# Commonsense Knowledge Augmentation for Low-Resource Languages via Adversarial Learning

**Bosung Kim**[1*], **Juae Kim**[2, 3 *], **Youngjoong Ko**[1], **Jungyun Seo**[2]
[1] Department of Computer Science and Engineering, Sungkyunkwan University
[2] Department of Computer Science and Engineering, Sogang University
[3] AIRS Company, Hyundai Motor Group
{bosung17, yjko}@skku.edu, {juaekim, seojy}@sogang.ac.kr

## Abstract

Commonsense reasoning is one of the ultimate goals of artificial intelligence research because it simulates the human thinking process. However, most commonsense reasoning studies have focused on English because available commonsense knowledge for low-resource languages is scarce due to high construction costs. Translation is one of the typical methods for augmenting data for low-resource languages; however, translation entails ambiguity problems, where one word can be translated into multiple words due to polysemes and homonyms. Previous studies have suggested methods to measure the validity of translated multiple triples by using additional metadata and manually labeled data. However, such handcrafted datasets are not available for many low-resource languages. In this paper, we propose a knowledge augmentation method using adversarial networks that does not require any labeled data. Our adversarial networks can transfer knowledge learned from a resource-rich language to low-resource languages and thus measure the validity score of translated triples even without labeled data. We designed experiments to demonstrate that high-scoring triples obtained by the proposed model can be considered augmented knowledge. The experimental results show that our proposed method for a low-resource language, Korean, achieved 93.7% precision@1 on a manually labeled benchmark. Furthermore, to verify our model for other low-resource languages, we introduced new test sets for knowledge validation in 16 different languages. Our adversarial model obtains strong results for all language test sets. We will release the augmented Korean knowledge and test sets for 16 languages.

## Introduction

Commonsense knowledge consists of facts known by almost every human. People use this knowledge as a basis for their thinking systems to understand, infer and communicate with someone. Therefore, commonsense reasoning is considered the ultimate goal of artificial intelligence because it simulates human thinking. To turn common sense into a form of knowledge graph that can be understood by computers, ConceptNet (Liu and Singh 2004) was launched. ConceptNet involves knowledge as a triple consisting of a head entity, tail entity, and one relation that connects the two entities. The general triple form is <head entity, relation, tail entity>. For example, the commonsense knowledge of "*A bank can keep your money.*" is represented as a triple form of <bank, CapableOf, keep money>.

ConceptNet supports 304 languages, but data size differences between English and low-resource languages are stark. In particular, the English vocabulary consists of 1.8M words, but 217 languages, including Uyghur, Uzbek, and Sanskrit, have fewer than 10K words. In the case of Korean, ConceptNet covers 47K words, but there are only 4,561 triples where both entities are Korean. This is not a sufficient data size for commonsense reasoning. However, it is very hard to construct commonsense knowledge for all low-resource languages as much as the number of English triples due to construction costs.

One of the typical methods for triple expansions is to translate English triples as a source side into low-resource languages as a target side. However, translation suffers from the ambiguity problem; one word can be translated into two or more due to polysemes and homonyms. For example, in Korean, the word "bank" can be translated as "eun-haeng[1]" which indicates a financial institution that accepts deposits and "duk," which is the sloping land along the side of a river. The phrase "keep money" is translated as "don-eul bogwanhada." For a given triple, <bank, CapableOf, keep money>, the translated Korean triple of <eun-haeng, CapableOf, don-eul bogwanhada> is valid, but <duk, CapableOf, don-eul bogwanhada> is invalid. Therefore, in this paper, we define the knowledge augmentation as the task of finding reliable triples with high validity scores among the translated candidates.

Several studies have proposed scoring methods to measure the validity of the translated triples to resolve the ambiguity problem (Feng et al. 2016; Otani et al. 2018; Anwar et al. 2019; Moussallem, Soru, and Ngomo 2019). The most common constraint in those studies is that the proposed methods require manually labeled data or handcrafted templates. For this reason, these studies only validated the effectiveness of their methods for a few languages. These meth-

*Equal Contribution



[1]We denote Korean words as a phonetic expression in English for readability.

