Interpretability in Natural Language Processing

Bennur Kaya

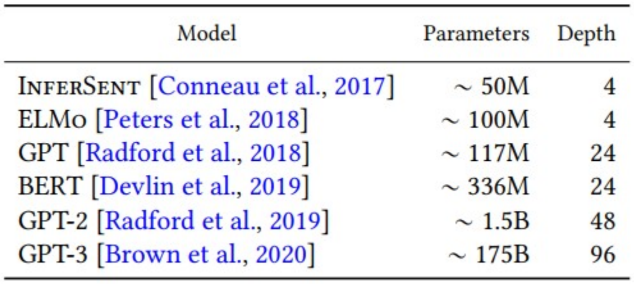
University of Mannheim
Mannheim, 68131, Germany
bekaya@mail.uni-mannheim.de

Abstract. Natural language processing enables computers to understand texts and spoken words in a way that humans can understand. Human language is full of polysemous words and creates ambiguities in sentence structure such as sarcasm, grammar, usage exceptions, and metaphors. Over the years, tremendous progress has been made in natural language processing tasks to better understand these uncertainties. However, the explosion in the number of parameters and the complexity of these state-of-the-art black box models significantly reduce the ability for humans to understand what exactly these architectures have learned and executed in the intermediate steps. In this work, the subject of interpretability is discussed over several different word embedding representations which are essential components of NLP architectures. To address interpretability problem, firstly, all word embedding representations are explained in details. Then, a paper comparing those representations is reviewed to learn more about interpretability. It also includes downstream task implementations to compare word embedding performances and detect lexical correlations.

Keywords: NLP · Word Embeddings · Interpretability

1 Introduction

Natural language processing (NLP) is a field of artificial intelligence in which computers analyze, understand, and extract meaning from human language in an intelligent and useful way. This can be used for a variety of tasks such as topic identification, sentiment analysis and automatic machine translation. For example, there are NLP models that can generate draft articles written of very similar quality to that of a human author. As another example, they may recognize and filter mails based on their content as spam, primary, social, or promotions. In addition, there are now intelligent assistants that recognize patterns in speech, then extract meaning and provide a helpful response thanks to voice recognition. Lastly, they can help autocomplete text by finishing the word or suggesting a relevant word based on what a person has written. Therefore, it is a very active research area with new techniques and applications constantly being developed. In recent years, there has been an accelerating growth in the complexity of state-of-the-art machine learning models in almost all areas of artificial intelligence, particularly in NLP. Transformer architectures such as famous BERT (Devlin et al., 2019) and GPT-n (Radford et al., 2018, 2019; Brown et al., 2020) models took the capabilities of NLP to another level. Moreover, some of the deep learning-based models achieve very high accuracy almost as much as humans. But then it reveals an important question regarding the black box nature of the machine learning models. Nearly, all popular models actively used in NLP today are so-called black box models. Their architecture are essentially non-transparent for direct human interpretation as a result of the explosion in the number of parameters and nonlinear interactions in the underlying mathematical models. As it can be seen from Table 1 below, the sizes and depths of recent popular NLP architectures are rapidly growing over the years with the help of new improvements. These architectures are usually trained with an end-to-end learning method. In the context of artificial intelligence and machine learning, end-to-end learning is a technique where the model learns all the steps starting from the initial stage to the final result without the need to get anything from a third party. This is a deep learning process where all the different parts are trained simultaneously rather than sequentially. So, it limits the capabilities of the practitioners and end users for the models' parameters, learning alghortims, functionality and hyper parameters during both training or inference stages.



Model	Parameters	Depth
InferSent [Conneau et al., 2017]	~ 50M	4
ELMo [Peters et al., 2018]	~ 100M	4
GPT [Radford et al., 2018]	~ 117M	24
BERT [Devlin et al., 2019]	~ 336M	24
GPT-2 [Radford et al., 2019]	~ 1.5B	48
GPT-3 [Brown et al., 2020]	~ 175B	96

Table 1. Parameters and depths in NLP architectures

From one point of view, this is the key strength of deep learning approaches as they skip a remarkable amount of human requirements for the task. Hence, they give better results in terms of evaluation metrics. From a different viewpoint, this is also a weakness, as it significantly reduces people's ability to understand exactly what these architectures learn and implement in these intermediate steps. Nevertheless, the researchers aim to improve the interpretability of the models for the purpose of increasing the performance of the model, generating different innovative ideas, and simply curiosity.

