Word2Box: Capturing Set-Theoretic Semantics of Words using Box Embeddings

Anonymous ACL submission

Abstract

Learning representations of words in a continuous space is perhaps the most fundamental task in NLP, a prerequisite for nearly all modern machine-learning techniques. Often the objective is to capture distributional similarity via vector dot product, however this is just one relation between word meanings we may wish to capture. It is natural to consider words as (soft) equivalence classes based on similarity, it is natural to expect the ability to perform set-theoretic operations (intersection, union, difference) on these representations. This is particularly relevant for words which are homographs - for example, "tongue"∩"body" should be similar to "mouth", while "tongue"∩"language" should be similar to "dialect". Box embeddings are a novel region-based representation which provide the capability to perform these set-theoretic operations. In this work, we provide a fuzzy-set interpretation of box embeddings, and train box embeddings with a CBOW objective where contexts are represented using intersection. We demonstrate improved performance on various word similarity tasks, particularly on less common words, and perform a quantitative and qualitative analysis exploring the additional unique expressivity provided by WORD2BOX.

1 Introduction

The concept of learning a distributed representation for a word has fundamentally changed the field of natural language processing. The introduction of efficient methods for training vector representations of words in Word2Vec (Mikolov et al., 2013), and later GloVe (Pennington et al.) as well as FastText (Bojanowski et al., 2017) revolutionized the field, paving the way for the recent wave of deep architectures for language modeling, all of which implicitly rely on this fundamental notion that a word can be effectively represented by a vector.

While now ubiquitous, the concept of representing a word as a single point in space is not particularly natural. All senses and contexts, levels of abstraction, variants and modifications which the word may represent are forced to be captured by the specification of a single location in Euclidean space. It is thus unsurprising that a number of alternatives have been proposed.

Gaussian embeddings (Vilnis and McCallum, 2015) propose modeling words using densities in latent space as a way to explicitly capture uncertainty. Poincaré embeddings (Tifrea et al., 2019) attempt to capture a latent hierarchical graph between words by embedding words as vectors in hyperbolic space. Trained over large corpora via similar unsupervised objectives as vector baselines, these models demonstrate an improvement on word similarity tasks, giving evidence to the notion that vectors are not capturing all relevant structure from their unsupervised training objective.

A more recent line of work explores region-based embeddings, which use geometric objects such as disks (Suzuki et al., 2019), cones (Vendrov et al., 2016; Lai and Hockenmaier, 2017; Ganea et al., 2018), and boxes (Vilnis et al., 2018) to represent entities. These models are often motivated by the need to express asymmetry, benefit from particular inductive biases, or benefit from calibrated probabilistic semantics. In the context of word representation, their ability to represent words using geometric objects with well-defined intersection, union, and difference operations is of interest, as we may expect these operations to translate to the words being represented in a meaningful way.

In this work, we introduce WORD2BOX, a region-based embedding for words where each word is represented by an n-dimensional hyperrectangle or "box". Of the region-based embeddings, boxes were chosen as the operations of intersection, union, and difference are easily calculable. Specifically, we use a variant of box embeddings known as Gumbel boxes, introduced in (Dasgupta et al., 2020). Our objective (both for training and inference) is inherently set-theoretic, not probabilistic, and as such we first provide a fuzzy-set interpretation of Gumbel boxes yielding rigorously defined mathematical operations for intersection, union, and difference of Gumbel boxes.

We train boxes on a large corpus in an unsupervised manner with a continuous bag of words (CBOW) training objective, using the intersection of boxes representing the context words as the representation for the context. The resulting model demonstrates improved performance compared to vector baselines on a large number of word similarity benchmarks. We also compare the models' abilities to handle set-theoretic queries, and find that the box model outperforms the vector model 90% of the time. Inspecting the model outputs qualitatively also demonstrates that WORD2BOX can provide sensible answers to a wide range of set-theoretic queries.

2 Background

Notation Let $V = \{v_i\}_{i=1}^N$ denote the vocabulary, indexed in a fixed but arbitrary order. A sentence $\mathbf{s} = (s_1, \ldots, s_j)$ is simply a (variable-length) sequence of elements in our vocab $s_i \in V$, and a document $\mathbf{d} = \{\mathbf{s}_i\}$ is a multiset[1] of sentences. We view our corpus $C = \{\mathbf{d}_i\}$ as a multiset of documents, and also consider the multiset $C_S = \{\mathbf{s} : \mathbf{s} \in \mathbf{d} \in C\}$ of all sentences in our corpus. Given some fixed "window size" ℓ, for each word s_i in a sentence $\mathbf{s}$ we can consider the window centered at i,

$$\mathbf{w}_i = [s_{i-\ell}, \ldots, s_i, \ldots, s_{i+\ell}],$$

where we omit any indices exceeding the bounds of the sentence. Given a window $\mathbf{w}_i$ we denote the center word using $\operatorname{cen}(w_i) = s_i$, and denote all remaining words as the context $\operatorname{con}(\mathbf{w}_i)$. We let C_W be the multiset of all windows in the corpus.

2.1 Fuzzy sets

Given any ambient space U a set $S \subseteq U$ can be represented by its characteristic function $\mathbb{1}_S : U \to \{0, 1\}$ such that $\mathbb{1}_S(u) = 1 \iff u \in S$. This definition can be generalized to consider functions $m : U \to [0, 1]$, in which case we call the pair $A = (U, m)$ a *fuzzy set* and $m = m_A$ is known as the *membership function* (Zadeh, 1965; Klir and Yuan, 1996). There is historical precedent for the use of fuzzy sets in computational linguistics (Zhelezniak et al., 2019; Lee and Zadeh, 1969), and more generally are naturally required any time we would like to learn a set representation in a gradient-based model, as hard assignments would not allow for gradient flow.

