Investigating Word-Class Distributions in Word Vector Spaces

Ryohei Sasano
Graduate School of Informatics
Nagoya University
sasano@i.nagoya-u.ac.jp

Anna Korhonen
Language Technology Lab
University of Cambridge
alk23@cam.ac.uk

Abstract

This paper presents an investigation on the distribution of word vectors belonging to a certain word class in a pre-trained word vector space. To this end, we made several assumptions about the distribution, modeled the distribution accordingly, and validated each assumption by comparing the goodness of each model. Specifically, we considered two types of word classes – the semantic class of direct objects of a verb and the semantic class in a thesaurus – and tried to build models that properly estimate how likely it is that a word in the vector space is a member of a given word class. Our results on selectional preference and WordNet datasets show that the centroid-based model will fail to achieve good enough performance, the geometry of the distribution and the existence of subgroups will have limited impact, and also the negative instances need to be considered for adequate modeling of the distribution. We further investigated the relationship between the scores calculated by each model and the degree of membership and found that discriminative learning-based models are best in finding the boundaries of a class, while models based on the offset between positive and negative instances perform best in determining the degree of membership.

1 Introduction

Several studies have been successful in representing the meaning of a word with a vector in a continuous vector space (e.g., Mikolov et al. 2013a; Pennington et al. 2014). These representations are useful for a range of natural language processing (NLP) tasks. The interpretation and geometry of the word embeddings have also attracted attention (e.g., Kim and de Marneffe 2013; Mimno and Thompson 2017). However, little attention has been paid to the distribution of words belonging to a certain word class in a word vector space, though empirical analysis of such a distribution provides a better understanding of word vector spaces and insight into algorithmic choices for several NLP tasks, including selectional preference acquisition and entity set expansion.

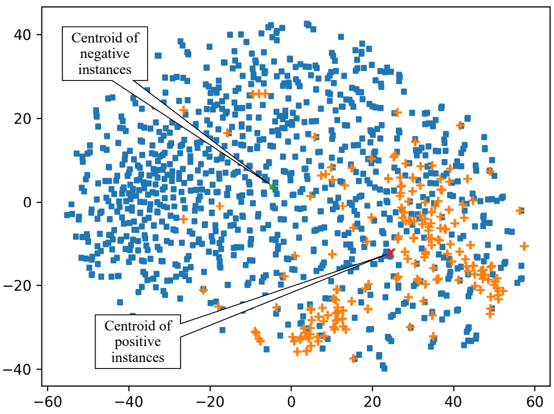


Figure 1: 2D t-SNE projection of GloVe vectors. The 200 plus symbols (+) represent the word vectors that can be a direct object of the verb *play* (positive instances) and the 1000 squares (■) represent other word vectors (negative instances).

Figure 1 shows a 2D projection of word embeddings. We extracted 200 words that can be a direct object of the verb *play* (positive instances) and 1000 other words (negative instances) and projected their GloVe vectors (Pennington et al., 2014) into two dimensions using t-distributed Stochastic Neighbor Embedding (t-SNE) (van der Maaten and Hinton, 2008). The plus symbols (+) represent the positive instances, and the squares (■) represent the negative instances. This figure shows that the positive instances tend to be densely distributed around their centroid but they are not evenly distributed near the centroid in the 2D spaces. In this study, we aimed to understand how these positive instances are distributed in the pre-trained word vector spaces built by three representative general-purpose models: CBOW, skip-gram (Mikolov et al.,

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 3657–3666
July 5 - 10, 2020. ©2020 Association for Computational Linguistics

2013a), and GloVe.

More specifically, we attempted to determine the following: whether or not a simple centroid-based approach can provide a reasonably good model, whether or not considering the geometry of the distribution and the existence of subgroups is useful for modeling the distribution, and whether or not considering the negative instances is essential to achieve adequate modeling. To this end, we first tackled properly modeling the vector distribution to distinguish a possible member of a word class from others when a subset of the class members is given. Note that although various approaches have been proposed to improve word vectors by taking knowledge related to word classes into account (Faruqui et al., 2015; Rothe and Schütze, 2015; Mrkšić et al., 2017), we explored ways to model the distribution of word vectors rather than attempting to improve the word vectors themselves.

We started with a centroid-based model, which is a simple but widely used way of representing a set of word vectors (e.g., Baroni et al. 2014; Woodsend and Lapata 2015) and assumes that how likely a word in the vector space is a member of a word class is proportional to the proximity to the centroid vectors of the class members. We then explored models that take the geometry of the distribution and the existence of subgroups into account. Here, we made two assumptions: vectors of words belonging to a certain word class are distributed with different variances depending on the direction, and most word sets will consist of several subgroups. We then explored the models that also consider negative instances. We assumed that the vectors of the words that do not belong to the target word class can be essential clues to distinguish a possible member of a word class from others. Specifically, we explored a model based on the offset between positive and negative instances and discriminative learning-based models to investigate the impact of negative instances.