Despite the numeric achievements, some of the models have been repeatedly shown to display a wide variety of overall undesirable behaviors such as weaknesses in handling basic and common linguistic phenomena. These issues can be related to the drawbacks of relying on simple metrics like accuracy on benchmark tasks to evaluate actual understanding of natural language in computer models. But also, there are usually no strong guarantees that the distribution of the instances in the datasets associated with a given task actually matches the distribution associated with the system which is intended to be modelled which can be classified as different forms of data set biases (Bourgeade, 2022). For these reasons, it is apparent there exists a strong need for methods which would enable more transparency and human understanding of algorithmic decisions, particularly from the very opaque deep learning models which are popular nowadays. While a lot of work and effort has been deployed to attempt to reach these goals, it is pretty difficult to give answers to those questions Lipton (2017) ask: What is interpretability and why is it important? Does interpretability simply mean a low-level mechanistic understanding of our models? If so, does it apply to the features, parameters, models, or training algorithms? Also, What constitutes transparency? Alternatively, by could consider the model's complexity, is it simple enough to be examined all at once by a human? Aside from the more challenging technical aspects of how to apply these concepts in practice, even these fundamental questions are yet to be fully conclusively answered.

2 Interpretability

There is no concrete definition of the interpretability of machine learning models. Different authors have tried to provide different definitions. It may defined as that the degree of understanding a model has of how a decision is made so that a model can respond to a user (Choudhary et al., 2022) . However, interpretability has no formal objective technical meaning. Also, it will be seen that different audience levels will interpret the model separately when trying to analyze the interpretability aspect from a user expectation. Lipton (2017) has listed trust, causality, transferability and informativeness as key elements when considering the interpretability research. As also observed in the paper, he considers the model properties in two categories. The first is transparency and it questions how the models work. The second is post-hoc explanations and it searches for the further information from the models. For example, it may reveal the importance of various parameters.

Model Transparency: Transparency is defined in terms of *simulateability, decomposability* and *algorithmic transparency.* First, simulateability means transparency at the level of the entire model. A person can contemplate the entire model at once and use input data with the model to reconstruct each computational step necessary to make the prediction. This allows one to understand the changes in model parameters caused by the training data. Secondly, decomposability indicates whether there is an intuitive explanation for all the model inputs, parameters and calculations. It indicates transparency at the level of individual components such as parameters. Lastly, algorithmic transparency is fundamentally the ability to explain the workings of the learning algorithm. So, it means algorithmic transparency at the level of the training algorithm. This may give some confidence that the model might behave well on previously unseen data.

Post-hoc Interpretability: Post-hoc interpretability is a different approach to extract information from the models. This is defined in terms of *textual description, visualization* and *local explanation.* Although it does not provide detailed information about how the model works, it may help to obtain useful information for practitioners and end users. Firstly, textual description provides a meaningful summary description of the model. They propose a system of multiple models, one capable of predicting and the other giving verbal explanations of the strategy. Secondly, visualization means explaining the model through visualization of the parameters. Lastly, local explanation indicates focusing on what a neural network depends on locally. A popular approach for deep neural networks is to compute a salience map. Typically, they take the gradient of the output corresponding to the correct class with respect to a given input vector.

3 Word Embeddings

Word embedding is one of the most powerful concepts of deep learning applied to Natural Language Processing. It can capture the context of a word in a document. The core concept of word embeddings is that every word used in a language can be represented by a set of real numbers in a vector. It learns representations of text in an n-dimensional space where words that have the same meaning have a similar representation. In this paper, Non-Negative Sparse Coding (NNSC), Non-Negative Sparse Embeddings (NNSE), Word2Sense, Sparse Over Complete Word Vector (SPOWV) and Sparse Interpretable Neural Embeddings (SPINE) methods will be examined.

Different approaches have been explored to achieve better interpretability in vectorized word representations. Constraint-based embedding models are widely used approach with two main constraint which are employed to enhance interpretability are *sparsity* and *non-negativity.* Bourgeade (2022) explains the necessity of sparsity with the example that people prefer to identify objects and concepts with a few specific and strongly related words rather than different but weaker related intonations. Models also can benefit from such constrained representations because it help to produce less biased models, as there are other properties of words which should not necessarily be taken into account (gender etc.). Sparsity leads to easier exhaustive analysis of the relevant components of an input, for the purpose of explanation methods. On the other hand, they mention that it is difficult to imagine what a negative value for a particular semantic component could mean. Non-negativity is mainly useful since it allows to understand the possible reasons of the non-zero components of a word embedding representation as a degree of “participation” of different parts. There are two ways to apply these constraints to word embedding approaches. One way is to start from scratch by modifying or creating a new coding process to include these constraints. These word embeddings are under the name of Priori Constrained Interpretable Embeddings. The second way is to start from an existing word embedding model and transform its vectors a posteriori via matrix factorization or fundamental rotation. Also, these word embeddings are under the name of Posteriori Constrained Interpretable Embeddings. In this section, various word embedding methods are presented in which such approaches with varying levels of complexity are presented. Subsequently, the same notations are defined for all word embedding representations for consistency and understandability in the formulation of each objective function (Bourgeade, 2022).

