Reflection-based Word Attribute Transfer

Yoichi Ishibashi, Katsuhito Sudoh, Koichiro Yoshino, Satoshi Nakamura
Nara Institute of Science and Technology
{ishibashi.yoichi.ir3, sudoh, koichiro, s-nakamura}@is.naist.jp

Abstract

Word embeddings, which often represent such analogic relations as $\overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman} \approx \overrightarrow{queen}$, can be used to change a word's attribute, including its gender. For transferring *king* into *queen* in this analogy-based manner, we subtract a difference vector $\overrightarrow{man} - \overrightarrow{woman}$ based on the knowledge that *king* is male. However, developing such knowledge is very costly for words and attributes. In this work, we propose a novel method for word attribute transfer based on reflection mappings without such an analogy operation. Experimental results show that our proposed method can transfer the word attributes of the given words without changing the words that do not have the target attributes.

1 Introduction

Word-embedding methods handle word semantics in natural language processing (Mikolov et al., 2013a,b; Pennington et al., 2014; Vilnis and McCallum, 2015; Bojanowski et al., 2017). Such word-embedding models as skip-gram with negative sampling (SGNS; Mikolov et al., 2013b) or GloVe (Pennington et al., 2014) capture such analogic relations as $\overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman} \approx \overrightarrow{queen}$. Previous work (Levy and Goldberg, 2014b; Arora et al., 2016; Gittens et al., 2017; Ethayarajh et al., 2019; Allen and Hospedales, 2019) offers theoretical explanation based on Pointwise Mutual Information (PMI; Church and Hanks, 1990) for maintaining analogic relations in word vectors.

These relations can be used to transfer a certain attribute of a word, such as changing *king* into *queen* by transferring its gender. This transfer can be applied to perform data augmentation; for example, rewriting *He is a boy* to *She is a girl*. It can be used to generate negative examples for natural language inference, for example. We tackle a novel task that transfers any word associated with certain attributes: *word attribute transfer*.

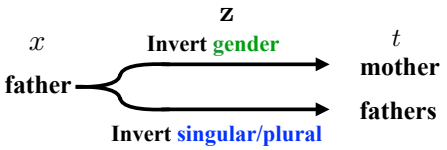

Figure 1: Examples of word attribute transfer

A naive way for word attribute transfer is to use a difference vector based on analogic relations, such as adding $\overrightarrow{woman} - \overrightarrow{man}$ to $\overrightarrow{king}$ to obtain $\overrightarrow{queen}$. This requires explicit knowledge whether an input word is male or female; we have to add a difference vector to a male word and subtract it from a female word for the gender transfer. We also have to avoid changing words without gender attributes, such as *is* and *a* in the example above, since they are non-attribute words. Developing such knowledge is very costly for words and attributes in practice. In this work, we propose a novel framework for a word attribute transfer based on *reflection* that does not require explicit knowledge of the given words in its prediction.

The contribution of this work is two-fold: (1) We propose a word attribute transfer method that obtains a vector with an inverted binary attribute without explicit knowledge. (2) The proposed method demonstrates more accurate word attribute transfer for words that have target attributes than other baselines without changing the words that do not have the target attributes.

2 Word Attribute Transfer Task

In this task, we focus on modeling the binary attributes (e.g. male and female[1]). Let x denote a word and let $\mathbf{v}_x$ denote its vector representation. We assume that $\mathbf{v}_x$ is learned in advance

[1]Gender-specific words are sometimes considered socially problematic. Here we use this as an example from the man-woman relation.

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pages 51–58

with an embedding model, such as skip-gram. In this task, we have two inputs, word x and vector $\mathbf{z}$, which represent a certain target attribute, and output word t with the inverted attribute of x for $\mathbf{z}$. In this paper, $\mathbf{z}$ is a 300-dimensional vector embedded from a target attribute ID using an embedding function of a deep learning framework. For example, given a set of attributes $\mathcal{Z} = \{\text{gender}, \text{antonym}\}$, we assign different random vectors $\mathbf{z}_{\text{gender}}$ for gender and $\mathbf{z}_{\text{antonym}}$ for antonym, respectively. Let $\mathcal{A}$ denote a set of triplets $(x, t, \mathbf{z})$, e.g., $(man, woman, \mathbf{z}_{\text{gender}}) \in \mathcal{A}$, and $\mathcal{N}$ denote a set of words without attribute $\mathbf{z}$, e.g., $(person, \mathbf{z}_{\text{gender}}) \in \mathcal{N}$. This task transfers input word vector $\mathbf{v}_x$ to target word vector $\mathbf{v}_t$ by transfer function $f_{\mathbf{z}}$ that inverts attribute $\mathbf{z}$ of $\mathbf{v}_x$:

$$\mathbf{v}_t \approx \mathbf{v}_y = f_{\mathbf{z}}(\mathbf{v}_x). \tag{1}$$

The following property must be satisfied: (1) attribute words $\{x|(x, t, \mathbf{z}) \in \mathcal{A}\}$ are transferred to their counterparts and (2) non-attribute words $\{x|(x, \mathbf{z}) \in \mathcal{N}\}$ are not changed (transferred back into themselves). For instance with $\mathbf{z}_{\text{gender}}$, given input word *man*, gender attribute transfer $f_{\mathbf{z}_{\text{gender}}}(\mathbf{v}_{man})$ should result in a vector close to $\mathbf{v}_{woman}$. Given another input word *person* as x, the results should be $\mathbf{v}_{person}$.

3 Analogy-based Word Attribute Transfer

Analogy is a general idea that can be used for word attribute transfer. PMI-based word embedding, such as SGNS and GloVe, captures analogic relations, including Eq. 2 (Mikolov et al., 2013c; Levy and Goldberg, 2014a; Linzen, 2016). By rearranging Eq. 2, Eq. 3 is obtained:

$$\mathbf{v}_{queen} \approx \mathbf{v}_{king} - \mathbf{v}_{man} + \mathbf{v}_{woman}, \tag{2}$$
$$\approx \mathbf{v}_{king} - (\mathbf{v}_{man} - \mathbf{v}_{woman}). \tag{3}$$

The analogy-based transfer function is

$$f_{\mathbf{z}}(\mathbf{v}_x) = \begin{cases} \mathbf{v}_x - \mathbf{d} & \text{if} \quad x \in \mathcal{M}, \\ \mathbf{v}_x + \mathbf{d} & \text{if} \quad x \in \mathcal{F}, \end{cases} \tag{4}$$

where $\mathcal{M}$ is a set of words with a target attribute (e.g., male) and $\mathcal{F}$ is a set of words with an inverse attribute (e.g., female). $\mathbf{d}$ is a difference vector, such as $\mathbf{v}_{man} - \mathbf{v}_{woman}$. Eq. 4 indicates that the operation changes depending on whether input word x belongs to $\mathcal{M}$ or $\mathcal{F}$. However, to transfer the word attribute by analogy, we need such explicit knowledge as attribute value ($\mathcal{M}$, $\mathcal{F}$ or others) that is contained by the input word.

4 Reflection-based Word Attribute Transfer

4.1 Ideal Transfer without Knowledge

What is ideal transfer function $f_{\mathbf{z}}$ for a word attribute transfer? The following are the ideal natures of such a transfer function:

$$\forall (m, w, \mathbf{z}) \in \mathcal{A}, \quad \mathbf{v}_m = f_{\mathbf{z}}(\mathbf{v}_w), \tag{5}$$
$$\forall (m, w, \mathbf{z}) \in \mathcal{A}, \quad \mathbf{v}_w = f_{\mathbf{z}}(\mathbf{v}_m), \tag{6}$$
$$\forall (u, \mathbf{z}) \in \mathcal{N}, \quad \mathbf{v}_u = f_{\mathbf{z}}(\mathbf{v}_u). \tag{7}$$

This function $f_{\mathbf{z}}$ enables a word to be transferred without explicit knowledge because operation $f_{\mathbf{z}}$ does not change depending on whether input word belongs to $\mathcal{M}$ or $\mathcal{F}$. By combining Eqs. 5, 6 and 7, we obtain the following formulas:

$$\forall (m, w, \mathbf{z}) \in \mathcal{A}, \quad \mathbf{v}_m = f_{\mathbf{z}}(\, f_{\mathbf{z}}(\mathbf{v}_m)\,), \tag{8}$$
$$\forall (m, w, \mathbf{z}) \in \mathcal{A}, \quad \mathbf{v}_w = f_{\mathbf{z}}(\, f_{\mathbf{z}}(\mathbf{v}_w)\,), \tag{9}$$
$$\forall (u, \mathbf{z}) \in \mathcal{N}, \quad \mathbf{v}_u = f_{\mathbf{z}}(\, f_{\mathbf{z}}(\mathbf{v}_u)\,). \tag{10}$$

Hence, the ideal transfer function is a mapping that becomes an identity mapping when we apply it twice for any $\mathbf{v}$. Such a mapping is called *involution* in geometry. For example, $f : \mathbf{v} \mapsto -\mathbf{v}$ is one example of an involution.