Furthermore, we investigated the relationship between the scores calculated by each model and the degree of membership using the Rosch (1975) dataset. The dataset contains typicality ratings for some instances of a category. Through experiments, we found that discriminative learning-based models perform better at distinguishing a possible member of a word class from others, while the offset-based model achieves higher correlations with the degree of membership.

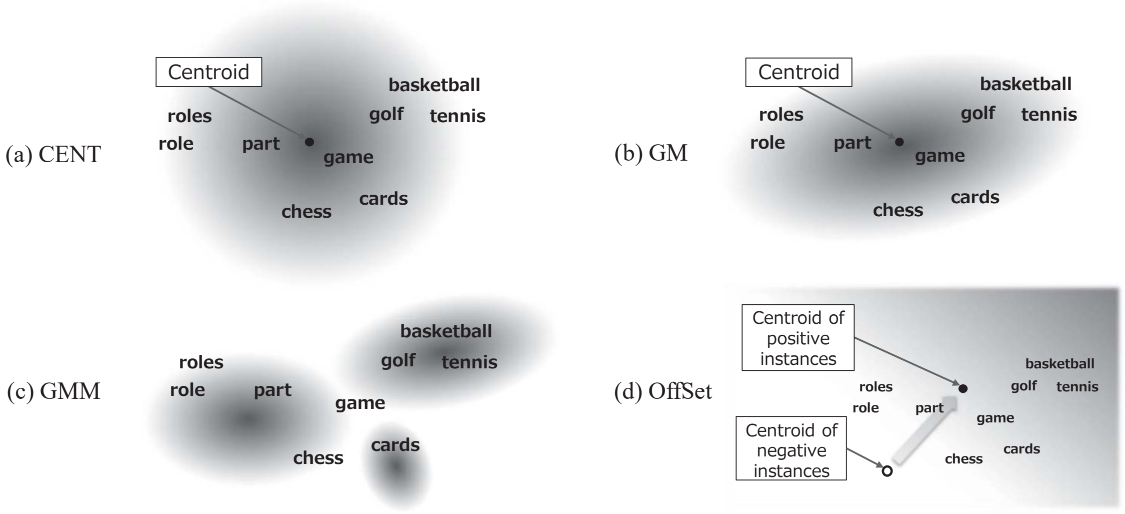
2 Related Work

The interpretation and geometry of word embeddings have attracted attention. Mimno and Thompson (2017) reported that vector positions trained with skip-gram negative sampling (SGNS) do not span the possible space uniformly but occupy a narrow cone instead. Mikolov et al. (2013b) showed that constant vector offsets of word pairs can represent linguistic regularities. Kim and de Marneffe (2013) demonstrated that vector offsets can be used to derive a scalar relationship amongst adjectives. Yaghoobzadeh and Schütze (2016) performed an analysis of subspaces in word embedding. These analyses suggest that a certain direction or subspace in the word vector space represents an aspect of the words and the possibility that a word class is distributed with different variances depending on the direction in the vector space.

While we investigated ways to model the distribution of a set of words in pre-trained word vector spaces to validate several assumptions about the distribution, various approaches have been proposed to improve word embeddings by considering knowledge related to word classes into account. For example, Faruqui et al. (2015) proposed a method of refining vector representations using relational information from semantic lexicons by encouraging linked words to have similar vector representations. Mrkšić et al. (2017) proposed an algorithm for improving the semantic quality of word vectors by injecting constraints extracted from lexical resources. Glavaš and Vulić (2018) use the linguistic constraints as training examples to learn an explicit specialization function with deep neural network architecture.

There are also several studies that expand the method for acquiring a word vector to consider the uncertainty of a word meaning via Gaussian models (Vilnis and McCallum, 2015; Athiwaratkun and Wilson, 2017) and word polysemy by introducing several vectors for each word (Chen et al., 2014; Neelakantan et al., 2014; Tian et al., 2014; Athiwaratkun et al., 2018). In this study, we only considered a vector for representing each word, but inspired by these studies, we explored models that can consider the geometry of the distribution and the existence of subgroups.

The problem we tackled is similar to a selectional preference acquisition task. There have been a number of studies on selectional preference acquisition. Resnik (1996) presented an information-



(a) CENT (b) GM (c) GMM (d) OffSet

Figure 2: Examples of distributions modeled by (a) CENT, (b) GM, (c) GMM, and (d) OffSet.

theoretic approach that inferred selectional preferences based on the WordNet hypernym hierarchy. Erk et al. (2010) described a method that uses corpus-driven distributional similarity metrics for selectional preference induction. Van de Cruys (2014) investigated the use of neural networks for selectional preference acquisition.

An entity set expansion task (Pantel et al., 2009) is also similar to our problem and has been well studied. For example, Sadamitsu et al. (2011) disambiguated entity word senses and alleviated semantic drift by extracting topic information from LDA for entity set expansion. Zhang et al. (2016) proposed a joint model for entity set expansion and attribute extraction. In this study, we seek to understand how these vectors are distributed in the pre-trained word vector space without using contextual or lexical information. A comparison with the state-of-the-art models for selectional preference induction and entity set expansion is beyond the scope of this work.