3.1 Priori Constrained Interpretable Embeddings

Priori Constrained Interpretable Embeddings rely on a construction method which imposes the constraints sparsity and non-negativity on the produced embedding space. One relatively simple way to achieve this is to use non-negative matrix factorization (NMF) techniques, as in LSA by applying this factorization on term context co-occurrences statistics. LSA indicates latent semantic analysis and it is a technique to analyze relationships between a set of documents and the terms by producing a set of concepts related to the documents and terms. It assumes that words that are close in meaning will occur in similar pieces of text. This property can be useful as a word similarity metric. So, efficient algorithms exist to compute such factorization, using multiplicative update rules like adding some additional terms to enforce new constraints to achieve non-negativity and sparsity.

Non-Negative Sparse Coding (NNSC): As observed in the original paper of Non-negative Sparse Coding (Hoyer et al., 2002), NNSC is described as a method for decomposing multivariate data into non-negative sparse components. They mention that the use of wavelet representations started to be common as well as Fourier analysis in signal representation, however these methods share significant disadvantages such as inability to adapt to the specific data being analyzed besides their advantages. Data-adaptive representations are learned directly from observed data by optimizing some measurements. Principal component analysis (PCA), independent component analysis (ICA), sparse coding, and non-negative matrix factorization (NMF) are the methods for analyzing large datasets containing a high number of dimensions, increasing the interpretability of data while preserving the maximum amount of information. Thus, they combine sparse coding and non-negative matrix factorization into non-negative sparse coding. The motivation comes from modeling neural information processing. Non-Negative Sparse Coding is the method to enforce both non-negativity and sparsity in the produced representations by using non-negative matrix factorization under some additional constraints. They proposed objective function is in Figure 1 below.

$$\arg\min_{W,H} C_X(W,H) = \sum_{i=1}^{V} (\|X_{i,:} - W_{i,:} \times H\|_F^2 + \lambda \|W_{i,:}\|_1)$$

Fig. 1. Objective function of NNSC

For the consistency, the formulation of each objective function is adapted so that they make use of the same notation. In the objective function, X represents input statistical observation data. W is the resulting sparse non-negative word embedding matrix. H is the second part of the factorization of X, often called the "dictionary" or basis matrix. Lastly, for any matrix A corresponds to a single element with i-th row and its j-th column. These notations will present the same meaning in every objective function.

The authors of the original paper applied non-negativity constraint to both matrices W which is sparse non-negative word embedding matrix and H which the second part of the factorization of X as they should be bigger than zero. Also, unit rescaling constraint is applied where lambda is a positive hyperparameter controlling the trade-of between the accuracy of the factorization the sparsity of the output embedding matrix W. The rescaling constraint is necessary to ensure the second term of the objective, which enforces the sparsity of parse non-negative word embedding matrix W. The authors experimented with this approach on image data and saw the addition of the sparsity constraint appears to improve the efficiency and add more robustness.

Non-Negative Sparse Embeddings (NNSE): Non-Negative Sparse Embedding (NNSE) method is a matrix factorization technique which is a variation on Non-Negative Sparse Coding (Murphy et al., 2012). They introduce an application of matrix factorization to produce corpus-derived, distributional models of semantics that demonstrate cognitive plausibility. After the application and examination of NNSE, it is found that word representations are sparse, effective, and highly interpretable. Non-Negative Sparse Embedding is convex with respect to each variable when the others are kept fixed. They concluded as the superiority of semantic models learned by NNSE over other state-of-the-art baselines. They also mentioned that there are still many ways in which they can extend and improve this new embedding method, for example, testing the method as a component of core NLP tasks (chunking, named-entity recognition, parsing etc.) or comparing the individual NNSE dimensions to other benchmarks that explicitly cover categories and properties.

$$\arg\min_{W,H} C_X(W,H) = \frac{1}{2}\|X - HW\|_F^2 + \lambda \sum_{i,j} W_{i,j}$$

Fig. 2. Objective function of NNSE

The proposed objective function Non-Negative Sparse Embeddings is in Figure 2 above. The authors applied non-negativity constraint only to W which is sparse non-negative word embedding matrix as it should be bigger than zero. The difference between NNSE and NNSC for non-negativity is that there is no constraint for H which is the second part of the factorization of X. Similar to Non-Negative Sparse Coding, unit rescaling constraints applied where lambda is a positive hyperparameter controlling the trade-of between the accuracy of the factorization the sparsity of the output embedding matrix W.

