Understanding the Semantic Content of Sparse Word Embeddings Using a Commonsense Knowledge Base

Vanda Balogh
University of Szeged
bvanda@inf.u-szeged.hu

Gábor Berend
University of Szeged
MTA-SZTE RGAI, Szeged
berendg@inf.u-szeged.hu

Dimitrios I. Diochnos
University of Oklahoma
diochnos@ou.edu

György Turán
University of Illinois at Chicago
MTA-SZTE RGAI, Szeged
gyt@uic.edu

Abstract

Word embeddings have developed into a major NLP tool with broad applicability. Understanding the semantic content of word embeddings remains an important challenge for additional applications. One aspect of this issue is to explore the interpretability of word embeddings. Sparse word embeddings have been proposed as models with improved interpretability. Continuing this line of research, we investigate the extent to which human interpretable semantic concepts emerge along the bases of sparse word representations. In order to have a broad framework for evaluation, we consider three general approaches for constructing sparse word representations, which are then evaluated in multiple ways. We propose a novel methodology to evaluate the semantic content of word embeddings using a commonsense knowledge base, applied here to the sparse case. This methodology is illustrated by two techniques using the ConceptNet knowledge base. The first approach assigns a commonsense concept label to the individual dimensions of the embedding space. The second approach uses a metric, derived by spreading activation, to quantify the coherence of coordinates along the individual axes. We also provide results on the relationship between the two approaches. The results show, for example, that in the individual dimensions of sparse word embeddings, words having high coefficients are more semantically related in terms of path lengths in the knowledge base than the ones having zero coefficients.

1 Introduction

Word embeddings have developed into a major tool in NLP applications. An important problem – receiving much attention in the past years – is to study, and possibly improve, the *interpretability* of word embeddings. As interpretability is a many-faceted notion which is hard to formalize, the evaluation of interpretability can take different forms. One approach is *intrusion detection* (Faruqui et al. 2015b; Murphy, Talukdar, and Mitchell 2012), where human evaluators test the coherence of groups of words found using word embeddings. A basic observation is that *sparsity* of word embeddings improves interpretability (Faruqui et al. 2015b; Subramanian et al. 2018).

Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

In order to perform a systematic study, we consider several methods to generate sparse word embeddings from dense embeddings for the purposes of the experiments. One family of word embeddings is obtained by *sparse coding* (Berend 2017), another family is obtained by *clustering*, and a third family is obtained by greedily choosing *almost orthogonal* bases.

Another important problem, also receiving much attention is to combine word embeddings and *knowledge bases*. Such a combination has the potential to improve performance on downstream tasks. The information contained in a knowledge base can be incorporated into a word embedding in different ways either during (Iacobacci, Pilehvar, and Navigli 2015; Osborne, Narayan, and Cohen 2016) or after (Faruqui et al. 2015a; Glavaš and Vulić 2018) the construction of the word embeddings.

A knowledge base provides different tools to explore the semantic content of directions, and thus of the basis vectors (also referred to as *semantic atoms*) in sparse word embeddings. These tools include *concepts* contained in a knowledge base and notions of *semantic relatedness* that can be derived from a knowledge base (Feng et al. 2017). The former can be *simple* or *composite* concepts, the latter can be relatedness notions based on *graph distances and edge labels, e.g., using spreading activation, label propagation or random walks*.

Knowledge bases give a principled computational approach for the two problems on word embeddings mentioned above (interpretability and knowledge bases), by providing explicit *"meanings"* with quantifiable validity, which capture the implicit coherence of groups of words in general. We focus on *commonsense* knowledge bases, in particular on ConceptNet (Speer and Havasi 2012), as commonsense knowledge seems to be a fundamental problem where progress coming from such a combination of statistical and symbolic approaches could be relevant.

In this paper we report recent results on a systematic study of *explicit* connections between word embeddings and knowledge bases. We make our source code for reproducing our experiments available online[1]. Our approach is schematized in Figure 1.

[1] https://github.com/begab/interpretability_aaai2020

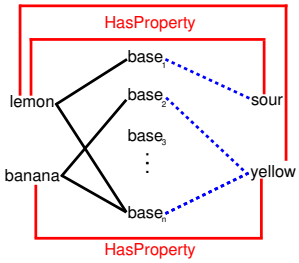


Figure 1: Tripartite graph presenting the connections between embedded words, bases and concepts. Connections indicated by solid lines are initially given, and we are interested in extracting the relationships between bases and commonsense concepts marked by the dashed connections.

We first review related work in Section 2. Section 3 then describes three types of sparse word embeddings discussed, and compares them in terms of incoherence and the overlap between word vectors and semantic atoms. Section 4 introduces the algorithm for assigning ConceptNet concepts to bases in word embeddings and the different quantities from information retrieval measuring the quality of the assignments. The experiments performed evaluate the assignments for the three types of embeddings. Results are also given on how the sparsity parameter influences the results. Section 5 develops the tool for the other evaluation approach: using ConceptNet to measure coherence or semantic relatedness of a set of words by spreading activation. This is then used for experiments evaluating words corresponding to bases in the sparse embeddings. Section 6 brings the two approaches together by analyzing their correspondences.