3 Problem Formulation

First, let us introduce the notation. W_c is a subset of words that belong to the target word class c. W_o is a subset of words that do not belong to the word class. w_t is a target word that can be a member of the word class c but is not included in W_c. $v_w \in V_w$ is a pre-trained vector for word w. We normalize all the word vectors to unit length.[1] Note that we select the words in W_o to share the same grammatical category as the words in W_c.

Our objective is to distinguish the word w_t from the words in W_o, given W_c and V_w. More specifically, we aim to find a scoring function $\mathrm{f}(w, W_c)$ that assigns a higher score to w_t and lower scores to the words in W_o. For example, suppose c is a class of words that can be a direct object of the verb *play*; W_c, W_o and w_t will be as follows: $W_c = \{$*role, part, game, golf, tennis*$\}$, $W_o = \{$*school, apple, milk, arch, idea*$\}$, and $w_t =$ *basketball*. Our objective is to find a scoring function that assigns a higher score to *basketball* than to *school*, *apple*, *milk*, *arch*, and *idea*.

4 Models

We will start with a centroid-based model (**CENT**) that measures the score between a word w and a word set W_c by calculating the cosine similarity between the word vector and the centroid vector of the word vectors in the word set (Figure 2-(a)). The scoring function can be written as:

$$\mathrm{f}_{\mathrm{CENT}}(w, W_c) = \cos(v_w, \frac{1}{|W_c|} \sum_{w_c \in W_c} v_{w_c}). \quad (1)$$

CENT provides a reasonable baseline, but it does not take the geometry of the distribution of the word vectors into account. Therefore, we introduce a simple Gaussian model (**GM**) to represent the distribution of word vectors belonging to a word class c (Figure 2-(b)). The scoring function is as follows:

$$\mathrm{f}_{\mathrm{GM}}(w, W_c) = \mathcal{N}(v_w | \mu, \Sigma), \quad (2)$$

[1] We also performed experiments with original vectors but obtain similar results in most cases.

where mean μ and covariance matrix Σ are estimated from $\{v_{w_c}|w_c \in W_c\}$. We select the constraint on covariance matrices for Gaussian distribution from {spherical, diagonal, full} by performing cross-validation on W_c. GM is identical with CENT when the covariance matrix is an identity matrix.

Next, we introduce a Gaussian mixture model (**GMM**) to take the existence of subgroups in a word class c into account (Figure 2-(c)). The scoring function can be written as:

$$\mathrm{f}_{\text{GMM}}(w, W_c) = \sum_{k=1}^{K} \pi_k \mathcal{N}(v_w|\mu_k, \Sigma_k), \qquad (3)$$

where weights π_k, means μ_k, and covariance matrices Σ_k are estimated from $\{v_{w_c}|w_c \in W_c\}$. We select the number of components of a Gaussian mixture K from $\{1, 2, \dots, 10\}$ and the constraint on covariance matrices from {spherical, diagonal, full} by performing cross-validation on W_c. GMM can be considered an extension of CENT because it is identical to the CENT when K is 1 and the covariance matrix is an identity matrix.

Furthermore, we will consider another extension of CENT that only considers the existence of subgroups. Since all word vectors are normalized to unit length, $\mathrm{f}_{\text{CENT}}(w, W_c)$ can also be written as:

$$\mathrm{f}_{\text{CENT}}(w, W_c) = \frac{\alpha_{W_c}}{|W_c|} \sum_{w_c \in W_c} \cos(v_w, v_{w_c}), \qquad (4)$$

where α_{W_c} is a normalization term depending only on W_c and thus does not affect the ranking. That is, we can consider that CENT takes the average of the cosine similarities between a word vector v_w and all word vectors in the given word set W_c. If the words in the word set consist of several subgroups, it would be more plausible to consider only the top-k most similar words for scoring. Accordingly, we introduce the k-nearest neighbor model (***k*NN**), which takes the average of only the top k similar vectors. The scoring function can be written as:

$$\mathrm{f}_{k\text{NN}}(w, W_c) = \frac{1}{k} \sum_{w_c \in k\text{NN}_w(W_c)} \cos(v_w, v_{w_c}), \qquad (5)$$

where $k\text{NN}_w(W_c)$ is a function returning a set of words in W_c that take the top-k highest cosine similarities against the word w. The number of k is selected from $\{1, 2, 2^2, \dots, |W_c|\}$ by performing cross-validation on W_c. kNN is identical to CENT when $|W_c|$ is selected as k.

As the last model without negative instances, we adopt a one-class support vector machine (SVM) (Schölkopf et al., 2001)-based model (**1-SVM**) to clarify the importance of the negative instances. We select the kernel from {linear, cubic polynomial, RBF} and tune the parameter $nu \in \{0.05, 0.10, \dots, 0.50\}$ by performing cross-validation. Note that models without negative instances learn a decision function for outlier detection: classifying new data as similar or different to the given positive instances.