In order to extend the notion of intersection to fuzzy sets, it is necessary to define a *t-norm*, which is a binary operation $\top : [0, 1] \times [0, 1] \to [0, 1]$ which is commutative, monotonic, associative, and equal to the identity when either input is 1. The $\min$ and product operations are common examples of t-norms. Given any t-norm, the intersection of fuzzy sets A and B has membership function $m_{A\cap B}(x) = \top(m_A(x), m_B(x))$. Any t-norm has a corresponding t-conorm which is given by $\bot(a, b) = 1 - \top(1 - a, 1 - b)$; for $\min$ the t-conorm is $\max$, and for product the t-conorm is the probabilistic sum, $\bot_{\text{sum}}(a, b) = a + b - ab$. This defines the union between fuzzy sets, where $m_{A\cup B}(x) = \bot(m_A(x), m_B(x))$. Finally, the complement of a fuzzy set simply has member function $m_{A^c}(x) = 1 - m_A(x)$.

2.2 Box embeddings

Box embeddings, introduced in (Vilnis et al., 2018), represent elements $\mathbf{x}$ of some set X as a Cartesian product of intervals,

$$\begin{aligned}\operatorname{Box}(\mathbf{x}) &:= \prod_{i=1}^{d} [x_i^-, x_i^+] \\ &= [x_1^-, x_1^+] \times \cdots \times [x_d^-, x_d^+] \subseteq \mathbb{R}^d.\end{aligned} \quad (1)$$

The volume of a box can be calculated as

$$|\operatorname{Box}(\mathbf{x})| = \prod_{i=1}^{d} \max(0, x_i^+ - x_i^-),$$

and when two boxes intersect, their intersection is

$$\begin{aligned}&\operatorname{Box}(\mathbf{x}) \cap \operatorname{Box}(\mathbf{y}) \\ &\quad= \prod_{i=1}^{d} [\max(x_i^-, y_i^-), \min(x_i^+, y_i^+)].\end{aligned}$$

Boxes are trained via gradient descent, and these hard min and max operations result in large areas of the parameter space with no gradient signal. Dasgupta et al. (2020) addresses this problem by modeling the corners of the boxes $\{x_i^\pm\}$ with Gumbel random variables, $\{X_i^\pm\}$, where the probability

[1] A *multiset* is a set which allows for repetition, or equivalently a sequence where order is ignored.

of any point $\mathbf{z} \in \mathbb{R}^d$ being inside the box $\text{Box}_G(\mathbf{x})$ is given by

$$P(\mathbf{z} \in \text{Box}_G(\mathbf{x})) = \prod_{i=1}^{d} P(z_i > X_i^-) P(z_i < X_i^+).$$

For clarity, we will denote the original ("hard") boxes as Box, and the Gumbel boxes as Box_G. The Gumbel distribution was chosen as it was min/max stable, thus the intersection $\text{Box}_G(\mathbf{x}) \cap \text{Box}_G(\mathbf{y})$ which was defined as a new box with corners modeled by the random variables $\{Z_i^\pm\}$ where

$$Z_i^- \coloneqq \max(X_i^-, Y_i^-) \text{ and } Z_i^+ \coloneqq \min(X_i^+, Y_i^+)$$

is actually a Gumbel box as well. Boratko et al. observed that

$$P(\mathbf{z} \in \text{Box}_G(\mathbf{x}) \cap \text{Box}_G(\mathbf{y})) = \\ P(\mathbf{z} \in \text{Box}_G(\mathbf{x}))P(\mathbf{z} \in \text{Box}_G(\mathbf{y})), \quad (2)$$

and also provided a rigorous probabilistic interpretation for Gumbel boxes when embedded in a space of finite measure, leading to natural notions of "union" and "intersection" based on these operations of the random variables (Boratko et al.).

In this work, we do not embed the boxes in a space of finite measure, but instead interpret them as *fuzzy sets*, where the above probability acts as a soft membership function.

3 Fuzzy Sets of Windows

In this section, we describe the motivation for using fuzzy sets to represent words, starting with an approach using traditional sets.

First, given a word $v \in V$, we can consider the windows centered at v,

$$\text{cen}_W(v) \coloneqq \{w \in W : \text{cen}(w) = v\},$$

and the set of windows whose context contains v,

$$\text{con}_W(v) \coloneqq \{w \in W : \text{con}(w) \ni v\}.$$

A given window is thus contained inside the intersection of the sets described above, namely

$$[w_{-j}, \ldots, w_0, \ldots, w_j] \\ \in \text{cen}_W(w_0) \cap \bigcap_{i \neq 0} \text{con}_W(w_i).$$

As an example, the window

$$\mathbf{w} = \text{"quick brown fox jumps over"},$$

is contained inside the $\text{cen}_W(\text{"fox"})$ set, as well as $\text{con}_W(\text{"quick"})$, $\text{con}_W(\text{"brown"})$, $\text{con}_W(\text{"jumps"})$, $\text{con}_W(\text{"over"})$. With this formulation, the intersection of the con_W sets provide a natural choice of representation for the context. We might hope that $\text{cen}_W(v)$ provides a reasonable representation for the word v itself, however for any $u \neq v$ we have $\text{cen}_W(u) \cap \text{cen}_W(v) = \emptyset$.