2 Related Work

Faruqui et al. (2015b) and Subramanian et al. (2018) are seminal papers on sparse word embeddings. In particular, Subramanian et al. (2018) mention that "sparsity and non-negativity are desirable characteristics of representations, that make them interpretable" as a hypothesis. Investigating this hypothesis using quantitative evaluation is one of the objectives of our paper.

Tsvetkov et al. (2015) introduced the evaluation measure QVEC to evaluate the quality of a word embedding space. QVEC computes a correlation between the dimensions of a word embedding space and the semantic categories obtained from SemCor (Miller et al. 1993). QVEC-CCA (Tsvetkov, Faruqui, and Dyer 2016) was introduced as an improvement over standard QVEC, relying on canonical correlation analysis (Hotelling 1936). Compared to our paper, both QVEC and QVEC-CCA provide an overall statistical measure rather than an explicit interpretation, and interpretations are given in terms of a relatively small number of lexical categories. QVEC correlates positively with performance on downstream tasks, i.e., word embeddings that are more interpretable (in the QVEC sense) perform better.

Şenel et al. (2017) consider explicit assignments to word embedding dimensions, and propose specific interpretability scores to measure semantic coherence. This is perhaps the paper most closely related to our approach. They introduce a new dataset (SEMCAT) of 6,500 words described with 110 categories as the knowledge base. (Senel et al. 2017) considers dense word embeddings. In contrast, our paper investigates sparse word embeddings from multiple aspects, and it is based on ConceptNet, which is much larger and richer but also noisier than SEMCAT.

Osborne et al. (2016) introduced an algorithm for determining word representations that also encode prior knowledge into the learned embeddings besides the distributional information originating from raw text corpora. Alsuhaibani et al. (2018) consider a learning process where a word embedding and a knowledge base are learned together. The knowledge base is incorporated into the embedding in an *implicit* manner by integrating it into the objective function (i.e., vectors of words being in a relation are supposed to be close). Several papers take a similar approach to utilize background knowledge in deep learning, e.g., TransE (Border et al. (2013)). In the other direction, similarity of vectors is used for updating the knowledge base. Gardner et al. (2014) uses word embeddings similarity to aid finding paths for new relation tuple prediction. Evaluations are typically performed on downstream tasks. Explicit concept assignment – proposed in this paper – could be considered as an additional tool for all these approaches.

Path-based methods for semantic relatedness are surveyed among other methods, e.g., in Feng et al. (2017). Harrington (2010) considers spreading activation-based methods in ASKNet semantic networks. Berger-Wolf et al. (2013) considers spreading activation in ConceptNet 4 for question answering.

3 Sparse Word Models

We created sparse word representations based on multiple strategies. Here we introduce the different approaches employed during our experiments.

Dictionary Learning-Based Sparse Coding (DLSC) The first approach we employed was dictionary learning-based sparse coding (DLSC). DLSC is a traditional technique for decomposing a matrix $X \in \mathbb{R}^{v \times m}$ into the product of a sparse matrix $\alpha \in \mathbb{R}^{v \times k}$ and a dictionary matrix $D \in \mathbb{R}^{k \times m}$, where k denotes the number of basis vectors (semantic atoms) to be employed. In our case X is a matrix of stacked word vectors, the rows of D form an overcomplete set of basis vectors and the sparse nonzero coefficients in the ith row of α indicate which basis vectors from D should be incorporated in the reconstruction of input signal $\mathbf{x}_i$. DLSC optimizes for

$$\min_{D \in \mathcal{C}, \alpha \in \mathbb{R}^{v \times k}_{\geq 0}} \frac{1}{2} \|X - \alpha D\|_F^2 + \lambda \|\alpha\|_1, \tag{1}$$

where $\mathcal{C}$ denotes the convex set of matrices with row norm at most 1 and the sparse coefficients in α are required to be non-negative. We imposed the non-negativity constraint on α as it has been reported to provide increased interpretability (Murphy, Talukdar, and Mitchell 2012). We used the

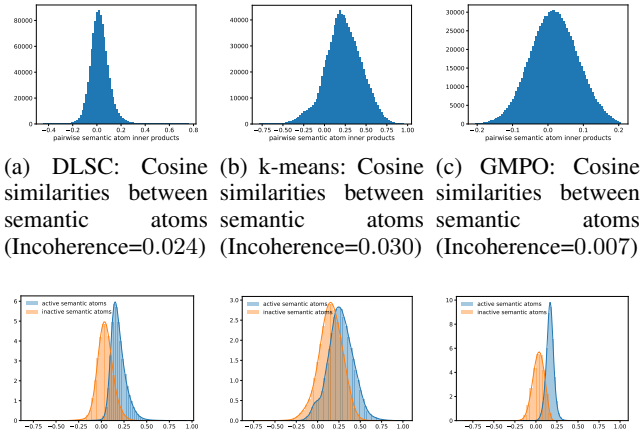
SPAMS library (Mairal et al. 2009) to solve the above optimization problem.