Next, we explore models that also leverage negative instances. Here, we introduce a word set W_n as negative instances, where W_n consists of words that are not included in either W_c or W_o. We select the words in W_n to share the same grammatical category as the words in W_c as well as W_o. Both W_o and W_n consist of words that are not included in W_c, but their roles are different. While words in W_o are used as negative instances in the estimation, words in W_n are used as negative instances for modeling the word-class distribution.

As the first model with negative instances, we introduce a model based on the offset between positive and negative instances (**OffSet**). This model is inspired by the Kim and de Marneffe (2013)'s work, which demonstrates that vector offsets can be used to derive adjectival scales. We assume that the vector offset between the centroid of the positive instances and that of the negative instances represents the degree of membership in the vector space (Figure 2-(d)). The scoring function of OffSet is as follows:

$$\mathrm{f}_{\text{OffSet}}(w, W_c, W_n) = \cos(v_w, \frac{v_{\Sigma c}}{|v_{\Sigma c}|} - \frac{v_{\Sigma n}}{|v_{\Sigma n}|}), \qquad (6)$$

$$\text{where} \;\; v_{\Sigma c} = \sum_{w_c \in W_c} v_{w_c}, \;\; v_{\Sigma n} = \sum_{w_n \in W_n} v_{w_n}.$$

Now let us move on to discriminative learning-based models. In this study, we chose a support vector machine with a linear kernel (**SVM**$_L$) or a radial basis function (RBF) kernel (**SVM**$_R$). We only used word vectors as the input of these models and regard the decision function as the scoring function. We tuned the parameter $C \in \{0.1, 0.2, 0.5, 1, 2, 5, 10\}$ and class weight for positive instances $P \in \{1, 2, 4, 8\}$ for SVM$_L$ and the parameter $C \in \{0.2, 0.5, 1, 2, 5\}$, $\gamma \in \{0.2, 0.5, 1, 2\}$, and class weight for positive instances $P \in \{1, 2, 4, 8\}$ for SVM$_R$ by performing cross-validation on W_c and W_n. Note that we wanted to determine the usefulness of negative instances in modeling the distribution of word vectors; thus we make no assertions that these are optimal models.

5 Experiments

5.1 Word embeddings

We used three publicly available pre-trained word vectors for English: the 300-dimensional embeddings trained on the Google News corpus with the CBOW model (CBOW),[2] the 300-dimensional embeddings trained on Wikipedia with the skip-gram model (SGNS),[3] and the 300-dimensional embeddings trained on Wikipedia and Gigaword with the GloVe model (GloVe).[4] For Japanese, we trained 300-dimensional embeddings on an approximately 1.5 billion word corpus collected from the Web, with the CBOW model (CBOW), the skip-gram model (SGNS),[5] and the GloVe model (GloVe).[6] We also trained 50-, 100-, and 200-dimensional embeddings on the same corpus for each model in order to investigate the effect of the vector size.

5.2 Datasets

For the evaluation, we used two types of datasets for English and Japanese, respectively.

5.2.1 SP dataset

As the first type, we used word sets that consist of words which can be a direct object of a certain verb. For example, suppose a word set consists of {*role*, *part*, *game*, *golf*, *tennis*, *etc.*}, where each word can be a direct object of the verb *play*. We did not use the verb itself for evaluation but we can regard this as a selectional preference (SP) task.

For the English SP dataset, we extracted pairs of verbs and their direct objects from the Google Books Syntactic N-grams dataset (Goldberg and Orwant, 2013). We first extracted verbs with the POS tag of VBD, VBP or VBZ that have direct objects at a rate of more than 40%. We decided on a threshold of 40% empirically to extract transitive verbs only. Then, we listed the extracted verbs in descending order of the number of the different direct objects and chose the top 1,000 of them.

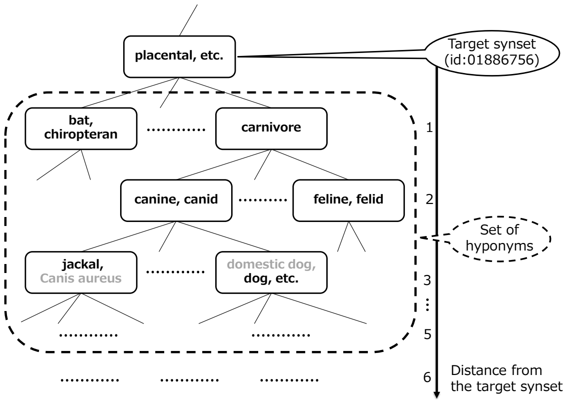


Figure 3: The pair of a synset and a set of its hyponyms in a distance of at most five. The hyponyms are surrounded by a broken line.

For the Japanese SP dataset, we extracted pairs of verbs and their accusative arguments from the predicate-argument data used by Sasano and Okumura (2016). First, we extracted verbs that have accusative arguments at a rate of more than 70%. Again, we decided on a threshold of 70% empirically to extract transitive verbs only. Then, we listed the extracted verbs in descending order of the number of the different accusative arguments and chose the top 1,000 of them.

