Noname manuscript No.
(will be inserted by the editor)

Exploiting Latent Semantic Subspaces to Derive Associations for Specific Pharmaceutical Semantics

Janus Wawrzinek · José María González Pinto · Oliver Wiehr · Wolf-Tilo Balke

Received: date / Accepted: date

Abstract State-of-the-art approaches in the field of neural-embedding models (NEMs) enable progress in the automatic extraction and prediction of semantic relations between important entities like active substances, diseases, and genes. In particular, the *prediction property* is making them valuable for important research-related tasks such as hypothesis generation and drug-repositioning. A core challenge in the biomedical domain is to have *interpretable* semantics from NEMs that can distinguish, for instance, between the following two situations: a) drug x *induces* disease y and b) drug x *treats* disease y. However, NEMs alone cannot distinguish between associations such as treats or induces. Is it possible to develop a model to learn a *latent representation* from the NEMs capable of such disambiguation? To what extent do we need domain knowledge to succeed in the task? In this paper, we answer both questions and show that our proposed approach not only succeeds in the *disambiguation* task but also advances current growing research efforts to find real predictions using a sophisticated retrospective analysis. Furthermore, we investigate which type of associations are generally better contextualized and therefore probably have a stronger influence in our disambiguation task. In this context, we present an approach to extract an interpretable *latent semantic subspace* from the original embedding space in which *therapeutic* drug-disease associations are more likely.

J. Wawrzinek, J.M.G. Pinto, O. Wiehr, W.T. Balke
Institute for Information Systems, TU-Braunschweig
Mühlenpfordstrasse 23
38106 Braunschweig
Germany
Tel.: +49 (5 31) 3 91 - 74 42
Fax: +49 (5 31) 3 91 - 32 98
E-mail: wawrzinek@ifis.cs.tu-bs.de
E-mail: pinto@ifis.cs.tu-bs.de
E-mail: wiehr@tu-bs.de
E-mail: balke@ifis.cs.tu-bs.de

Keywords Digital libraries · Association mining · Information extraction · Neural embeddings · Semantic enrichment · Semantic subspaces

1 Introduction

Today's digital libraries have to manage the exponential growth of scientific publications [5], which results in faster-growing data holdings. To illustrate the effects of this growth, consider as an example Sara, a young scientist from the pharmaceutical field who wants to find drugs related to "Diabetes" to design a new hypothesis that might link an existing drug with "Diabetes" that has not yet been discovered (not published in a paper). Indeed, this is a complex information need, and in this context, a term-based search in the digital library PubMed leads to 39,000 hits for the year 2019 alone. Due to these data amounts, Sara will have to dedicate considerable time to analyse each paper and take some other steps to satisfy her information need. Given this complicated situation, we believe that this problem makes innovative access paths beyond term-based searches necessary.

One of the most effective ways to help users like Sara is based on the automatic extraction of entity relations that are embedded in literature, i.e. such as those that exist between drugs and diseases [10, 12, 14]. Considering pharmaceutical research drug-disease associations (DDAs) play a crucial role because they are considered candidates for drug-repurposing [3]. The central idea behind drug-repurposing is to use an already known and well-studied drug for the treatment of another disease. In addition, drug-repurposing generally leads to lower risk in terms of adverse side effects [3]. However, it is not only the therapeutic application that is of interest, but also whether a drug induces a disease and may therefore be life-threatening for patients [8, 16].

From this motivation, numerous computer-based approaches have been developed in recent years which attempt not only to extract but also to predict DDAs. Here, for the majority of the methods, entity-similarities form the basis [7, 8, 16]. For example, one of the most important assumptions is that drugs with a similar chemical (sub) structure also have similar (therapeutic) properties [15]. Moreover, while structural similarity is extremely useful for screening, it does not capture other important semantic features.

Scientific literature is one of the primary sources in the investigation of new drugs [10], which is why newer approaches use Neural Embedding Models (NEMs) to calculate linguistic or lexical similarities between entities in order to deduce their properties, semantics, and relationships [1,9]. The use of NEMs in this area is based on a (context) hypothesis [1], where words which share numerous similar surrounding word-contexts are spatially positioned closer to each other in a high dimensional space. This property leads to the fact, that with increasing similarity (i.e. cosine-similarity) also a possible semantic relationship between two entities can be deduced.

This contextualization property can be used to predict complex chemical properties decades in advance [9, 17] or novel therapeutic drug-properties [25].

However, using NEMs to disambiguate the type (i.e. “treats”, “induces”) of a drug-disease association is a challenging task; after all, what semantic is behind a cosine similarity between such entity-pairs? All we know is that they appear in similar contexts, but we do not know how to interpret it. Despite NEMs being an excellent foundation for several tasks, little is known about how to find –if it exists- a feature space with-in the NEMs that allow us to disambiguate associations between drug-disease pairs such as treats or induces. In this paper, we hypothesize that such disambiguation is possible. In particular, we first propose to apply deep learning to learn a *latent feature* space from the NEMs and, thus, disambiguate associations between pharmaceutical entities. Here we first prove that this latent feature space with high probability exists. Therefore we assume that a combination of dimensions (a subspace) probably exists in which e.g. therapeutic associations are better represented compared to the original embedding space. Better in this therapeutic context means that with increasing similarity (e.g., cosine-similarity) the probability for a therapeutic association is higher compared to the original embedding space. For this purpose we extend our work from [32] and focus on the extraction of a semantically optimized subspace and additionally we investigate which DDA semantics are actually present in the embedding space and to what extent. In brief are therapeutic or rather induced associations better learned/contextualized? With respect to a certain semantic, the identification of a semantic subspace would have the advantage of indeed being better interpretable. To identify a semantic subspace we use the approach presented in [30]. Here, the main idea is to consider dimensions of the original embedding space as features. Using feature selection approaches, the authors determine a combination of dimensions in which a given drug-semantic is better represented, e.g. drugs are better grouped by therapeutic properties in a subspace compared to the original embedding space. Yet, in contrast to [30] we do not search for a subspace for entities of the same type (e.g., only drugs) but for associations between entities of different types. Here, drugs and diseases are not similar per se but only associated. Therefore, in our specific case the associations are rather binary-associations (treats or induces). The novelty in this paper is that we adapt and apply the approach presented in [30] to extract and analyse a semantically optimized subspace for drug-disease associations.