Word2Sense: Word2Sense is another word embedding technique which is developed around the question of whether an interpretable embedding which coordinates make a clear sense to humans is possible (Panigrahi et all., 2019). They present an unsupervised method to generate Word2Sense aiming to compute the embedding of a polysemous word as a distribution over its senses in a corpus. For the shortcomings of prior works on unsupervised word embeddings of single prototype embedding, they don't explicitly capture sense information (Word2Vec, Glove), require extra computation for introducing interpretability (SPINE, SPOWV) and they lack explicit algorithm for polysemy detection in context. For the multi-prototype embeddings, the main shortcoming is that it is computationally expensive. Word2sense brings the advantage of that rather than constructing a per-word representation of senses, it construct a global pool of senses from which the senses a word takes in the corpus are inferred. In this case, sense is a set of semantically similar words that collectively evoke a bigger picture than individual words in the reader's mind. So, Word2Sense means probability distributions over senses. Latent Dirichlet Allocation (LDA) generative statistical model is used. LDA explains a set of observations through unobserved groups, and each group explains why some parts of the data are similar. Generative model for the co-occurrence matrix recovers senses from a corpus and represents word embeddings as probability distributions over senses. Co-occurrence matrix associates each word with a sense distribution which are *Thetas* (ϑ). Then, it forms a context around a target word by sampling senses according to sense distributions and sampling words from the distribution of sense. All in all, after comparing the performance of Word2Sense with other word embedding models like Word2Vec (Mikolov et al., 2013), Word2GM (Athiwaratkun & Wilson, 2019), SPOWV (Faruqui et all., 2015) and SPINE (Pruthi et all, 2018) by applying various downstream tasks such as word intrusion test, sentiment analysis, news classification, question classification, the authors concluded Word2Sense embeddings works well and it is compareable to other word embeddings generated by unsupervised methods. Word2Sense embeddings are at least as sparse and fast to compute as prior art. For the future work, sense distribution may extended to sentences instead of only words.

Contrary to the previous methods, the co-occurrence matrix is assumed to follow a generative model where a sense model is inferred as a set of Dirichlet distributions over the words. Thetas are the values between zero and one. When a word in the corpus is not associated with any sense, it gets its thetas value as zero. If the word is associated with any or more senses, it can take a value up to 1 depending on the strength of the association. In this case, the non-negativity constraint has already been satisfied. Also, any word can be encoded as sparse in dimensional Dirichlet distribution over these learned senses.

3.2 Posteriori Constrained Interpretable Embeddings

Instead of starting from scratch from text corpora statistics, a method is used to transform an existing dense word embedding matrix into a non-negative or a sparser form in order to improve interpretability. Sparse Over complete Word Vector (SPOWV) and Sparse Interpretable Neural Embeddings (SPINE) are considered as Posteriori Constrained Interpretable Embeddings.

Sparse Over Complete Word Vector (SPOWV): Faruqui et al. (2015) propose a method which transform word vectors into sparse vectors and optionally binary word vectors, and name it as Sparse Over complete Word Vector Representations. A sparse vector is a vector having a relatively small number of nonzero elements. Because the vectors are highly sparse, they are computationally easy to work with. They inspired by the idea of introducing sparsity in word vector dimensions has been shown to improve dimension interpretability (Murphy et al., 2012; Fyshe et al., 2014) and usability of word vectors as features in downstream tasks (Guo et al., 2014). Word vectors can be derived directly from raw and unannotated corpora and intrinsic evaluations on the tasks guides methods which captures different information about lexical semantics. However, the representations described in lexical semantic theory do not represent word vectors. It is an important task to conceptualize word meanings symbolically for theoretical understanding and explainability, even though it is costly. As an contribution to these discussions, the researchers came up with Sparse Over Complete Word Vector Representations. As a result of this work, they found consistent benefits of the method on standard benchmark evaluation tasks. After evaluating the word vectors in a word intrusion experiment, they found that sparse vectors are more interpretable than the original vectors. For that reason, they highly support their discovery since they think SPOWV plays an important role in statistical NLP models.

