Intrinsic Evaluation of Grammatical Information within Word Embeddings

Daniel Edmiston
Department of Linguistics
University of Chicago
Chicago, IL, USA
danedmiston@uchicago.edu

Taeuk Kim
Department of Computer Science and Engineering
Seoul National University
Seoul, Korea
taeuk@europa.snu.ac.kr

Abstract

This work presents a proof-of-concept study for a framework of intrinsic evaluation of continuous embeddings as used in NLP tasks. This evaluation method compares the geometry of such embeddings with ground-truth embeddings in a linguistically-inspired, discrete feature space. Using model distillation (Hinton et al., 2015) as a means of extracting morphological information from models with no explicit morphological awareness (e.g. word-atomic models), we train multiple learner networks which do model morpheme composition so as to compare the amount of grammatical information different models capture. We use Korean affixes as a case-study, as they encode multiple types of linguistic information (phonological, syntactic, semantic, and pragmatic), and allow us to investigate specific types of linguistic generalizations models may or may not be sensitive to.

1 Introduction

While NLP systems built with neural network architectures have dominated the field in recent years, it is often lamented that their improved performance has come at the cost of understanding the models. Furthermore, recent work on natural language inference (McCoy et al., 2019) has cast doubt on the ability of such models to generalize linguistic patterns effectively. Particularly, carefully selected examples are shown to fool these systems, suggesting they learn heuristics for performing well on data sets rather than truly capturing linguistic information. As such, methods of probing the information within these models are of growing importance. This work proposes such a method by comparing the geometry of ground-truth, discrete-space embeddings against continuous embeddings learnt by neural networks. To do this, we extract morphological information from different embedding models via transparent model distillation (Tan et al., 2017) and examine the resulting morpheme embeddings for grammatical information. By investigating the embeddings directy this falls under the rubric of intrinsic evaluation. This complements extrinsic evaluation methods, where embeddings' ability to serve as input for classifiers on linguistic tasks is seen as a proxy for their linguistic content.

As is, intrinsic evaluation methods in the literature most often test for lexical semantics. For instance, the word analogy task tests similarities between pairs of words, e.g. *king* is to *queen* as *man* is to what? Here, since our ground-truth embeddings reflect grammatical properties, the similarities between them and the continuous representations can be taken to reflect *grammatical*, rather than *lexical* content in the continuous representations.

As an initial investigation, we compare the morphological information from multiple methods of embedding Korean words into continuous space, using three learner networks to distill morpheme representations for comparison. These are the StaffNet architecture of Edmiston and Stratos (2018), the morphological recursive neural network MRNN model of Luong et al. (2013), and a TreeLSTM (Tai et al., 2015) over morphological parses. By distilling into these networks, we extract explicit morpheme representations. We focus on the *affix* repre-

Pacific Asia Conference on Language, Information and Computation (PACLIC 33), pages 395-404, Hakodate, Japan, September 13-15, 2019

sentations, as they house the grammatical information we are testing for.

Korean affixes are the choice for this pilot study for multiple reasons. (i) Korean is an agglutinative language, meaning there is (largely) a one-to-one correspondence between affixes and meanings. (ii) The morphology is highly regular, which facilitates high-fidelity morphological parsing. (iii) Korean affixes display at least four distinct types of linguistic information. **Phonolgical**: Korean exhibits phonologically-driven allomorphy. **Syntactic:** Korean affixes contain syntactic information as they attach to different syntactic units. **Semantic:** affixes perform different semantic functions, e.g. logical operators vs. focus markers. **Pragmatic:** Certain affixes are indicative of formal language, and others display honorific features. This allows us to run focused experiments which probe what type of information different models are sensitive to.

Having distilled affix embeddings from various ground-truth models, we run our evaluation task by comparing the distilled representations of different models with ground-truth morpheme representations embedded in a discrete, linguistically-inspired feature space, and show that indeed neural models are picking up on at least some forms of *grammatical* meaning. Results suggest that semantic relationships are more difficult to capture than syntactic, and models appear insensitive to phonological/pragmatic information.

Contributions: **(i)** We introduce a new linguistically-inspired intrinsic evaluation task which probes for *grammatical* meaning, rather than *lexical* meaning. **(ii)** We show results that different types of linguistic information are captured to differing degrees, and may be of a differing nature from one another. **(iii)** To the authors' knowledge, this is the first application of transparent model distillation for interpretation in an NLP setting. **(iv)** We focus on an under-studied language in NLP, which also happens to be typologically far removed from those usually studied in the literature.