Both datasets consisted of 1,000 verbs with at least 250 unique direct objects. We selected 200 direct objects as W_c from the most frequent 250 direct objects and the other 50 direct objects as w_t for each verb. Thus, the number of tasks N was 50,000, i.e., 50 tasks for each of the 1,000 verbs. We used 2,000 negative instances against 200 positive instances to build models with negative instances.

5.2.2 WordNet datasets

We used word sets extracted from English and Japanese WordNet (Fellbaum, 1998; Isahara et al., 2008) as the second type. For example, a word set consists of {*dog*, *llama*, *hedgehog*, *wolf*, *etc.*}, which are all hyponyms of the same synonym set (synset n01886756, *placental*). We extracted the pair of a synset ID and a set of words in the synset and its hyponyms in a distance of at most five from the target synset in the WordNet hyponym tree, as shown in Figure 3. We did not use multiword expressions or words whose word vectors are not included in any of the three pre-trained word embeddings.

We extracted synsets that have at least 250 words. There are 109 word sets for English datasets and

[2] https://code.google.com/archive/p/word2vec/

[3] https://github.com/jhlau/doc2vec

[4] http://nlp.stanford.edu/data/glove.6B.zip

[5] https://code.google.com/archive/p/word2vec/. We used the default parameters except for the vector size.

[6] https://nlp.stanford.edu/projects/glove/. We used the same parameters as demo.sh except for setting the window size to 5 and the vector size to 300.

120 word sets for Japanese datasets. We selected 200 words as W_c and the other 50 words as w_t for each synset. The number of tasks N was 5,450, i.e., 50 tasks for each of the 109 synsets for English, and 6,000, i.e., 50 tasks for each of the 120 synsets for Japanese. We used 2,000 negative instances against 200 positive instances to build models with negative instances as well as the SP datasets.

5.3 Experimental settings

We compared eight models: CENT, GM, GMM, kNN, 1-SVM, OffSet, SVM$_L$, and SVM$_R$. For each dataset, we made W_o by extracting 999 words from the other word sets; that is, the number of words for scoring was 1,000, including the target word w_t. For OffSet, SVM$_L$, and SVM$_R$, we make W_n by extracting words from the other word sets subject to the constraint $W_o \cap W_n = \{\}$.

We regarded the problem as a ranking task and adopted the mean reciprocal rank (MRR) as the metric for evaluation. The MRR is calculated by the following equation:

$$\text{MRR} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{\text{rank}(w_{t_i})}, \qquad (7)$$

where rank(w_{t_i}) is the rank of the target word w_{t_i} for each task. We tune the parameters to maximize the MRR in parameter tuning.

We measured the statistical significance with an approximate randomization test (Chinchor, 1992) with 99,999 iterations and significance level $\alpha = 0.05$ after Bonferroni correction. To satisfy the independence assumption, we treated each verb (for the SP datasets) or synset (for the WordNet datasets) as the unit of a randomization test.

5.4 Experimental results

5.4.1 Results on the SP datasets

Tables 1 and 2 show the experimental results on the SP dataset for English and Japanese, respectively. In these tables, the best scores for each word embedding model and the scores with no significant difference from the best score are indicated in bold. In addition, the CENT score and the scores with no significant difference from the CENT score are italicized.

The results in these tables indicate that the models considering the geometry of the distribution or the existence of subgroups in the word class outperform the centroid-based model (CENT) for both the English and Japanese SP datasets. In particular, a simple Gaussian model (GM) performed the best among the models that only depend on positive instances. This indicates that these word sets are distributed with different variances depending on the direction in the vector space and it is useful to consider the geometry of the distribution.

Model	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
CBOW	*.1642*	.2539	.2360	.2097	.1726	.2782	.3397	**.3905**
SGNS	*.1887*	.2461	.2308	*.1918*	.2252	.2189	.3365	**.3608**
GloVe	*.1925*	.2596	.2462	.2245	.2295	.1150	.3554	**.3800**
Ave.	*.1818*	.2532	.2377	.2087	.2091	.2040	.3439	**.3771**

Table 1: Results on the English SP dataset.

Model	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
CBOW	*.2600*	.3151	.2947	.2783	.2812	.2516	.4371	**.4922**
SGNS	*.0789*	.2231	.2039	.1757	.1249	.2594	.4173	**.4510**
GloVe	*.1643*	.2489	.2377	.2016	.1927	.2088	.3264	**.3632**
Ave.	*.1677*	.2624	.2454	.2185	.1996	.2399	.3936	**.4355**

Table 2: Results on the Japanese SP dataset.

The two discriminative learning-based models with negative instances, SVM$_L$ and SVM$_R$, achieved much higher performance, whereas 1-SVM yielded a limited improvement over CENT. This demonstrates that modeling the distribution with only positive instances has an obvious limitation, and it is essential to leverage the negative instances as well. OffSet with CBOW or SGNS achieved a relatively good performance, but OffSet with GloVe did not, which suggests that the usefulness of the offset depends on the word embedding model.