We would like the representation of u to overlap with v if u has "similar meaning" to v, i.e. we would like to consider

$$\widetilde{\text{cen}_W}(v) \coloneqq \{w \in W : \text{cen}(w) \text{ similar to } v\}.$$

A crisp definition of *meaning* or *similarity* is not possible (Hill et al., 2015; Finkelstein et al., 2001) due to individual subjectivity. Inner-annotator agreement for Hill et al. (2015) is only 0.67, for example, which makes it clear that $\widetilde{\text{cen}_W}(v)$ could not possibly be represented as a traditional set. Instead, it seems natural to consider $\widetilde{\text{cen}_W}(v)$ as represented by a fuzzy set (W, m), where $m(w) \in [0, 1]$ can be thought of as capturing graded similarity between v and $\text{cen}(w)$.[2] In the same way, we can define

$$\widetilde{\text{con}_W}(v) \coloneqq \{w \in W : \text{con}(v) \ni w \text{ similar to } v\},$$

which would also be represented as a fuzzy set.

As we wish to capture these similarities with a machine learning model, we now must find trainable representations of fuzzy sets.

Remark 1. Our objective of learning trainable representations for these sets provides an additional practical motivation for using fuzzy sets - namely, the hard assignment of elements to a set is not differentiable. Any gradient-descent based learning algorithm which seeks to represent sets will have to consider a smoothed variant of the characteristic function, which thus leads to fuzzy sets.

4 Gumbel Boxes as Fuzzy Sets

In this section we will describe how we model fuzzy sets using Gumbel boxes (Dasgupta et al., 2020). As noted in Section 2.2, the Gumbel Box model represents entities $\mathbf{x} \in X$ by $\text{Box}_G(\mathbf{x})$ with corners modeled by Gumbel random variables $\{X_i^\pm\}$. The probability of a point $\mathbf{z} \in \mathbb{R}^d$ being

[2] For an even more tangible definition, we can consider $m(w)$ the percentage of people who consider u to be similar to $\text{cen}(w)$ when used in context $\text{con}(w)$.

inside this box is

$$P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x})) = \prod_{i=1}^{d} P(z_i > X_i^-) P(z_i < X_i^+).$$

Since this is contained in $[0, 1]$, we have that $(\mathbb{R}^d, P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}))$ is a fuzzy set. For clarity, we will refer to this fuzzy set as $\mathrm{Box}_F(\mathbf{x})$.

The set complement operation has a very natural interpretation in this setting, as $\mathrm{Box}_F(\mathbf{x})^c$ has membership function $1 - P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}))$, that is, the probability of $\mathbf{z}$ not being inside the Gumbel box. The product t-norm is a very natural choice as well, as the intersection $\mathrm{Box}_F(\mathbf{x}) \cap \mathrm{Box}_F(\mathbf{y})$ will have membership function $P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}))P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{y}))$, which is precisely the membership function associated with $\mathrm{Box}_G(\mathbf{x}) \cap \mathrm{Box}_G(\mathbf{y})$, where here the intersection is between Gumbel boxes as defined in Dasgupta et al. (2020). Finally, we find that the membership function for the union $\mathrm{Box}_F(\mathbf{x}) \cup \mathrm{Box}_F(\mathbf{y})$ is given (via the t-conorm) by

$$\begin{aligned} &P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x})) + P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{y})) - \\ &\qquad P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}) P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{y})). \end{aligned} \tag{3}$$

Remark 2. Prior work on Gumbel boxes had not defined a union operation on Gumbel boxes, however (3) has several pleasing properties apart from being a natural consequence of using the product t-norm. First, it can be directly interpreted as the probability of $\mathbf{z}$ being inside $\mathrm{Box}_G(\mathbf{x})$ or $\mathrm{Box}_G(\mathbf{y})$. Second, if the Gumbel boxes were embedded in a space of finite measure, as in Boratko et al., integrating (3) would yield the probability corresponding to $P(\mathrm{Box}(\mathbf{x}) \cup \mathrm{Box}(\mathbf{y}))$.

To calculate the size of the fuzzy set $\mathrm{Box}_F(\mathbf{x})$ we integrate the membership function over $\mathbb{R}^d$,

$$|\,\mathrm{Box}_F(\mathbf{x})| = \int_{\mathbb{R}^d} P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}))\, d\mathbf{z}.$$

The connection between this integral and that which was approximated in (Dasgupta et al., 2020) is provided by Lemma 3 of (Boratko et al.), and thus we have

$$|\,\mathrm{Box}_F(\mathbf{x})| \approx \prod_{i=1}^{d} \beta \log\left(1 + \exp\left(\frac{\mu_i^+ - \mu_i^-}{\beta} - 2\gamma\right)\right)$$

where μ_i^-, μ_i^+ are the location parameters for the Gumbel random variables X_i^-, X_i^+, respectively. As mentioned in Section 2.2, Gumbel boxes are closed under intersection, i.e. $\mathrm{Box}_G(\mathbf{x}) \cap \mathrm{Box}_G(\mathbf{y})$ is also a Gumbel box, which implies that the size of the fuzzy intersection

$$\begin{aligned} &|\,\mathrm{Box}_F(\mathbf{x}) \cap \mathrm{Box}_F(\mathbf{y})| \\ &= \int_{\mathbb{R}^d} P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}))P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{y}))\, d\mathbf{z} \\ &= \int_{\mathbb{R}^d} P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}) \cap \mathrm{Box}_G(\mathbf{y}))\, d\mathbf{z} \end{aligned}$$

can be approximated as well. As both of these are tractable, integrating (3) is also possible via linearity. Similarly, we can calculate the size of fuzzy set differences, such as

$$\begin{aligned} &|\,\mathrm{Box}_F(\mathbf{x}) \setminus \mathrm{Box}_F(\mathbf{y})| = \\ &\int_{\mathbb{R}^d} P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{x}))[1 - P(\mathbf{z} \in \mathrm{Box}_G(\mathbf{y}))]\, d\mathbf{z}. \end{aligned}$$

By exploiting linearity and closure under intersection, it is possible to calculate the size of arbitrary fuzzy intersections, unions, and set differences, as well as any combination of such operations.