We utilized 300-dimensional Glove embeddings (Pennington, Socher, and Manning 2014) pre-trained on 6 billion tokens for our experiments. The embeddings consist of the 400,000 most frequent lowercased English words based on a 2014 snapshot of Wikipedia and the Gigaword 5 corpus. We set $k = 1000$, i.e., the dictionary matrix contained 1000 basis vectors. Unless stated otherwise, the regularization coefficient λ was set to 0.5 in our experiments. We chose the hypereparameters k and λ based on similar choices made previously in the literature (Faruqui et al. 2015b; Berend 2017).

Determining Semantic Atoms Based on Clustering As semantic atoms can be also viewed as representative *meta-word vector*s, we also constructed D by performing k-means clustering of the actual word vectors as well. Note that k-means can also be considered as a special case of the k-SVD sparse coding algorithm (Aharon, Elad, and Bruckstein 2006). We set $k = 1000$ similar to DLSC and determined the semantic atoms comprising D as the cluster representatives, i.e., the centroids of the identified clusters.

Determining Almost Pairwise Orthogonal Semantic Atoms from Actual Word Vectors As the semantic atoms can be regarded as prototype vectors in the original embedding space, we introduced an approach which treats actual word vectors originating from the embedding matrix X as entries of the dictionary matrix D. Since the dictionary learning literature regards the incoherence of dictionary matrices as a desirable property, we defined such a procedure which explicitly tries to optimize to that measure. The proposed algorithm chooses the dense word vector corresponding to the most frequent word from the embedding space as the first vector to be included in D. Then in $k-1$ subsequent steps, the dictionary matrix gets extended by $\mathbf{x} \in X$ which minimizes the score $\max_{\mathbf{d_i} \in D} |\langle \mathbf{x}, \mathbf{d_i} \rangle|$. We shall refer to this procedure as the **g**reedy **m**aximization for the **p**airwise **o**rthogonality of the semantic atoms, or GMPO for short.

Comparison of the Different Approaches The formal notion of incoherence (Arora, Ge, and Moitra 2013) gives us a tool to quantitatively measure the diversity of a dictionary matrix $D \in \mathbb{R}^{k \times m}$, according to $\max_{\mathbf{d_i} \neq \mathbf{d_j}} \langle \mathbf{d_i}, \mathbf{d_j} \rangle / \sqrt{k}$, with $\langle \cdot, \cdot \rangle$ denoting the inner product. As incoherence of the dictionary matrix has been reported to be an important aspect in sparse coding, we analyzed D from that perspective. Figure 2a illustrates the pairwise inner products between the semantic atoms from the dictionary matrix D in the case of the DLSC method. We can observe that the semantic atoms are diverse, i.e., the inner products concentrate around zero. From the perspective of incoherence, the dictionary matrix obtained by performing k-means clustering has a lower quality (higher incoherence score) as also illustrated by the pairwise inner products of the semantic atoms in Figure 2b. Figure 2c demonstrates that keeping the pairwise orthogonality of the semantic atoms in mind (cf. GMPO) indeed results in a more favorable incoherence score of 0.007.



(a) DLSC: Cosine similarities between semantic atoms (Incoherence=0.024)

(b) k-means: Cosine similarities between semantic atoms (Incoherence=0.030)

(c) GMPO: Cosine similarities between semantic atoms (Incoherence=0.007)

(d) DLSC: Cosine similarities of word vectors and semantic atoms

(e) k-means: Cosine similarities of word vectors and semantic atoms

(f) GMPO: Cosine similarities of word vectors and semantic atoms

Figure 2: Characteristics of matrices D and α when different approaches are used for determining D.

We now define active and inactive semantic atoms with respect to some word vector $\mathbf{x_i}$. We say that a semantic atom $\mathbf{d_j}$ is active with respect to $\mathbf{x_i}$, if $\mathbf{d_j}$ takes part in the reconstruction of $\mathbf{x_i}$, i.e., when $\alpha_{ij} > 0$. Additionally, we define the semantic overlap between a semantic atom $\mathbf{d_j}$ and a dense word vector $\mathbf{x_i}$ as $\langle \mathbf{x_i}, \mathbf{d_j} \rangle$, i.e., the projection of $\mathbf{x_i}$ onto $\mathbf{d_j}$. We can see in Figure 2d that the semantic overlap of word vectors towards active semantic atoms tend to be higher than for inactive ones, suggesting that we managed to learn meaningful sparse representations. As semantic atoms are less dissimilar from each other in the case of the k-means approach, we observed that the distribution of the active and inactive (semantic atom, dense word vector) pairs is also less distinguishable from each other (cf. Figure 2e). In accordance with the low incoherence score for GMPO, Figure 2f reveals that the difference in the distribution of the semantic overlap between active and inactive semantic atoms towards the dense input vectors is the most pronounced for GMPO.

We also compared the sparsity levels obtained by the different approaches. Table 1 contains the number of nonzero coefficients a word form received on average. We can see that the k-means approach had the tendency of producing fewer nonzero coefficients per word form on average when using the same regularization coefficient of $\lambda = 0.5$. The second row of Table 1 reveals that the higher sparsity level of the k-means representations comes at the price of performing worse in the reconstruction of the original dense embeddings.

Table 1: The number of nonzero coefficients assigned to a word on average and the total reconstuction error incurred during the reconstruction of the embedding matrix X.