5.4.2 Results on the WordNet datasets

Tables 3 and 4 show the experimental results on the WordNet dataset for English and Japanese, respectively. The meaning of bold and italic fonts is identical to that on the SP dataset.

The two discriminative learning-based models with negative instances and OffSet with CBOW or SGNS achieved a relatively high performance. This demonstrates that the negative instances must be taken into account to model the distribution properly. On the other hand, in contrast with the SP datasets, there were no significant improvements when the geometry of the distribution and the existence of subgroups were considered.

The scores were generally lower than those of the SP datasets. We conjecture that this is because WordNet is developed manually and reflects human

Model	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
CBOW	*.1435*	*.1320*	*.1460*	*.1473*	*.1541*	.2263	.2564	**.2678**
SGNS	*.1767*	*.1679*	.1573	.1625	*.1704*	.1998	**.2292**	**.2357**
GloVe	*.1792*	*.1694*	.1562	*.1744*	*.1684*	.1310	.2075	**.2264**
Ave.	*.1665*	.1564	.1532	.1614	.1643	.1857	.2310	**.2433**

Table 3: Results on the English WordNet dataset.

Model	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
CBOW	*.1996*	*.1991*	*.1918*	.2169	*.2082*	.2656	.2730	**.2961**
SGNS	*.0466*	.0521	.0774	.0768	.0701	.2367	.2686	**.2862**
GloVe	*.1055*	*.1050*	*.1021*	*.0987*	*.0984*	.0681	.2033	**.2189**
Ave.	*.1172*	.1187	.1238	.1308	.1256	.1901	.2483	**.2671**

Table 4: Results on the Japanese WordNet dataset.

Size	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
50	.1686	.2360	.2055	.1909	.1825	.1769	.2842	**.3568**
100	.1738	.2557	.2177	.2075	.1954	.2189	.3366	**.4044**
200	.1724	.2697	.2233	.2178	.2005	.2363	.3813	**.4340**
300	.1677	.2624	.2454	.2185	.1996	.2399	.3936	**.4355**

Table 5: The average scores of different vector size with the Japanese SP dataset.

$\|W_c\|$	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
25	.1563	.1728	.1522	.1635	.1562	.1880	.2326	**.2600**
50	.1612	.2008	.1779	.1795	.1722	.2144	.2898	**.3157**
100	.1652	.2388	.2098	.1988	.1880	.2307	.3475	**.3790**
200	.1677	.2624	.2454	.2185	.1996	.2399	.3936	**.4355**

Table 6: The average scores of different word set size with the Japanese SP dataset.

intuition, whereas the SP datasets are automatically built from the corpus and are highly compatible with the pre-trained word vectors. In addition, we examined which types of words tend to rank low and found that words extracted from a synset corresponding to their infrequent sense such as *stock* in the sense of livestock tend to rank low. We leave further exploration for future work.

5.4.3 Discussion

It is interesting that although SVM$_L$ is effectively just a linear classifier, SVM$_L$ achieves a relatively high performance. This is likely due to the relatively large vector size compared to the number of positive instances and indicates that the positive instances occupy a certain span in the vector space though such a span cannot be determined by only using positive instances. We confirmed two desirable properties of the discriminative learning-based models with negative instances for practical applications. One is that since we used simple models, they do not require much training time. The other is that their performance is relatively stable among the different word embeddings and datasets compared to the other models.

We also investigated the effect of the vector size and the number of positive instances on the Japanese SP dataset. Table 5 shows the averaged CBOW, SGNS, and GloVe scores for different vector dimensions, 50, 100, 200, and 300. We found that while CENT and 1-SVM were not affected much by the vector size, the other models, particularly OffSet, SVM$_L$, and SVM$_R$, were significantly affected by the vector size. Table 6 shows the averaged CBOW, SGNS, and GloVe scores for the different number of positive instances, 25, 50, 100, and 200. We can conclude that all the models perform at a higher level based on the larger number of positive instances, especially for GM, GMM, SVM$_L$, and SVM$_R$. This is not surprising, since these models have a large number of parameters and can extract a rich variety of information from the large number of positive instances. Similar tendencies were also observed with the other dataset. These results demonstrate that we can obtain relatively high performance by using discriminative learning-based models with a large enough vector and training data size.

5.5 Degree of membership

Rosch (1975) developed the prototype concept and proved that not all members of a category are equally representative of the category. Here, we are interested in the relationship between the scores calculated by each model and the degree of membership. We thus investigated how consistent the score calculated by each model is with human intuition on the degree of membership.

