Intrinsic Subspace Evaluation of Word Embedding Representations

Yadollah Yaghoobzadeh and **Hinrich Schütze**
Center for Information and Language Processing
University of Munich, Germany
yadollah@cis.lmu.de

Abstract

We introduce a new methodology for intrinsic evaluation of word representations. Specifically, we identify four fundamental criteria based on the characteristics of natural language that pose difficulties to NLP systems; and develop tests that directly show whether or not representations contain *the subspaces necessary to satisfy these criteria*. Current intrinsic evaluations are mostly based on the overall similarity or *full-space similarity* of words and thus view vector representations as *points*. We show the limits of these point-based intrinsic evaluations. We apply our evaluation methodology to the comparison of a count vector model and several neural network models and demonstrate important properties of these models.

1 Introduction

Distributional word representations or *embeddings* are currently an active area of research in natural language processing (NLP). The motivation for embeddings is that knowledge about words is helpful in NLP. Representing words as vocabulary indexes may be a good approach if large training sets allow us to learn everything we need to know about a word to solve a particular task; but in most cases it helps to have a representation that contains distributional information and allows inferences like: "above" and "below" have similar syntactic behavior or "engine" and "motor" have similar meaning.

Several methods have been introduced to assess the quality of word embeddings. We distinguish two different types of evaluation in this paper: (i) *extrinsic evaluation* evaluates embeddings in an NLP application or task and (ii) *intrinsic evaluation* tests the quality of representations independent of a specific NLP task.

Each single word is a combination of a large number of morphological, lexical, syntactic, semantic, discourse and other features. Its embedding should accurately and consistently represent these features, and ideally a good evaluation method must clarify this and give a way to analyze the results. The goal of this paper is to build such an evaluation.

Extrinsic evaluation is a valid methodology, but it does not allow us to understand the properties of representations without further analysis; e.g., if an evaluation shows that embedding A works better than embedding B on a task, then that is not an analysis of the causes of the improvement. Therefore, extrinsic evaluations do not satisfy our goals.

Intrinsic evaluation analyzes the generic quality of embeddings. Currently, this evaluation mostly is done by testing overall distance/similarity of words in the embedding space, i.e., it is based on viewing word representations as points and then computing *full-space similarity*. The assumption is that the high dimensional space is smooth and similar words are close to each other. Several datasets have been developed for this purpose, mostly the result of human judgement; see (Baroni et al., 2014) for an overview. We refer to these evaluations as *point-based* and as *full-space* because they consider embeddings as points in the space – sub-similarities in subspaces are generally ignored.

Point-based intrinsic evaluation computes a score based on the full-space similarity of two words: a single number that generally does not say anything about the underlying reasons for a lower or higher value of full-space similarity. This makes it hard to interpret the results of point-based evaluation and may be the reason that contradictory results have been published; e.g., based on

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 236–246,
Berlin, Germany, August 7-12, 2016.

point-based evaluation, some papers have claimed that count-based representations perform as well as learning-based representations (Levy and Goldberg, 2014a). Others have claimed the opposite (e.g., Mikolov et al. (2013), Pennington et al. (2014), Baroni et al. (2014)).

Given the limits of current evaluations, we propose a new methodology for intrinsic evaluation of embeddings by identifying generic fundamental criteria for embedding models that are important for representing features of words accurately and consistently. We develop corpus-based tests using supervised classification that directly show whether the representations contain the information necessary to meet the criteria or not. The fine-grained corpus-based supervision makes the sub-similarities of words important by looking at the subspaces of word embeddings relevant to the criteria, and this enables us to give direct insights into properties of representation models.

2 Related Work

Baroni et al. (2014) evaluate embeddings on different intrinsic tests: similarity, analogy, synonym detection, categorization and selectional preference. Schnabel et al. (2015) introduce tasks with more fine-grained datasets. These tasks are unsupervised and generally based on cosine similarity; this means that only the overall direction of vectors is considered or, equivalently, that *words are modeled as points* in a space and only their full-space distance/closeness is considered. In contrast, we test embeddings in a classification setting and *different subspaces of embeddings are analyzed*. Tsvetkov et al. (2015) evaluate embeddings based on their correlations with WordNet-based linguistic embeddings. However, correlation does not directly evaluate how accurately and completely an application can extract a particular piece of information from an embedding.

Extrinsic evaluations are also common (cf. (Li and Jurafsky, 2015; Köhn, 2015; Lai et al., 2015)). Li and Jurafsky (2015) conclude that embedding evaluation must go beyond human-judgement tasks like similarity and analogy. They suggest to evaluate on NLP tasks. Köhn (2015) gives similar suggestions and also recommends the use of supervised methods for evaluation. Lai et al. (2015) evaluate embeddings in different tasks with different setups and show the contradictory results of embedding models on different tasks. Idiosyncrasies of different downstream tasks can affect extrinsic evaluations and result in contradictions.

3 Criteria for word representations

Each word is a combination of different properties. Depending on the language, these properties include lexical, syntactic, semantic, world knowledge and other features. We call these properties *facets*. The ultimate goal is to learn representations for words that accurately and consistently contain these facets. Take the facet gender (GEN) as an example. We call a representation 100% *accurate* for GEN if information it contains about GEN is always accurate; we call the representation 100% *consistent* for GEN if the representation of every word that has a GEN facet contains this information.