2 Related Work

This work falls in the context of the emerging literature on the interpretability of neural network models using model distillation, and on the interpretability of linguistic representations learnt for NLP tasks.

In the broader context of interpreting the behavior of neural network models, model distillation has emerged as a viable method. Tan et al. (2018b) use so-called transparent model distillation to audit black-box risk scoring models. By distilling a black-box model into a transparent learner model, and comparing this learner model with a non-distilled transparent model trained on ground-truth data, they are able to gain insights into black-box models. Likewise, Zhang et al. (2018) use model distillation (which they call *knowledge distillation*) to extract human-interpretable features from the middle layers of convolutional networks trained on computer vision tasks. While similar, our method differs slightly from these approaches in that we use model distillation to induce representations for linguistic units which would otherwise be unavailable (affixes).

As for the interpretability of linguistic representations learnt for NLP tasks, extrinsic evaluation methods have focused both on morphological information (e.g. (Belinkov et al., 2017)) and syntactic information (e.g. (Adi et al., 2016)). As these are extrinsic methods, the task consists of learning representations, and then training classifiers on carefully designed supervised learning tasks using these representations as input. The tasks are meant to probe for linguistic properties (e.g. negation (Ettinger et al., 2016)), and performance on these tasks can be interpreted as a proxy for the amount of linguistic information contained in the original representations. Work in the intrinsic evaluation domain has largely focused on testing lexical semantics, as by comparing relations between e.g. countries and capitals (Mikolov et al., 2013a). By comparison, here we intrinsically test for what we call *grammatical information*, to be made specific in Section 3.3.

3 Background

3.1 Model Distillation

Model distillation (Hinton et al., 2015) is the technique of training one neural network (the *learner network*) to approximate the output of another (the *ground-truth network*). While the technique was originally designed to train relatively lightweight networks to approximate the outputs of larger, more cumbersome networks or ensembles, model distillation has recently been used to facilitate the interpretation of so-called "black-box" neural network models (Tan et al., 2017, 2018a,b; Zhang

et al., 2018).

The idea behind using model distillation for the interpretation of word-embeddings in this study is to train a learner network amenable to morphological interpretation to approximate a model which is otherwise not straight-forward to analyze morphologically. In this case, this includes word-atomic word-embedding models (such as *Word2Vec*), as well as word-embedding models which compose embeddings from sub-word constituents (e.g. syllable-embedding models). Assuming successful distillation, the morphological representations of the learner network can be seen as a proxy for the morphological information captured by the original ground-truth network. Here, model distillation of a ground-truth model into a learner network proceeds by iterating over the corpus the ground-truth model was originally trained on. For each word, the ground-truth embedding $\mathbf{y}$ is calculated, as is the learner network's embedding $\hat{\mathbf{y}}$. Training proceeds by optimizing the learner network's parameters to minimize the squared distance between $\mathbf{y}$ and $\hat{\mathbf{y}}$.

3.2 Learner Networks

3.2.1 StaffNet

The first learner network is the StaffNet architecture. Introduced in Edmiston and Stratos (2018), StaffNet ('Stem-Affix Net') is a dynamic neural network architecture designed to compose morpheme representations into full word-embeddings in a linguistically plausible way. The defining feature of StaffNet is its distinguishing of morphemes into stems and affixes, treating the former as vectors and the latter as functions over vectors. Here, we treat affixes as linear transformations and represent them as matrices in $\mathbb{R}^{d \times d}$.

StaffNet is a dynamic architecture, whose composition of a word-embedding varies with the morphological parse of the word. It composes a word-embedding in a three-step process. First, a word is decomposed into its constituent stems and affixes.[2] Second, the (potentially compound) stem representation is calculated as the convex combination of the outputs of a BiLSTM into which the stems are fed. Third, any affixes are iteratively applied to the compound stem representation—the convex combination from the previous step. Figure 1 illustrates this for the word *cheese-burger*.PL.NOM, or *cheeseburgers* marked with nominative case.

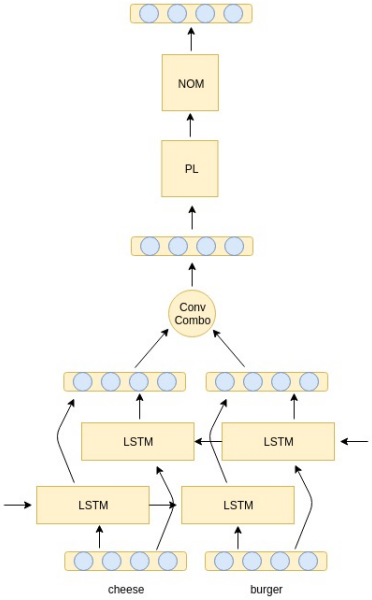

Figure 1: StaffNet architecture showing the dynamic composition of *cheese-burger*.PL.NOM.