	DLSC	k-means	GMPO
Avg. nnz in α per word	52.86	19.41	59.64
Error term ($\|X - \alpha D\|_F$)	2734.5	3286.9	2971.8

4 Base Assignment

Our first approach to investigate the interpretability of the dimensions of sparse embedding matrices is assigning each dimension human interpretable features, which is similar to our previous work on embeddings in Hungarian (Balogh et al. 2019). The rows of the embedding matrix correspond to sparse word vectors representing words. We call the columns (dimensions) of the sparse embedding matrix *bases*. As human interpretable features, we take concepts extracted from a semantic knowledge base, ConceptNet. We focus on the English words in ConceptNet 5.6 (later on we will simply refer to it as ConceptNet). The records of the knowledge base are called *assertions*. Each assertion associates two words (or phrases) – *start* and *end* nodes – with a semantically labelled, directed relation. A word (or phrase) in ConceptNet can either be a start node, an end node or both. In our setting, the start nodes correspond to *embedded words* and we call the end nodes *concepts*. We keep only those concepts that appear more than 40 times as end nodes in ConceptNet. Obviously, not all of the embedded words are present in ConceptNet, therefore in the following we will work with the 50k most popular words (based on total degrees) in ConceptNet that are also among the embedded words. Basically, we deal with a tripartite graph (see Figure 1) with words connected to bases and concepts. A word w is connected to $base_i$ if the ith coordinate of the sparse word vector corresponding to w is nonzero. Also, w is connected to a concept c if there exists an assertion in ConceptNet that associates w and c. We are interested in the relations between concepts and bases (dotted lines). In other words, our goal here is to analyze to what extent the sparse embedding is in accordance with the knowledge base.

4.1 Base Assignment Algorithm

The process of associating a base with a concept is divided into five phases that we describe below.

I. Produce Knowledge Base Matrix We consider ConceptNet as a bipartite graph whose two sets of vertices correspond to (embedded) words and concepts. A word can appear as a concept, too. The bipartite graph is represented as a biadjacency matrix C (which simply discards the redundant parts of a bipartite graph's adjacency matrix). Every embedded word w is associated with an indicator vector v_w where the ith coordinate of v_w is 1 if w is associated to the ith concept, 0 otherwise. At this point, words have two sparse representations: the vectors coming from sparse word embeddings and the binary vectors from ConceptNet.

II. Compute Product We binarize the nonnegative sparse embedding matrix α by thresholding it at 0, then we take the product of the transpose of C and this binarized matrix. The result is a matrix A, containing the co-occurrences of concept-base pairs.

III. Compute NPPMI We compute the normalized positive pointwise mutual information (NPPMI) for every element of A. We rely on this normalized version of PMI (Bouma 2009) as it handles co-occurrences of low frequency better. We compute the NPPMI for some concept c_i and base b_j as

$$\mathrm{NPPMI}(c_i, b_j) = \max\left(0; \ln \frac{\mathrm{P}(c_i, b_j)}{\mathrm{P}(c_i)\mathrm{P}(b_j)} \Big/ -\ln \mathrm{P}(c_i, b_j)\right)$$

where probabilities are approximated as relative frequencies of words as follows: $\mathrm{P}(c_i)$ is the relative frequency of words connected to the ith concept, $\mathrm{P}(b_j)$ takes the relative frequency of words whose jth coefficient in their embedded vector representation is nonzero and $\mathrm{P}(c_i, b_j)$ is the relative frequency of the co-occurrences of the words mentioned above. The result is a sparse matrix P whose columns and rows correspond to bases and concepts, respectively.

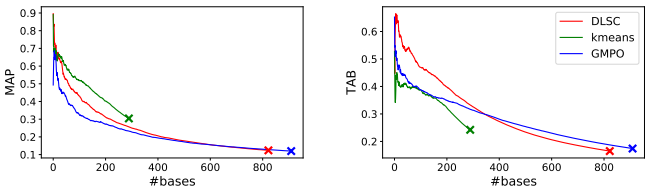
IV. Take Argmax By taking the arguments of the maximum values of every column in P we can associate a base with a concept. If the maximum value for a base is zero – implying no positive dependence to any concept – then no concept is assigned to it. We take the argmax focusing on bases, similarly to (Tsvetkov et al. 2015), allowing us to assign a concept to multiple bases.

V. Create and Assign Meta-Concepts As a post processing step we compute the NPPMI for concept pairs (based on concept co-occurrences) thus we have a notion of closeness for concepts. Alongside the associated concept c_i of a base b, the concepts that are close to c_i are also assigned to b, thus creating *meta-concepts*. The set of close concepts for c_i is defined as: $\mathrm{close}(c_i) = \{c_j | i \neq j, \mathrm{NPPMI}(c_i, c_j) \geq 0.5, \mathrm{NPPMI}(c_i, c_j) \geq 0.95 * \max_{k \neq i}(\mathrm{NPPMI}(c_i, c_k)\}$. After assigning close concepts, there were on average 2.55, 2.56 and 2.39 concepts assigned to each base in DLSC, GMPO and k-means embeddings, respectively.

4.2 Evaluation

To evaluate the associations between bases and concepts, we employ metrics from the information retrieval literature (Manning, Raghavan, and Schütze 2008). We would like to measure if the *dominant words* of a base, i.e., the words for which the given base is active (as defined in Section 3), are in relation with the concepts associated to the base according to ConceptNet.