4.2 Reflection

Reflection $\text{Ref}_{\mathbf{a},\mathbf{c}}$ is an ideal function because this mapping is an involution:

$$\forall \mathbf{v} \in \mathbb{R}^n, \quad \mathbf{v} = \text{Ref}_{\mathbf{a},\mathbf{c}}(\, \text{Ref}_{\mathbf{a},\mathbf{c}}(\mathbf{v})\,). \tag{11}$$

Reflection reverses the location between two vectors in a Euclidean space through an hyperplane called a *mirror*. Reflection is different from inverse mapping. When m and w are paired words, reflection can transfer $\mathbf{v}_m$ and $\mathbf{v}_w$ each other with identical reflection mapping as in Eqs. 5 and 6, but an inverse mapping cannot. Given vector $\mathbf{v}$ in Euclidean space $\mathbb{R}^n$, the formula for the reflection in the mirror is given:

$$\text{Ref}_{\mathbf{a},\mathbf{c}}(\mathbf{v}) = \mathbf{v} - 2\frac{(\mathbf{v} - \mathbf{c}) \cdot \mathbf{a}}{\mathbf{a} \cdot \mathbf{a}}\mathbf{a}, \tag{12}$$

where $\mathbf{a} \in \mathbb{R}^n$ is a vector orthogonal to the mirror and $\mathbf{c} \in \mathbb{R}^n$ is a point through which the mirror passes. $\mathbf{a}$ and $\mathbf{c}$ are parameters that determine the mirror.

4.3 Proposed method: Reflection-based Word Attribute Transfer

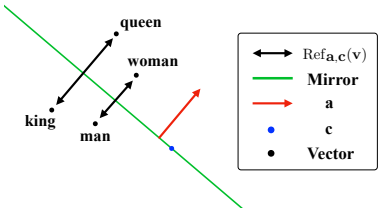

Figure 2: Reflection-based word attribute transfer with a single mirror

We apply reflection to the word attribute transfer. We learn a mirror (hyperplane) in a pre-trained embedding space using training word pairs with binary attribute $\mathbf{z}$ (Fig. 2). Since the mirror is uniquely determined by two parameter vectors, $\mathbf{a}$ and $\mathbf{c}$, we estimate $\mathbf{a}$ and $\mathbf{c}$ from target attribute $\mathbf{z}$ using fully connected multi-layer perceptrons:

$$\mathbf{a} = \mathrm{MLP}_{\theta_1}(\mathbf{z}), \quad (13)$$
$$\mathbf{c} = \mathrm{MLP}_{\theta_2}(\mathbf{z}), \quad (14)$$

where θ is a set of trainable parameters of MLP_θ. Here, θ_1 and θ_2 are optimized for each attribute dataset. Transferred vector $\mathbf{v}_y$ is obtained by inverting attribute $\mathbf{z}$ of $\mathbf{v}_x$ by reflection:

$$\mathbf{v}_y = \mathrm{Ref}_{\mathbf{a},\mathbf{c}}(\mathbf{v}_x). \quad (15)$$

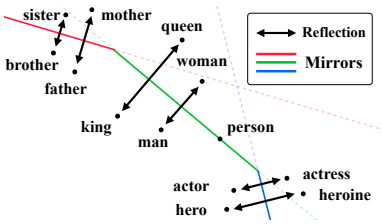

Figure 3: Reflection with parameterized mirrors

Reflection with a mirror by Eqs. 13 and 14 assumes a single mirror that only depends on $\mathbf{z}$. Previous discussion assumed pairs that share a stable pair, such as *king* and *queen*. However, since gendered words often do not come in pairs, gender is not stable enough to be modeled by a single mirror. For example, although *actress* is exclusively feminine, *actor* is clearly neutral in many cases. Thus, *actor* is not obviously a masculine counterpart like *king*. In fact, bias exists in gender words in the embedding space (Zhao et al., 2018; Kaneko and Bollegala, 2019). This phenomenon can occur not only with gender attributes but also with other attributes. The single mirror assumption forces the mirror to be a hyperplane that goes through the midpoints for all the word vector pairs. However, the vector pair *actor-actress* shown on the right in Fig. 3 cannot be transferred well since the single mirror (the green line) does not satisfy this constraint due to the bias of the embedding space. To solve this problem, we propose *parameterized mirrors*, based on the idea of using different mirrors for different words. We define mirror parameters $\mathbf{a}$ and $\mathbf{c}$ using word vector $\mathbf{v}_x$ to be transferred in addition to attribute vector $\mathbf{z}$:

$$\mathbf{a} = \mathrm{MLP}_{\theta_1}([\mathbf{z}; \mathbf{v}_x]), \quad (16)$$
$$\mathbf{c} = \mathrm{MLP}_{\theta_2}([\mathbf{z}; \mathbf{v}_x]), \quad (17)$$

where $[\cdot;\cdot]$ indicates the vector concatenation in the column. The *parameterized mirrors* are expected to work more flexibly on different words than a single mirror because *parameterized mirrors* dynamically determine similar mirrors for similar words. For instance, as shown in Fig. 3, suppose we learned the mirror (the blue line) that transfers $\mathbf{v}_{hero}$ to $\mathbf{v}_{heroine}$ in advance. If input word vector $\mathbf{v}_{actor}$ resembles $\mathbf{v}_{hero}$, a mirror that resembles the one for $\mathbf{v}_{hero}$ should be derived and used for the transfer.