3.2.2 MRNN

The second learner network is the morphological RNN model of Luong et al. (2013), which constructs a word's embedding from constituent morpheme embeddings by means of a recursive neural network over the binary tree of the word's morphological parse. Parent nodes in the network are functions of their children nodes, and are calculated as $p = f(W[c_1; c_2] + b)$. That is, they are the result of a non-linearity (here *tanh*) applied to the output of an affine transformation over the concatenated children embeddings. An example is as in Figure 2.

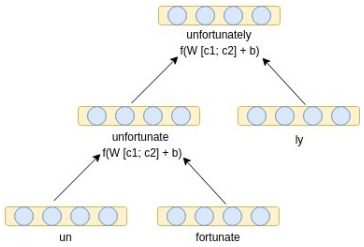

Figure 2: Morphological RNN architecture showing the composition of *un-fortunate-ly*

[2] All morphological parsing done with the *Komoran* POS-tagger available with the KoNLPy package. http://konlpy.org/en/latest/

3.2.3 TREELSTM

The final learner network is the N-ary Tree-LSTM of Tai et al. (2015). The transition equations follow the original paper, where k indexes the kth child of parent node p.

$$i_p = \sigma(W^{(i)}x_p + \sum_{l=1}^{N} U_l^{(i)} h_{pl} + b^{(i)})$$

$$f_{pk} = \sigma(W^{(f)}x_p + \sum_{l=1}^{N} U_{kl}^{(f)} h_{pl} + b^{(f)})$$

$$o_p = \sigma(W^{(o)}x_p + \sum_{l=1}^{N} U_l^{(o)} h_{pl} + b^{(o)})$$

$$u_p = tanh(W^{(u)}x_p + \sum_{l=1}^{N} U_l^{(u)} h_{pl} + b^{(u)})$$

$$c_p = i_p \odot u_p + \sum_{l=1}^{N} f_{pl} \odot c_{pl}$$

$$h_p = o_p \odot tanh(c_p)$$

Here, N is restricted to 2, as this model also operates over binary morphological parses.

3.3 Korean Affixes as a Case Study

As mentioned above, Korean affixes exhibit at least four different types of information: phonological, syntactic, semantic, and pragmatic. Phonological information is shown through phonologically-driven allomorphy. For example, the accusative marker -을 '*-eul*' attaches to nominals ending in a coda, while its allomorph -를 '*-leul*' attaches to nominals ending in vowels. A familiar analogy from English would be the choice between '*an*' and '*a*', e.g. '*an animal*' vs. '*a dog*.' Syntactic information is shown through place of attachment (Cho and Sells, 1995). There are many semantic dimensions along which affixes vary, e.g. some contribute focus semantics, others serve as logical operators. Multiple pragmatic dimensions of meaning are also evident in Korean affixation. Some affixes are reserved for written usage, others indicate the relationship between speaker and hearer, such as honorifics.

Having formalized the feature set of 107 Korean affixes, we embed each affix into the binary feature space $\{0, 1\}^n$, the dimensions of which are interpreted as linguistic features (e.g. [+CODA] vs. [-CODA]) with 1 indicating the presence of that feature, 0 otherwise. These embeddings serve as 'ground-truth' representations of Korean affixes, and geometrically can be interpreted as the corners of an $n-$dimensional hypercube. We define distance in this binary feature space with Hamming distance, where $d_H(x, y)$ is the number of dimensions along which x and y differ.

4 Methodology

4.1 Ground-truth Models and Distillation

We propose a method of intrinsically evaluating the extent to which different word-embedding models capture the meaning of affixes in Korean, and therefore how well they capture phonological, syntactic, semantic, and pragmatic information. The models we distill and compare are as follows. **Character-level:** *fastText* (Bojanowski et al., 2017),[3] Naver's *kor2vec* model (based on Kim et al. (2016))[4], **Syllable:** The model of Choi et al. (2017), **Word:** *Skip-gram* and *CBOW* models of Mikolov et al. (2013a), and *GloVe* Pennington et al. (2014).