Figure 3 above shows the objective function of Sparse Over Complete Word Vector where λ and τ respectively control the L1-norm sparsity constraint on W

$$\arg\min_{W,H} C_X(W,H) = \sum_{i=1}^{V}(\|X_{i,:} - W_{i,:} \times H\|_F^2 + \lambda\|W_{i,:}\|_1 + \tau\|H\|_F^2)$$

Fig. 3. Objective function of SPOWV

and the L2-norm soft bounding constraint on H. Unlike the previous methods, this method uses a specialized variant of online adaptive gradient descent (AdaGrad) specially adapted to handle the L1 regularization term while also clipping the negative terms in W which parse non-negative word embedding matrix to 0 for non-negativity.

Sparse Interpretable Neural Embeddings (SPINE): Sparse Interpretable Neural Embeddings is another word embedding method developed with the need for explainability. Pruthi et al. (2018) propose a denoising new variant of k-sparse auto encoders that generate highly efficient and interpretable distributed word representations. They explain k-sparse auto encoder (Makhyani, 2014) as it is an auto encoder with high probability and at most k hidden units are active for any given input. They compare SPINE with Glove (Pennington et al., 2014), Word2Vec and SPOWV. Distributed representations map words to vectors of real numbers in a continuous space. Mostly, the entire corpus is scanned, and the vector creation process is performed by determining which words the target word occurs with more often. In this way, the semantic closeness of the words to each other is also revealed. It is a way to give ability to computers understanding humans, words or lexical meanings via real numbers. However, humans find difficult to understand the word vectors with high density representations.

They start with a little task and they observe GloVe, Word2Vec and Sparse Over Complete Word Vectors (SPOWV) are not interpretable with randomly selected words since the top participating words do not form a semantically coherent and meaningful group. They argue on if interpretability can help in gaining better understanding of neural representations and models since it can provide cues to make them more efficient and robust. They decide to use sparsity and non-negativity to make embeddings more interpretable. Starting from the question ”How does one transform word representations to a new space where they are more interpretable, possibly by exploiting the principles of sparsity and non-negativity?”, they make two main contribution. First is employing a denoising k-sparse auto encoder to obtain Sparse Interpretable Neural Embeddings which is a transformation of input word embeddings. They train the auto encoder using a novel learning objective and activation function to attain interpretable and efficient representations. Secondly, they evaluate SPINE using a large scale intrusion detection test, along with the downstream tasks. As a result, they reported that their word embedding result is way better and interpretable than the original embeddings. They concluded that SPINE outperform popular word embeddings available in a suite of comparative downstream tasks. For future studies, they aim to investigate the effect of triggering different amounts of sparsity across multiple hidden layers in larger and more complex networks, and use the GloVe/Word2Vec framework along with our loss formulations to enforce sparsity and non-negativity.

$$
\begin{aligned}
\arg\min_{W,H} C_X(W,H) = & \underbrace{\frac{1}{V}\sum_{i=1}^{V}\Big(\|X_{i,:} - \mathrm{Dec}\big(\mathrm{Enc}(X_{i,:})\big)\|_F^2\Big)}_{\mathrm{RL}} \\
& \underbrace{+ \lambda_1 \sum_{h=1}^{d} \max\left(0, \Big(\frac{1}{V}\sum_{i=1}^{V}\mathrm{Enc}(X_{i,:})_{:,h}\Big) - \rho\right)^2}_{\mathrm{ASL}} \\
& \underbrace{+ \lambda_2 \frac{1}{m}\sum_{i=1}^{V}\sum_{h=1}^{d}\Big(\mathrm{Enc}(X_{i,:})_{:,h} \times \big(1 - \mathrm{Enc}(X_{i,:})_{:,h}\big)\Big)}_{\mathrm{PSL}}
\end{aligned}
$$

Fig. 4. Objective function of SPOWV

The proposed objective function of Sparse Interpretable Neural Embeddings is in Figure 4 above. It is a three-part objective function. Enc() and Dec() are respectively the encoding and decoding functions of the auto-encoder. $\lambda 1$, $\lambda 2$ and ρ respectively the hyper parameters controlling the Average Sparsity Loss (ASL) term, the Partial Sparsity Loss (PSL) term, and the desired sparsity factor. The Average Sparsity Loss (ASL) term pushes each dimension in the output representations towards a sparsity factor (or lower). To be able to provide non-negativity in the output embeddings, activation functions that produce non-negative values for all possible inputs values are the potentials. Rectified Linear Units (ReLU) and Sigmoid units can be suitable for this purpose. For strict sparsity, the authors eliminated the Sigmoid activation function due to its asymptotic nature. As a result, they estimate the interpretability of the dimensions with word intrusion detection task and other benchmark downstream tasks. They found that their precision scores are notably higher than those of the original vectors and SPINE works competitively well on all benchmark tasks.