For this experiment, we used the typicality data by Rosch (1975). Rosch asked 209 college students to use a 7-point scale to rate the extent to which each instance represents their idea or image of the meaning of the category term, and reported the rank orders with the mean ratings for ten categories.[7] For example, for the *Furniture* category, 60 examples are ranked with the mean ratings, *chair* and *sofa* are top-ranked with the score of 1.04, and

[7] To test the reliability of ratings, Rosch (1975) obtained Spearman rank-order correlations and Pearson product-moment correlations between sub-groups of students and reported that consistency was extremely high.

Category	Synset ID	$\lvert W_R \rvert$	$\lvert W_c \rvert$	$\lvert W_R \cap W_c \rvert$
Furniture	n03405725	60	89	26
Fruit	n13134947	51	165	41
Vehicle	n04524313	50	346	34
Weapon	n04565375	60	119	19
Vegetable	n07707451	56	102	27
Bird	n01503061	54	330	51
Sport	n00523513	59	106	33
Clothing	n03051540	55	409	31

Table 7: Statistics of the typicality dataset.

Model	CENT	GM	GMM	kNN	1-SVM	OffSet	SVM$_L$	SVM$_R$
					ρ			
CBOW	.1736	.1905	.1706	.2417	.1160	**.3224**	.3176	.2562
SGNS	.2848	.3194	**.4024**	.3221	.1924	.2940	.3363	.3121
GloVe	.1458	.1949	.1448	.3204	.1780	**.4383**	.3367	.2702
Ave.	.2014	.2349	.2393	.2947	.1621	**.3516**	.3302	.2795
					τ			
CBOW	.1230	.1373	.1198	.1833	.0728	**.2400**	.2289	.1855
SGNS	.2101	.2400	**.2945**	.2355	.1400	.2066	.2390	.2180
GloVe	.1012	.1401	.1080	.2254	.1266	**.3038**	.2391	.1908
Ave.	.1448	.1725	.1741	.2147	.1131	**.2501**	.2357	.1981

Table 8: Averaged rank correlation coefficients against the typicality data by Rosch.

stove is ranked as 50th with the score of 5.4.

In this study, we used eight categories that have a corresponding synset in WordNet. Table 7 shows the statistics of the dataset. In the table, $|W_R|$ denotes the number of examples in Rosch's dataset, $|W_c|$ denotes the number of words in the synset and its hyponyms in the WordNet, and $|W_R \cap W_c|$ is the number of words included in both W_R and W_c, which we try to rank here.

In this experiment, the objective was not to distinguish a possible member from others but to rank the positive member w_c in W_c according to the degree of membership. That is, we first formed the scoring function by using W_c and W_n and then applied the function to each member of $W_R \cap W_c$ to predict the typicality ranking. We evaluated the ranking by calculating Spearman's rank correlation coefficient (ρ) and Kendall's rank correlation coefficient (τ) against the ranking of the goodness-of-example in Rosch's dataset. We computed the average rank correlation coefficient over the eight categories for ρ and τ. Table 8 shows the experimental results.

In contrast with the previous experiments, the highest scores were achieved by OffSet. These results suggest that the vector offsets can be used to derive the degree of membership. We can say that, while discriminative learning-based models, especially SVM$_R$, can find the boundary of a category in a vector space with high accuracy, the vector offset between the centroid of positive instances and that of negative instances can properly represent the degree of membership in a category.

When we focused on each combination of the embedding and distribution models, we found that the highest and second highest scores were achieved by OffSet with GloVe and GMM with SGNS, respectively. In contrast, both achieved relatively low performance in distinguishing a possible member of a word class from others, as shown in Table 3. These results demonstrate that the proper models for finding the boundaries of a class and those for determining the degree of membership are different and that choosing a proper model depending on the task is essential.

6 Conclusion and Future Work

We investigated the distribution of words that belong to a certain word class in a pre-trained general-purpose word vector space. The experimental results show that a centroid-based approach cannot provide a reasonably good model and considering the geometry of the distribution and the existence of subgroups is useful for modeling the distribution in some cases. However, the impact is limited, and the negative instances must be taken into account for adequate modeling. The results indicate that just observing the distribution of positive instances is not enough to understand the geometry of word embedding spaces. Furthermore, we investigated the relationship between the score calculated by each model and the degree of membership and demonstrated that, while discriminative learning-based models can distinguish a possible member of a word class from others, the offset-based model achieves higher correlations with the degree of membership.

The investigation in this study leveraged only general-purpose word vectors to represent the meaning of a word. However, several studies have expanded the method for acquiring a word vector to account for the uncertainty of word meanings and word polysemy (e.g., Athiwaratkun et al. 2018). In addition, contextualized word embeddings have been shown to be very effective on a range of NLP tasks (Peters et al., 2018; Devlin et al., 2019). Furthermore, Gong et al. (2018) reported that word embeddings learned in several tasks are biased towards word frequency: the embeddings of high-frequency and low-frequency words lie in different subregions of the embedding space. Thus, in the future, we will take the uncertainty, polysemy, and context sensitivity of the word meanings and the frequency of words into account and explore better ways of modeling the word-class distributions in semantic vector spaces.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 16K16110 and 18H03286.