All models were trained on a Korean Wikidump with vocabulary size limited to 10,000, and were trained to produce embeddings in 300 dimensions, as suggested by Choi et al. (2016). Other hyperparameters followed the suggestions of the original publications where applicable. Each model was then distilled into each of the learner architectures, embedding affixes as vectors in $\mathbb{R}^{300}$ or in STAFFNET's case, as matrices in $\mathbb{R}^{300\times300}$.

4.2 Comparing Affix Representations for Intrinsic Evaluation

One standard method of performing intrinsic evaluation of high-dimensional word-embeddings is analogy tests (Mikolov et al., 2013b). In such a test, word-sets are assembled of the form *a:b, c:d*, where d is withheld. Given an embedding model, an estimation of withheld d is given by $\hat{d} = \vec{b} - \vec{a} + \vec{c}$. An example is counted as correct if the vector $\vec{d}$ is the closest—via cosine similarity—in the embedding space to the model's calculated $\hat{d}$, and incorrect otherwise.

In the case where affixes are represented by vectors, here too we make use of cosine similarity. Where affixes are represented as matrices, cosine similarity is not applicable, and instead we compare them via subspace similarity, as described in Algorithm 1 (Mu et al., 2017).

$\mathbf{M}_{aff1}$ and $\mathbf{M}_{aff2}$ represent the matrix representations of the affixes to be compared. N is an inte-

[3] https://fasttext.cc/docs/en/pretrained-vectors.html

[4] https://github.com/naver/kor2vec

Algorithm 1 Subspace Similarity

Input: $\mathbf{M}_{aff1}$, $\mathbf{M}_{aff2}$, N
Output: $score \in [0, 1]$

$X \leftarrow [pc(\mathbf{M}_{aff1})_1; ...; pc(\mathbf{M}_{aff1})_N]$
$Y \leftarrow [pc(\mathbf{M}_{aff2})_1; ...; pc(\mathbf{M}_{aff2})_N]$
$Z \leftarrow X^\top Y$
$score \leftarrow \sqrt{\sum_{t=1}^{N} \sigma_t^2}/N$
return(*score*)

ger signifying the number of principal directions to use. σ_t represents the t^{th} singular value of Z. The algorithm proceeds by performing PCA on the matrix inputs, and stacking the first N principal directions into matrices in $\mathbb{R}^{d \times N}$. The sum of squares of the N singular values of the product of these matrices, divided by N, results in a similarity score in $[0, 1]$. The geometric intuition behind this metric is that similar affixes should map stems to similar subspaces.

To evaluate the distilled representations, we compare the continuous embeddings against the discrete embeddings in the following way. Given the set of all affixes $\mathcal{A}$, consider the subset $\mathcal{Y}_{\text{Aff}} = \underset{x \in \mathcal{A}, x \neq \text{Aff}}{argmin}\, d_H(x, \text{Aff})$, the set of closest affixes to any given affix in the ground truth discrete feature space, where $|\mathcal{Y}_{\text{Aff}}| = k$. In the continuous space, we define the $k-$closest affixes to any given affix with $\hat{\mathcal{Y}}_{\text{Aff}} = \underset{x \in \mathcal{A}, x \neq \text{Aff}}{k\text{-}argmax}\, sim(x, \text{Aff})$, where $sim(x, y)$ is cosine similarity or subspace similarity, depending on architecture. By design $\mathcal{Y}_{\text{Aff}}$ and $\hat{\mathcal{Y}}_{\text{Aff}}$ are of the same cardinality.

Given $\mathcal{Y}_{\text{Aff}}$ and $\hat{\mathcal{Y}}_{\text{Aff}}$, we define two scores. The first is the percentage of overlap—the percentage of the k-closest affixes in the continuous representation which are k-closest with regard to the ground-truth embeddings.[5] The second measures error; $avg(\{d_H(x, \text{Aff}) \mid x \in \hat{\mathcal{Y}}_{\text{Aff}}\}) - \underset{x \in \mathcal{A}, x \neq \text{Aff}}{min}\, d_H(x, \text{Aff})$; that is, the true error with regard to an affix, as calculated by the average hamming distance between Aff and x for $x \in \hat{\mathcal{Y}}_{\text{Aff}}$, minus the minimum possible error. We label this penalty the *Hamming offset*. Note that percentage of overlap and the Hamming offset provide two graded measures of success with regard to an affix, unlike all-or-nothing diagnostics like e.g. analogy tests.

[5] Since it is always the case that $|\mathcal{Y}_{\text{Aff}}| = |\hat{\mathcal{Y}}_{\text{Aff}}|$ this amounts to the F_1 measure on cluster analysis.