On the other hand, the reflection works as an identity mapping for a vector on the mirror (e.g., $\mathbf{v}_{person}$ in Fig 3). That is, the proposed method assumes that non-attribute word vectors are located on the mirror. Since we used a 300-dimensional embedded space in the experiment, we assume that the non-attribute word vector exists in a 299-dimensional subspace.

Here, it should be noted that Eq. 11 may not hold for parameterized mirrors. In reflection with a single mirror, it is true that $\mathbf{v} = \mathrm{Ref}_{\mathbf{a},\mathbf{c}}(\mathrm{Ref}_{\mathbf{a},\mathbf{c}}(\mathbf{v}))$. However, with the $\mathbf{v}$-parameterized reflection $\mathrm{Ref}_{\mathbf{a}_\mathbf{v},\mathbf{c}_\mathbf{v}}(\mathbf{v})$, this is not guaranteed. Because mirror parameters $\mathbf{a}_\mathbf{v}$ and $\mathbf{c}_\mathbf{v}$ depend on an input word vector as Eqs. 16 and 17. Thus, we exclude this constraint and employ the constraints given by Eqs. 5-7 for our loss function.

The following property must be satisfied in word attribute transfer: (1) words with attribute $\mathbf{z}$ are transferred and (2) words without it are not transferred. Thus, loss $L(\theta_1, \theta_2)$ is defined:

$$L(\theta_1, \theta_2) = \frac{1}{|\mathcal{A}|} \sum_{(x,t,\mathbf{Z})\in\mathcal{A}} (\mathbf{v}_y - \mathbf{v}_t)^2 \quad (18)$$
$$+ \frac{1}{|\mathcal{N}|} \sum_{(x,\mathbf{Z})\in\mathcal{N}} (\mathbf{v}_y - \mathbf{v}_x)^2, \quad (19)$$

where Eq. 18 is a term that draws target word vector $\mathbf{v}_{t_i}$ closer to corresponding transferred vector $\mathbf{v}_{y_i}$ and Eq. 19 is a term that prevents words without a target attribute from being moved by transfer function. $\mathbf{v}_y$ is the output of a reflection (Eq. 15).

5 Experiment

We evaluated the performance of the word attribute transfer using data with four different attributes. We used 300-dimensional word2vec and GloVe as the pre-trained word embedding. We used four different datasets of word pairs with four binary attributes: Male-Female, Singular-Plural, Capital-Country, and Antonym (Table 1). These word pairs were collected from analogy test sets (Mikolov et al., 2013a; Gladkova et al., 2016) and the Internet. Noun antonyms were taken from the literature (Nguyen et al., 2017). For non-attribute dataset $\mathcal{N}$, we sampled words from the vocabulary of word embedding. We sampled from 4 to 50 words for training and 1000 for the test ($|\mathcal{N}_{\text{test}}| = 1000$).

Table 1: Statistics of binary attribute word pair datasets (in number of word pairs)

Dataset $\mathcal{A}$	Train	Val	Test	Total
Male-Female (MF)	29	12	12	53
Singular-Plural (SP)	90	25	25	140
Capital-Country (CC)	59	25	25	109
Antonym (AN)	1354	290	290	1934

5.1 Evaluation Metrics

We measured the accuracy and stability performances of the word attribute transfer. The accuracy measures how many input words in $\mathcal{A}_{\text{test}}$ were transferred correctly to the corresponding target words. The stability score measures how many words in $\mathcal{N}_{\text{test}}$ were not mapped to other words. For example, in the Male-Female transfer, given *man*, the transfer is regarded as correct if *woman* is the closest word to the transferred vector; otherwise it is incorrect. Given *person*, the transfer is regarded as correct if *person* is the closest word to the transferred vector; otherwise it is incorrect. The accuracy and stability scores are calculated by the following formula:

$$\delta(\mathbf{v}_y, t) = \begin{cases} 1 & \text{if} \quad \arg\max_{k \in \mathcal{V}} (\cos(\mathbf{v}_y, \mathbf{v}_k)) = t \\ 0 & \text{otherwise,} \end{cases} \tag{20}$$

$$\text{Accuracy} = \frac{1}{|\mathcal{A}_{\text{test}}|} \sum_{(x,t,\mathbf{Z}) \in \mathcal{A}_{\text{test}}} \delta(\mathbf{v}_y, t), \tag{21}$$

$$\text{Stability} = \frac{1}{|\mathcal{N}_{\text{test}}|} \sum_{(x,\mathbf{Z}) \in \mathcal{N}_{\text{test}}} \delta(\mathbf{v}_y, x), \tag{22}$$

where $\mathcal{V}$ is the vocabulary of the word embedding model and $\cos(\mathbf{v}_y, \mathbf{v}_k)$ is the cosine similarity measure, defined as: $\cos(\mathbf{v}_y, \mathbf{v}_k) = \frac{\mathbf{v}_y \cdot \mathbf{v}_k}{\|\mathbf{v}_y\| \|\mathbf{v}_k\|}$.

5.2 Methods and Configurations

In our experiment, we compared our proposed method with the following baseline methods[2]:

REF Reflection-based word attribute transfer with a single mirror. We used a fully connected 2-layer MLP with 300 hidden units and ReLU (Glorot et al., 2011) to estimate $\mathbf{a}$ and $\mathbf{c}$.

REF+PM Reflection-based word attribute transfer with *parameterized mirrors*. We used the same architecture of MLP as the REF.

MLP Fully connected MLP with 300 hidden units and ReLU: $\mathbf{v}_y = \text{MLP}([\mathbf{v}_x; \mathbf{z}])$. The highest accuracy models in SGNS are a 2-layer MLP for Capital-Country and 3-layer MLP for the other datasets. The highest accuracy models in GloVe are a 2-layer MLP for Singular-Plural and 3-layer MLP for the other datasets.

DIFF Analogy-based word attribute transfer with a difference vector: $\mathbf{d} = \mathbf{v}_m - \mathbf{v}_w$, where m and w are in the training data of $\mathcal{A}$. We chose $\mathbf{d}$ that achieved the best accuracy in the validation data of $\mathcal{A}$. We determined whether to add or subtract $\mathbf{d}$ to $\mathbf{v}_x$ based on the explicit knowledge (Eq. 4). DIFF$^+$ and DIFF$^-$ transfer with a difference vector regardless of the explicit knowledge. $^+$ and $^-$ add or subtract the difference vector to any input word vector.

MEANDIFF Analogy-based word attribute transfer with a mean difference vector $\bar{\mathbf{d}}$: $\bar{\mathbf{d}} = \frac{1}{|\mathcal{A}_{\text{train}}|} \sum_{(m_i,w_i,\mathbf{Z}) \in \mathcal{A}_{\text{train}}} (\mathbf{v}_{m_i} - \mathbf{v}_{w_i})$. We determined whether to add or subtract $\bar{\mathbf{d}}$ to $\mathbf{v}_x$ based on the explicit knowledge (Eq. 4).

For proposed methods, we used the Adam optimizer (Kingma and Ba, 2015) with $\alpha = 10^{-4}$ for Male-Female, Singular-Plural and Capital-Country,

[2]Our code and datasets are available at: https://github.com/ahclab/reflection

and $\alpha = 15^{-3}$ for Antonym (the other hyperparameters were the same as the original one (Kingma and Ba, 2015)). We did not use such regularization methods as dropout (Srivastava et al., 2014) or batch normalization (Ioffe and Szegedy, 2015) because they did not show any improvement in our pilot test. We implemented REF, REF+PM and MLP with Chainer (Tokui et al., 2019), which is one of the best deep learning frameworks.

5.3 Evaluation in Accuracy and Stability

Table 2 shows the accuracy and stability results. Different pre-trained word embeddings GloVe or word2vec gave similar results. REF+PM achieved the best accuracy among the methods that did not use explicit attribute knowledge. For example, the accuracy of REF+PM was 76% in Capital-Country, but the accuracy of DIFF$^+$ was 26%. For stability, reflection-based transfers achieved outstanding stability scores that exceeded 99%. The results show that our proposed method transfers an input word if it has a target attribute and does not transfer an input word with better score than the baselines otherwise, even though the proposed method does not use attribute knowledge of the input words. MLP worked poorly both in accuracy and stability. On the antonym dataset, although the transfer accuracy by the proposed method was a bit lower than that by MLP, the proposed methods stability was 100% and that of MLP was extremely poor: about 1%.