References

Ben Athiwaratkun and Andrew Wilson. 2017. Multimodal word distributions. In *Proc. of ACL'17*, pages 1645–1656.

Ben Athiwaratkun, Andrew Wilson, and Anima Anandkumar. 2018. Probabilistic fastText for multi-sense word embeddings. In *Proc. of ACL'18*, pages 1–11.

Marco Baroni, Georgiana Dinu, and German Kruszewski. 2014. Don't count, predict! a systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proc. of ACL'14*, pages 238–247.

Xinxiong Chen, Zhiyuan Liu, and Maosong Sun. 2014. A unified model for word sense representation and disambiguation. In *Proc. of EMNLP'14*, pages 1025–1035.

Nancy Chinchor. 1992. The statistical significance of the MUC-4 results. In *Proceedings of the 4th Message Understanding Conference (MUC)*, pages 30–50.

Tim Van de Cruys. 2014. A neural network approach to selectional preference acquisition. In *Proc. of EMNLP'14*, pages 26–35.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. of NAACL-HLT'19*, pages 4171–4186.

Katrin Erk, Sebastian Padó, and Ulrike Padó. 2010. A flexible, corpus-driven model of regular and inverse selectional preferences. *Computational Linguistics*, 36(4):723–763.

Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proc. of NAACL-HLT'15*, pages 1606–1615.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Goran Glavaš and Ivan Vulić. 2018. Explicit retrofitting of distributional word vectors. In *Proc. of ACL'18*, pages 34–45.

Yoav Goldberg and Jon Orwant. 2013. A dataset of syntactic-ngrams over time from a very large corpus of English books. In *Proc. of *SEM'13*, pages 241–247.

Chengyue Gong, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2018. Frage: Frequency-agnostic word representation. In *Proc. of NIPS'18*, pages 1334–1345.

Hitoshi Isahara, Francis Bond, Kiyotaka Uchimoto, Masao Utiyama, and Kyoko Kanzaki. 2008. Development of the Japanese WordNet. In *Proc. of LREC'08*, pages 2420–2423.

Joo-Kyung Kim and Marie-Catherine de Marneffe. 2013. Deriving adjectival scales from continuous space word representations. In *Proc. of EMNLP'13*, pages 1625–1630.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In *Proc. of NIPS'13*, pages 3111–3119.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continuous space word representations. In *Proc. of NAACL-HLT'13*, pages 746–751.

David Mimno and Laure Thompson. 2017. The strange geometry of skip-gram with negative sampling. In *EMNLP'17*, pages 2873–2878.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017. Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics*, 5:309–324.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient nonparametric estimation of multiple embeddings per word in vector space. In *Proc. of EMNLP'14*, pages 1059–1069.

Patrick Pantel, Eric Crestan, Arkady Borkovsky, Ana-Maria Popescu, and Vishnu Vyas. 2009. Web-scale distributional similarity and entity set expansion. In *Proc. of EMNLP'09*, pages 938–947.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proc. of EMNLP'14*, pages 1532–1543.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL-HLT'18*, pages 2227–2237.

Philip Resnik. 1996. Selectional constraints: An information-theoretic model and its computational realization. *Cognition*, 61:127–159.

Eleanor Rosch. 1975. Cognitive representations of semantic categories. *Journal of Experimental Psychology: General*, 104:192–233.

Sascha Rothe and Hinrich Schütze. 2015. AutoExtend: Extending word embeddings to embeddings for synsets and lexemes. In *Proc. of ACL-IJCNLP'15*, pages 1793–1803.

Kugatsu Sadamitsu, Kuniko Saito, Kenji Imamura, and Genichiro Kikui. 2011. Entity set expansion using topic information. In *Proc. of ACL-HLT'11*, pages 726–731.

Ryohei Sasano and Manabu Okumura. 2016. A corpus-based analysis of canonical word order of Japanese double object constructions. In *Proc. of ACL'16*, pages 2236–2244.

Bernhard Schölkopf, John C. Platt, John C. Shawe-Taylor, Alex J. Smola, and Robert C. Williamson. 2001. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471.

Fei Tian, Hanjun Dai, Jiang Bian, Bin Gao, Rui Zhang, Enhong Chen, and Tie-Yan Liu. 2014. A probabilistic model for learning multi-prototype word embeddings. In *Proc. of COLING'14*, pages 151–160.

Luke Vilnis and Andrew McCallum. 2015. Word representations via gaussian embedding. In *Proc. of ICLR'15*.

Kristian Woodsend and Mirella Lapata. 2015. Distributed representations for unsupervised semantic role labeling. In *Proc. of EMNLP'15*, pages 2482–2491.

Yadollah Yaghoobzadeh and Hinrich Schütze. 2016. Intrinsic subspace evaluation of word embedding representations. In *Proc. of ACL'16*, pages 236–246.

Zhenzhong Zhang, Le Sun, and Xianpei Han. 2016. A joint model for entity set expansion and attribute extraction from web search queries. In *Proc. of AAAI'16*, pages 3101–3107.