Remark 3. If our boxes are embedded in a space of finite measure, as in (Boratko et al.), the sizes of these fuzzy sets correspond to the intersection, union, and negation of the binary random variables they represent.

5 Training

In this section we describe our method of training fuzzy box representations of words, which we refer to as WORD2BOX.

In Section 3 we defined the fuzzy sets $\widetilde{\mathrm{cen}}_W(v)$ and $\widetilde{\mathrm{cen}}_W(v)$, and in Section 4 we established that Gumbel boxes can be interpreted as fuzzy sets, thus for WORD2BOX we propose to learn center and context box representations

$$\begin{aligned} \mathrm{cen}_B(v) &:= \mathrm{Box}_F(\widetilde{\mathrm{cen}}_W(v)) \\ \mathrm{con}_B(v) &:= \mathrm{Box}_F(\widetilde{\mathrm{cen}}_W(v)). \end{aligned}$$

Given a window, $\mathbf{w} = [w_{-j}, \dots, w_0, \dots, w_j]$, we noted that $\mathbf{w}$ must exist in the intersection,

$$\widetilde{\mathrm{cen}}_W(w_0) \cap \bigcap_{i \neq 0} \widetilde{\mathrm{con}}_W(w_i) \tag{4}$$

and thus we consider a max-margin training objective where the score for a given window is given as

$$f(\mathbf{w}) := \left|\mathrm{cen}_B(w_0) \cap \bigcap_{i \neq 0} \mathrm{cen}_B(w_i)\right|. \tag{5}$$

To create a negative example $\mathbf{w}'$ we follow the same procedure as CBOW from Mikolov et al. (2013), replacing center words with a word sampled from the unigram distribution raised to the $3/4$. We also subsample the context words as in (Mikolov et al., 2013). As a vector baseline, we compare with a WORD2VEC model trained in CBOW-style. We attach the source code with supplementary material.

6 Experiments and Results

We evaluate both WORD2VEC and WORD2BOX on several quantitative and qualitative tasks that cover the aspects of semantic similarity, relatedness, lexical ambiguity, and uncertainty. Following the previous relevant works (Athiwaratkun and Wilson, 2018; Meyer and Lewis, 2020; Baroni et al., 2012), we train on the lemmatized WaCkypedia corpora (Baroni et al., 2009) which, after pre-processing (details in Appendix A) contains around 0.9 billion tokens, with just more than 112k unique tokens in the vocabulary. Noting that an n-dimensional box actually has $2n$ parameters (for min and max coordinates), we compare 128-dimensional WORD2VEC embeddings and 64-dimensional WORD2BOX embeddings for all our experiments. We train over 60 different models for both the methods for 10 epochs using random sampling on a wide range of hyperparameters (please refer to appendix A for details including learning rate, batch size, negative sampling, sub-sampling threshold etc.). In order to ensure that the only difference between the models was the representation itself, we implemented a version of WORD2VEC in PyTorch, including the negative sampling and subsampling procedures recommended in (Mikolov et al., 2013), using the original implementation as a reference. As we intended to train on GPU, however, our implementation differs from the original in that we use Stochastic Gradient Descent with varying batch sizes. We provide our source code with the supplementary materials.

6.1 Word Similarity Benchmarks

We primarily evaluate our method on several word similarity benchmarks: SimLex-999 (Hill et al., 2015), WS-353 (Finkelstein et al., 2001), YP-130 (Yang and Powers, 2006), MEN (Bruni et al., 2014), MC-30 (Miller and Charles, 1991), RG-65 (Rubenstein and Goodenough, 1965), VERB-143 (Baker et al., 2014), Stanford RW (Luong et al., 2013), Mturk-287 (Radinsky et al., 2011) and Mturk-771 (Halawi et al., 2012). These datasets consist of pairs of words (both noun and verb pairs) that are annotated by human evaluators for semantic similarity and relatedness.

In table 1 we compare the WORD2BOX and WORD2VEC models which are best performing on the similarity benchmarks. We observe that WORD2BOX outperforms WORD2VEC (as well as the results reported by other baselines) in the majority of the word similarity tasks. We outperform WORD2VEC by a large margin in Stanford RW and YP-130, which are the rare-word datasets for noun and verb respectively. Noticing this effect, we enumerated the frequency distribution of each dataset. The datasets fall in different sections of the frequency spectrum, e.g., Stanford RW (Luong et al., 2013) only contains rare words which make its median frequency to be 5,683, where as WS-353 (Rel) (Finkelstein et al., 2001) contains many more common words, with a median frequency of 64,490. We also observe that we we achieve a much better score on other datasets which have low to median frequency words, e.g. MC-30, MEN-Tr-3K, and RG-65, all with median frequency less than 25k. The order they appear in the table and the subsequent plots is lowest to highest frequency, left to right. Please refer to Appendix B for details.

In figure 1, we see that WORD2BOX outperforms WORD2VEC more significantly with less common words. In order to investigate further, we selected four datasets (RW-Stanford (rare words), Simelex-999, SimVerb-3500,WS-353 (Rel)), truncated them at a frequency threshold, and calculated the correlation for different levels of this threshold. In Figure 2, we demonstrate how the performance gap between WORD2BOX and WORD2VEC changes as increasing amount frequent words are added to these similarity datasets. We posit that the geometry of box embeddings is more flexible in the way it handles sets of mutually disjoint words (such as rare words) which all co-occur with a more common word. Boxes have exponentially many corners, relative to their dimension, allowing extreme flexibility in the possible arrangements of intersection to achieve complicated co-occurrance models.