4.3 Data Sets

This study makes use of two data sets. The first is a Korean WikiDump used to train the original models, and also used as the corpus for model distillation. The second data set was hand-constructed for this study, and constitutes the ground-truth representations used in the experiments below. It is the embedding of 107 Korean affixes into a discrete, binary feature space, consisting of 62 dimensions. These 62 dimensions correspond to linguistic features along which Korean affixes vary, and include features such as the aforementioned [+CODA], and [+HONORIFIC]. We divided the linguistic features into four distinct feature subsets, one for each of phonological, syntactic, semantic, and pragmatic features, so as to run tests on feature subsets. Over the affixes, there are five distinct phonological configurations, eight distinct syntactic configurations, 55 distinct semantic configurations, and five distinct pragmatic configurations.

5 Experiments

5.1 Verifying Distillations

As we are testing models based on their distilled representations rather than their original representations, the first question to ask is whether the distilled embeddings are a faithful recreation of the models they are meant to emulate.

MODEL	STAFFNET	MRNN	TREELSTM
KOR2VEC	0.971	0.840	0.952
FASTTEXT	0.949	0.824	0.946
SYLLABLE	0.967	0.869	0.979
W2V-SG	0.953	0.755	0.920
W2V-CBOW	0.941	0.786	0.878
GLOVE	0.935	0.690	0.875

Table 1: Average cosine similarity between ground-truth embeddings and distilled embeddings over 10k vocabulary.

The results in Table 1 show that each of the models was able to reproduce the original embeddings to a very high degree of accuracy. This is especially true given that the overwhelming majority of volume in $\mathbb{R}^{300}$ is orthogonal to any given point. We take this to mean that the models have been successfully distilled, and our distilled representations can serve as faithful representatives of their ground-truth models.

5.2 Closest Affix

This section tabulates the scores for each of the models—as well as a random baseline—on the intrinsic task described in Section 4.2. We test each model distilled into each architecture, and for the StaffNet architecture we also test for different values of subspace rank N. We chose the rank of the subspace to be the minimum number of principal components which accounted for 25%, 50%, and 75% of the average variance of the affixes for each model. As mentioned above, we calculate the average percent of overlap between the k-closest in the continuous and discrete spaces, and also calculate the average Hamming offset. The results are in Table 2, where the figures represent the scores for each model averaged over performance on all 107 affixes.

As can be seen in the results, no model was able to achieve a high level of accuracy on the task, but all models significantly outperform the random baseline, and the MRNN and TreeLSTM models fare better than the StaffNet distillations with regard to both average percentage of overlap and average hamming offset.

5.3 Feature Subsets

Given that Korean affixes contain linguistic information of different sorts, we can perform a variant of our experiment from above using only subsets of features. For example, given the feature-makeup described above, affixes fall into one of five phonological configurations, which we can describe as +CODA, -CODA, +LOW, -LOW, or NONE. For these experiments, similar to before define $\mathcal{Y}_{\text{Aff}} = \underset{x \in \mathcal{A}, x \neq \text{Aff}}{argmin}\, d_H(phon(x), phon(\text{Aff}))$, or the set of affixes which are closest to a certain affix in the discrete space considering only phonological features. We then define $\hat{\mathcal{Y}}_{\text{Aff}}$ as before, $\hat{\mathcal{Y}}_{\text{Aff}} = \underset{x \in \mathcal{A}, x \neq \text{Aff}}{argmax}\, sim(x, \text{Aff})$. Percentage of overlap and Hamming offset are as before. Results can be interpreted as: for the k-closest affixes in the continuous space, how many behave the same with regard to, e.g. phonological features? This should help identify what type of linguistic information models are sensitive to.

The results for the phonological, syntactic, semantic, and pragmatic subset tests are in Tables 3-6.

6 Discussion

Before discussing individual test results, it is noteworthy that the scores of the tables vary significantly from one another, and the random baseline shows particularly strong results for certain subsets, particularly the pragmatic subset. This is the result of fluctuations in the average k (i.e. cluster size) and average *Hamming offset* for each subset. Table 7 lists these figuers. Dividing the average k by the total number of affixes $|\mathcal{A}|$ gives the expected random score for percentage of overlap. For each subset, the random baselines roughly reflect these figures. For interpreting the results, performance relative to the random baseline is what is important, not the percentage figure or Hamming offset figures themselves.