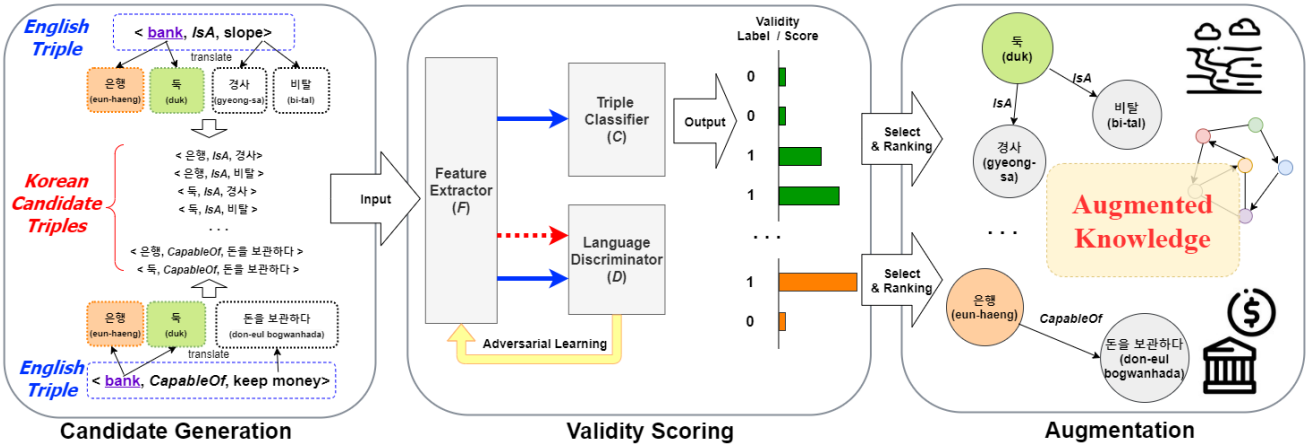


Figure 1: Overview of the proposed knowledge augmentation method.

ods are more limited in the case of low-resource languages because manual labeling is very costly.

Therefore, we propose a knowledge augmentation method using adversarial learning that does not require any labeled data for low-resource languages. Our adversarial model predicts whether the triples are valid or not and learns the independent language features simultaneously. Thereby, the model can transfer information learned from resource-rich languages to low-resource languages and can measure the validity of the candidate triples for any language without labeled data.

In the experiments, we applied our augmentation method to Korean, a low-resource language. We evaluated the performance of the proposed model on a manually labeled Korean benchmark test set and achieved 73.17% at mean average precision (MAP). Furthermore, to demonstrate that our model is language-agnostic, we devised a method to automatically construct test dataset that can verify the model on various languages without manual annotation costs. We experimented on 16 languages, and the results showed that our validity scoring model is useful regardless of the language.

In summary, the contributions of this paper are as follows:

- We propose a knowledge augmentation method that does not require additional labeled data for low-resource languages. To the best of our knowledge, this is the first attempt at applying adversarial learning to data augmentation and validity measurements for commonsense knowledge.

- Our model is language-agnostic, so it can be validated for various languages. To show the effectiveness of the proposed method, we introduced a method to automatically generate a test set for the validation of knowledge.

- The augmented knowledge in Korean, comprising 626,681 triples with a confidence of 93.7%, will be publicly released for further research. In addition, we will open automatically generated test sets in 16 languages.

## Related Works

Several studies to reduce the number of differences in knowledge base between resource-rich languages and low-resource languages have been conducted (Feng et al. 2016; Otani et al. 2018; Anwar et al. 2019; Moussallem, Soru, and Ngomo 2019). To expand upon low-resource knowledge, the authors proposed a machine translation (MT) based approach, a method for finding the triple with the highest validity score among translated candidates from one source triple. Feng et al. proposed a training method to map source triples and target triples into the same semantic vector space. The validity score of the translated triple is calculated using the distance from the source triple in the vector space. This method requires manually aligned data with the same meaning in the source and target triples for training. The works of Otani et al. and Anwar et al. are the most closely related to our study. In the work by Otani et al., they combined the MT score that was converted into string by a handcrafted template with the validity score measured from pre-trained knowledge base completion (KBC) model. They obtained Japanese and Chinese triples from English triples with ConceptNet with high precision. Anwar et al. proposed a back-translation-based approach to resolve the ambiguity problem for Chinese to Uyghur triple translation. Moussallem, Soru, and Ngomo, also suggested MT-based KB expansion, but they focused on the translator unlike other previous methods that used existing translators or translation link information. They trained their own translator with aligned bilingual triples and a parallel corpus. All of the above mentioned studies mentioned above have in common that they use manually labeled parallel corpus or handcrafted templates.

Adversarial networks have shown remarkable success in the natural language processing area, such as domain adaptation (Ganin and Lempitsky 2015; Ganin et al. 2016), and cross-lingual transfer learning (Zhang et al. 2017; Kim et al. 2017; Chen et al. 2018). Ganin et al. proposed adversarial training to overcome the insufficient labeled data problem in various domains for classification tasks. They applied reverse gradients from the domain classifier to learn the domain-independent features for the target task. Chen

et al. introduced a language-adversarial training approach for multi-lingual sentiment classification. They showed outstanding performance in a low-resource language without labeled data since their adversarial approach transfers information learned from resource-rich languages to low-resource languages. Our approach to measure the validity score was inspired by their study.

## Methods

Our goal for knowledge augmentation is to obtain valid low-resource triples from the resource-rich knowledge. Figure 1 presents the process for expanding Korean triples from an English triple. Our data augmentation is composed of three parts: candidate generation, validity scoring, and augmentation. Given an English triple, <bank, IsA, slope> and <bank, CapableOf, keep money>, we first generate the candidate triples in a low-resource language. Then, the candidate triples are then entered into the proposed adversarial model to measure the validity score. In the augmentation stage, the triples that the model predicts as valid are selected at the first stage. Next, we sort candidate triples by their logit scores. Lastly, the top-N triples are obtained as augmented knowledge.