6.2 Set Theoretic Operations

All the senses, contexts and abstractions of a word can not be captured captured accurately using a point vector, and must be captured with sets. In

	Stanford RW	RG-65	YP-130	MEN	MC-30	Mturk-287	SimVerb-3500	SimLex-999	Mturk-771	WS-353 (Sim)	WS-353 (All)	WS-353 (Rel)	VERB-143
*Poincaré	—	75.97	—	—	80.46	—	18.90	31.81	—	—	62.34	—	—
*Gaussian	—	71.00	41.50	71.31	70.41	—	—	32.23	—	76.15	65.49	58.96	—
WORD2VEC	40.25	66.80	43.77	68.45	75.57	61.83	23.58	37.30	59.90	75.81	**69.01**	**61.29**	31.97
WORD2BOX	**45.08**	**81.45**	**51.6**	**73.68**	**87.12**	**70.62**	**29.71**	**38.19**	**68.51**	**78.60**	68.68	60.34	**48.03**

Table 1: Similarity: We evaluate our box embedding model WORD2BOX against a standard vector baseline WORD2VEC. For comparison, we also include the reported results for Gaussian and Poincaré embeddings, however we note that these may not be directly comparable as many other aspects (eg. corpus, vocab size, sampling method, training process, etc.) may be different between these models.

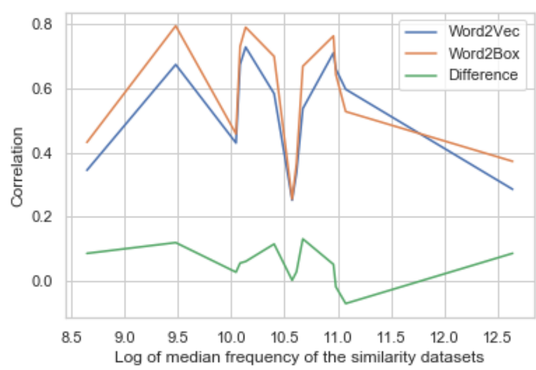

Figure 1: This plot depicts the gain in correlation score for WORD2BOX against WORD2VEC is much higher for the low and mid frequency range.

Vector \ Box	$A \cap B$	$A \setminus B$	$A \cup B$
Addition	0.90	0.92	0.98
Subtraction	0.90	0.65	0.80
Max Pooling	0.95	0.86	0.86
Min Pooling	0.90	0.75	0.92
Score Max Pooling	0.95	0.84	0.94
Score Min Pooling	1.0	0.80	0.84

Table 2: Percentage of times the Box Embeddings set operations are better than different vector operations. Thus more than 0.5 means that boxes are better. The Intersection, Union and Difference can be performed with Boxes as they originally are, however, we choose an exhaustive list of similar vector operations.

this section, we evaluate our models capability of representing sets by performing set operations on the trained models.

6.2.1 Quantitative Results

Homographs, words with identical spelling but distinct meanings, and polysemous words are ideal choice of stimuli for this purpose. We constructed set theoretic logical operations on words based on common polysemous words and homographs (Nelson et al., 1980). For example, the word 'property' will have association with words related both "asset' and 'attribute', and thus the union of the later two should be close to the original 'word' property. Likewise, intersection set of 'property' and 'math' should contain many words related to properties of algebra and geometry. Our dataset consists of triples (A, B, C) where $A \circ B$ should yield a set similar to C. In this task, given two words A and B and a set theoretic operation $\circ$, we try to find the rank of word C in the sorted list based on the set similarity (vector similarity scores for the vectors) score between $A \circ B$ and all words in the vocab. The dataset consists of 52 examples for both Union and Negation, 20 examples for Intersection. The details of the dataset can be found in appendix B. In table 2, we report the percentage of times the WORD2BOX outperformes WORD2VEC, i.e., the model yields better rank for the word C. Note that, it is not evidently clear how to design the union, difference or the intersection operations with vectors. Thus, in this work, we compare with a comprehensive list of operations for them. We observe that almost of all the values are more than 0.9, which means WORD2BOX gets better rank for 90 out of 100 examples. This empirically validates that our model is indeed capturing the underlying set theoretic aspects of the words in the corpus.

Here, the addition, subtraction, max pool, min pool are point wise vector operations between vector for word A and B. We also propose score max and score min operations where, we select the $\max(A \cdot X, B \cdot X)$ and $\min(A \cdot X, B \cdot X)$, where X is any word. The purpose of this design of operation if to mimic the essence of union and intersection in the vector space, however, it is evident that the trained vector geometry is not harmonious to this construction as well.

6.2.2 Qualitative Analysis

In this section, we present some interesting examples of set theoretic queries on words, with different degrees of complexities. For all the tables in this section, we perform the set-operations on the query words then look at the ranked list of most similar

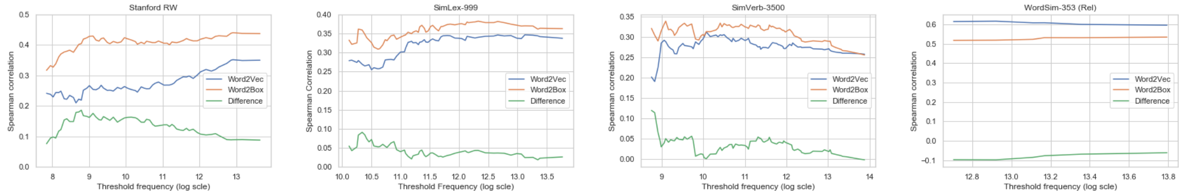

Figure 2: We plot the Spearman's correlation score vs Threshold frequency in log scale for Stanford RW, Simelex-999 SimVerb-3500, WS-353 (Rel). The correlation value is calculated on the word pairs where both of them have frequency less than the threshold frequency.

words to the output query. Many of these queries are based on words with multiple senses which is very instrumental for the inspection of the models.