Examining Table 2, the MRNN and TreeLSTM models outperformed the StaffNet models by a large margin. A plausible hypothesis would be to attribute this difference to the lack of non-linearity in the StaffNet-derived affixes. While StaffNet does derive stem representations via back-propagation through a BiLSTM, the affix representations always apply post-non-linearity during forward propagation, and as such are unable to learn any potentially non-linear relationships.

Regarding Table 3, while nearly all models outperformed the random baseline, none did so significantly. It is furthermore surprising that character and syllable-level models did not significantly outperform word-atomic models, to which phonological information of the kind driving allomorphy in Korean should be unavailable. This may suggest that the models examined here are not sensitive to phonological information in any significant way.

The syntactic results in Table 4 show all models outperforming the random baseline, suggesting it is possible for models to capture syntactic information. Furthermore, the distilling architectures performed relatively similarly, with a StaffNet distillation achieving the highest score. This suggests that non-linearity may not be necessary to capture syntactic relations.

For the semantic subset results in Table 5, while StaffNet distillations performed similar to the random baseline, the models deriving affix representations via non-linearity showed relatively strong results. This suggests two things. (i) It is possible for modern neural network models to capture *grammatical* meaning of a semantic nature,

Architecture	StAffNet-25% variance		StAffNet-50% variance		StAffNet-75% variance		MRNN		TreeLSTM	
Model	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$
kor2vec	12%	3.35	11%	3.49	9%	3.39	16%	2.82	**19%**	2.62
fastText	13%	3.23	12%	3.22	9%	3.31	18%	**2.59**	16%	2.63
Syllable	13%	3.18	11%	3.42	8%	3.36	17%	2.88	17%	3.29
w2v-sg	12%	3.17	8%	3.57	7%	3.40	18%	**2.59**	15%	3.01
w2v-cbow	13%	3.19	9%	3.46	7%	3.56	18%	2.63	16%	2.86
GloVe	10%	3.30	10%	3.34	7%	3.47	16%	2.80	11%	3.13
Random	2%	4.23	3%	4.38	3%	4.47	2%	4.32	1%	4.35

Table 2: Percent correct and average Hamming Distance for models mapping to subspaces of different sizes. Best scores in bold, worst scores in red.

Architecture	StAffNet-25% variance		StAffNet-50% variance		StAffNet-75% variance		MRNN		TreeLSTM	
Model	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$
kor2vec	31%	0.89	30%	0.91	30%	0.91	**32%**	0.91	**32%**	0.90
fastText	**32%**	0.87	31%	0.89	31%	0.89	29%	0.95	29%	0.95
Syllable	**32%**	0.86	31%	0.90	31%	0.90	31%	0.88	29%	0.96
w2v-sg	**32%**	**0.85**	30%	0.89	30%	0.89	29%	0.95	28%	0.96
w2v-cbow	31%	0.88	30%	0.91	30%	0.90	31%	0.92	31%	0.93
GloVe	32%	0.87	29%	0.91	29%	0.90	29%	0.95	28%	0.94
Random	26%	0.97	26%	0.99	27%	0.97	28%	0.96	29%	0.96

Table 3: Percent correct and Average Hamming offset: Restricting to **Phonological** features.

Architecture	StAffNet-25% variance		StAffNet-50% variance		StAffNet-75% variance		MRNN		TreeLSTM	
Model	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$
kor2vec	36%	1.27	38%	1.24	39%	1.22	37%	1.27	39%	1.23
fastText	37%	1.27	40%	1.21	**41%**	**1.19**	38%	1.23	37%	1.25
Syllable	35%	1.30	37%	1.26	39%	1.22	35%	1.29	38%	1.25
w2v-sg	37%	1.26	37%	1.26	39%	1.21	37%	1.27	36%	1.29
w2v-cbow	36%	1.29	38%	1.24	38%	1.24	38%	1.23	39%	1.23
GloVe	36%	1.28	37%	1.26	38%	1.24	35%	1.30	34%	1.32
Random	28%	1.43	28%	1.45	29%	1.43	29%	1.42	30%	1.41

Table 4: Percent correct and Average Hamming offset: Restricting to **Syntacitc** features.