We use mean average precision (MAP) as a precision oriented metric during our evaluation. MAP is calculated for the first 50 words that have the highest nonzero values for every base. If a base has no concept assigned to it, the average precision and the reciprocal rank of that base is set to zero. As for recall oriented metrics, similarly to (Senel et al. 2017), train and test words are randomly selected (60%, 40%) for each concept before the assignment takes place. On average each concept has 40 test words. The assignments are obtained from train words (described in Section 4.1), and for each concept its test words are removed. Afterwards, the



(a) MAP (b) TAB

Figure 3: Cumulative evaluation scores for MAP and TAB. The horizontal axis shows bases cumulatively ordered in ascending order with respect to their highest NPPMI values. After the crosses NPPMI values are zero, meaning that no new concept assignment took place afterwards.

percentages of unseen test words are calculated in two different ways. The first one measures accuracy of the test words according to bases and it is called *test accuracy by bases* (TAB). Formally,

$$\mathrm{TAB}(b) = \frac{|\{w \in D_b \cap \mathrm{test}(c)\}|}{|\{w \in V | (w,c) \in KB \wedge w \notin \mathrm{train}(c)\}|},$$

where D_b is the set of nonzero coefficient words in base b, c is the concept assigned to base b, V is the set of all words, KB stands for the knowledge base, furthermore $\mathrm{test}(c)$ and $\mathrm{train}(c)$ are the set of test and train words for concept c, respectively. The other metric we use measures *test accuracy by concepts* (TAC) and it is calculated for some concept c as

$$\mathrm{TAC}(c) = \frac{|\{w \in (\cup_b \{D_b | b \text{ has } c \text{ assigned }\}) \cap \mathrm{test}(c)\}|}{|\{w \in V | (w,c) \in KB \wedge w \notin \mathrm{train}(c)\}|}.$$

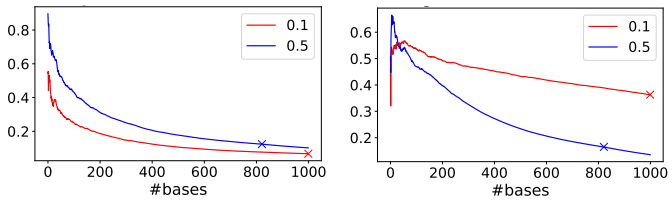
The average is taken over all bases for TAB and all concepts in the case of TAC. Finally, in order to combine the precision and the recall-oriented views, we compute an F-score-like metric by treating MAP as precision and TAB as recall.

Figure 3 shows the results of MAP and TAB cumulatively. The bases are always in ascending order according to NPPMI values. The evaluation metric with respect to all the bases is always the value at the end of the horizontal axis. Generally (as seen in the monotone behaviour of curves in Figure 3), *the NPPMI values correlate with the evaluation metrics*. As long as k-means has bases that have assigned concepts (shown as a cross in the figures), it performs the best in terms of MAP. However, DLSC and GMPO have a lot more bases that have concepts assigned to them. On the long run, GMPO slightly outperforms DLSC at MAP. Figure 3b and Table 2 reveals that DLSC and GMPO tend to perform similarly and better than the clustering-based approach for the further evaluation metrics.

Finally, we evaluated the effects of applying the less conservative regularization coefficient $\lambda = 0.1$. For space considerations, we report it for the DLSC approach only. Decreasing the regularization coefficient form $\lambda = 0.5$ to $\lambda = 0.1$ caused the average number of nonzero coefficients per a word to increase from 52.9 to 186.9. Figure 4 illustrates that sparser representations favor evaluation towards MAP, while TAB performances are better in the case of representations with lower sparsity.

Table 2: Mean and standard deviation of TAC computed for all assigned concepts and F-score taking MAP as precision and TAB as recall.

Approach	Mean$_{TAC}$	Std dev$_{TAC}$	F-score
DLSC	0.498	0.241	0.105
k-means	0.450	0.201	0.072
GMPO	0.497	0.228	0.117



(a) MAP (b) TAB

Figure 4: Comparison of evaluation scores on DLSC sparse embedding with different regularization coefficients. Precision related metrics tend to favour sparser solutions (λ=0.5), recall oriented metrics gravitate towards less sparse representations (λ=0.1).

5 Spreading Activation and ConceptNet

Collins and Quillian (1969) were the first to show supporting evidence that categories of objects form a hierarchical network in the human memory and through this hierarchy meaning could be given to different words. Various applications on knowledge bases build on such hierarchical structure in order to find semantic similarity between words, semantic relatedness, meaning, as well as for question answering. Among the main tools used in various such applications are *label propagation* (Quillian 1969) and *spreading activation* methods (Collins and Loftus 1975); e.g., (Salton and Buckley 1988; Harrington 2010; Nooralahzadeh et al. 2016; Berger-Wolf et al. 2013).