### Candidate Generation

We adopt the MT-based method[2] to obtain the candidates for knowledge augmentation in low-resource language. To generate as many translated candidates as possible, we use the both Bing[3] and Google Translate[4] translator APIs. Google Translate presents the most appropriate translation result by using context. However, what we need to translate in ConceptNet is entities, which are words or phrases, thus the context cannot be reflected in the translation. Meanwhile, Bing can output all possible translation results using a dictionary. Therefore, it is more appropriate to use the multiple translation results provided by Bing to augment the various triples. Thus, we first translate the entities in an English triple with Bing. However, Bing may be unable to provide translation results for some entities that are not registered in the dictionary. In this case, the entities that cannot be translated by Bing are fed into the Google Translate. Therefore most of the English entities can be translated into low-resource languages by using two translators.

In addition, we applied the heuristic rule that if a translated entity has different part-of-speech (POS) tags to an input English entity, we exclude this entity result from the candidate entities to reduce the number of obviously incorrect translations. The POS tag information for translation results is obtained from the Bing translator. This heuristic rule does not apply when Bing cannot provide POS tags, or the translated candidate is obtained using a Google Translate.

After translation, we consider combinations of translated entities as the candidate triples. For example in Figure 1, given the English triple, <bank, IsA, slope>, the head entity "bank" translates to "eun-haeng" and "duk," and then the tail entity "slope" translates to "gyeong-sa" and "bi-tal.", which are different terms with the same meaning in Korean. Therefore, we can obtain $2 * 2 = 4$ candidate triples. For the triple <bank, CapableOf, keep money>, the tail entity "keep money" has one translation result, "don-eul bogwanhada," thus generating two candidates.

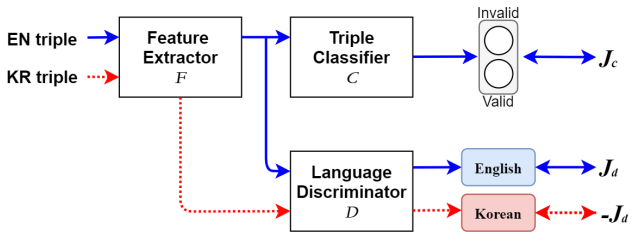


Figure 2: Architecture of adversarial learning.

### Validity Scoring by Adversarial Networks

To decide whether the input triple is valid, we adopt the adversarial learning approach. The proposed method is inspired by the study of Chen et al. because his work achieved good performance with a large amount of data in a resource-rich language on a low-resource language. Our adversarial networks consist of three parts: a feature extractor ($F$), a triple classifier($C$), and a language discriminator ($D$). Training is formalized as a work in which the classifier predicts the correct validity label and the discriminator cannot distinguish the input language. After training, the feature extractor is trained to learn language-agnostic features.

In Figure 2, the flow of data usage is indicated by arrows: the labeled resource-rich language triple (EN) is used to train a triple classifier and language discriminator with training objectives $J_c$ and $J_d$ (blue solid arrows). The unlabeled target language triple (KR) is trained for the only language discriminator with $J_d$ (red dotted arrow) because this triple does not have the validity labels.

**Feature Extractor** ($F$) The feature extractor represents the input triple on a vector space. This space is shared in both resource-rich and low-resource languages. The triples vectorized by the feature extractor are entered in the triple classifier and the language discriminator. The parameters of the feature extractor are adjusted from the losses incurred by these two modules. The purpose of adversarial learning is that the feature extractor learns language-agnostic features. Any neural network model can be used as a feature extractors, and we present the performances of when using various DNN architectures in the experiment section.

**Triple Classifier** ($C$) This is the primary classifier used for predicting the validity of the input triple. The label for

[2]The MT-based method is a typical way to generate candidates; other candidate-generation methods such as automatic entity linking and distant supervision can also be acceptable in our system.

[3]We used the "Dictionary Lookup" provided by translator v3.0 to generate the translated candidates. Please refer to https://docs.microsoft.com/en-us/azure/cognitive-services/translator/

[4]"Cloud Translation - Basic(v2)" was used to generate candidates. More detailed information is included at https://cloud.google.com/translate/docs/

validity is defined as a binary value that indicates whether the input triple is valid or not. For training the triple classifier, we use only English triples as resource-rich language. The valid English triples are simply obtained from ConceptNet. The invalid triples, which refers to negative triples, are automatically generated by replacing the head or tail entity of the valid triple with the random entities. In inference time, the validity score of the input is specified as the logit score which is derived from this classifier. The classifier is developed by a simple feed-forward architecture that consists of two hidden layers and softmax layer for binary classification with ReLu activation function.

**Language Discriminator ($D$)** The purpose of the language discriminator is to distinguish whether the language of the input triples is English or another low-resource language. The output of the language discriminator is the scalar value that indicates the language score. The architecture of the language discriminator is the same with the triple classifier except that it has a linear layer for output instead of softmax layer.