In summary, the questions that guided our research were the following:

- RQ1: Do word embeddings contain the information of the type of an embedded drug-disease association, and if yes, to which extent?
- RQ2: Distance in the embedding space has an effect on semantics [17]. In this context we want to answer what is the impact of the distance between entities in the embedding space in the disambiguation task?
- RQ3: Does domain-specific knowledge have an impact to uncover the embedding space needed to disambiguate predicted drug-disease associations?
- RQ4: Is it possible to identify a latent feature space from the original embedding space where a certain drug-disease semantic is predominant?

To answer the first two questions, we investigate different Deep-Learning models and compare the results with a baseline-approach. We show that distance has indeed an effect on accuracy-quality, but not all models are affected by it in the same strength. Hereafter, we propose different semantic enriched Deep-Leaning models and using a retrospective analysis we can show, that our semantic enriched models lead to improved results in disambiguating the DDA-Type ("treats", "induces") for real DDA predictions.

To answer our last research question, we demonstrate that we can extract a semantically optimized subspace for therapeutic DDAs. In addition we show that therapeutic associations are disproportionately better contextualized. This allows the assumption that therapeutic associations are in the foreground while induced associations are much less discussed in bio-medical publications. Based on this results, we assume that the therapeutic context is the key factor in the previous disambiguation tasks.

2 Related Work

Entities like active substances, diseases, genes, and their interrelationships are of central interest for bio-medical digital libraries [4]. In this context, manual curation is a key-point that guarantees a high quality in today's digital libraries. Arguably, one of the best bio-medical databases for curated associations is the Comparative Toxicogenomics Database[1] (CTD). In the case of DDAs, information about the specific type of a DDA association is curated in the CTD, i.e., either a drug is used to treat a disease, or it induces a disease. Due to the high quality, we use the manually curated drug-disease associations from the CTD as ground truth in our work.

On the one side, manual curation leads to the highest quality, but on the other side it is also time-consuming and often tends to be incomplete [19]. Considering these problems, numerous entity-centric methods have been developed for the automatic extraction and prediction of DDAs [3, 8, 16]. Entity specific similarities form the basis for these approaches, e.g., in the case of active substances similar therapeutic properties can be inferred [8] by calculating a molecular/chemical similarity between different drugs. In addition to these entity-centric approaches, which mostly require information from specialized databases [16], also literature-based methods exist, e.g., like the co-occurrence approach [20] that can be used for detection and prediction of DDAs. Usually, with this approach, entities in the documents are first recognized using Named Entity Recognition. Afterwards, in a next step, a co-occurrence of different entities in documents is counted. The hypothesis is that if two entities co-occur in documents, then a relationship between them can be assumed. Besides, it can be assumed that with an increasing number of co-occurrences, the probability of an association also increases [20]. In our work, we use the co-occurrence approach in combination with a retrospective analysis to determine DDA predictions. Newer literature-based techniques in the field of neural

[1] http://ctdbase.org/

embedding models use state-of-the-art approaches like Word2Vec [6, 11] to efficiently learn and predict semantic associations based on word contexts [21]. In comparison to a co-occurrence approach, entities do not necessarily have to occur in the same document but rather can have similar word contexts. Therefore, we use the Word2Vec (Skip-gram) implementation from the open source Deep-Learning-for-Java[2] library in our investigation. Recent work in the field of NLP shows that the use of word embeddings in various NLP classification tasks outperforms previous (classical) approaches [13]. That finding led to an increasing interest using word embeddings also in the bio-medical field [2]. For example, in the work of Patrick et al. [25], the authors are using word embeddings as features to predict a therapeutic effect of drugs on a specific group of diseases. In our case, we not only try to predict a therapeutic association but also whether a drug can cause a disease in the sense of a side effect. Furthermore, we do not limit the body of documents or our evaluation to certain pharmaceutical entities, but consider all DDAs curated in the CTD.

Neural embedding models like Word2Vec are generally considered to be difficult to interpret [26, 27]. Due to this problem, one branch of research focuses on the creation of a semantically optimized embedding space to increase interpretability. In this context new approaches like [26,27] rely on transforming the original embedding space into a semantically optimized space using transformation matrices. For these approaches a large amount of expert knowledge is required to train/create the transformation matrices. A disadvantage of these approaches is that semantic information available in the original space can be lost [26, 27] and furthermore the use of expert knowledge influences the original word-semantics. Such a modified space therefore hardly allows any conclusions about the entity semantics that are present in the document corpus. Our aim is not only to extract an optimized space, but also to investigate to what extent semantics have been learned in the original embedding space, which in turn allows conclusions regarding the content of the corpus. For these reasons we use a feature selection based approach [30] to extract a semantically optimized space from the original embedding space. The main idea is to find a combination of dimensions (features) in which a certain entity-semantic is more pronounced. However, the approach was only presented for entities of the same type (active substances) and not for binary associations as in our case. Therefore, our goal is to adapt the approach to binary associations like DDAs.