Evidently, our the results from WORD2BOX look much better. Note that, from table, we observe that set difference of 'property' and 'land' yields a set of words that are related to attributes of science subjects, they are mostly "chemical-property" , "algebraic-property" etc. Thus, we wanted to examine how to this resulting query of 'property' - 'finance', relate to algebra and chemistry. We observe that the outputs indeed correspond to properties of those sub fields of science. We can observe such consistency of WORD2BOX with all the example logical queries.

A	B	Model	Operation	X
girl	boy	WORD2BOX	$A \cap B \cap X$	kid girls schoolgirl teenager woman boys child baby teenage orphan
		WORD2VEC	$(A + B) \cdot X$	shoeshine nanoha soulja schoolgirl yeller beastie jeezy crudup 'girl rahne
property	burial	WORD2BOX	$A \cap B \cap X$	cemetery bury estate grave interment tomb dwelling site gravesite sarcophagus
		WORD2VEC	$(A + B) \cdot X$	interment moated interred dunams ceteris burials catafalque easement deeded inhumation
	historical	WORD2BOX	$A \cap B \cap X$	historic estate artifact archaeological preserve ownership patrimony heritage landmark site
		WORD2VEC	$(A + B) \cdot X$	krajobrazowy burgage easement kravis dilapidation tohono intangible domesday moated laertius
	house	WORD2BOX	$A \cap B \cap X$	estate mansion manor residence houses tenement building premise buildings site
		WORD2VEC	$(A + B) \cdot X$	leasehold mansion tenements outbuildings estate burgage bedrooms moated burgesses manor
tongue	body	WORD2BOX	$A \cap B \cap X$	eye mouth ear limb lip forehead anus neck finger penis
		WORD2VEC	$(A + B) \cdot X$	tubercle ribcage meatus diverticulum forelegs radula tuberosity elastin foramen nostrils
	language	WORD2BOX	$A \cap B \cap X$	dialect idiom pronunciation meaning cognate word accent colloquial speaking speak
		WORD2VEC	$(A + B) \cdot X$	fluently dialects vowels patois languages loanwords phonology lingala tigrinya fluent

A	B	Model	Operation	X
algebra	finance	WORD2BOX	$(A\ \ B) \cap X$	homomorphism isomorphism automorphism abelian algebraic bilinear topological morphism spinor homeomorphism
		WORD2VEC	$(A - B) \cdot X$	homeomorphic unital homomorphisms nilpotent algebraically projective holomorphic propositional nondegenerate endomorphism
bank	finance	WORD2BOX	$(A\ \ B) \cap X$	wensum junction neman mouth tributary downstream corner embankment forks sandwich
		WORD2VEC	$(A - B) \cdot X$	shaddai takla thrombus gauley paria epenthetic chibchan urubamba foremast bolshaya
	river	WORD2BOX	$(A\ \ B) \cap X$	barclays hsbc banking citigroup citibank firm ipo brokerage interbank kpmg
		WORD2VEC	$(A - B) \cdot X$	cheques tymoshenko receivables citibank eurozone brinks defrauded courtaulds refinance mortgage
chemistry	finance	WORD2BOX	$(A\ \ B) \cap X$	biochemistry superconductor physics physic eutectic heat isotope fluorescence yttrium spectroscopy
		WORD2VEC	$(A - B) \cdot X$	augite alkyne desorption phosphorylating dimorphism fumarate hypertrophic empedocles hydratase enantiomer
property	land	WORD2BOX	$(A\ \ B) \cap X$	homotopy isomorphism involution register bijection symplectic eigenvalue idempotent compactification lattice
		WORD2VEC	$(A - B) \cdot X$	brst stieltjes l'p repressor absurdum doesn conjugates nonempty didn wouldn

A	B	C	Model	Operation	X
property	finance	algebra	WORD2BOX	$((A \setminus B) \cap C) \cap X$	laplacian nilpotent antiderivative lattice surjective automorphism invertible homotopy integer integrand
			WORD2VEC	$(A - B + C) \cdot X$	expropriate extort refco underwrite reimburse refinance parmalat refinancing brokerage privatizing
		chemistry	WORD2BOX	$((A \setminus B) \cap C) \cap X$	eutectic desiccant allotrope phenocryst hardness solubility monoclinic hygroscopic nepheline trehalose
			WORD2VEC	$(A - B + C) \cdot X$	refinance brokerage burgage stockbroking refinancing warranties reimburse madoff privatizing valorem

7 Related Work

Learning distributional vector representations from a raw corpus was introduced in Mikolov et al. (2013), quickly followed by various improvements (Pennington et al.; Bojanowski et al., 2017). More recently, vector representations which incorporate contextual information have shown significant im-

bank ∩ finance	bank ∪ finance	bank \ finance	bank + finance	bank - finance	max(bank, finance)	min(bank, finance)	max_score(bank, finance)	min_score(bank, finance)
investment	banking	wensum	subprime	shaddai	refinance	securities	refinance	securities
banking	treasury	junction	securities	takla	laundering	subprime	laundering	subprime
investor	investor	neman	refinance	thrombus	reimbursements	jpmorgan	reimbursements	jpmorgan
financing	investment	mouth	liquidity	gauley	superannuation	citigroup	superannuation	citigroup
fund	business	tributary	laundering	paria	liquidity	equities	liquidity	equities
government	economy	downstream	kaupthing	epenthetic	debit	ebrd	debit	ebrd
corporation	management	corner	underwrite	chibchan	controllata	kaupthing	controllata	kaupthing
treasury	firm	embankment	receivables	urubamba	subprime	mortgage	subprime	mortgage
citigroup	fund	forks	ibrd	foremast	underwrite	refinance	underwrite	refinance
firm	financial	sandwich	equities	bolshaya	disbursement	debentures	disbursement	debentures