**Training** If the language discriminator achieves good performance in differentiating the input languages, the feature extractor learns the distinctive features of each language. That is, the language discriminator is adversarial to the feature extractor whose purpose is to learn language-independent features. Therefore, to allow the feature extractor to learn language-agnostic features, the discriminator should be trained in a way that cannot distinguish between input languages. The objective function of the discriminator is shown in Equation 1. The feature extractor, triple classifier and language discriminator are denoted as $F$, $C$, and $D$ respectively. The discriminator $D$ is trained with the resource-rich language triple $x$ and the low-resource triple $x'$. $F(x)$ indicates the vector which is represented by the feature extractor for input $x$.

$$J_d(\theta_f) \equiv \max_{\theta_d} \mathop{\mathbb{E}}_{F(x)}[D(F(x))] - \mathop{\mathbb{E}}_{F(x')}[D(F(x'))] \quad (1)$$

where $F(x)$ and $F(x')$ follow the distribution of resource-rich languages and low-resource languages, respectively. To coordinate the training of the triple classifier and language discriminator, we first train the discriminator with $k$ iterations which is a hyper-parameter. Afterward, the parameters of triple classifier are updated by the loss denoted as Equations (2). In Equation, $L(\hat{y}, y)$ indicates the cross-entropy loss where $y$ is the answer validity label and $\hat{y}$ is the output of the triple classifier.

$$J_c(\theta_f) \equiv \min_{\theta_f, \theta_c} \mathop{\mathbb{E}}_{(x,y)}[L(C(F(x)), y)] \quad (2)$$

Then, the feature extractor is trained to accurately predict the validity label and at the same time learn the language-agnostic features by minimizing the loss denoted as follow equation:

$$J_f \equiv J_c + \lambda J_d \quad (3)$$

where $\lambda$ is a hyper-parameter that controls the influence of the triple classifier and language discriminator for updating the feature extractor.

# Experiments

In our experiments, we first show the results of the knowledge augmentation on one of the low-resource languages, Korean. Next, we introduce automatically generated test sets on 16 languages. Lastly, we demonstrate that our method can be applied to different languages.

## Low-resource Knowledge Augmentation on Korean

While ConceptNet has 34M triples, there are only 4,561 Korean triples. In this section, we first demonstrate that our adversarial learning approach can validate the Korean triples with high reliability when the labeled Korean data does not exist. Next, we show how many Korean triples can be obtained via the proposed knowledge augmentation method.

**(1) Datasets** The datasets needed to train our adversarial model are three-fold: labeled resource-rich language (English) data, unlabeled target language (Korean) data, and labeled target language data for evaluation.

- **Labeled English Data**: There are over 3.4M English triples in ConceptNet. We extracted 18,690 highly weighted triples, and divided them into 17,690, 500, and 500 for the training, development, and test sets, respectively. In order to train the triple classifier, which predicts whether the triples are valid or not, negative triples are needed. We generated negative triples by replacing the head or tail entities with a random entity while preventing duplication. As a result, 106,140, 3,000, and 3,000 English triples were used for the training, development, and test sets, respectively.
- **Unlabeled Korean Data**: Since there are only 4,561 valid Korean triples available, we used a translator to build unlabeled Korean triples. We translated 17,690 English triples to 217,411 Korean triples using Bing and Google Translate.
- **Labeled Korean Data**: To evaluate the proposed method's performance on Korean, we manually constructed development and test sets. First, we translated each 500 labeled English development and test triples to 5,055 and 4,509 Korean candidate triples, respectively. Then, three native Koreans annotated each triple as valid or invalid.

**(2) Baseline Models** To demonstrate our method's effectiveness, we conducted experiments on three different architectures: LSTM, CNN, and Multilingual BERT (Devlin et al. 2019) as a feature extractor. These baselines are trained to minimize the cross-entropy loss in Equation 2 without adversarial learning. In addition, we compared our method with the KBC methods, such as TransE (Bordes et al. 2013), which can be applied to measure the validity of triples in the knowledge graph.

- **TransE**: Using the KBC method is a common approach to validate a triple's reliability. The score function suggested by Bordes et al. provides a high score when the triple is valid. We used the TransE method to score the validity of candidate triples.

- **MT+KBC**: This model follows the implementation described in Otani et al.; they suggested a knowledge augmentation method that combines the MT score and KBC score. To obtain the MT score, we annotated templates to transform the Korean triple to the sentence. The converted sentence was entered into the MT model trained with two English-Korean parallel corpora, the 2018 version of OpenSubtitles (Lison, Tiedemann, and Kouylekov 2018) and Ted talks released in IWSLT2017 (Cettolo et al. 2017), all of which total 1.6M sentences. We used TransE as a KBC model. Translated Korean triples are fed into the KBC model to compute the KBC score. Finally, we measured the validity score by combining the MT score and KBC score with a multi-layer perceptron.
- **LSTM**: We performed comparisons with the proposed adversarial method and several feature extractor models without adversarial learning. We used Bi-LSTM with a dot attention method (Luong, Pham, and Manning 2015) as a feature extractor, and bilingual word embeddings (BWEs) for English and the target language are used as initial inputs.
- **CNN**: A feature extractor with a CNN (Kim 2014) model was used. The input and output architectures are the same as those of LSTM baseline.
- **Multilingual BERT (M-BERT)**: M-BERT is a multilingual version of BERT (Devlin et al. 2019). M-BERT is trained on multilingual corpora in 104 languages and has shown good performance at various cross-lingual tasks. We used M-BERT as a feature extractor, and we added single feed-forward layers for the triple classifier and language discriminator, respectively.

**(3) Experimental Settings** When we trained the triple classifier ($C$), the negative triples are generated by replacing the true triple's head or tail entity with other random entities. Experimental results show that models perform best when the number of negative samples is five times the number of positive samples. Thus, we made five invalid triples according to each positive triple.