3 Problem Formulation and Methodology

In this section, we first define the problem and provide definitions to accomplish our goal: a deep-learning based approach to disambiguate semantic relations between entities in the biomedical field. Hereafter, we present a method to extract semantic subspaces in which, e.g., drug-disease associations of the type “treats” are more likely.

[2] https://deeplearning4j.org/

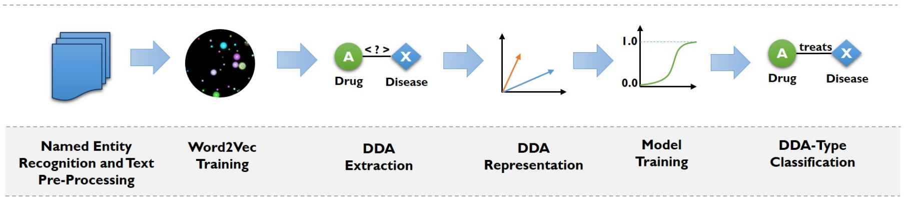

Fig. 1 Method overview: We start with NER and text pre-processing, followed by Word2Vec training. Afterward, we extract Drug-Disease Associations from the embedding space. As next, we generate different DDA representations and afterward we use them for training a deep-learning model in order to disambiguate a DDA-type.

3.1 Semantic Classification of Drug-Disease Relationships

Problem Definition: Given a neural embedding model M over a suitable collection of pharmaceutical documents, two n-dimensional entity embeddings $Drugs_M$ and $Diseases_M$ can be learned by embedding techniques (e.g., Word2Vec), such that:

- $Drugs_M$ collects all entity representations where the respective entities correspond to drugs identified by some controlled vocabulary (e.g., MeSH identifier)
- $Diseases_M$ collects all entity representations corresponding to diseases again identified by some controlled vocabulary.

Then, given a set Rel of labels $r1, \ldots, rk$ for possible and clearly disambiguated semantic relationships between drugs and diseases (e.g., treats, induces, etc.), the semantic classification problem of drug-disease relationships means to learn a classifier $f : \mathbb{R}^n \times \mathbb{R}^n \to Rel$ with $(e_i, e_j) \mapsto r_m$, where $e_i \in Drugs_M$, $e_j \in Diseases_M$, and $r_m \in Rel$.

Since there is a variety of alternatives to learn such a classification function from data in a supervised fashion, and we will explore some in this work. In the following, we present our approach as well as briefly describe the used deep-learning models and provide the details of their implementation in our experiments.

Methodology: Our method consists of the following six steps (Fig. 1):

Steps 1-2. Text Pre-processing and NEM Learning. Our document corpus consists of scientific publications from the medical field. First, we remove stopwords and apply Named Entity Recognition to identify drug- as well as disease entities in the documents. After this initial pre-processing we train a Neural Embedding Model (i.e., Word2Vec) on this corpus resulting in a matrix of entity embeddings.

Step 3. Drug-Disease Association Extraction. Using a k-Nearest-Neighbor approach we extract for a given drug-entity the k-Nearest-Disease-Neighbors. For example, for $k = 10$ we will extract ten DDAs, where the association-type i.e. “treats” or “induces” is unknown.

Step 4. DDA-Representations. In our investigations we create different DDA representations for model training. For example, given a pair of entity-vectors (e_i, e_j) we create a (DDA) representation by i.e. concatenating or averaging the two vectors. Using taxonomic information from pharmaceutical classifications systems, we also create a semantic representation of a DDA.

Steps 5-6. Model Training and DDA-Type Classification. Next, we use the different representations to train a deep-learning model to classify a DDA-type. In this context, we investigate the following two deep-learning models:

- Multilayer Perceptron (MLP): The multilayer perceptron represents one of the most straightforward architectures available. Fixed length input vectors are handed over from layer to layer sequentially, and no recursion is used.
- Convolutional Neural Networks (CNN): Convolutional neural networks (CNN) are known best for their use on image data, such as object detection, handwritten character classification, or even face recognition. Recently, they have also shown to achieve state of the art results on natural language tasks. Goodfellow, I. et al. [22] have emphasized that three essential ideas motivate the use of CNNs in different machine learning tasks, including our disambiguation task: sparse interactions, parameter sharing, and equivariant representations. For our task, sparse interactions allow for learning automatically -without manual feature engineering- patterns from d-dimensional spaces; parameter sharing influences computation storage requirements; equivariant representation allows for robustness in the patterns learned.

3.2 Identifying Semantic Subspaces

The general problem with using projections to create a semantically optimized subspace from a d-dimensional one is that $2^d - 1$ possible combinations exist. To efficiently identify a semantic subspace we use the method presented in [30] but in addition we adapt it for DDAs. In the following we describe the method and our adjustments.

As already described in section 2, the approach is based on a feature selection by means of regression approaches. In this context, the authors of [30] first adapt the notation of the general Multiple Linear Regression Model $(y = X\beta + \varepsilon)$ as follows:

$$y_s = W\beta + \varepsilon$$

where y_s represents a vector of semantic observations y_{s_i} $(1 \leq i \leq n)$, W represents a word embedding matrix with the predictor variables (dimensions) in the columns, β represents a $(n+1)$-dimensional vector with regression coefficients, and ε represents an n-dimensional vector with the error terms.