Label propagation methods starting with two nodes having two distinct labels, proceed in iterations where a label is propagated to neighbors that obtained the label in the previous round. Ultimately, a node (or a set of nodes) is reached where both starting labels appear on that node. Such nodes are important as they allow the formation of a short path between the two starting nodes without looking at the entire network. Spreading activation methods build on this idea; in each round apart from propagating labels, activation values are propagated along the relations connecting the various words. Different variants of spreading activation can arise; e.g., one can think of the nodes firing once (or continuously) after they first receive a label, propagate decayed activation values to neighbors, etc. Thus, the activation values that propagate among neighboring nodes, allow in the end additional filtering on the activated network so that *heavy* short paths are found connecting the starting nodes.

In our second approach we employ spreading activation in ConceptNet 5.6 (Speer and Havasi 2012) to investigate

Table 3: Results obtained using spreading activation on ConceptNet 5.6. APL_t and APL_b correspond to the average path length for pairs of top and bottom words respectively. The last column titled n/a counts bases for which we could not complete the experiments due to memory constraints.

approach		size of activated network				comparing average path lengths			
		min	median	average	max	$APL_t < APL_b$	$APL_t > APL_b$	ties	n/a
DLSC	top smaller	740	661	630	554	657	300	23	20
	bottom smaller	238	319	350	426				
	ties	2	0	0	0				
k-means	top smaller	768	761	703	580	667	299	13	21
	bottom smaller	209	218	276	399				
	ties	2	0	0	0				
GMPO	top smaller	766	685	651	563	731	238	18	13
	bottom smaller	219	302	336	424				
	ties	2	0	0	0				

Table 4: Coherent top words in some bases of the DLSC embedding and the assigned concepts. APL_t and APL_b show the average path length for the top 10 and bottom 10 words, respectively.

Concepts assigned	Top words	APL_t	APL_b
china, prefecture	china changchun chongqing tianjin wuhan liaoning xinjiang shenyang shenzhen nanjing	1.84	3.40
farm, farmer	maize crops wheat grain crop soybean sugarcane corn livestock cotton	1.87	3.96
drug, pharmaceutical drug	antidepressant drug tamoxifen drugs statin painkiller aspirin stimulant antiviral estrogen	2.00	4.07
death, funeral, die	slaying murder stabbing murdering death beheading killing murderer hanged manslaughter	1.96	3.58
payment, pay	payment deductible expenses taxes pay pension refund tax tuition money	1.73	3.40

the coherence of the dominant words in each base. Whereas earlier we were interested in English words solely, this time we allow non-English words to be activated and appear in this search process and in fact we give such an example at the end of the current section. We are interested if the dominant words in a base make a semantically coherent group compared to the words with zero coefficients. With this goal in mind, 10 words with the largest nonzero coefficients are selected from each base (if possible) and also, 10 words with zero coordinates are randomly chosen. We call these two sets of words *top* and *bottom* words of a base, which always come from the 50k most popular embedded words (having the highest total degree) that appear in ConceptNet.

Table 3 presents findings from our experiments. For the paths found the average path length among pairs of top words (APL_t) is less than the average path length among pairs of bottom words (APL_b) in about $66\% - 73\%$ of the bases. Interestingly, *the network activated while searching for a path is typically smaller for pairs of top words compared to the one obtained for pairs of bottom words.*

On the Average Path Lengths When APL_t has a value of *3.044 or less* then that value is always smaller than APL_b. This is true for *all three algorithms*. Furthermore, when APL_t has a value of about *2.5 or less*, then such words are very well aligned and all of them are typically members of a broader group. As APL_t increases, the coherence among the top words fades out. Table 4 in Section 6 provides some examples.

On the Spreading Activation Variant The spreading activation variant we use behaves similarly to label propagation. In almost all cases the path connecting a pair of words is one of the shortest found in the knowledge base and the activation helps us identify a heavy such short path. This approach is in accordance to our basic intuition that words that have good alignment with particular bases should form coherent groups and we would expect this coherence to be exemplified by short paths connecting such pairs of words. As we mention in Section 7 an interesting future direction is to explore other variants of spreading activation.

On the Alignment In some cases the top 10 aligned words with a particular base do not form a (very) coherent group. For example, with the DLSC dictionary, in base 609, the top words are: contiguity, plume, maghreb, tchaikovsky, acuminate, maglev, trnava, interminably, snowboarder, and convalesce. In fact this is an example where the top words have average path length *more than* that of the bottom words (4.044 vs 3.644); so the incoherence of the top aligned words is reflected in the path lengths.

On Polysemy In several cases it is the phenomenon of polysemy that gives the path which is short and heavy. This issue can happen when looking at paths for both top and bottom words and regardless of the overall coherence of the words in the group. For example, when using the k-means dictionary, for base 48, the top words *trad* and *volcanologist* are found to be connected with the path: /c/en/trad – /c/en/music – /c/en/rock – /c/fr/géologie – /c/en/volcanology – /c/en/volcanologist.

6 Discussion and Synthesis of Results

Now we bring together the evaluation of the base assignment with coherence analysis. The words come from the 50k highest degree words in ConceptNet. The qualitative results are in accordance with the quantitative ones.