In the experiments for the LSTM and CNN feature extractor models, we used the AdamW optimizer (Loshchilov and Hutter 2019) and set the batch size to 128, learning rate to 1e-4, and clip to (-0.01, 0.01) for language discriminator $D$. The size of the hidden layers of Bi-LSTM is 900, and the kernel sizes of CNN are 3, 4, and 5 with 400 feature maps.

In training M-BERT, we set the batch size to 64, learning rate to 2e-5, and clipped gradient norm as 1.0 to avoid exploding gradients. $\lambda$ in Equation 3 was chosen to be between $\{0.01, 0.1, 0.2, 0.4\}$, and $k$ was chosen to be between $\{0, 3, 5, 10\}$. We adapted KG-BERT's (Yao, Mao, and Luo 2019) training method and the input of M-BERT is a text sequence of each triple. All hyper-parameters were tuned based on the development.

When we compared the baseline models, which are trained without adversarial learning, where labeled Korean triples are needed. Thus, we composed data for supervised learning using the same method for generating labeled English data. For English-Korean BWE, we used the published aligned word vectors made by Joulin et al..

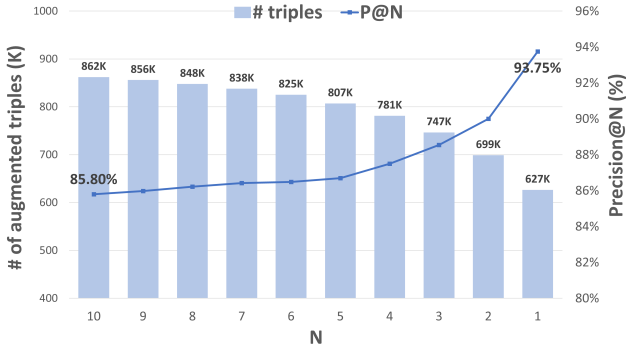


Figure 3: Precision@N on the Korean test set, and the number of augmented triples according to the result of top-N augmentation.

**(4) Evaluation of Triple Classifier on Korean** We first show the results of the triple classifier on Korean. As shown in Figure 1, each English triple is translated to several candidates due to the translation ambiguity problem. Moreover, multiple candidates can be true because words can be translated with synonyms. Therefore, MAP and accuracy are used for evaluating ranking and triple classification performance, respectively.

Table 1 shows the performance of the triple classification on the Korean test sets. Firstly, the results of TransE method did not meet expectations, despite having shown good performance in many datasets (Sun et al. 2019). We observed that if labeled data are not sufficient, such as low-resource language datasets, models cannot learn the relational properties in the knowledge graph.

We also performed comparisons with three baseline models. While the LSTM and CNN baselines showed low performance, our adversarial method successfully discerned the Korean triples without any labeled Korean data, improving both accuracy and MAP. Notably, our method obtained a significant performance gain by 13.74 in MAP with the CNN feature extractor, and improved about 4 points in both LSTM and M-BERT.

Moreover, we conducted further experiments with the assumption of having small labeled data, for example, 4,561 Korean triples in ConceptNet. We added 4,561 triples to training data for the triple classifier. The results show that small labeled data are helpful for improving the performance, but not significantly. Our adversarial network still works well even when a small batch of labeled data is given.

Experiments were also conducted on M-BERT. M-BERT with a small batch of labeled Korean data outperformed other feature extractor models; in particular, it improved MAP significantly, by over 13 points for each baseline model.

**(5) Knowledge Augmentation on Korean** Using the trained triple classifier model, we evaluated how reliable triples can be generated. Figure 3 shows the precision@N of our knowledge augmentation results on the Korean test

| | Dev | | Test | |
|---|---|---|---|---|
| | Accuracy | MAP | Accuracy | MAP |
| TransE (Bordes et al. 2013) | 0.6271 (± 0.11) | 0.4920 (± 0.08) | 0.6088 (± 0.10) | 0.5385 (± 0.06) |
| MT+KBC (Otani et al. 2018) | 0.5273 (± 0.17) | 0.5532 (± 0.02) | 0.5456 (± 0.21) | 0.5744 (± 0.04) |
| LSTM (Luong, Pham, and Manning 2015) | 0.3041 (± 0.00) | 0.3764 (± 0.08) | 0.3553 (± 0.00) | 0.3952 (± 0.06) |
| Adv-LSTM | 0.6663 (± 0.08) | 0.4272 (± 0.04) | 0.6147 (± 0.06) | 0.4343 (± 0.04) |
| Adv-LSTM (+Kr) | 0.6537 (± 0.02) | 0.4348 (± 0.05) | 0.6386 (± 0.05) | 0.4262 (± 0.05) |
| CNN (Kim 2014) | 0.3041 (± 0.00) | 0.5126 (± 0.01) | 0.3512 (± 0.00) | 0.5447 (± 0.00) |
| Adv-CNN | 0.6734 (± 0.02) | 0.6946 (± 0.02) | 0.6323 (± 0.02) | 0.6766 (± 0.01) |
| Adv-CNN (+Kr) | 0.6726 (± 0.02) | 0.7024 (± 0.01) | 0.6398 (± 0.00) | 0.6724 (± 0.02) |
| M-BERT (Devlin et al. 2019) | 0.6990 (± 0.02) | 0.7076 (± 0.02) | 0.6684 (± 0.00) | 0.7180 (± 0.00) |
| Adv-M-BERT | 0.7098 (± 0.04) | 0.7012 (± 0.02) | 0.6723 (± 0.05) | 0.7329 (± 0.02) |
| Adv-M-BERT (+Kr) | **0.7163** (± 0.02) | **0.7162** (± 0.02) | **0.6781** (± 0.02) | **0.7335** (± 0.02) |