We now introduce four important criteria that a representation must satisfy to represent facets accurately and consistently. These criteria are applied across different problems that NLP applications face in the effective use of embeddings.

Nonconflation. A word embedding must keep the evidence from different local contexts separate – "do not conflate" – because each context can infer specific facets of the word. Embeddings for different word forms with the same stem, like plural and singular forms or different verb tenses, are examples vulnerable to conflation because they occur in similar contexts.

Robustness against sparseness. One aspect of natural language that poses great difficulty for statistical modeling is sparseness. Rare words are common in natural language and embedding models must learn useful representations based on a small number of contexts.

Robustness against ambiguity. Another central problem when processing words in NLP is lexical ambiguity (Cruse, 1986; Zhong and Ng, 2010). Polysemy and homonymy of words can make it difficult for a statistical approach to generalize and infer well. Embeddings should fully represent all senses of an ambiguous word. This criterion becomes more difficult to satisfy as distributions of senses become more skewed, but a robust model must be able to overcome this.

Accurate and consistent representation of multifacetedness. This criterion addresses settings with large numbers of facets. It is based on the following linguistic phenomenon, a phenomenon that occurs frequently crosslinguistically

(Comrie, 1989). (i) Words have a large number of facets, including phonetic, morphological, syntactic, semantic and topical properties. (ii) Each facet by itself constitutes a small part of the overall information that a representation should capture about a word.

4 Experimental setup and results

We now design experiments to directly evaluate embeddings on the four criteria. We proceed as follows. First, we design a probabilistic context free grammar (PCFG) that generates a corpus that is a manifestation of the underlying phenomenon. Then we train our embedding models on the corpus. The embeddings obtained are then evaluated in a classification setting, in which we apply a linear SVM (Fan et al., 2008) to classify embeddings. Finally, we compare the classification results for different embedding models and analyze and summarize them.

Selecting embedding models. Since this paper is about developing a new evaluation methodology, the choice of models is not important as long as the models can serve to show that the proposed methodology reveals interesting differences with respect to the criteria.

On the highest level, we can distinguish two types of distributional representations. *Count vectors* (Sahlgren, 2006; Baroni and Lenci, 2010; Turney and Pantel, 2010) live in a high-dimensional vector space in which each dimension roughly corresponds to a (weighted) count of cooccurrence in a large corpus. *Learned vectors* are learned from large corpora using machine learning methods: unsupervised methods such as LSI (e.g., Deerwester et al. (1990), Levy and Goldberg (2014b)) and supervised methods such as neural networks (e.g., Mikolov et al. (2013)) and regression (e.g., Pennington et al. (2014)). Because of the recent popularity of learning-based methods, we consider one count-based and five learning-based distributional representation models.

The learning-based models are: (i) vLBL (henceforth: LBL) (vectorized log-bilinear language model) (Mnih and Kavukcuoglu, 2013), (ii) SkipGram (henceforth: SKIP) (skipgram bag-of-word model), (iii) CBOW (continuous bag-of-word model (Mikolov et al., 2013), (iv) Structured SkipGram (henceforth SSKIP), (Ling et al., 2015) and CWindow (henceforth CWIN) (continuous window model) (Ling et al., 2015). These models learn word embeddings for input and target spaces using neural network models.

1	$P(aVb\|S)$	=	1/4	
2	$P(bVa\|S)$	=	1/4	
3	$P(aWa\|S)$	=	1/8	
4	$P(aWb\|S)$	=	1/8	
5	$P(bWa\|S)$	=	1/8	
6	$P(bWb\|S)$	=	1/8	
7	$P(v_i\|V)$	=	1/5	$0 \leq i \leq 4$
8	$P(w_i\|W)$	=	1/5	$0 \leq i \leq 4$

Figure 1: Global conflation grammar. Words v_i occur in a subset of the contexts of words w_i, but the global count vector signatures are the same.

For a given context, represented by the input space representations of the left and right neighbors $\vec{v}_{i-1}$ and $\vec{v}_{i+1}$, LBL, CBOW and CWIN predict the target space $\vec{v}_i$ by combining the contexts. LBL combines $\vec{v}_{i-1}$ and $\vec{v}_{i+1}$ linearly with position dependent weights and CBOW (resp. CWIN) combines them by adding (resp. concatenation). SKIP and SSKIP predict the context words v_{i-1} or v_{i+1} given the input space $\vec{v}_i$. For SSKIP, context words are in different spaces depending on their position to the input word. In summary, CBOW and SKIP are learning embeddings using bag-of-word (BoW) models, but the other three, CWIN, SSKIP and LBL, are using position dependent models. We use `word2vec`[1] for SKIP and CBOW, `wang2vec`[2] for SSKIP and CWIN, and Lai et al. (2015)'s implementation[3] for LBL.