We investigated the relation between the training data size of the non-attribute words, and the stability of the learning-based methods by conducting an additional experiment that varied $|\mathcal{N}_{\text{train}}|$. The stability scores by MLP did not improve (Table 3). On the other hand, REF+PM achieved high stability scores with $|\mathcal{N}_{\text{train}}| = 0$ and maintained the accuracy. We hypothesized that the high stability came from the distance between the word and its mirror. If non-attribute words are distributed on the mirror, they will not be transferred. We investigated the distance between input word vector $\mathbf{v}_x$ and its mirror (Fig. 4). The result shows that non-attribute words are close to the mirror, and attribute words are distributed away from it. In Male-Female and Singular-Plural, the distance is not significantly farther than Antonym and Capital-Country. If the distance between paired words is very small, the distance between the word and its mirror is also small. Fig. 5 shows the distribution of the distance between input $\mathbf{v}_x$ and target word vector $\mathbf{v}_t$. The distance of Male-Female and Singular-Plural is much smaller than Capital-Country and Antonym.

5.4 Visualization of Parameterized Mirrors

Figure 6 shows the t-SNE results of mirror parameter $\mathbf{a}$ obtained for the test words. Paired mirror, $(\mathbf{a}_x, \mathbf{a}_t)$, is connected by a line segment. Fig. 6 suggests that the mirror parameters of the paired words are similar to each other and that those with the attribute form a cluster; words with the same attribute have similar mirror parameters $\mathbf{a}$.

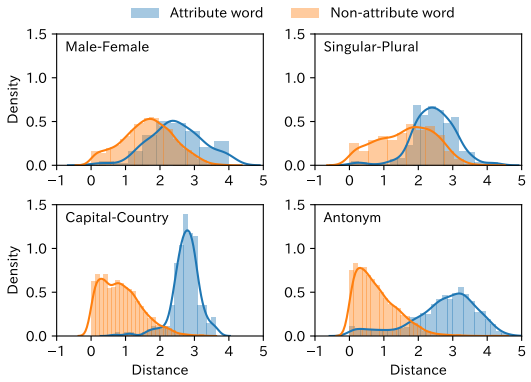

Figure 4: Distribution of distance between input word vector and its mirror $\frac{|(\mathbf{v}_x - \mathbf{c}) \cdot \mathbf{a}|}{\|\mathbf{a}\|}$ learned by REF+PM. Non-attribute words are close to the mirror, and attribute words are distributed away from it.

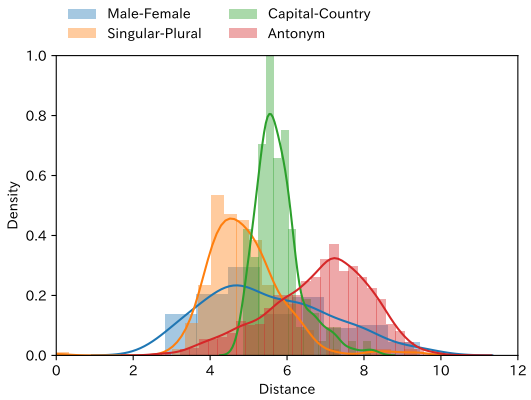

Figure 5: Distribution of distance between input word vector $\mathbf{v}_x$ and target word vector $\mathbf{v}_t$

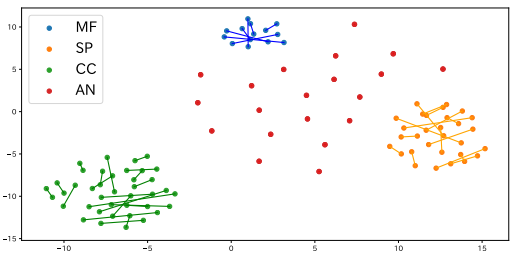

Figure 6: Two-dimensional t-SNE projection of $\mathbf{a}$

Table 2: Results in accuracy and stability scores: MF, SP, CC, and AN are datasets.