Word	Model	Similarity
bank	WORD2BOX	population median age female race family poverty every career census
	WORD2VEC	debit depositors securities kaupthing interbank subprime counterparty citibank fdic nasdaq
economics	WORD2BOX	population median age female race family poverty every career census
	WORD2VEC	microeconomic keynesian microeconomics minored macroeconomics econometrics sociology thermodynamics evolutionism structuralist
microeconomics	WORD2BOX	population median age female race family poverty every career census
	WORD2VEC	microeconomic initio germline instantiation zachman macroeconomics oxoglutarate glycemic noncommutative pubmed
property	WORD2BOX	population median age female race family poverty every career census
	WORD2VEC	easement infringes burgage krajobrazowy chattels policyholder leasehold intestate liabilities ceteris
rock	WORD2BOX	population median age female race family poverty every career census
	WORD2VEC	shoegaze rhyolitic punk britpop mafic outcrops metalcore bluesy sedimentary quartzite

provements (Peters et al., 2018; Devlin et al., 2019; Radford et al., 2019; Brown et al., 2020). As these models require context, however, Word2Vec-style approaches are still relevant in settings where such context is unavailable.

Hyperbolic representations (Nickel and Kiela, 2017; Ganea et al., 2018; Chamberlain et al., 2017) have become popular in recent years. Most related to our setting, Tifrea et al. (2019) propose a hyperbolic analog to GloVe, with the motivation that the hyperbolic embeddings will discover a latent hierarchical structure between words.[3] Vilnis and McCallum (2015) use Gaussian distributions to represent each word, and KL Divergence as a score function. [4] Athiwaratkun and Wilson (2018) extended such representations by adding certain thresholds for each distribution. For a different purpose, Ren and Leskovec (2020) use Beta Distributions to model logical operations between words. Our work can be seen as a region-based analog to these models.

Of the region-based embeddings, Suzuki et al. (2019) uses hyperbolic disks, and Ganea et al. (2018) uses hyperbolic cones, however these are not closed under intersection nor are their intersections easily computable. Vendrov et al. (2016) and Lai and Hockenmaier (2017) use an axis-aligned cone to represent a specific relation between words/sentences, for example an entailment relation. Vilnis et al. (2018) extends Lai and Hockenmaier (2017) by adding an upper-bound, provably increasing the representational capacity of the model. Li et al. (2019) and Dasgupta et al. (2020) are improved training methods to handle the difficulties inherent in gradient-descent based region learning. Ren et al. (2020) and Abboud et al. (2020) use a box-based adjustment of their loss functions, which suggest learning per-entity thresholds are beneficial. (Chen et al., 2021) use box embeddings to model uncertain knowledge graphs, and (Onoe et al., 2021) use boxes for fined grained entity typing.

8 Conclusion

In this work we have demonstrated that box embeddings can not only effectively train to represent pairwise similarity but also the it can capture the rich set theoretic logical structure of the words. The expressivity of box models allows them to capture cooccurrances is such a distributed set theoretic way which is inaccessible to vector models.

[3]Reported results are included in table 1 as “Poincaré”

[4]Reported results are included in table 1 as “Gaussian”

References

Ralph Abboud, İsmail İlkan Ceylan, Thomas Lukasiewicz, and Tommaso Salvatori. 2020. Boxe: A box embedding model for knowledge base completion. In *Proceedings of the 34th Annual Conference on Neural Information Processing Systems, NeurIPS*.

Ben Athiwaratkun and Andrew Gordon Wilson. 2018. Hierarchical density order embeddings. In *International Conference on Learning Representations*.

Simon Baker, Roi Reichart, and Anna Korhonen. 2014. An unsupervised model for instance level subcategorization acquisition. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 278–289, Doha, Qatar. Association for Computational Linguistics.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 23–32, Avignon, France. Association for Computational Linguistics.

Marco Baroni, Silvia Bernardini, A. Ferraresi, and E. Zanchetta. 2009. The wacky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43:209–226.

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.

Michael Boratko, Javier Burroni, Shib Sankar Dasgupta, and Andrew McCallum. Min/Max Stability and Box Distributions. page 10.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. In *Advances in Neural Information Processing Systems*.

Elia Bruni, Nam Khanh Tran, and Marco Baroni. 2014. Multimodal distributional semantics. *J. Artif. Int. Res.*, 49(1):1–47.

Benjamin Paul Chamberlain, James R. Clough, and Marc Peter Deisenroth. 2017. Neural embeddings of graphs in hyperbolic space. *ArXiv*, abs/1705.10359.

Xuelu Chen, Michael Boratko, Muhao Chen, Shib Sankar Dasgupta, Xiang Lorraine Li, and Andrew McCallum. 2021. Probabilistic box embeddings for uncertain knowledge graph reasoning. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*.

Shib Sankar Dasgupta, Michael Boratko, Dongxu Zhang, Luke Vilnis, Xiang Lorraine Li, and Andrew McCallum. 2020. Improving local identifiability in probabilistic box embeddings. In *Advances in Neural Information Processing Systems*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota. Association for Computational Linguistics.

Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*, pages 406–414.

Octavian-Eugen Ganea, Gary Bécigneul, and Thomas Hofmann. 2018. Hyperbolic entailment cones for learning hierarchical embeddings. In *International Conference on Machine Learning*.

Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. KDD '12, page 1406–1414, New York, NY, USA. Association for Computing Machinery.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695.

George J Klir and Bo Yuan. 1996. Fuzzy sets, fuzzy logic, and fuzzy systems: selected papers by lotfi a. zadeh.

Alice Lai and Julia Hockenmaier. 2017. Learning to predict denotational probabilities for modeling entailment. In *EACL*.

E.T. Lee and L.A. Zadeh. 1969. Note on fuzzy languages. *Information Sciences*, 1(4):421–434.