Here, the dimensions of the embedding matrix $W$are regarded as the actual features (predictors). The n-dimensional vector y_s is also called semantic vector. Each entry of y_s contains the relative position of an entity e_i so that semantics can be expressed between the entities: Entities with similar properties (e.g., therapeutic) are positioned close to each other while entities with dissimilar properties are at most far away from each other. The hypothesis is that a combination of the predictors, which can best predict the observations [28, 29], form the semantic subspace [30]. In the desired subspace, the similarity $sim_A(e_i, e_j)$, e.g. cosine-similarity between a pair of embedded entities $e_i, e_j \in E$, develops proportionally to a second similarity $sim_B(e_i, e_j)$, e.g. a therapeutic similarity between this pair [30]. Whereby, taxonomic information was used in [30] to first calculate a semantic representation of the entities. In the next sections we denote $sim_A(e_i, e_j)$ as a similarity measure between a pair of entity-embeddings and $sim_B(e_i, e_j)$ as a *binary similarity measure* that expresses if an association is of a certain type. Overall, we can summarize the approach, including our adjustments for DDAs, in the following three steps:

Step-1: In the first two steps the semantic vector y_s is generated. For this purpose, first a dimension is chosen in which the proportionality between $sim_A(e_i, e_j)$ and $sim_B(e_i, e_j)$ is best represented. To determine this proportionality the mean squared error is calculated for all dimensions and for each pair of entities (DDA) as follows:

$$\frac{1}{n\,(n-1)} \sum_{e_i, e_j \in E} \left(sim_A(e_i, e_j) - sim_B(e_i, e_j)\right)^2$$

Since the MSE is determined for each dimension separately, we use the Euclidean Distance in the one-dimensional space for $sim_A(e_i, e_j)$. As described in [30] we perform also a rescaling, so that a value of 1 means that two entities have no distance to each other and a value of 0 means that a pair of entities is maximally far away from each other. At this point we adapt the approach for the binary associations (DDAs) and we calculate $sim_B(e_i, e_j)$ per semantic (treats/induces): To determine a therapeutic subspace, we set the value for each DDA which is of the type "treats" to 1 (is similar), otherwise to 0 (not similar). We proceed analogously when we calculate an "induces" subspace. Here we set the value to 1 if the DDA is of type "induces" and to the value 0 otherwise. Therefore, this step is done separately for the treats and induces associations so that one dimension per association type is determined.

Step-2: To identify the entities that show best average pairwise proportionality in the selected dimension, we first determine a list of DDAs that fulfil the following condition:

$$|sim_A(e_i, e_j) - sim_B(e_i, e_j)|^2 \leq \lambda$$

where λ represents a threshold. Afterward, we select the top-k entities that appear most frequently in this list. These entities form the sample (y_s vector) for the following training step. As shown in [30] the previous steps are essential for the quality of the semantic subspace.

Step-3: In the last step, the generated sample is used to determine the dimensions that can best predict the observations using Least Angle Regression (LAR). For this step we first remove the dimension used for the semantic vector generation and train LAR with the remaining dimensions. LAR has the additional advantage of ranking the features according to their predictive power [28, 29]. This is expressed in the coefficient values and predictors with a coefficient value of $\beta_i = 0$ can be removed. To improve the (subspace) quality further, the authors of [30] remove also predictors with low (absolute) coefficient values. For this task they propose a standard deviation heuristic. First, they calculate the standard deviation of all coefficients. Afterwards they select only predictors x_i where $|\beta_i| \geq \alpha * sd, \alpha \in R$. The remaining dimensions form the semantically optimized subspace.

4 Experimental Investigation

In this section, we will first describe our pharmaceutical text corpus and the necessary experimental set-up decisions. Afterward, we define for each research-question the quality criteria that our proposed models should fulfill, followed by our evaluation.

Experimental Setup.

Evaluation corpus. In the biomedical field PubMed[3] is one of the most comprehensive digital libraries. For the most publications a full-text access is not available and therefore we collected only abstracts for our experiments. Furthermore, all collected abstracts were published between 1900-01-01 and 2019-06-01. Word embedding algorithms usually train on single words, resulting in one vector per word and not per entity. This is a problem, because disease and drug names often consist of several words (e.g., ovarian cancer). Therefore, we first use PubTator[4] to identify the entities in documents. Afterward, we place a unique identifier at the entity's position in the text. Retrospective Analysis Evaluation Corpora. In order to detect predicted DDAs using a retrospective analysis, we divide our evaluation corpus into two corpora: 1900.01.01-1988.31.12 (1989 corpus) and 1900.01.01-2019.06.01 (2019 corpus). Each corpus contains only the documents for the respective time period.

Query Entities. As query entities for the evaluation, we selected all Drugs from the DrugBank[5] collection, which can be also be found (using a MeSH-Id) in CTD Database[6] as well as in the pre-processed documents. Therefore, our final document set for evaluation contains 29 million abstracts for 1700 drugs. As ground truth, we selected for each drug all manually curated drug-disease associations from CTD, resulting in a data set of 33541 inducing and 18664 therapeutic drug-disease associations.

Experiment implementation and parameter settings.

[3] https://www.ncbi.nlm.nih.gov/pubmed/

[4] https://www.ncbi.nlm.nih.gov/CBBresearch/Lu/Demo/PubTator/

[5] https://www.drugbank.ca/

[6] http://ctdbase.org/

Text Pre-processing. In an initial document pre-processing step, we removed stop-words and performed stemming using Lucene's Porter Stemmer[7] implementation. Here we made sure that the drug and disease identifiers were not affected.

Word Embeddings. After document pre-processing, word embeddings were created with DeepLearning4J's Word2Vec[8] implementation. A larger window-size can lead to improved results in learning (pharmaceutical) associations [2,18]. Therefore, to train the neural embedding model, we set the word window size in our investigations to 50. Further, we set the layer size to 200 features per word, and we used a minimum word frequency of 5 occurrences.