4 Evaluating Interpretability in Interpretable Word Embeddings

Bourgeade (2022) compared the word embedding methods discussed in the previous section with various downstream tasks. In this section, those tasks will be introduced and the performance evaluation results will be critiqued to better understand and comment on interpretability.

4.1 Baseline Interpretable Embedding Model: NMF300

The researchers found it necessary to have a common baseline model to compare word embedding methods. To be able to have an appropriate baseline model which will display some level of interpretability, they want to introduce sparsity and non-negativity to the model. Non-negative matrix factorization techniques produce output matrices which display these properties by their nature. So, they built the NMF300 baseline model to use it as a point of reference comparing with more complex models from the literature by using the non-negative factorization algorithm for the Kullback-Leibler divergence reconstruction cost function (Lee et al., 2001). Figure 5 shows objective function of the baseline model NMF300.

$$\arg\min_{W,H} D_{KL}(X\|WH) = \sum_{i,j=1,1}^{V,C} (X_{ij} \log \frac{X_{ij}}{(WH)_{ij}} - X_{ij} + (WH)_{ij})$$

Fig. 5. Objective function of the baseline model NMF300

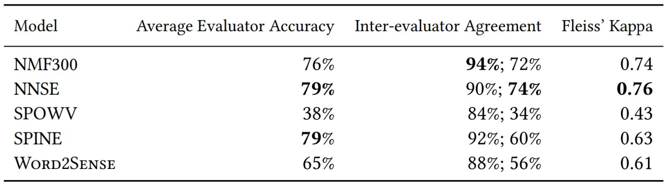
The authors of the original paper applied non-negativity constraint to both matrices W which is sparse non-negative word embedding matrix and H which the second part of the factorization of X should be bigger than zero. Also, unit rescaling constraint is applied on the columns to W. This baseline model is constructed on a co-occurrence matrix collected on over a 2.2 billion word Wikipedia dump from May 2017, with a vocabulary size equals to 100000 words.

4.2 Downstream Tasks

There are 9 different tasks to compare representations. Firstly, BoolQ indicates Boolean Questions which is a yes/no question answering data set. It incorporates with 3000 yes/no questions and passages. The task is considered as unexpectedly challenging as they require looking for potentially complex information in the accompanying passage with regards to the question. Second one is Emergent. The task is to classify the journalistic stance with regards to a claim sentence, from articles headlines related to the claim, in which each article can be labelled either for, against, or simply observing the claim. It contains 300 claims and 2595 associated news articles. Third one is IMDB. It is a sentiment analysis for collected user movie reviews from the popular Internet Movie Database website with binary sentiment labels. It is on a stars scale and neutral reviews are not included. Fourth one is SST which stands for the Stanford Sentiment Treebank data set which is for sentiment analysis based on the movie reviews. It contains 11855 single sentences, which are broken down into a total of 215154 unique phrases constituted into parse trees. It is annotated by 3 human judges into one of five polarity classes ("very negative", "negative", "neutral", "positive", "very positive"). Fifth one is Sarcasm data set which is annotated for the presence of sarcasm. It is mainly either generic sarcasm, rhetorical questions, or hyperbolg. Sixth one is the UR-FUNNY which is a multi modal humor detection data set, incorporating textual, visual and acoustic modalities, extracted from publicly available TED talks videos. It is a balanced data set containing 88257 multi modal (punchline, context, labels) instances for each class (humorous, non-humorous), spanning over 1741 speakers, 1866 videos and 417 topics. The next one is SNLI which is provided by Stanford Natural Language Inference corpus and is predicting whether a hypothesis sentence logically and semantically follows from a premise sentence with three possible classes of relationships to predict as entailment, contradiction and neutral. The next one is PDTB. Discourse relations, attempt to formally characterize the textual relations between two segments of a discourse. These relations can either be realized explicitly, through some type of linking word or phrase, most often conjunctions (and, or, because, when, although etc.). Last one is Word Intrusion Task. Given set of 5 word, it is necessary to find the odd one out. As first shortcomings, it was basically impossible to tell the intruder apart from the most active words for such dense representations and they found the task is not challenging enough even when changing the threshold hyper parameters. So, they propose a modification of the sampling process which significantly increases the difficulty of the task.