Table 1: Performance of triple classifier on Korean development and test sets. Adv- denotes baseline model with adversarial learning, and (+Kr) means small labeled Korean data is added to train data.

set; as well as the number of triples when the top-N triples are augmented. We achieved 93.7% precision@1 with our adversarial method with M-BERT, implying that if labeled resource-rich languages are given, we can obtain target language knowledge by picking the highest-scoring triple with 93.7% reliability.

We translated 3,414,063 English triples in ConceptNet into Korean, and 7,475,644 candidate triples were generated. Next, we measured the validity of the Korean candidate triples and selected the highest scoring Korean triples. As a result, we obtained 626,681 Korean triples at 93.7% precision@1 without any labeled Korean data.

One encouraging fact is that the proposed method performed at more than 85% even at precision@10. In some cases, such as when the quantity of data is more important than the quality and target data is low-resource language, our method is a compelling option for expanding knowledge. In Korean, we obtained 862,304 fairly reliable triples with confidence of over 85%.

**(6) Human Evaluation** In addition, we conducted human evaluation on the automatically generated Korean triples. We randomly selected 1,000 triples among 626,681 Korean triples that were automatically generated by our best model. Then, two annotators tagged with *valid*, *invalid*, or *neutral*. When *neutral* triples were considered invalid, the accuracy was 83.6%, and when they were considered valid triples, the accuracy was 96.6%. Most of the *invalid* triples were semantically wrong but had text similarities between the head and tail entities. e.g., the generated Korean triple `<eun-haeng namu, FormOf, eun-haeng>`[5] is invalid but its confidence score is high because of the text similarity of `eun-haeng`. Setting neutral results aside, our model showed high accuracy without labeled Korean data.

## Multilingual Experiments

**(1) Multilingual Automatically Generated Test Sets** To show that the proposed methods are language-independent, evaluations on different languages should be provided. However, labeled data is scarce for most low-resource languages, and manually constructing data for all different languages is extremely costly. Therefore, we designed an evaluation method that can be used for validating knowledge in several languages and introduced automatically built test sets using ConceptNet.

Although ConceptNet contains triples in 304 languages, most triples are in resource-rich languages. We devised automatic test sets using "`Synonym`" relation in ConceptNet. In ConceptNet, synonyms in different languages are connected with the relation "`Synonym`" as an *inter-language link*. For example, the entities "`dog`" and "`perro`" (dog in Spanish) are connected as "`Synonym`". Using this information, we can build new triple sets by connecting entities with the same meaning in English and other languages. For example, as shown in Figure 4, when we make the Spanish test

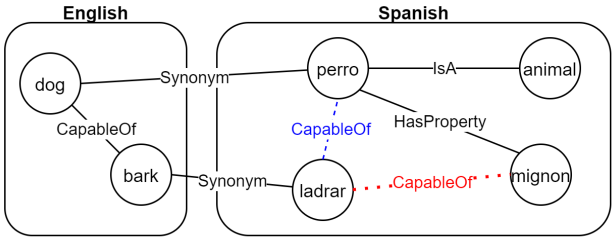


Figure 4: Example of automatically generated test sets.

| Language | # triples | Language | # triples |
|---|---|---|---|
| fr (French) | 38,887 | de (German) | 30,574 |
| es (Spanish) | 1,758 | ar (Arabic) | 360 |
| zh (Chinese) | 91,832 | sv (Swedish) | 2,316 |
| ru (Russian) | 3,905 | ms (Malay) | 410 |
| it (Italian) | 4,563 | sl (Slovenian) | 90 |
| pt (Portuguese) | 2,923 | sk (Slovak) | 222 |
| nl (Dutch) | 2,093 | hi (Hindi) | 112 |
| ja (Japanese) | 59,000 | tr (Turkish) | 539 |

Table 2: Statistics for 16 languages test sets

[5]means `<ginkgo, FormOf, ginkgo nut>`, and eun-haeng has several meanings in Korean including *bank* and *ginkgo nut*

set, the process is as follows:

1. Triples with both head and tail entities as Spanish synonym links are chosen. e.g. <dog, CapableOf, bark>
2. New Spanish triples are obtained by replacing each English entity with Spanish, e.g. <perro, CapableOf, ladrar> (blue dashed-line in Figure 4)
3. Negative triples are generated by replacing one of head and tail entities with its 1-hop neighbor, e.g. <migon, CapableOf, ladrar> (red dotted-line in Figure 4)

Note that when we generated negative triples, we excluded the neighbors that have relations such as {RelatedTo, Synonym, FormOf, IsA, PartOf, DefinedAs, and SimilarTo} with head or tail entities. For example, "animal" is a neighbor of perro in Figure 4, but <animal, CapableOf, ladrar> is not included as a negative triple because the relation is "IsA". In many cases, neighbors in such relations can be regarded as valid triples.