The count-based model is position-sensitive PPMI, Levy and Goldberg (2014a)'s explicit vector space representation model.[4] For a vocabulary of size V, the representation $\vec{w}$ of w is a vector of size $4V$, consisting of four parts corresponding to the relative positions $r \in \{-2, -1, 1, 2\}$ with respect to occurrences of w in the corpus. The entry for dimension word v in the part of $\vec{w}$ corresponding to relative position r is the PPMI (positive pointwise mutual information) weight of w and v for that relative position. The four parts of the vector are length normalized. In this paper, we use only two relative positions: $r \in \{-1, 1\}$, so each $\vec{w}$ has two parts, corresponding to immediate left and right neighbors.

[1] `code.google.com/archive/p/word2vec`
[2] `github.com/wlin12/wang2vec`
[3] `github.com/licstar/compare`
[4] `bitbucket.org/omerlevy/hyperwords`

4.1 Nonconflation

Grammar. The PCFG grammar shown in Figure 1 generates v_i words that occur in two types of contexts: a-b (line 1) and b-a (line 2); and w_i words that also occur in these two contexts (lines 4 and 5), but in addition occur in a-a (line 3) and b-b (line 6) contexts. As a result, the set of contexts in which v_i and w_i occur is different, but if we simply count the number of occurrences in the contexts, then v_i and w_i cannot be distinguished.

Dataset. We generated a corpus of 100,000 sentences. Words that can occur in a-a and b-b contexts constitute the positive class, all other words the negative class. The words v_3, v_4, w_3, w_4 were assigned to the test set, all other words to the training set.

Results. We learn representations of words by our six models and train one SVM per model; it takes a word representation as input and outputs +1 (word can occur in a-a/b-b) or -1 (it cannot). The SVMs trained on PPMI and CBOW representations assigned all four test set words to the negative class; in particular, w_3 and w_4 were incorrectly classified. Thus, the accuracy of classification for these models (50%) was not better than random. The SVMs trained on LBL, SSKIP, SSKIP and CWIN representations assigned all four test set words to the correct class: v_3 and v_4 were assigned to the negative class and w_3 and w_4 were assigned to the positive class.

Discussion. The property of embedding models that is relevant here is that PPMI is an *aggregation model*, which means it calculates aggregate statistics for each word and then computes the final word embedding from these aggregate statistics. In contrast, all our learning-based models are *iterative models*: they iterate over the corpus and each local context of a word is used as a training instance for learning its embedding.

For iterative models, it is common to use composition of words in the context, as in LBL, CBOW and CWIN. Non-compositional iterative models like SKIP and SSKIP are also popular. Aggregation models can also use composite features from context words, but these features are too sparse to be useful. The reason that the model of Agirre et al. (2009) is rarely used is precisely its inability to deal with sparseness. All widely used distributional models employ individual word occurrences as basic features.

The bad PPMI results are explained by the fact

1	$P(AVB\|S)$	=	1/2	
2	$P(CWD\|S)$	=	1/2	
3	$P(a_i\|A)$	=	1/10	$0 \leq i \leq 9$
4	$P(b_i\|B)$	=	1/10	$0 \leq i \leq 9$
5	$P(c_i\|C)$	=	1/10	$0 \leq i \leq 9$
6	$P(d_i\|D)$	=	1/10	$0 \leq i \leq 9$
7	$P(v_i\|V)$	=	1/10	$0 \leq i \leq 9$
8	$P(w_i\|W)$	=	1/10	$0 \leq i \leq 9$
9	$L' =$	$L(S)$		
10		$\cup \{a_i u_i b_i \| 0 \leq i \leq 9\}$		
11		$\cup \{c_i x_i d_i \| 0 \leq i \leq 9\}$		

Figure 2: In language L', frequent v_i and rare u_i occur in a-b contexts; frequent w_i and rare x_i occur in c-d contexts. Word representations should encode possible contexts (a-b vs. c-d) for both frequent and rare words.

that it is an aggregation model: the PPMI model cannot distinguish two words with the same global statistics – as is the case for, say, v_3 and w_3. The bad result of CBOW is probably connected to its weak (addition) composition of context, although it is an iterative compositional model. Simple representation of context words with iterative updating (through backpropagation in each training instance), can influence the embeddings in a way that SKIP and SSKIP get good results, although they are non-compositional.

As an example of conflation occurring in the English Wikipedia, consider this simple example. We replace all single digits by "7" in tokenization. We learn PPMI embeddings for the tokens and see that among the one hundred nearest neighbors of "7" are the days of the week, e.g., "Friday". As an example of a conflated feature consider the word "falls" occurring immediately to the right of the target word. The weekdays as well as single digits often have the immediate right neighbor "falls" in contexts like "Friday falls on a public holiday" and "2 out of 3 falls match" – tokenized as "7 out of 7 falls match" – in World Wrestling Entertainment (WWE). The left contexts of "Friday" and "7" are different in these contexts, but the PPMI model does not record this information in a way that would make the link to "falls" clear.

4.2 Robustness against sparseness

Grammar. The grammar shown in Figure 2 generates frequent v_i and rare u_i in a-b contexts (lines 1 and 9); and frequent w_i and rare x_i in c-d contexts (lines 2 and 10). The language generated by the PCFG on lines 1–8 is merged on lines 9–11 with the ten contexts $a_0u_0b_0 \ldots a_9u_9b_9$ (line 9)

and the ten contexts $c_0x_0d_0 \ldots c_9x_9d_9$ (line 10); that is, each of the u_i and x_i occurs exactly once in the merged language L', thus modeling the phenomenon of sparseness.