Xiang Li, Luke Vilnis, Dongxu Zhang, Michael Boratko, and Andrew McCallum. 2019. Smoothing the geometry of probabilistic box embeddings. *ICLR*.

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, Sofia, Bulgaria. Association for Computational Linguistics.

Francois Meyer and Martha Lewis. 2020. Modelling lexical ambiguity with density matrices. pages 276–290.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *NIPS*.

George A Miller and Walter G Charles. 1991. Contextual correlates of semantic similarity. *Language & Cognitive Processes*, 6(1):1–28.

Douglas Nelson, Cathy Mcevoy, John Walling, and Joseph Wheeler. 1980. The university of south florida homograph norms. *Behavior research methods, instruments, computers: a journal of the Psychonomic Society, Inc*, 12:16–37.

Maximilian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Neural Information Processing Systems*.

Yasumasa Onoe, Michael Boratko, and Greg Durrett. 2021. Modeling fine-grained entity types with box embeddings. *Association for Computational Linguistics*.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. New Orleans, Louisiana. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Kira Radinsky, Eugene Agichtein, Evgeniy Gabrilovich, and Shaul Markovitch. 2011. A word at a time: Computing word relatedness using temporal semantic analysis. WWW '11, page 337–346, New York, NY, USA. Association for Computing Machinery.

Hongyu Ren, Weihua Hu, and Jure Leskovec. 2020. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *8th International Conference on Learning Representations*. OpenReview.net.

Hongyu Ren and Jure Leskovec. 2020. Beta embeddings for multi-hop logical reasoning in knowledge graphs. *arXiv preprint arXiv:2010.11465*.

Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Commun. ACM*, 8(10):627–633.

Ryota Suzuki, Ryusuke Takahama, and Shun Onoda. 2019. Hyperbolic disk embeddings for directed acyclic graphs. In *International Conference on Machine Learning*, pages 6066–6075. PMLR.

Alexandru Tifrea, Gary Becigneul, and Octavian-Eugen Ganea. 2019. Poincare glove: Hyperbolic word embeddings. In *International Conference on Learning Representations*.

Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2016. Order-embeddings of images and language. In *International Conference on Learning Representations*.

Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. 2018. Probabilistic embedding of knowledge graphs with box lattice measures. In *Association for Computational Linguistics*.

Luke Vilnis and Andrew McCallum. 2015. Word representations via gaussian embedding. In *ICLR*.

Dongqiang Yang and David M. W. Powers. 2006. Verb similarity on the taxonomy of wordnet. In *In the 3rd International WordNet Conference (GWC-06), Jeju Island, Korea*.

L.A. Zadeh. 1965. Fuzzy sets. *Information and Control*, 8(3):338–353.

Vitalii Zhelezniak, Aleksandar Savkov, April Shen, Francesco Moramarco, Jack Flann, and Nils Y. Hammerla. 2019. Don't settle for average, go for the max: Fuzzy sets and max-pooled word vectors. In *International Conference on Learning Representations*.

A Preprocessing

The WaCKypedia corpus has been tokenized and lemmatized. We used the lemmatized version of the corpus, however it was observed that various tokens were not split as they should have been (eg. “1.5billion” -> “1.5 billion”). We split tokens using regex criteria to identify words and numbers. All punctuation was removed from the corpus, all numbers were replaced with a “<num>” token, and all words were made lowercase. We also removed any words which included non-ascii symbols. After this step, the entire corpus was tokenized once more, and any token occurring less than 100 times was dropped.

B Dataset Analysis

Dataset	Median Frequency
Men-Tr-3K	23942
Mc-30	25216.5
Mturk-771	43128.5
Simlex-999	40653.0
Verb-143	309192.0
Yp-130	23044.0
Rw-Stanford	5683.5
Rg-65	13088.0
Ws-353-All	58803.0
Ws-353-Sim.	57514.0
Ws-353-Rel	64490.0
Mturk-287	32952
Simverb-3500	39020

Table 3: Median Frequency of each similarity dataset.

C Hyperparameters

As discussed in Section 6, we train on 128 dimensional WORD2VEC and 64 dimensional WORD2BOX models for 10 epochs. We ran at least 60 runs for each of the models with random seed and randomly chose hyperparamter from the following range - batch_size:[2048, 4096, 8192, 16384, 32768], learning rate log_uniform[exp(-1), exp(-10)], Window_size: [5, 6, 7, 8, 9, 10], negative_samples: [2, 5, 10, 20], sub_sampling threshold: [0.001, 0.0001].

D Set Theoretic Queries

A	B	AB
table	chair	furniture
car	plane	transportation
city	village	location
wolf	bear	animal
shirt	pant	clothes
computer	phone	Electronics
red	blue	color
movie	book	entertainment
school	college	education
doctor	engineer	Profession
box	circle	shape
big	small	size
dog	tree	bark
fish	tone	bass
sports	wing	bat
carry	animal	bear
sadness	color	blue
bend	weapon	bow
hit	food	buffet
combine	building	compound
happy	list	content
acquire	agreement	contract
location	organise	coordinate
hot	leave	desert
information	food	digest
furry	lower	down
entry	bewitch	entrance
exhibition	judgement	fair
good	charge	fine
luck	whale	fluke
odor	angry	incense
crotch	race	lap
thin	slant	lean
sleep	wrong	lie
broadcast	life	live
small	time	minute
overlook	woman	miss
thing	oppose	object
target	thing	object
air	turn	wind
category	keyboard	type
mercy	type	kind
truck	teach	train
topic	impose	subject
jump	miss	skip
first	time	second
move	drink	shake
surface	ordinary	plain
bravery	remove	pluck
luggage	beer	porter
create	vegetables	produce
rise	flower	rose