4.3 Downstream Tasks Performance Evaluation

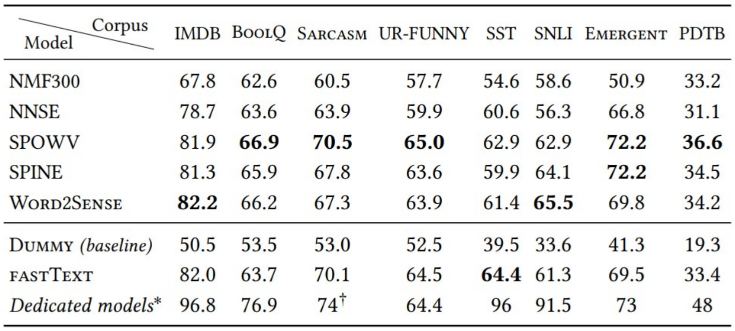
Table 2 below displays both the evaluators average accuracy and the inter-evaluator agreement metrics on the Word Intrusion Detection task to compare the word embedding methods. They realized that the baseline model NMF300 performs well despite its simplicity. They interpreted this as showing that the addition of sparsity and non-negative constraints is highly effective. Researchers have found that most embedding methods are relatively good at associating with a particular lexical aspect, except SPOWV. On the other hand, they also noticed that the models seemed to capture a few frequency artifacts such as brands and sports players.



Model	Average Evaluator Accuracy	Inter-evaluator Agreement	Fleiss' Kappa
NMF300	76%	**94%**; 72%	0.74
NNSE	**79%**	90%; **74%**	**0.76**
SPOWV	38%	84%; 34%	0.43
SPINE	**79%**	92%; 60%	0.63
Word2Sense	65%	88%; 56%	0.61

Table 2. Results on Word Intrusion Detection task

For the Downstream task evaluation, they compare the results with the task-dedicated models which achieve state-of-the-art performance from the literature under the name of *Dedicated models*. They trained continuous-bag-of-words (CBOW) softmax regression classifier. For the tasks with two separate input texts, they used the sentence-encoding architecture *InferSent* (Conneau et al., 2017). Also, they used the average of the interpretable embeddings of all the words in an instance as an input to their model by benefiting from simple *fastText-based linear* for both training and evaluating (Joulin et al., 2017). Lastly, they use a *dummy classifier* that produces random predictions, weighted by the class distribution of the task.



Corpus Model	IMDB	BoolQ	Sarcasm	UR-FUNNY	SST	SNLI	Emergent	PDTB
NMF300	67.8	62.6	60.5	57.7	54.6	58.6	50.9	33.2
NNSE	78.7	63.6	63.9	59.9	60.6	56.3	66.8	31.1
SPOWV	81.9	**66.9**	**70.5**	**65.0**	62.9	62.9	**72.2**	**36.6**
SPINE	81.3	65.9	67.8	63.6	59.9	64.1	**72.2**	34.5
Word2Sense	**82.2**	66.2	67.3	63.9	61.4	**65.5**	69.8	34.2
Dummy *(baseline)*	50.5	53.5	53.0	52.5	39.5	33.6	41.3	19.3
fastText	82.0	63.7	70.1	64.5	**64.4**	61.3	69.5	33.4
*Dedicated models**	96.8	76.9	$74^{\dagger}$	64.4	96	91.5	73	48

Table 3. Results of the Downstream Tasks

As it can be seen in Table 3, the interpretable embedding models SPOWV, SPINE, Word2Sense perform relatively well, while NNSE significantly below. Also, the trained classifiers have high accuracies and even better for most of the tasks in some interpretable embedding models with the fastText model. Also, researchers generate global explanation reports for each of the trained models. They commented that NNSE is the most influential dimension for each class and it has a strong positive and negative groups logically make sense. With NMF300, however, the second positive class is more questionable as it appears to be associated with the names of public celebrities. For the IMDB task, which is the data set with the highest performance results, several dimensions associated with a large number of surnames and first names which appear to be strong predictors of the "positive" class such as Shakira according to NMF300 model. On BoolQ data set, the most contributing dimensions seem to focus on debated themes and conspiracy theories for the false answers and science, history, geography, or politics for the true answers. For the Sarcasm and UR-FUNNY tasks, positive-class dimensions seems to be associated with music and musical artists and negative-class dimensions focus more on medical themes, legal themes and a lot of technical-themed dimensions.