Dataset. We generated a corpus of 100,000 sentences using the PCFG (lines 1–8) and added the 20 rare sentences (lines 9–11). We label all words that can occur in c-d contexts as positive and all other words as negative. The singleton words u_i and x_i were assigned to the test set, all other words to the training set.

Results. After learning embeddings with different models, the SVM trained on PPMI representations assigned all twenty test words to the negative class. This is the correct decision for the ten u_i (since they cannot occur in a c-d context), but the incorrect decision for the x_i (since they can occur in a c-d context). Thus, the accuracy of classification was 50% and not better than random. The SVMs trained on learning-based representations classified all twenty test words correctly.

Discussion. Representations of rare words in the PPMI model are sparse. The PPMI representations of the u_i and x_i only contain two nonzero entries, one entry for an a_i or c_i (left context) and one entry for a b_i or d_i (right context). Given this sparseness, it is not surprising that representations are not a good basis for generalization and PPMI accuracy is random.

In contrast, learning-based models learn that the a_i, b_i, c_i and d_i form four different distributional classes. The final embeddings of the a_i after learning is completed are all close to each other and the same is true for the other three classes. Once the similarity of two words in the same distributional class (say, the similarity of a_5 and a_7) has been learned, the contexts for the u_i (resp. x_i) look essentially the same to embedding models as the contexts of the v_i (resp. w_i). Thus, the embeddings learned for the u_i will be similar to those learned for the v_i. This explains why learning-based representations achieve perfect classification accuracy.

This sparseness experiment highlights an important difference between count vectors and learned vectors. Count vector models are less robust in the face of sparseness and noise because they base their representations on individual contexts; the overall corpus distribution is only weakly taken into account, by way of PPMI weighting. In contrast, learned vector models make much better use of the overall corpus distribution and they can leverage second-order effects for learning improved representations. In our example, the second order effect is that the model first learns representations for the a_i, b_i, c_i and d_i and then uses these as a basis for inferring the similarity of u_i to v_i and of x_i to w_i.

1	$P(AV_1B\|S)$	=10/20	
2	$P(CW_1D\|S)$	=9/20	
3	$P(CW_2D\|S)$	$=\beta\cdot$1/20	
4	$P(AW_2B\|S)$	$=(1-\beta)\cdot$1/20	
5	$P(a_i\|A)$	=1/10	$0 \leq i \leq 9$
6	$P(b_i\|B)$	=1/10	$0 \leq i \leq 9$
7	$P(c_i\|C)$	=1/10	$0 \leq i \leq 9$
8	$P(d_i\|D)$	=1/10	$0 \leq i \leq 9$
9	$P(v_i\|V_1)$	=1/50	$0 \leq i \leq 49$
10	$P(w_i\|W_1)$	=1/45	$5 \leq i \leq 49$
11	$P(w_i\|W_2)$	=1/5	$0 \leq i \leq 4$

Figure 3: Ambiguity grammar. v_i and $w_5 \ldots w_{49}$ occur in a-b and c-d contexts only, respectively. $w_0 \ldots w_4$ are ambiguous and occur in both contexts.

4.3 Robustness against ambiguity

Grammar. The grammar in Figure 3 generates two types of contexts that we interpret as two different meanings: a-b contexts (lines 1,4) and c-d contexts (lines 2, 3). v_i occur only in a-b contexts (line 1), $w_5 \ldots w_{49}$ occur only in c-d contexts (line 2); thus, they are unambiguous. $w_0 \ldots w_4$ are ambiguous and occur with probability β in c-d contexts (line 3) and with probability $(1-\beta)$ in a-b contexts (lines 3, 4). The parameter β controls the skewedness of the sense distribution; e.g., the two senses are equiprobable for $\beta = 0.5$ and the second sense (line 4) is three times as probable as the first sense (line 3) for $\beta = 0.25$.

Dataset. The grammar specified in Figure 3 was used to generate a training corpus of 100,000 sentences. Label criterion: A word is labeled positive if it can occur in a c-d context, as negative otherwise. The test set consists of the five ambiguous words $w_0 \ldots w_4$. All other words are assigned to the training set.

Linear SVMs were trained for the binary classification task on the train set. 50 trials of this experiment were run for each of eleven values of β: $\beta = 2^{-\alpha}$ where $\alpha \in \{1.0, 1.1, 1.2, \ldots, 2.0\}$. Thus, for the smallest value of α, $\alpha = 1.0$, the two senses have the same frequency; for the largest value of α, $\alpha = 2.0$, the dominant sense is three times as frequent as the less frequent sense.

Results. Figure 4 shows accuracy of the classi-

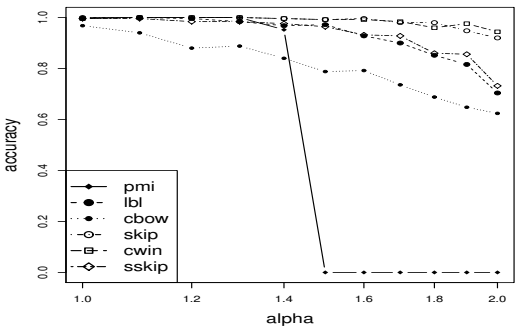

Figure 4: SVM classification results for the ambiguity dataset. X-axis: $\alpha = -\log_2 \beta$. Y-axis: classification accuracy:

fication on the test set: the proportion of correctly classified words out of a total of 250 (five words each in 50 trials).