Similarity-Measure. To measure a similarity between drug and disease embeddings we choose cosine similarity in all experiments. A value of 1 between two vectors means a perfect similarity, and the value 0 means a maximum dissimilarity. An exception are the experiments for the semantic subspaces (Section 4.5). Here, we additionally use the Euclidean distance in one-dimensional spaces. However, we perform a rescaling, so that a value of 1 means that two entities have no distance to each other and a value of 0 means that a pair of entities is maximally far away from each other.

Model Training and Evaluation Settings. For all experiments related to the disambiguation task, cross-validation is applied by creating ten identical models, where each of them is trained on a randomly selected and balanced data set, containing 50% inducing associations and 50% therapeutic associations. The selected data set is then randomly split into 90% training data, and 10% test data in a stratified way, meaning both training and test set will also consist of 50% inducing associations and 50% therapeutic associations. As a measure for the performance of the model, the average test accuracy of the ten models on each test data set is measured. For the neural networks, the average of the maximum accuracy overall epochs of all models will serve as the measure for comparison.

4.1 Quality Criteria

First, we investigate whether, and to what extent, word-embeddings are suitable for learning the DDA type (i.e. "treats", "induces"). In addition, we investigate if latent features can be learned and therefore dimensions where semantics of an association type are probably expressed. In this context, the following quality criterion should be fulfilled to answer RQ1:

- *Disambiguation Suitability:* If latent features exist in an entity-vector representation, that indicate the type of a DDA, then Machine-Learning approaches that are able to weight certain features higher should lead to better results (i.e. increased classification accuracy) compared to methods

[7] https://lucene.apache.org/

[8] https://deeplearning4j.org/word2vec

that asses all features equally. In this context, a sufficiently good quantitative result (i.e. high accuracy) should be achieved.

Distance between two embedded entities can affect the quality of a DDA predictions [17]. Therefore, we investigate whether distance can also affect the quality of a DDA-type prediction. In this context, the following criterion should be fulfilled to answer RQ2:

- *Disambiguation Stability:* As the distance between a drug and a disease increases, the accuracy disambiguating a DDA should not decrease substantially. This would further indicate that a certain latent subspace has been learned by the Deep-Learning approaches. Therefore, with increasing distance between a drug and a disease vector-representation, always a sufficiently good quantitative result (i.e. high classification accuracy) should be achieved.

In our next investigation, we propose semantically enhanced deep learning models that can determine the type of (real) DDA predictions. In this context, the following criterion should be fulfilled:

- *Prediction Accuracy with Semantic Enhancement*: So far, we have only evaluated the classification of existing DDAs on current datasets, i.e. DDAs and their association type are available in the CTD and in addition the DDAs can be found in publications. However, our primary interest is to classify the type of (real) DDA predictions. For this task, our proposed semantic-enhanced deep learning models should lead to improved results compared to modes trained without semantic information.

In our last investigation, we apply and investigate the proposed approach for the identification of a latent feature space from the original embedding space. In this context, the following criterion should be fulfilled:

- *Semantic Subspace Properties*: The desired semantic subspace should have the property that DDA semantics are more pronounced compared to the original embedding space. More pronounced means that, e.g., in a therapeutic subspace the probability of a therapeutic DDA is not only proportional to the similarity (e.g., cosine similarity) in the space but in addition, this probability should increase faster compared to the original embedding space. On the other side, for low similarity values (e.g., < 0.5) the probability should decrease faster in comparison to the original space. The probability can be determined by calculating the proportion of *therapeutic* and *induces* associations in a particular similarity range.

4.2 General Suitability of Word Embeddings to Disambiguate Drug-Disease Associations

In our first experiment, we investigate RQ1 and the hypothesis that latent information about the type of a DDA is encoded in certain areas of the vectors, hence certain dimensions, and can be learned using deep learning approaches. In this context, we will first describe the used data-set followed by our Baseline description. Afterward, we describe the used Deep-Learning models as well as their implementation details followed by our experiments.

Experimental Data-Set. In this experiment, we use all drug-disease vectors that can also be found as a curated drug-disease association in CTD. Thus, our data set contains 33541 inducing associations as well as 18664 therapeutic associations. Since the classes are not strongly skewed, representative results can be expected when training on a balanced data set. Therefore, our data-set contains 18644 therapeutic as well as the same number of induce associations.

Baseline Construction. The work of Lev et al. [13] demonstrates that vector pooling techniques, applied to Word2Vec vectors, can outperform various literature-based algorithms in different NLP tasks and therefore can be seen as a method to construct strong baselines. As an example, a common pooling technique is the calculation of a mean vector v for N different vectors with:

$$v = \frac{1}{N}\sum_{i=1}^{n} x_i$$

In our case, we use a drug vector and a disease vector for the pooling approach. However, since this ‘mean’-pooling approach, resulting in 200-dimensional vectors, blurs part of the information contained in the entity vectors, two more approaches will be tested. For the ’concat’-pooling, both vectors are concatenated, resulting in 400-dimensional association vectors. Finally, for the ’stack’-polling, the drug and disease vectors are stacked, resulting in association matrices of the shape 200x2. Afterward, using our new vector representations, we train a scikit-learn’s Support Vector Classifier (SVC) on our data-set to learn the drug-disease association types “treats” and “induces”. As for the SVC-parameters, a degree of 3 has proven to be the best choice in our experiments. In addition, the kernel is a radial basis function and for all other parameters we used the default values. Next, we describe the investigated Deep-Learning models and their implementation details.