Method	Knowledge	word2vec								GloVe							
		Accuracy (%)				Stability (%)				Accuracy (%)				Stability (%)			
		MF	SP	CC	AN	MF	SP	CC	AN	MF	SP	CC	AN	MF	SP	CC	AN
REF		20.8	0.0	36.0	0.0	99.8	**100.0**	**99.8**	**100.0**	12.5	2.0	26.0	0.0	**100.0**	**100.0**	**100.0**	**100.0**
REF+PM		**41.7**	**22.0**	**58.0**	28.8	**99.9**	99.4	99.4	**100.0**	**45.8**	**50.0**	**76.0**	33.5	99.7	99.1	99.2	**100.0**
MLP		8.3	4.0	12.0	**35.9**	2.2	0.0	2.7	1.9	4.2	10.0	18.0	**36.7**	5.1	7.0	5.2	1.2
DIFF $^+$		25.0	2.0	32.0	-	72.1	77.9	53.9	-	25.0	2.0	26.0	-	99.3	94.2	99.3	-
DIFF $^-$		25.0	2.0	30.0	-	49.6	78.2	56.3	-	25.0	2.0	24.0	-	**100.0**	99.9	99.5	-
MEANDIFF $^+$		4.2	0.0	22.0	-	98.6	99.4	87.6	-	0.0	0.0	22.0	-	**100.0**	**100.0**	**100.0**	-
MEANDIFF $^-$		8.3	0.0	14.0	-	97.2	99.3	92.4	-	0.0	0.0	0.0	-	**100.0**	**100.0**	**100.0**	-
DIFF	✓	62.5	4.0	64.0	-	-	-	-	-	50.0	4.0	44.0	-	-	-	-	-
MEANDIFF	✓	12.5	0.0	36.0	-	-	-	-	-	0.0	0.0	0.0	-	-	-	-	-

Table 3: Relation among size of $|\mathcal{N}_{\text{train}}|$ and stability of learning-based methods

		Accuracy (%)				Stability (%)			
		$\lvert\mathcal{N}_{\text{train}}\rvert$				$\lvert\mathcal{N}_{\text{train}}\rvert$			
		0	4	10	50	0	4	10	50
MF	REF	12.5	12.5	12.5	12.5	**100.0**	**100.0**	**100.0**	**100.0**
	REF+PM	**45.8**	**41.7**	**37.5**	**41.7**	99.7	99.9	99.9	99.9
	MLP	0.0	4.2	0.0	4.2	0.0	0.4	1.0	5.0
SP	REF	0.0	0.0	2.0	0.0	**100.0**	**100.0**	**100.0**	**100.0**
	REF+PM	**48.0**	**40.0**	**50.0**	**46.0**	53.3	99.1	99.1	99.8
	MLP	4.0	6.0	6.0	10.0	0.0	0.5	1.7	7.0
CC	REF	24.0	26.0	24.0	20.0	**100.0**	**100.0**	**100.0**	**100.0**
	REF+PM	**76.0**	**72.0**	**74.0**	**74.0**	99.2	**100.0**	99.9	99.9
	MLP	16.0	10.0	14.0	18.0	0.0	0.4	1.0	5.2
AN	REF	0.0	0.0	0.0	0.0	**100.0**	**100.0**	**100.0**	**100.0**
	REF+PM	26.9	26.7	33.5	25.7	**100.0**	**100.0**	**100.0**	**100.0**
	MLP	**29.5**	**29.7**	**36.7**	**36.6**	0.1	0.5	1.2	4.6

5.5 Transfer Example

Table 4 shows the gender transfer results for a tiny example sentence. Here the attribute transfer was applied to every word in the sentence. MLP made many wrong transfers. Analogy-based transfers can transfer only in one direction. REF+PM can transfer only attribute words. Table 5 shows that words with different target attributes were transferred by each reflection-based transfer.

Table 4: Comparison of gender transfers. Each method transfers words in a sentence one by one.

X	the woman got married when you were a boy.
REF	the **woman** got married when you were a **boy**.
REF+PM	the **man** got married when you were a **girl**.
DIFF $^+$	the **man** got married when you were a **boy**.
DIFF $^-$	she **woman** got married she you were a **girl**.
MLP	By_Katie_Klingsporn **girlfriend** Valerie_Glodowski fiancee Doughty_Evening_Chronicle ma'am Bob_Grossweiner_& a **mother**.

Table 5: Transfer of different attributes with REF+PM

X	the rich actor wants to visit the beautiful city in tokyo.
+ MF	the rich **actress** wants to visit the beautiful city in tokyo.
+ SP	the rich **actresses** wants to visit the beautiful **cities** in tokyo.
+ CC	the rich actresses wants to visit the beautiful cities in **japan**.
+ AN	the **poor** actresses wants to visit the beautiful cities in japan.

6 Related Work

The theory of analogic relations in word embeddings has been widely discussed (Levy and Goldberg, 2014b; Arora et al., 2016; Gittens et al., 2017; Ethayarajh et al., 2019; Allen and Hospedales, 2019; Linzen, 2016). In our work, we focus on the analogic relations in a word embedding space and propose a novel framework to obtain a word vector with inverted attributes. The style transfer task (Niu et al., 2018; Prabhumoye et al., 2018; Logeswaran et al., 2018; Jain et al., 2019; Dai et al., 2019; Lample et al., 2019) resembles ours. In style transfer, the text style of the input sentences is changed. For instance, Jain et al. (2019) transferred from formal to informal sentences. These style transfer tasks use sentence pairs; our word attribute transfer task uses word pairs. Style transfer changes sentence styles, but our task changes the word attributes. Soricut and Och (2015) studied morphological transformation based on character information. Our work aims for more general attribute transfer, such as gender transfer and antonym, and is not limited to morphological transformation.