All models perform well for balanced sense frequencies; e.g., for $\alpha = 1.0, \beta = 0.5$, the SVMs were all close to 100% accurate in predicting that the w_i can occur in a c-d context. PPMI accuracy falls steeply when α is increased from 1.4 to 1.5. It has a 100% error rate for $\alpha \geq 1.5$. Learning-based models perform better in the order CBOW (least robust), LBL, SSKIP, SKIP, CWIN (most robust). Even for $\alpha = 2.0$, CWIN and SKIP are still close to 100% accurate.

Discussion. The evaluation criterion we have used here is a classification task. The classifier attempts to answer a question that may occur in an application – can this word be used in this context? Thus, the evaluation criterion is: does the word representation contain a specific type of information that is needed for the application.

Another approach to ambiguity is to compute multiple representations for a word, one for each sense. We generally do not yet know what the sense of a word is when we want to use its word representation, so data-driven approaches like clustering have been used to create representations for different usage clusters of words that may capture some of its senses. For example, Reisinger and Mooney (2010) and Huang et al. (2012) cluster the contexts of each word and then learn a different representation for each cluster. The main motivation for this approach is the assumption that single-word distributional representations cannot represent all senses of a word well (Huang et al., 2012). However, Li and Jurafsky (2015) show that simply increasing the dimensionality of single-representation gets comparable results to using multiple-representation. Our results confirm that a single embedding can be robust against ambiguity, but also show the main challenge: skewness of sense distribution.

1	$P(NF_n\|S)$	=1/4	
2	$P(AF_a\|S)$	=1/4	
3	$P(NM_n\|S)$	=1/4	
4	$P(AM_f\|S)$	=1/4	
5	$P(n_i\|N)$	=1/5	$0 \leq i \leq 4$
6	$P(a_i\|A)$	=1/5	$0 \leq i \leq 4$
7	$P(x_i^{\text{nf}} U_i^{\text{nf}}\|F_n)$	=1/5	$0 \leq i \leq 4$
8	$P(f\|U_i^{\text{nf}})$	=1/2	
9	$P(\mu(U_i^{\text{nf}})\|U_i^{\text{nf}})$	=1/2	
10	$P(x_i^{\text{af}} U_i^{\text{af}}\|F_a)$	=1/5	$0 \leq i \leq 4$
11	$P(f\|U_i^{\text{af}})$	=1/2	
12	$P(\mu(U_i^{\text{af}})\|U_i^{\text{af}})$	=1/2	
13	$P(x_i^{\text{nm}} U_i^{\text{nm}}\|M_n)$	=1/5	$0 \leq i \leq 4$
14	$P(m\|U_i^{\text{nm}})$	=1/2	
15	$P(\mu(U_i^{\text{nm}})\|U_i^{\text{nm}})$	=1/2	
16	$P(x_i^{\text{am}} U_i^{\text{am}}\|M_f)$	=1/5	$0 \leq i \leq 4$
17	$P(m\|U_i^{\text{am}})$	=1/2	
18	$P(\mu(U_i^{\text{am}})\|U_i^{\text{am}})$	=1/2	

Figure 5: This grammar generates nouns ($x_i^{\text{n}\cdot}$) and adjectives ($x_i^{\text{a}\cdot}$) with masculine (x_i^{m}) and feminine (x_i^{f}) gender as well as paradigm features u_i. μ maps each U to one of $\{u_0 \ldots u_4\}$. μ is randomly initialized and then kept fixed.

4.4 Accurate and consistent representation of multifacetedness

Grammar. The grammar shown in Figure 5 models two syntactic categories, nouns and adjectives, whose left context is highly predictable: it is one of five left context words n_i (resp. a_i) for nouns, see lines 1, 3, 5 (resp. for adjectives, see lines 2, 4, 6). There are two grammatical genders: feminine (corresponding to the two symbols F_n and F_a) and masculine (corresponding to the two symbols M_n and M_a). The four combinations of syntactic category and gender are equally probable (lines 1–4). In addition to *gender*, nouns and adjectives are distinguished with respect to *morphological paradigm*. Line 7 generates one of five feminine nouns (x_i^{nf}) and the corresponding paradigm marker U_i^{nf}. A noun has two equally probable right contexts: a context indicating its gender (line 8) and a context indicating its paradigm (line 9). μ is a function that maps each U to one of five morphological paradigms $\{u_0 \ldots u_4\}$. μ is randomly initialized before a corpus is generated and kept fixed.

The function μ models the assignment of

paradigms to nouns and adjectives. Nouns and adjectives can have different (or the same) paradigms, but for a given noun or adjective the paradigm is fixed and does not change. Lines 7–9 generate gender and paradigm markers for feminine nouns, for which we use the symbols x_i^{nf}. Lines 10–18 cover the three other cases: masculine nouns (x_i^{nm}), feminine adjectives (x_i^{af}) and masculine adjectives (x_i^{am}).