Multilayer Perceptron (MLP). Using Keras’ Sequential class, this model consists of three densely connected layers of decreasing size. The first two layers use a reactive linear unit as the activation function. The final layer uses a sigmoid function to produce the binary classification output and the loss function used is a binary cross-entropy loss function. The optimizer is Adam [23], with a learning rate of 1e-4 and the batch size is set to 8.

Convolutional Neural Network (CNN). This model is built on Keras’ Sequential class as well and also uses a single-neuron sigmoid layer as the final layer. The hidden layers describe an underlying CNN architecture. A dropout [24] layer with a rate of 0.1 is applied to combat overfitting. The kernel size is set to value of 3 for the ’mean’ and ’concat’ vectors and 3x2 for the ’stack’ vectors. Similar to the MLP, the loss function is a binary cross-entropy loss function, and the optimizer is Adam [23] with a learning rate of 1e-4. The batch size is eight, as well.

For a comparison, the two deep-learning models were trained with the different pooling vector-representations, where the "stack" vectors were exclusively used for training the CNN. The average Accuracies in a 10-fold cross validation are presented in Table 1:

Table 1 Accuracies achieved with different models and pooling-approaches. Best values in bold.

	mean	concat	stack
SVC	71.84	**73.84**	-
MLP	80.75	**81.64**	-
CNN	80.40	**81.61**	81.54

Results and Results Interpretation. We can observe in Table 1 that both deep learning approaches learned a feature space capable of disambiguating between "treats" and "induces". Moreover, the concatenation of the vectors delivered the best results overall, achieving more than 81% of accuracy. Compared to the baseline, we can recognize an increase in accuracy of up to 8%. In summary, we obtained empirical evidence that answers our first research question: deep learning models can find a latent space to disambiguate associations between pharmaceutical entities.

However, further investigation is needed to assess their performance. In particular, given the findings of [17] regarding the impact of the distance in the embedding space between entities to find associations, we would like to test the stability of our proposed methods in the following section.

4.3 Distance Relationship and Learning

In our next experiment, we investigate the fact that with increasing entity distance, the accuracy quality using k-NN approaches can decrease [17]. Therefore, we assume that with increasing distance also the semantic disambiguation accuracy (SDA) might show the same characteristic, and thus we lose the information for classifying a DDA Type.

Evaluation Dataset. To test this assumption, we select for each drug the k-nearest disease neighbors (k-NDNs), where $k = 10, 20, 50$. With increasing k also the distance between two entity-vectors will increase. In addition, we test only with DDAs that are curated in the CTD i.e. the association type in known. This selection results in data sets of the following sizes:

Since the amount of training data will probably influence a model's accuracy, each model is trained and tested on data sets of equal size to achieve comparable results. Each subset of each data set will, therefore, contain exact 688 inducing as well as 688 therapeutic associations. We use the same models as in the previous experiment in combination with a concatenation approach. The results achieved with a 10-fold cross validation are presented in Table 3:

Table 2 Evaluation Datasets for distance related evaluation.

k-NDNs	10	20	50
inducing	688	1172	2264
therapeutic	1939	3044	4812

Table 3 Investigation of the influence of distance for DDA-type disambiguation. Accuracies achieved for different k-Nearest-Disease-Neighbors sets.

k-NDNs	10	20	50
SVC	77.32	72.97	73.26
MLP	82.61	80.58	77.17
CNN	81.38	79.93	79.93

Results and Result Interpretations. We can observe in Table 3 that the SVC accuracy has a declining tendency with an increasing number of NDNs. The effect from 10 to 50 NDNs reaches up to 4% for the SVC. The MLP model shows similar results with a decreasing rate in accuracy of 5%. Finally, with a change of 2% the CNN model shows a rather stable performance. This is a somewhat surprising result, given that the dataset used is small, which tends to lead to more volatile results in the model's accuracy. A possible explanation of the stability of the CNN model could be found in the rationale behind using CNNs in machine learning tasks. In Goodfellow, I. et al. [22], in particular, two properties from CNNs: sparse interactions and equivariant representation. For our task, the results confirm that sparse interactions allow for learning automatically -without manual feature engineering- patterns from d-dimensional spaces and equivariant representation allows for robustness in the patterns learned.

Given our experimental findings, we can claim that the performance of the CNN model is not affected by the distance in the original embedding space. Thus, our model has learned a robust latent representation that succeeds to disambiguate associations between entities. In the next section, we build on our findings to assess the impact of incorporating domain knowledge by introducing a hybrid neural network architecture and performing a sophisticated retrospective analysis to assess model performance when facing real DDA predictions.

4.4 Impact of Domain Knowledge to Disambiguate Predicted Drug-Disease Associations

In our previous sections, we have empirically proven that DDAs extracted from the year 2019 can indeed be classified with higher accuracy. In this section, we will verify that this is not only true for already discovered/existing DDAs, but also for real predictions using retrospective analysis. Moreover, we answer our third research question in the difficult task of real predictions by introducing domain knowledge into the models. In this context, we first describe how we identify real DDA predictions. Hereafter, we present an approach to create a semantic representation of a DDA using medical classification systems. Afterward, we present our proposed semantically enriched Deep-Learning models for DDA type disambiguation and compare the results with all previously tested models. With this retrospective approach, we want to simulate today's situation, where we have on the one hand possible DDA predictions and in addition we have access to rich taxonomic data that can be used as a source for semantic information.