Through this process, we constructed test sets for 16 languages, including those from different language families, such as Germanic, Romance, Slavic, and non-Indo-European. The statistics for each dataset are presented in Table 2.

To demonstrate that our method can be used language-independently, we performed experiments on 16 automatically built test sets proposed in the previous section.

**(2) Experimental Settings** In practice, the knowledge of low resource languages is scarce, and tagging all the triples in different languages is laborious. To simulate these constraints, we limited the training examples to 30K, which is the average for the 30 languages with the fewest resource in ConceptNet. English labeled data was used as resource-rich language data, and the training and development data for the target languages were built in the same way used for building the English training/development datasets; we extracted valid triples from ConceptNet and randomly generated negative triples.

The training and development datasets have 30K examples each, except for Hindi, Swedish, and Turkish. These languages have fewer than 10K triples in ConceptNet; thus, we used smaller-sized training and development datasets. Details of the datasets are described in the Appendix. We used the M-BERT model for the multilingual evaluation with the same settings as the Korean experiments.

**(3) Evaluation on Multilingual Automatically Generated Test Sets** Table 3 shows the evaluation results for the multilingual automated test sets. We compared the M-BERT models with small labeled monolingual training sets and M-BERT using adversarial learning (Adv-M-BERT) without labeled data in the target languages. Our adversarial method outperformed the baseline for all language test sets. We achieved improvements of over 10 points for five languages (Chinese, Russian, Portuguese, German, and Slovak); and of over 20 points for Dutch.

Although our adversarial learning approach improved the performance over the M-BERT on the proposed test sets, the results for most languages are still unsatisfactory, with $\leq$ 50% accuracy, especially in low-resource languages, such as Turkish, Slovenian, and Malay. Considering the necessity of knowledge for low-resource languages, achieving high performance on the proposed test sets would encourage the development of further research. In the same vein, obtaining high performance on all language test sets would be highly challenging.

| | Accuracy (%) | |
|---|---|---|
| | M-BERT | Adv-M-BERT |
| fr (French) | 69.80 ± 0.42 | 71.07 ± 0.82 (+1.27) |
| es (Spanish) | 40.58 ± 0.77 | 43.14 ± 0.21 (+2.56) |
| zh (Chinese) | 75.88 ± 1.36 | 88.89 ± 1.24 (+13.01) |
| ru (Russian) | 42.04 ± 0.08 | 57.42 ± 0.74 (+15.38) |
| it (Italian) | 43.98 ± 0.56 | 50.74 ± 0.23 (+6.75) |
| pt (Portuguese) | 51.96 ± 0.11 | 62.25 ± 0.89 (+10.28) |
| nl (Dutch) | 42.67 ± 1.21 | 62.93 ± 1.15 (+20.25) |
| ja (Japanese) | 68.38 ± 0.65 | 77.15 ± 0.04 (+8.77) |
| de (German) | 56.46 ± 2.06 | 67.93 ± 1.15 (+11.47) |
| ar (Arabic) | 34.96 ± 0.06 | 45.58 ± 1.61 (+10.62) |
| hi (Hindi) | 39.74 ± 0.62 | 45.91 ± 0.58 (+6.21) |
| sv (Swedish) | 41.87 ± 0.51 | 48.23 ± 1.46 (+6.36) |
| sk (Slovak) | 42.15 ± 0.90 | 53.63 ± 0.58 (+11.48) |
| tr (Turkish) | 36.08 ± 2.23 | 43.96 ± 1.05 (+7.88) |
| sl (Slovenian) | 33.50 ± 1.33 | 35.33 ± 0.33 (+1.83) |
| ms (Malay) | 39.89 ± 1.57 | 49.58 ± 0.80 (+9.69) |

Table 3: Results on 16 automatically generated test sets.

## Conclusion & Future Work

In this paper, we propose a knowledge augmentation method for low-resource languages. Our augmentation task consists of two steps: candidate generation and scoring validity. First, we translate triples from resource-rich languages into low-resource languages. In general, translation leads to more than one result due to polysemes and homonyms. To resolve this ambiguity problem, we adopted the adversarial learning approach to measure the validity of candidate triples. In adversarial learning, knowledge learned from a resource-rich language can be transferred to low-resource languages; thus, this approach is suitable for languages for which it is difficult to obtain labeled data. The experimental results showed that applying adversarial learning is effective. Our proposed method for the low-resource language, Korean, achieved 93.7% precision@1 on a manually labeled benchmark. As a result, we obtained 626,681 triples of high-reliable Korean knowledge from 3.4M English triples. Additionally, to show that the proposed method is language-agnostic, we introduced new test sets for 16 languages and validated our model on them. Our adversarial method also obtained strong results on all language test sets.

In future, we plan to expand our method with more languages and to improve the learning method and models to achieve high performance on our test sets. In addition, we will conduct research on how to efficaciously use the knowledge we have generated. Therefore, we expect that future work will address the difficulties related to starting various research efforts for low-resource languages.

## Acknowledgements

This work was supported by Institute for Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00368, A Neural-Symbolic Model for Knowledge Acquisition and Inference Techniques).