Dataset. We perform 10 trials. In each trial, μ is initialized randomly and a corpus of 100,000 sentences is generated. The train set consists of the feminine nouns (x_i^{nf}, line 7) and the masculine nouns (x_i^{nm}, line 13). The test set consists of the feminine (x_i^{af}) and masculine (x_i^{am}) adjectives.

Results. Embeddings have been learned, SVMs are trained on the binary classification task feminine vs. masculine and evaluated on test. There was not a single error: accuracy of classifications is 100% for all embedding models.

Discussion. The facet gender is indicated directly by the distribution and easy to learn. For a noun or adjective x, we simply have to check whether f or m occurs to its right anywhere in the corpus. PPMI stores this information in two dimensions of the vectors and the SVM learns this fact perfectly. The encoding of "f or m occurs to the right" is less direct in the learning-based representation of x, but the experiment demonstrates that they also reliably encode it and the SVM reliably picks it up.

It would be possible to encode the facet in just one bit in a manually designed representation. While all representations are less compact than a one-bit representation – PPMI uses two real dimensions, learning-based models use an activation pattern over several dimensions – it is still true that most of the capacity of the embeddings is used for encoding facets other than gender: syntactic categories and paradigms. Note that there are five different instances each of feminine/masculine adjectives, feminine/masculine nouns and u_i words, but only two gender indicators: f and m. This is a typical scenario across languages: words are distinguished on a large number of morphological, grammatical, semantic and other dimensions and each of these dimensions corresponds to a small fraction of the overall knowledge we have about a given word.

Point-based tests do not directly evaluate specific facets of words. In similarity datasets, there is no individual test on facets – only full-space similarity is considered. There are test cases in analogy that hypothetically evaluate specific facets like gender of words, as in king-man+woman=queen. However, it does not consider the impact of other facets and assumes the only difference of "king" and "queen" is gender. A clear example that words usually differ on many facets, not just one, is the analogy: London:England $\sim$ Ankara:Turkey. *political-capital-of* applies to both, *cultural-capital-of* only to London:England since Istanbul is the cultural capital of Turkey.

To make our argument more clear, we designed an additional experiment that tries to evaluate gender in our dataset based on similarity and analogy methods. In the *similarity evaluation*, we search for the nearest neighbor of each word and accuracy is the proportion of nearest neighbors that have the same gender as the search word. In the *analogy evaluation*, we randomly select triples of the form $<x_i^{c_1 g_1}, x_j^{c_1 g_2}, x_k^{c_2 g_2}>$ where $(c_1, c_2) \in \{(\text{noun}, \text{adjective}), (\text{adjective}, \text{noun})\}$ and $(g_1, g_2) \in \{(\text{masculine}, \text{feminine}), (\text{feminine}, \text{masculine})\}$. We then compute $\vec{s} = \vec{x}_i^{c_1 g_1} - \vec{x}_j^{c_1 g_2} + \vec{x}_k^{c_2 g_2}$ and identify the word whose vector is closest to $\vec{s}$ where the three vectors $\vec{x}_i^{c_1 g_1}$, $\vec{x}_j^{c_1 g_2}$, $\vec{x}_k^{c_2 g_2}$ are excluded. If the nearest neighbor of $\vec{s}$ is of type $\vec{x}_l^{c_2 g_1}$, then the search is successful; e.g., for $\vec{s} = \vec{x}_i^{\text{nf}} - \vec{x}_j^{\text{nm}} + \vec{x}_k^{\text{am}}$, the search is successful if the nearest neighbor is feminine. We did this evaluation on the same test set for PPMI and LBL embedding models. Error rates were 29% for PPMI and 25% for LBL (similarity) and 16% for PPMI and 14% for LBL (analogy). This high error, compared to 0% error for SVM classification, indicates it is not possible to determine the presence of a low entropy facet accurately and consistently when full-space similarity and analogy are used as test criteria.

5 Analysis

In this section, we first summarize and analyze the lessons we learned through experiments in Section 4. After that, we show how these lessons are supported by a real natural-language corpus.

5.1 Learned lessons

(i) Two words with clearly different context distributions should receive different representations. Aggregation models fail to do so by calculating

	all entities		head entities		tail entities	
	MLP	1NN	MLP	1NN	MLP	1NN
PPMI	61.6	44.0	69.2	63.8	43.0	28.5
LBL	63.5	51.7	72.7	66.4	44.1	32.8
CBOW	63.0	53.5	71.7	69.4	39.1	29.9
CWIN	66.1	53.0	73.5	68.6	46.8	31.4
SKIP	64.5	**57.1**	69.9	**71.5**	**49.8**	**34.0**
SSKIP	**66.2**	52.8	**73.9**	68.5	45.5	31.4

Table 1: Entity typing results using embeddings learned with different models.

global statistics.

(ii) Embedding learning can have different effectiveness for sparse vs. non-sparse events. Thus, models of representations should be evaluated with respect to their ability to deal with sparseness; evaluation data sets should include rare as well as frequent words.