Architecture	StAffNet-25% variance		StAffNet-50% variance		StAffNet-75% variance		MRNN		TreeLSTM	
Model	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$
kor2vec	6%	2.54	5%	2.52	5%	2.53	23%	1.85	25%	1.80
fastText	6%	2.39	5%	2.40	5%	2.43	**33%**	**1.65**	24%	1.72
Syllable	5%	2.47	6%	2.51	5%	2.64	18%	1.96	19%	2.04
w2v-sg	5%	2.48	5%	2.46	5%	2.48	26%	1.77	29%	1.78
w2v-cbow	6%	2.56	5%	2.54	5%	2.6	24%	1.89	19%	2.04
GloVe	5%	2.49	5%	2.49	4%	2.57	24%	1.89	23%	1.92
Random	3%	2.79	3%	2.70	2%	2.71	2%	2.72	2%	2.67

Table 5: Percent correct and Average Hamming offset: Restricting to **Semantic** features.

Architecture	StaffNet-25% variance		StaffNet-50% variance		StaffNet-75% variance		MRNN		TreeLSTM	
Model	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$	%Overlap	Offset$_{d_H}$
kor2vec	69%	0.33	69%	0.33	70%	0.31	71%	0.31	**72%**	**0.29**
fastText	69%	0.32	69%	0.33	70%	0.32	71%	0.31	69%	0.32
Syllable	71%	0.31	69%	0.33	70%	0.33	70%	0.32	70%	0.31
w2v-sg	70%	0.33	69%	0.34	70%	0.32	70%	0.32	69%	0.33
w2v-cbow	68%	0.33	68%	0.34	69%	0.34	70%	0.32	70%	0.32
GloVe	68%	0.34	68%	0.34	68%	0.34	70%	0.32	69%	0.32
Random	71%	0.31	70%	0.34	69%	0.33	70%	0.31	69%	0.34

Table 6: Percent correct and Average Hamming offset: Restricting to **Pragmatic** features.

Subset	Avg. k	Avg. k / $\lvert\mathcal{A}\rvert$	Avg. d_H
All	2.21	0.02	5.56
Phon	29.93	0.28	0.96
Syn	30.92	0.29	1.42
Sem	3.25	0.03	3.00
Prag	73.91	0.69	0.32

Table 7: Average size of nearest k affixes for each feature subset.

and (ii) non-linearity is required to capture these meanings.

For the pragmatic results in Table 6, all models perform virtually indistinguishably from the random baseline. This suggests that the word-embedding models examined here are not sensitive to pragmatic features, at least not when distilled into another model.[6]

As this is a proof-of-concept study, the principal takeaway is that this method of intrinsic evaluation is possible, and reveals interesting characteristics of neural representations. Specifically, the underlying geometry of ground-truth discrete embeddings are at least to some extent being captured in the geometry of the continuous representations learnt by different neural-network models. Furthermore, the results in this study show that not only are neural network models sensitive to grammatical information—as distinct from lexical information—they are sensitive to different types of grammatical information to differing degrees. Syntactic information can apparently be captured by linear relations, while semantic information requires non-linearities. This result is perhaps not surprising, as even in the theoretical linguistics literature semantic analyses often require more complex algebraic structures built on top of syntactic parses. Finally, neural models seem insensitive to phonological and pragmatic information, with all models here performing virtually the same as the random baseline on these sub-tests.

Finally, in addition to the test results themselves, the fact that any models significantly outperformed the random baseline shows that transparent model distillation can serve as a viable means of extracting sub-atomic information from otherwise atomic representations.

7 Conclusion

This study has been a proof-of-concept for an intrinsic evaluation method which probes grammatical, rather than lexical information in word-embeddings. To do this, we studied the case of Korean affixes, which display multiple types of grammatical information. In order to derive affix representations, we used Transparent Model Distillation to extract morpheme representations where they otherwise did not exist. Our study has shown that neural network models are sensitive to grammatical information, and that the geometry of the continuous representations learnt by neural network models reflects to some degree the geometry of ground-truth discrete embeddings.

We put forward such a process as a framework for the fine-grained intrinsic analysis of the high-dimensional continuous embeddings which are often used to help solve natural language processing tasks.

[6] Though it is of note that all models were trained on text which is ostensibly academic in character (a WikiDump file), and which therefore is almost devoid of pragmatically-marked language like honorifics.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2016. Fine-Grained Analysis of Sentence Embeddings using Auxiliary Prediction Tasts. ArXiv preprint arXiv:1608.04207.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do Neural Machine Translation Models Learn about Morphology. ArXiv preprint arXiv:1704.03471.

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.

Young-Mee Yu Cho and Peter Sells. 1995. A Lexical Account of Inflectional Suffixes in Korean. *Journal of East Asian Linguistics*, 4(2):119–174.