7 Conclusion

This research aims to transfer word binary attributes (e.g., gender) for applications such as data augmentation of a sentence. We can transfer the word attribute with analogy of word vectors, but it requires explicit knowledge whether the input word has the attribute or not (e.g., *man* $\in$ gender, *woman* $\in$ gender, *person* $\notin$ gender). The proposed method transfers binary word attributes using reflection-based mappings and keeps non-attribute words unchanged, without attribute knowledge in inference time. The experimental results showed that the proposed method outperforms analogy-based and MLP baselines in transfer accuracy for attribute words and stability for non-attribute words.

References

Carl Allen and Timothy M. Hospedales. 2019. Analogies Explained: Towards Understanding Word Embeddings. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 223–231.

Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2016. A Latent Variable Model Approach to PMI-based Word Embeddings. *Transactions of the Association for Computational Linguistics*, 4:385–399.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Kenneth Ward Church and Patrick Hanks. 1990. Word Association Norms, Mutual Information, and Lexicography. *Computational Linguistics*, 16(1):22–29.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style Transformer: Unpaired Text Style Transfer without Disentangled Latent Representation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 5997–6007.

Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. 2019. Towards Understanding Linear Word Analogies. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3253–3262.

Alex Gittens, Dimitris Achlioptas, and Michael W. Mahoney. 2017. Skip-Gram - Zipf + Uniform = Vector Additivity. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 69–76.

Anna Gladkova, Aleksandr Drozd, and Satoshi Matsuoka. 2016. Analogy-based detection of morphological and semantic relations with word embeddings: what works and what doesn't. In *Proceedings of the Student Research Workshop, SRW@HLT-NAACL 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, pages 8–15.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. 2011. Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2011, Fort Lauderdale, USA, April 11-13, 2011*, pages 315–323.

Sergey Ioffe and Christian Szegedy. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 448–456.

Parag Jain, Abhijit Mishra, Amar Prakash Azad, and Karthik Sankaranarayanan. 2019. Unsupervised Controllable Text Formalization. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019.*, pages 6554–6561.

Masahiro Kaneko and Danushka Bollegala. 2019. Gender-preserving Debiasing for Pre-trained Word Embeddings. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1641–1650.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Guillaume Lample, Sandeep Subramanian, Eric Michael Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2019. Multiple-Attribute Text Rewriting. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*.

Omer Levy and Yoav Goldberg. 2014a. Linguistic Regularities in Sparse and Explicit Word Representations. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning, CoNLL 2014, Baltimore, Maryland, USA, June 26-27, 2014*, pages 171–180.

Omer Levy and Yoav Goldberg. 2014b. Neural Word Embedding as Implicit Matrix Factorization. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2177–2185.

Tal Linzen. 2016. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, RepEval@ACL 2016, Berlin, Germany, August 2016*, pages 13–18.

Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada.*, pages 5108–5118.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic Regularities in Continuous Space Word Representations. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, June 9-14, 2013, Westin Peachtree Plaza Hotel, Atlanta, Georgia, USA*, pages 746–751.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Distinguishing Antonyms and Synonyms in a Pattern-based Neural Network. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 76–85.

Xing Niu, Sudha Rao, and Marine Carpuat. 2018. Multi-Task Neural Models for Translating Between Styles Within and Across Languages. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1008–1021.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543.

Shrimai Prabhumoye, Yulia Tsvetkov, Ruslan Salakhutdinov, and Alan W. Black. 2018. Style Transfer Through Back-Translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 866–876.

Radu Soricut and Franz Josef Och. 2015. Unsupervised Morphology Induction Using Word Embeddings. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1627–1637.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958.

Seiya Tokui, Ryosuke Okuta, Takuya Akiba, Yusuke Niitani, Toru Ogawa, Shunta Saito, Shuji Suzuki, Kota Uenishi, Brian Vogel, and Hiroyuki Yamazaki Vincent. 2019. Chainer: A Deep Learning Framework for Accelerating the Research Cycle. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2019, Anchorage, AK, USA, August 4-8, 2019*, pages 2002–2011.

Luke Vilnis and Andrew McCallum. 2015. Word Representations via Gaussian Embedding. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning Gender-Neutral Word Embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4847–4853.