Table 5: The 5 highest nonzero coefficient words for assigned concepts in the three sparse embeddings. The words that appear in ConceptNet alongside the assigned concept are **bold**.

concept(s)	DLSC	k-means	GMPO
car, cars	**sedan chevrolet bmw audi toyota**	**sedan hatchback coupe** lexus **roadster**	tesla **roadster** musk **electric volt**
disease, pathology	**disease diseases encephalitis pneumonia meningitis**	**measles polio diphtheria meningitis tetanus**	**polio measles** immunization **vaccination diphtheria**
greek mythology, greek god	porgy tchaikovsky bluebeard falstaff **ariadne**	**zeus theseus** odin **agamemnon hephaestus**	juno award gemini jupiter emmy
law, legal	**judge court appellate judges** supreme	**appellate court** supreme **injunction judges**	**waiver** retroactive infielder **waive** signed
mathematics	**polynomial** integer **invertible affine quadratic**	**abelian topological affine isomorphic**	integer **factorization polynomial modulo** divisible

Table 6: Pearson correlations (ρ) between the assignment evaluations (MAP, TAB) and the average path length of top words for sparse word models. We report p-values for the ρ in parenthesis.

	DLSC	k-means	GMPO
ρ_{MAP}	-0.60 (1.1e-98)	-0.58 (6.1e-88)	-0.53 (3.2e-73)
ρ_{TAB}	-0.60 (3.0e-97)	-0.59 (8.0e-93)	-0.53 (1.3e-62)

Generally, *the concepts that were assigned to bases reflect their dominant words*. Table 4 shows bases where the average path length among the dominant words was much lower than among the non-dominant ones (zero coefficient words), which implies the coherence of the base. Clearly, there is a strong connection between assigned concepts, dominant words and average path lengths of top words in bases. Table 6 shows the Pearson correlations between the average path length of top words and the assignment evaluations (MAP, TAB). The moderate negative correlation implies that the quantities move in opposite directions (as expected).

Polysemous words occur in all sparse embeddings with their multiple meanings reflected by the assigned concepts. For example, *court* is a dominant word of bases that are assigned to meta-concepts {*law, legal*} and {*sport*}. Likewise, *virus* is dominant for bases assigned to meta-concepts {*computer, network, desktop*} and {*disease, pathology*}.

Altogether, there are 63 meta-concepts (corresponding to 119 separate concepts) that were assigned to some base in all of the embeddings. Comparison of the three sparse embedding approaches with respect to concepts can be seen in Table 5. K-means tends to have bases where the words with the highest coefficients are actually associated with the assigned concept in ConceptNet. This shows correspondence with the quantitative results (see Section 4.2). On the other hand, as seen in Table 4, GMPO seems to have bases with dominant words that are not connected to the assigned concept of a given base, but there is a semantic relation between them (*tesla* is an automotive company, *juno* is the Roman equivalent of Hera, *retroactive* is a type of law). Also, Table 5 shows an example for DLSC where the concept assignment is wrong: *porgy, tchaikovsky, bluebeard, falstaff* are rather connected to opera and not Greek mythology or Greek god.

7 Conclusions and Future Work

In this paper we analyzed the extent to which the bases of sparse word embeddings overlap with commonsense knowledge. We provided an algorithm for labeling the most dominant semantic connotations that the individual bases convey relying on ConceptNet. Our qualitative experiments suggest that there is substantial semantic content captured by the bases of sparse embedding spaces. We also demonstrated the semantic coherence of the individual bases via analysing the paths between concepts in ConceptNet and quantified the correlation between the two types of evaluations.

Our experiments suggest several directions. Construction methods for sparse word embeddings which combine the approaches studied, such as k-SVD, could be added to the current ones for comparison. We are planning to expand our analysis to dense embeddings as well. Concept assignment could be extended to include other forms of composite concepts and bases. Spreading activation and network analysis methods going beyond path lengths could be used to determine semantic relatedness, taking into account the "heaviness" information obtained, edge labels, combination with random walks, neighborhood analysis and other techniques; for example Diochnos in (2013) explores several properties of ConceptNet 4 with the tools of network analysis and some of these findings can potentially be associated with providing meaning to word embeddings using more recent versions of ConceptNet.

Experiments are planned on extending current techniques for downstream NLP tasks and knowledge base analysis using the explicit information found in the word embeddings.

Acknowledgements

This research was partly funded by the project "Integrated program for training new generation of scientists in the fields of computer science", no EFOP-3.6.3-VEKOP-16-2017-0002, supported by the EU and co-funded by the European Social Fund. This work was in part supported by the National Research, Development and Innovation Office of Hungary through the Artificial Intelligence National Excellence Program (grant no.: 2018-1.2.1-NKP-2018-00008).

References

[2006] Aharon, M.; Elad, M.; and Bruckstein, A. 2006. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse

Representation. *Trans. Sig. Proc.* 54(11):4311–4322.

[2018] Alsuhaibani, M.; Bollegala, D.; Maehara, T.; and ichi Kawarabayashi, K. 2018. Jointly learning word embeddings using a corpus and a knowledge base. *PLOS One* 13(3):1–26.

[2013] Arora, S.; Ge, R.; and Moitra, A. 2013. New algorithms for learning incoherent and overcomplete dictionaries. *CoRR* abs/1308.6273.

[2019] Balogh, V.; Berend, G.; Diochnos, D. I.; Farkas, R.; and Turán, G. 2019. Interpretability of Hungarian embedding spaces using a knowledge base. In *XV. Conference on Hungarian Computational Linguistics (2019)*, 49–62. JATE Press.