Sanghyuk Choi, Taeuk Kim, Jinseok Seol, and Sanggoo Lee. 2017. A Syllable-based Technique for Word Embeddings of Korean Words. In *Proceedings of the First Workshop on Subword and Character Level Models in NLP*, pages 36–40.

Sanghyuk Choi, Jinseok Seol, and Sang-goo Lee. 2016. On Word Embedding Models and Parameters Optimized for Korean. In *Proceedings of the 28th Annual Conference on Human and Cognitive Language Technology (In Korean)*.

Daniel Edmiston and Karl Stratos. 2018. Compositional Morpheme Embeddings with Affixes as Functions and Stems as Arguments. In *Proceedings of the Workshop on the Relevance of Linguistic Structure in Neural Architectures for NLP*, pages 1–5.

Allyson Ettinger, Ahmed Elgohary, and Philip Resnik. 2016. Probing for Semantic Evidence of Composition by Means of Simple Classification Tasks. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, pages 134–139.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the Knowledge in a Neural Network. In *NIPS Deep Learning and Representation Learning Workshop*.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander Rush. 2016. Character-Aware Neural Language Models. In *Proceedings of AAAI*, pages 2741–2749.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better Word Representations with Recursive Neural Networks for Morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. ArXiv preprint arXiv:1902.01007.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeff Dean. 2013a. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*.

Tomas Mikolov, Wen-Tau Yih, and Geoffrey Zweig. 2013b. Linguistic Regularities in Continuous Space Word Representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 764–751.

Jiaqi Mu, Suma Bhat, and Pramod Viswanath. 2017. Representing Sentences as Low-rank Subspaces. ArXiv preprint arXiv:1704.05358.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Kai Sheng Tai, Richard Socher, and Christopher Manning. 2015. Improved Semantic Representations from Tree-Structured Long Short-Term Memory Networks. ArXiv preprint arXiv:1503.00075.

Sarah Tan, Rich Caruana, Giles Hooker, Paul Koch, and Albert Gordo. 2018a. Learning Global Additive Explanations for Neural Nets Using Model Distillation. ArXiv preprint arXiv:1801.08640.

Sarah Tan, Rich Caruana, Giles Hooker, and Yin Lou. 2017. Detecting Bias in Black-box Models Using Transparent Model Distillation. ArXiv preprint arXiv:1710.06169.

Sarah Tan, Rich Caruana, Giles Hooker, and Yin Lou. 2018b. Distill-and-Compare: Auditing Black-Box Models Using Transparent Model Distillation. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 303–310.

Quanshi Zhang, Yu Yang, Yuchen Liu, Ying-Nian Wu, and Zhu Song-Chun. 2018. Unsupervised Learning of Neural Networks to Explain Neural Networks. ArXiv preprint arXiv:1805.07468.

Appendix A: Feature subsets

Table 8 displays the feature values which make up the dimensions of the discrete space. The combination of all feature values from the subsets comprise the full discrete space, upon which the experiment in Table 2 was run.

SUBSET	FEATURE VALUES
PHONOLOGICAL FEATURES	[+CODA], [-CODA], [+LOW], [-LOW]
SYNTACTIC FEATURES	[+POSTPOSITION], [+CONJUNCTIVE], [+X-LIMITER], [+Z-LIMITER], [+V1], [+V2], [+V3], [+V4]
SEMANTIC FEATURES	[+DECLARATIVE], [+NOMINATIVE], [+TOPIC], [+LOCATIVE], [+DIRECTIVE], [+GENATIVE], [+ACCUSATIVE], [+PAST], [+CONJUNCTION], [+ADVERBIAL], [+ALSO], [+TAG], [+NOMINALIZER], [+FROM], [+CONDITIONAL], [+LIKE], [+ESSIVE], [+COMPARATOR], [+COMPLEMENTIZER], [+RELATIVIZER], [+PAST], [+RETROSPECTIVE], [+FUTURE], [+PLURAL], [+PRESENT], [+BECAUSE], [+COPULA], [+QUOTATIVE], [+ABLATIVE], [+INTENT], [+MUST], [+RESULT], [-ANIMATE], [+ANIMATE], [+INSTRUMENTAL], [+INTERROGATIVE], [+DATIVE], [+GOAL], [+EVEN], [+COHORTATIVE], [+ONLY], [+DISJUNCTIVE], [+EACH], [+DURATION]
PRAGMATIC FEATURES	[+FORMAL], [-FORMAL], [+HONORIFIC], [+FAMILIAR]

Table 8: Description of feature subsets.