(iii) Our results in Section 4.3 suggest that single-representation approaches can indeed represent different senses of a word. We did a classification task that roughly corresponds to the question: does this word have a particular meaning? A representation can fail on similarity judgement computations because less frequent senses occupy a small part of the capacity of the representation and therefore have little impact on full-space similarity values. Such a failure does not necessarily mean that a particular sense is not present in the representation and it does not necessarily mean that single-representation approaches perform poor on real-world tasks. However, we saw that even though single-representations do well on balanced senses, they can pose a challenge for ambiguous words with skewed senses.

(iv) Lexical information is complex and multifaceted. In point-based tests, all dimensions are considered together and their ability to evaluate specific facets or properties of a word is limited. The full-space similarity of a word may be highest to a word that has a different value on a low-entropy facet. Any good or bad result on these tasks is not sufficient to conclude that the representation is weak. The valid criterion of quality is whether information about the facet is consistently and accurately stored.

5.2 Extrinsic evaluation: entity typing

To support the case for sub-space evaluation and also to introduce a new extrinsic task that uses the embeddings directly in supervised classification, we address a *fine-grained entity typing* task.

Learning taxonomic properties or types of words has been used as an evaluation method for word embeddings (Rubinstein et al., 2015). Since available word typing datasets are quite small (cf. Baroni et al. (2014), Rubinstein et al. (2015)), entity typing can be a promising alternative, which enables to do supervised classification instead of unsupervised clustering. Entities, like other words, have many properties and therefore belong to several semantic types, e.g., "Barack Obama" is a POLITICIAN, AUTHOR and AWARD_WINNER. We perform entity typing by learning types of knowledge base entities from their embeddings; this requires looking at subspaces because each entity can belong to multiple types.

We adopt the setup of Yaghoobzadeh and Schütze (2015) who present a dataset of Freebase entities;[5] there are 102 types (e.g., POLITICIAN FOOD, LOCATION-CEMETERY) and most entities have several. More specifically, we use a multilayer-perceptron (MLP) with one hidden layer to classify entity embeddings to 102 FIGER types. To show the limit of point-based evaluation, we also experimentally test an entity typing model based on *cosine similarity* of entity embeddings. To each test entity, we assign all types of the entity closest to it in the train set. We call this approach 1NN (kNN for $k = 1$).[6]

We take part of ClueWeb, which is annotated with Freebase entities using automatic annotation of FACC1[7] (Gabrilovich et al., 2013), as our corpus. We then replace all mentions of entities with their Freebase identifier and learn embeddings of words and entities in the same space. Our corpus has around 6 million sentences with at least one annotated entity. We calculate embeddings using our different models. Our hyperparameters: for learning-based models: dim=100, neg=10, iterations=20, window=1, sub=10^{-3}; for PPMI: SVD-dim=100, neg=1, window=1, cds=0.75, sub=10^{-3}, eig=0.5. See (Levy et al., 2015) for more information about the meaning of hyperparameters.

Table 1 gives results on test for all (about 60,000 entities), head (freq $>$ 100; about 12,200 entities) and tail (freq $<$ 5; about 10,000 entities). The MLP models consistently outperform 1NN on

[5] cistern.cis.lmu.de/figment

[6] We tried other values of k, but results were not better.

[7] lemurproject.org/clueweb12/FACC1

all and tail entities. This supports our hypothesis that only part of the information about types that is present in the vectors can be determined by similarity-based methods that use the overall direction of vectors, i.e., full-space similarity.

There is little correlation between results of MLP and 1NN in all and head entities, and the correlation between their results in tail entities is high.[8] For example, for all entities, using 1NN, SKIP is 4.3% (4.1%) better, and using MLP is 1.7% (1.6%) worse than SSKIP (CWIN). The good performance of SKIP on 1NN using cosine similarity can be related to its objective function, which maximizes the cosine similarity of cooccuring token embeddings.

The important question is not similarity, but whether the information about a specific type exists in the entity embeddings or not. Our results confirm our previous observation that a classification by looking at subspaces is needed to answer this question. In contrast, based on full-space similarity, one can infer little about the quality of embeddings. Based on our results, SSKIP and CWIN embeddings contain more accurate and consistent information because MLP classifier gives better results for them. However, if we considered 1NN for comparison, SKIP and CBOW would be superior.

6 Conclusion and future work

We have introduced a new way of evaluating distributional representation models. As an alternative to the common evaluation tasks, we proposed to identify generic criteria that are important for an embedding model to represent properties of words accurately and consistently. We suggested four criteria based on fundamental characteristics of natural language and designed tests that evaluate models on the criteria. We developed this evaluation methodology using PCFG-generated corpora and applied it on a case study to compare different models of learning distributional representations.

While we showed important differences of the embedding models, the goal was not to do a comprehensive comparison of them. We proposed an innovative way of doing intrinsic evaluation of embeddings. Our evaluation method gave direct insight about the quality of embeddings. Additionally, while most intrinsic evaluations consider word vectors as points, we used classifiers that identify different small subspaces of the full space. This is an important desideratum when designing evaluation methods because of the multifacetedness of natural language words: they have a large number of properties, each of which only occupies a small proportion of the full-space capacity of the embedding.