## References

Anwar, A.; Li, X.; Yang, Y.; and Wang, Y. 2019. Constructing Uyghur Commonsense Knowledge Base by Knowledge Projection. *Applied Sciences* 9(16): 3318.

Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. In Burges, C. J. C.; Bottou, L.; Welling, M.; Ghahramani, Z.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 26*, 2787–2795. Curran Associates, Inc. URL http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf.

Cettolo, M.; Federico, M.; Bentivogli, L.; Jan, N.; Sebastian, S.; Katsuitho, S.; Koichiro, Y.; and Christian, F. 2017. Overview of the iwslt 2017 evaluation campaign. In *International Workshop on Spoken Language Translation*, 2–14.

Chen, X.; Sun, Y.; Athiwaratkun, B.; Cardie, C.; and Weinberger, K. Q. 2018. Adversarial Deep Averaging Networks for Cross-Lingual Sentiment Classification. *Trans. Assoc. Comput. Linguistics* 6: 557–570. URL https://transacl.org/ojs/index.php/tacl/article/view/1413.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics. doi:10.18653/v1/n19-1423. URL https://doi.org/10.18653/v1/n19-1423.

Feng, X.; Tang, D.; Qin, B.; and Liu, T. 2016. English-Chinese Knowledge Base Translation with Neural Network. In Calzolari, N.; Matsumoto, Y.; and Prasad, R., eds., *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, 2935–2944. ACL. URL https://www.aclweb.org/anthology/C16-1276/.

Ganin, Y.; and Lempitsky, V. S. 2015. Unsupervised Domain Adaptation by Backpropagation. In Bach, F. R.; and Blei, D. M., eds., *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, 1180–1189. JMLR.org. URL http://proceedings.mlr.press/v37/ganin15.html.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. S. 2016. Domain-Adversarial Training of Neural Networks. *J. Mach. Learn. Res.* 17: 59:1–59:35. URL http://jmlr.org/papers/v17/15-239.html.

Joulin, A.; Bojanowski, P.; Mikolov, T.; Jégou, H.; and Grave, E. 2018. Loss in Translation: Learning Bilingual Word Mapping with a Retrieval Criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.

Kim, J.; Kim, Y.; Sarikaya, R.; and Fosler-Lussier, E. 2017. Cross-Lingual Transfer Learning for POS Tagging without Cross-Lingual Resources. In Palmer, M.; Hwa, R.; and Riedel, S., eds., *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, 2832–2838. Association for Computational Linguistics. doi:10.18653/v1/d17-1302. URL https://doi.org/10.18653/v1/d17-1302.

Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1746–1751. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1181. URL https://www.aclweb.org/anthology/D14-1181.

Lison, P.; Tiedemann, J.; and Kouylekov, M. 2018. OpenSubtitles2018: Statistical Rescoring of Sentence Alignments in Large, Noisy Parallel Corpora. In Calzolari, N.; Choukri, K.; Cieri, C.; Declerck, T.; Goggi, S.; Hasida, K.; Isahara, H.; Maegaard, B.; Mariani, J.; Mazo, H.; Moreno, A.; Odijk, J.; Piperidis, S.; and Tokunaga, T., eds., *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2018/summaries/294.html.

Liu, H.; and Singh, P. 2004. ConceptNet—a practical commonsense reasoning tool-kit. *BT technology journal* 22(4): 211–226.

Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Luong, T.; Pham, H.; and Manning, C. D. 2015. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1412–1421. Lisbon, Portugal: Association for Computational Linguistics. doi:10.18653/v1/D15-1166. URL https://www.aclweb.org/anthology/D15-1166.

Moussallem, D.; Soru, T.; and Ngomo, A. N. 2019. THOTH: Neural Translation and Enrichment of Knowledge Graphs. In Ghidini, C.; Hartig, O.; Maleshkova, M.; Svátek, V.; Cruz, I. F.; Hogan, A.; Song, J.; Lefrançois, M.; and Gandon, F., eds., *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part I*, volume 11778 of *Lecture Notes in Computer Science*, 505–522. Springer. doi:

10.1007/978-3-030-30793-6_29. URL https://doi.org/10.1007/978-3-030-30793-6_29.

Otani, N.; Kiyomaru, H.; Kawahara, D.; and Kurohashi, S. 2018. Cross-lingual Knowledge Projection Using Machine Translation and Target-side Knowledge Base Completion. In Bender, E. M.; Derczynski, L.; and Isabelle, P., eds., *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, 1508–1520. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/C18-1128/.

Sun, Z.; Deng, Z.-H.; Nie, J.-Y.; and Tang, J. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. *arXiv preprint arXiv:1902.10197* .

Yao, L.; Mao, C.; and Luo, Y. 2019. KG-BERT: BERT for Knowledge Graph Completion. *arXiv preprint arXiv:1909.03193* .

Zhang, M.; Liu, Y.; Luan, H.; and Sun, M. 2017. Adversarial Training for Unsupervised Bilingual Lexicon Induction. In Barzilay, R.; and Kan, M., eds., *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, 1959–1970. Association for Computational Linguistics. doi:10.18653/v1/P17-1179. URL https://doi.org/10.18653/v1/P17-1179.