[2017] Berend, G. 2017. Sparse coding of neural word embeddings for multilingual sequence labeling. *Transactions of the Association for Computational Linguistics* 5:247–261.

[2013] Berger-Wolf, T.; Diochnos, D. I.; London, A.; Pluhár, A.; Sloan, R. H.; and Turán, G. 2013. Commonsense knowledge bases and network analysis. In *11th International Symposium on Logical Formalizations of Commonsense Reasoning*.

[2013] Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating embeddings for modeling multi-relational data. In *27th Annual Conference on Neural Information Processing Systems 2013*, 2787–2795.

[2009] Bouma, G. 2009. Normalized (pointwise) mutual information in collocation extraction. In *Proceedings of GSCL*, 31–40.

[1975] Collins, A. M., and Loftus, E. F. 1975. A Spreading-Activation Theory of Semantic Processing. *Psychological review* 82(6):407.

[1969] Collins, A. M., and Quillian, M. R. 1969. Retrieval Time from Semantic Memory. *Journal of Verbal Learning and Verbal Behavior* 8(2):240–247.

[2013] Diochnos, D. I. 2013. Commonsense Reasoning and Large Network Analysis: A Computational Study of ConceptNet 4. *CoRR* abs/1304.5863.

[2015a] Faruqui, M.; Dodge, J.; Jauhar, S. K.; Dyer, C.; Hovy, E.; and Smith, N. A. 2015a. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1606–1615.

[2015b] Faruqui, M.; Tsvetkov, Y.; Yogatama, D.; Dyer, C.; and Smith, N. A. 2015b. Sparse overcomplete word vector representations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1491–1500.

[2017] Feng, Y.; Bagheri, E.; Ensan, F.; and Jovanovic, J. 2017. The state of the art in semantic relatedness: a framework for comparison. *Knowledge Eng. Review* 32:e10.

[2014] Gardner, M.; Talukdar, P. P.; Krishnamurthy, J.; and Mitchell, T. M. 2014. Incorporating vector space similarity in random walk inference over knowledge bases. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014*, 397–406.

[2018] Glavaš, G., and Vulić, I. 2018. Explicit retrofitting of distributional word vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 34–45.

[2010] Harrington, B. 2010. A Semantic Network Approach to Measuring Relatedness. In *COLING 2010, 23rd International Conference on Computational Linguistics, Posters Volume*, 356–364.

[1936] Hotelling, H. 1936. Relations Between Two Sets of Variates. *Biometrika* 28(3/4):321–377.

[2015] Iacobacci, I.; Pilehvar, M. T.; and Navigli, R. 2015. SensEmbed: Learning sense embeddings for word and relational similarity. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 95–105.

[2009] Mairal, J.; Bach, F.; Ponce, J.; and Sapiro, G. 2009. Online dictionary learning for sparse coding. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, 689–696.

[2008] Manning, C. D.; Raghavan, P.; and Schütze, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

[1993] Miller, G. A.; Leacock, C.; Tengi, R.; and Bunker, R. T. 1993. A semantic concordance. In *Proceedings of the Workshop on Human Language Technology*, HLT '93, 303–308.

[2012] Murphy, B.; Talukdar, P.; and Mitchell, T. 2012. Learning effective and interpretable semantic models using non-negative sparse embedding. In *Proceedings of COLING 2012*, 1933–1950.

[2016] Nooralahzadeh, F.; Lopez, C.; Cabrio, E.; Gandon, F.; and Segond, F. 2016. Adapting Semantic Spreading Activation to Entity Linking in Text. In *International Conference on Applications of Natural Language to Information Systems*, 74–90. Springer.

[2016] Osborne, D.; Narayan, S.; and Cohen, S. 2016. Encoding prior knowledge with eigenword embeddings. *Transactions of the Association for Computational Linguistics* 4:417–430.

[2014] Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543.

[1969] Quillian, M. R. 1969. The Teachable Language Comprehender: A Simulation Program and Theory of Language. *Communications of the ACM* 12(8):459–476.

[1988] Salton, G., and Buckley, C. 1988. On the Use of Spreading Activation Methods in Automatic Information Retrieval. In *SIGIR'88, Proceedings of the 11th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval 1988*, 147–160.

[2017] Senel, L. K.; Utlu, I.; Yücesoy, V.; Koç, A.; and Çukur, T. 2017. Semantic structure and interpretability of word embeddings. *CoRR* abs/1711.00331.

[2012] Speer, R., and Havasi, C. 2012. Representing general relational knowledge in conceptnet 5. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA).

[2018] Subramanian, A.; Pruthi, D.; Jhamtani, H.; Berg-Kirkpatrick, T.; and Hovy, E. H. 2018. SPINE: sparse interpretable neural embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*, 4921–4928.

[2015] Tsvetkov, Y.; Faruqui, M.; Ling, W.; Lample, G.; and Dyer, C. 2015. Evaluation of word vector representations by subspace alignment. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2049–2054.

[2016] Tsvetkov, Y.; Faruqui, M.; and Dyer, C. 2016. Correlation-based intrinsic evaluation of word vector representations. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, 111–115.