zh task.

roborate the fact that the translation quality for a long sentence is lower, in general, than that of a shorter sentence.

5. Concluding remarks

This paper described a framework for extending the monolingual STS task setting to multiple cross-lingual settings, while detailing the resources developed for enabling cross-lingual STS tasks involving English, Japanese, and Chinese. The comparisons with the individual monolingual tasks confirmed that the "monolingual similarity after translation" approach works reasonably well, in the light of common insights gained in the field of cross-language information retrieval. However, as expected, it is highlighted that the quality of translation would impact the end-to-end performances.

Future work thus can be discussed in terms of the fundamental approach for *crossing* language barriers. If we still pursue the "monolingual similarity after off-the-shelf translation" approach, we obviously need to develop similarity features that are more robust to somewhat collapsed machine-translated sentences. Additionally, we might need to take advantages of the translation redundancy yielded by the use of multiple translation engines.

If we do not directly incorporate any translation process, we need to project each of the target sentence into a common semantic space, and compare them in the space. Two approaches both employing distributional/distributed representation could be possible: (1) first map the words in the target sentences into a common space (Faruqui et al., 2014), and then compare the semantic sentence vectors, which would be compositionally composed (Mitchell and Lapata, 2008); (2) simultaneously map the whole sentences into a common space (Pham et al., 2015), where the semantic vectors can be directly compared.

6. Acknowledgments

This work was supported by JSPS KAKENHI Grant Number #25280117.

7. Bibliographical References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 Task 6: A Pilot on semantic textual similarity. *Proc. of *SEM 2012*, pp.385–393.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *SEM 2013 shared task: Semantic textual similarity. *Proc. of *SEM 2013: The*

First Joint Conference on Lexical and Computational Semantics, pp.32–43.

Agirre, E., Banca,C., Cardie, C., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W., Mihalcea, R., Rigau, G., and Wiebe, J. 2014. SemEval-2014 Task 10: Multilingual semantic textual similarity. *Proc. of the 8th International Workshop on Semantic Evaluation (SemEval'14)*.

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. UKP: Computing semantic textual similarity by combining multiple content similarity measures. *Proc. of *SEM 2012*, pp.435–440.

Clarke, D. 2012. A context-theoretic framework for compositionality in distributional semantics. *Computational Linguistics*, 38(1): 41–71.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations Using multilingual correlation. *Procs. of EACL 2014*, pp.462–471.

Juri Ganitkevitch, BenjaminVan Durme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. *Proc. of NAACL-HLT*, pp.758–764.

Hideki Isozaki, Tsutomu Hirao, Kevin Duh, Katsuhito Sudoh, and Hajime Tsukada. 2010. Automatic evaluation of translation quality for distant language pairs. *Proc. of EMNLP 2010*, pp.944–952.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representation of words and phrases and their compositionality. *Proc. of NIPS 2013*, pp.3111–3119.

George Miller and Christiane Fellbaum. 2007. WordNet then and now. *Language Resources and Evaluation*, Volume 41, Issue 2, pp.209–214.

Jeff Mitchel, and Mirella Lapata. 2008. Vector-based models of semantic composition. *Proc. of ACL 08: HLT*, pp.236–244.

Hieu Pham, Minh-Thang Luong, and Christoper D. Manning. 2015. Learning distributed representations for multilingual text sequences. *Proc. of NAACL-HLT 2015*, pp.88–94.

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. TakeLab: Systems for measuring semantic text similarity. *Proc. of *SEM 2012*, pp.441–448.

Md Arafat Sultan, Steven Bethard, and Tamara Sumner. 2014. Back to basics for monolingual alignment: Exploiting word similarity and contextual evidence. *Trans. of the ACL*, Vol.2, pp.219–230.