Based on this paper, there are serveral lines of investigation we plan to conduct in the future. (i) We will attempt to support our results on artificially generated corpora by conducting experiments on *real natural language data*. (ii) We will study the *coverage of our four criteria* in evaluating word representations. (iii) We modeled the four criteria using separate PCFGs, but they could also be modeled by one single unified PCFG. One question that arises is then to what extent the four criteria are orthogonal and to what extent interdependent. A single unified grammar may make it harder to interpret the results, but may give additional and more fine-grained insights as to how the performance of embedding models is influenced by different fundamental properties of natural language and their interactions.

Finally, we have made the simplifying assumption in this paper that the best conceptual framework for thinking about embeddings is that the embedding space can be *decomposed into subspaces*: either into completely orthogonal subspaces or – less radically – into partially "overlapping" subspaces. Furthermore, we have made the assumption that the smoothness and robustness properties that are the main reasons why embeddings are used in NLP can be reduced to *similarities in subspaces*. See Rothe et al. (2016) and Rothe and Schütze (2016) for work that makes similar assumptions.

The fundamental assumptions here are decomposability and linearity. The smoothness properties could be much more complicated. However even if this was the case, then much of the general framework of what we have presented in this paper would still apply; e.g., the criterion that a particular facet be fully and correctly represented is as important as before. But the validity of the assumption that embedding spaces can be decomposed into "linear" subspaces should be investigated in the future.

Acknowledgments. This work was supported by DFG (SCHU 2246/8-2).

[8] The spearman correlation between MLP and 1NN for all=0.31, head=0.03, tail=0.75.

References

Eneko Agirre, Enrique Alfonseca, Keith B. Hall, Jana Kravalova, Marius Pasca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, May 31 - June 5, 2009, Boulder, Colorado, USA*, pages 19–27.

Marco Baroni and Alessandro Lenci. 2010. Distributional memory: A general framework for corpus-based semantics. *Computational Linguistics*, 36(4):673–721.

Marco Baroni, Georgiana Dinu, and Germán Kruszewski. 2014. Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014*, pages 238–247.

Bernard Comrie. 1989. *Language universals and linguistic typology: Syntax and morphology*. Blackwell, 2nd edition.

D. A. Cruse. 1986. *Lexical Semantics*. Cambridge University Press, Cambridge, MA.

Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *Journal of Machine Learning Research*, 9:1871–1874.

Evgeniy Gabrilovich, Michael Ringgaard, and Amarnag Subramanya. 2013. Facc1: Freebase annotation of clueweb corpora.

Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *The 50th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, July 8-14, 2012, Jeju Island, Korea - Volume 1: Long Papers*, pages 873–882.

Arne Köhn. 2015. What?s in an embedding? analyzing word embeddings through multilingual evaluation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2067–2073, Lisbon, Portugal, September.

Siwei Lai, Kang Liu, Liheng Xu, and Jun Zhao. 2015. How to generate a good word embedding? *CoRR*, abs/1507.05523.

Omer Levy and Yoav Goldberg. 2014a. Linguistic regularities in sparse and explicit word representations. In *CoNLL*.

Omer Levy and Yoav Goldberg. 2014b. Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2177–2185.

Omer Levy, Yoav Goldberg, and Ido Dagan. 2015. Improving distributional similarity with lessons learned from word embeddings. *TACL*, 3:211–225.

Jiwei Li and Dan Jurafsky. 2015. Do multi-sense embeddings improve natural language understanding? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1722–1732, Lisbon, Portugal, September.

Wang Ling, Chris Dyer, Alan W. Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *NAACL HLT 2015, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, Colorado, USA, May 31 - June 5, 2015*, pages 1299–1304.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. In *Proceedings of ICLR*.

Andriy Mnih and Koray Kavukcuoglu. 2013. Learning word embeddings efficiently with noise-contrastive estimation. In *NIPS*, pages 2265–2273.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Joseph Reisinger and Raymond J Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 109–117. Association for Computational Linguistics.

Sascha Rothe and Hinrich Schütze. 2016. Word embedding calculus in meaningful ultradense subspaces. In *ACL*.

Sascha Rothe, Sebastian Ebert, and Hinrich Schütze. 2016. Ultradense embeddings by orthogonal transformation. In *NAACL*.

Dana Rubinstein, Effi Levi, Roy Schwartz, and Ari Rappoport. 2015. How well do distributional models capture different types of semantic knowledge? In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 2: Short Papers*, pages 726–730.

Magnus Sahlgren. 2006. *The Word-Space Model*. Ph.D. thesis, Stockholm University.

Tobias Schnabel, Igor Labutov, David Mimno, and Thorsten Joachims. 2015. Evaluation methods for unsupervised word embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 298–307, Lisbon, Portugal, September.

Yulia Tsvetkov, Manaal Faruqui, Wang Ling, Guillaume Lample, and Chris Dyer. 2015. Evaluation of word vector representations by subspace alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2049–2054, Lisbon, Portugal, September.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *J. Artif. Intell. Res. (JAIR)*, 37:141–188.

Yadollah Yaghoobzadeh and Hinrich Schütze. 2015. Corpus-level fine-grained entity typing using contextual information. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 715–725, Lisbon, Portugal, September.

Zhi Zhong and Hwee Tou Ng. 2010. It makes sense: A wide-coverage word sense disambiguation system for free text. In *ACL 2010, Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, July 11-16, 2010, Uppsala, Sweden, System Demonstrations*, pages 78–83.