5 Conclusion

In this paper, why the concept of interpretability is necessary and how it can contribute to us is discussed together with the related studies. Different word embedding methods are introduced from their original papers. Next, methods and insights from an article comparing various word embedding methods with the help of downstream tasks are examined. In my opinion, when trying to improve interpretability with sparsity and non-negativity, it is very interesting that a basic base-line model can yield such good results compared to various word embedding models supported and developed by many other studies. In addition to these, the biggest ongoing limitation is that both technical and theoretical definitions of interpretability are not clear and black-box models are still not fully interpretable for humans to understand. Though these studies will shed light on future studies, this subject is still open to development. If I had a chance to contribute to this work, I would like to try to improve the interpretability of the models with different restrictions instead of/together with restrictions mentioned which are sparsity and non-negativity.

References

1. Devlin, J., Chang, M., Lee, K., Toutanova K. (2019) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Journal: arXiv preprint arXiv:1810.04805v2

2. Radford A., Narasimhan, K., Salimans, T., Sutskever, I. (2019). Improving language understanding by generative pre-training. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

3. Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., . . . Amodei, D. (2020) Language models are few-shot learners., Journal: arXiv preprint arXiv:2005.14165v4

4. Bourgeade, T. (2022). From text to trust : A priori interpretability versus post hoc explainability in Natural Language Processing. https://theses.hal.science/tel-03770191

5. Lipton, Z.,: The Mythos of Model Interpretability (2017), Journal: arXiv preprint arXiv:1606.03490v3

6. Choudhary, S., Chatterjee, N., Saha S. (2022). Interpretation of Black Box NLP Models: A Survey, Journal: arXiv preprint arXiv:2203.17081v1

7. Lou, Y., Caruana, R., Gehrke, J. (2013). Accurate Intelligible Models with Pairwise Interactions. https://www.cs.cornell.edu/ yinlou/papers/lou-kdd13.pdf

8. Hoyer, P. (2002). NON-NEGATIVE SPARSE CODING. https://www.academia.edu/504127/Non_negative_sparse_coding

9. Murphy, B., Talukdar P. P., Mitchell, T. (2012) Learning Effective and Interpretable Semantic Models using Non-Negative Sparse Embedding. https://www.cs.cmu.edu/ bmurphy/NNSE/

10. Panigrahi, A., Simhadri, H., Bhattacharyya, C. (2019). Word2Sense : Sparse Interpretable Word Embeddings. https://aclanthology.org/P19-1570.pdf

11. Mikolov, T., Chen, K., Corrado, G., Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space, Journal: arXiv preprint arXiv:1301.3781v3

12. Athiwaratkun, B., Wilson, A. (2019). Multimodal Word Distributions, Journal: arXiv preprint arXiv:1704.08424v2

13. Faruqui, M., Tsvetkov, Y., Yogatama, D., Dyer, C., Smith, N. (2015). Sparse Overcomplete Word Vector Representations, Journal: arXiv preprint arXiv:1506.02004v1

14. Fyshe, A., Talukdar, P., Murphy, B., Mitchell, T. (2014). Interpretable Semantic Vectors from a Joint Model of Brain- and Text- Based Meaning. https://aclanthology.org/P14-1046.pdf

15. Guo, J., Che, W., Wang, H., Liu, T. (2014). Revisiting Embedding Features for Simple Semi-supervised Learning. https://aclanthology.org/D14-1012/

16. Pruthi, D., Jhamtani, H., Subramanian, A., Berg-Kirkpatrick, T., Hovy, E. (2018). SPINE: SParse Interpretable Neural Embeddings. https://www.cs.cmu.edu/ hovy/papers/18AAAI-SPINE.pdf

17. Makhyani, A., Frey, B. (2014). k-Sparse Autoencoders, Journal: arXiv preprint arXiv:1312.5663v2

18. Pennington, J., Socher, R., Manning, C. (2014). GloVe: Global Vectors for Word Representation. https://nlp.stanford.edu/pubs/glove.pdf

19. Lee, D., Seung, H. (2001). Algorithms for Non-negative Matrix Factorization. https://papers.nips.cc/paper/2000/hash/f9d1152547c0bde01830b7e8bd60024c-Abstract.html

20. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A. (2018). Supervised Learning of Universal Sentence Representations. Natural Language Inference Data, Journal: arXiv preprint arXiv:1705.02364v5

21. Joulin, A., Grave, E., Bojanowski, P., Mikolov, T. (2017). Bag of Tricks for Efficient Text Classification.https://aclanthology.org/E17-2068.pdf