Evaluation Dataset for Retrospective Analysis. To detect real predictions, first we train our Word2Vec model with the historical corpus (Publication date $<$ 1989). Next, we extract DDAs from the resulting embedded space using a k-Nearest-Neighbor approach. Afterward, using a co-occurrence approach [20], we first check if a DDA does not exists, i.e., does not appear in at least three publications in our historical corpus. Then we check if a non-existing DDAs will appear in the actual corpus 2019 and/or can also be found in CTD. With this approach, we identify DDA predictions within the k-NDNs sets of each drug (where $k = 10, 20, 50$). In addition, we identify and train our proposed models on existing DDAs (DDA appears in documents where publication date $<$ 1989) and test these models with the predicted DDAs sets. This yields to the data sets shown in Table 4.

Table 4 Number of real predicted as well as existing DDAs extracted using a k-Nearest-Disease-Neighbors (k-NDN) approach.

	k-NDNs	10	20	50
Test (predicted)	inducing	88	168	402
	therapeutic	91	195	474
Train (existing)	inducing	687	1161	2311
	therapeutic	1408	2182	3535

Entity Specific Semantic Information. In order to semantically enrich drug as well as disease entities, we use (medical) classification systems as a source. Considering pharmaceutical entities, there are a couple of popular classification systems such as the Medical Subject Headings (MeSH) Trees[9] or the Anatomical Therapeutic Chemical (ATC) Classification System[10]. The ATC subdivides drugs (hierarchically) according to their anatomical, therapeutic/pharmacologic, and chemical features. For example, the cancer related drug 'Cisplatin' has the class label 'L01XA01'. In this context the first letter indicates the anatomical main group, where 'L' stands for 'Antineoplastic and immunomodulating agents'. The next level consists of two digits '01' expressing the therapeutic subgroup 'Antineoplastic agents'. Each further level classifies the object even more precisely, until the finest level usually uniquely

[9] https://www.nlm.nih.gov/mesh/intro_trees.html

[10] https://www.whocc.no/atc_ddd_index/

identifies a drug. We use the ATC-Classification system exclusively for drug-entities. To collect semantic information also for diseases as well as for drugs we use the Medical Subject Headings (MeSH). MeSH is a controlled vocabulary with a hierarchical structure and serves as general classification system for biomedical entities (Table 5).

Table 5 Example: Classes in different classification systems for the drug 'Cisplatin'.

Classification System	Assigned Classes
ATC	L01XA01
MeSH Trees	D01.210.375, D01.625.125, D01.710.10

Semantically Enriched DDA Representations. In our previous experiments we have generated a DDA vector representation by concatenation of a drug vector with a disease vector. Afterwards we trained the different approaches with these DDA representations. In order to additionally use semantic information from classification systems, we also need a DDA vector representation using the MeSH trees and the ATC classes. For this task we proceed as follows: MeSH trees consist of a letter in the beginning, followed by groups of digits from 0 to 9, separated by dots (Table 5). The leading letter signifies one of 16 categories, e.g., C for diseases and D for drugs and chemicals. Since the categories will not be mixed in any data sets, this letter can be omitted in our approach. Furthermore, the dots separating the levels provide no additional information, since each level is of the same length. After removing the dots and leading letters, a string of digits remains. To use this string as meaningful input to the neural networks, two more steps are necessary. First, the strings need to be of equal length. This is achieved through the padding of all strings, which are not of the maximum occurring length with zeros. Finally, since we will have a sparse representation, we use a Keras embedding layer to build a latent (dense) representation for each MeSH Tree. The structure of the ATC class labels consists of a string, containing both letters and digits. To prepare the class-strings for processing in a neural network, we mapped each character to a unique number. The resulting vector of numbers is then padded with zeros in the same way as the PubMed Vectors to achieve equal lengths. The resulting vectors can then be processed through an embedding layer to build a latent representation as well.

Semantically Enriched Deep-Learning Models. At this point we have two semantically different DDA representations. To achieve meaningful processing of these two fundamentally different information sources, we introduce a new type of neural network architecture. We will refer to it hereafter as a "hybrid network". Herein, a hybrid network is a neural network that is split at the input level and the first hidden layer and then merged into a single network through a merging layer, which is followed by further hidden layers and, finally, the output layer. By using the previously investigated MLP and CNN networks we propose and evaluate the following three hybrid architectures:

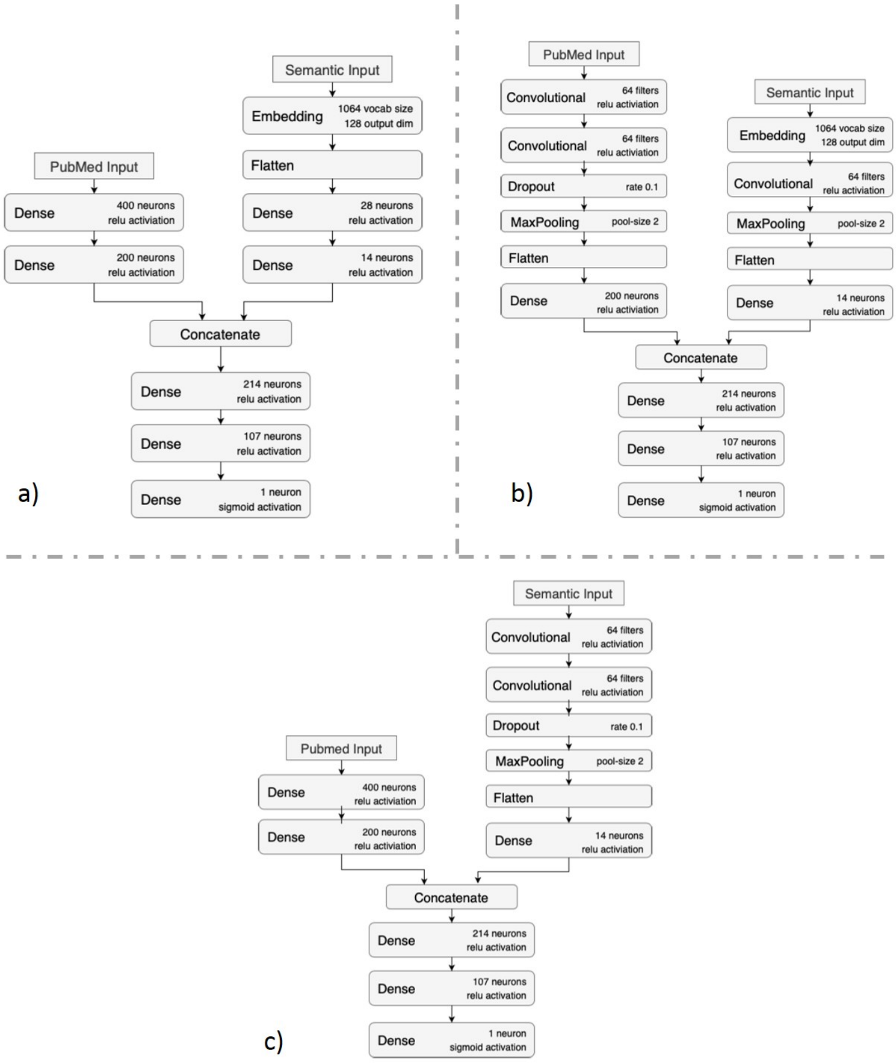
a)
Semantic Input
Embedding 1064 vocab size 128 output dim
Flatten
Dense 28 neurons relu activation
Dense 14 neurons relu activation
PubMed Input
Dense 400 neurons relu activiation
Dense 200 neurons relu activiation
Concatenate
Dense 214 neurons relu activation
Dense 107 neurons relu activation
Dense 1 neuron sigmoid activation
b)
PubMed Input
Convolutional 64 filters relu activiation
Convolutional 64 filters relu activiation
Dropout rate 0.1
MaxPooling pool-size 2
Flatten
Dense 200 neurons relu activiation
Semantic Input
Embedding 1064 vocab size 128 output dim
Convolutional 64 filters relu activiation
MaxPooling pool-size 2
Flatten
Dense 14 neurons relu activiation
Concatenate
Dense 214 neurons relu activation
Dense 107 neurons relu activation
Dense 1 neuron sigmoid activation
c)
Semantic Input
Convolutional 64 filters relu activiation
Convolutional 64 filters relu activiation
Dropout rate 0.1
MaxPooling pool-size 2
Flatten
Dense 14 neurons relu activiation
Pubmed Input
Dense 400 neurons relu activiation
Dense 200 neurons relu activiation
Concatenate
Dense 214 neurons relu activation
Dense 107 neurons relu activation
Dense 1 neuron sigmoid activation

Fig. 2 Hybrid Architectures Detail Overview. As an input for the left branch (PubMed Input) Word2Vec DDA representations (Concatenated vectors) were used and for the right branch (Semantic Input) we used the semantic DDA representations.

- Hybrid MLP (Fig. 2a.) - This architecture consists of two symmetrical densely connected networks that merge into one densely connected network through a concatenation layer.
- Hybrid CNN (Fig. 2b.) - The lower part of this network is identical to the Hybrid MLP. The two upper branches, however, have been replaced by CNNs.

– Hybrid CNN & MLP (Fig. 2c.) - This architecture is a mix of the previously introduced hybrid networks. The semantic input is pre-processed through a CNN, while PubMed vectors are pre-processed through an MLP.

Model Evaluation and Result Interpretation. For the one branch of a hybrid network, we will use the NEM (PubMed) vector representation of a DDA as input and for the other branch, we will use the semantic DDA representation as input. In this context, drugs, as well as diseases, can be assigned to multiple MeSH trees and ATC classes. If there are multiple MeSH trees or ATC classes for a drug or a disease, an additional semantic representation is created for each possible combination of MeSH trees and ATC classes. Therefore, in such cases for the same DDA (PubMed) vector representation we can have multiple semantic DDA representations as input for our enhanced models. Furthermore, to avoid overfitting the learning rate is adjusted to 5e-5 for the Hybrid MLP and to 2e-5 for the Hybrid CNN. We compare our enhanced models with all previously introduced models (Table 6). We show in Table 6 that our proposed semantic enhanced models lead to overall improved results for all investigated datasets, including the previously presented models.

Table 6 Accuracies of the different approaches achieved for DDAs extracted from different k-Nearest-Disease-Neighbors sets. Best values in bold.

k-NDNs	10	20	50
SVC	66.93	70.54	71.53
MLP	74.83	76.25	77.05
CNN	75.74	74.79	76.44
Hybrid MLP	**76.80**	79.14	77.44
Hybrid CNN	75.99	**80.60**	**77.65**
Hybrid CNN & MLP	75.34	79.46	77.21

Compared to the SVC baseline, Accuracy can be improved by up to 10% (Hybrid CNN, k = 20). Moreover, in comparison to the deep-learning models we achieve improvements of Accuracy by up to 6% (CNN vs. Hybrid CNN). We conclude from this experiment that semantic information allows substantial improvements in the prediction of the DDA type.

Furthermore, CNNs are less sensitive to distances (Table 3) and, besides, the Hybrid-CNN leads to the best average values. We conclude from this observation, that (hybrid) CNNs are generally better suited for the classification of the DDA type. Compared to the previous experiments (Table 3, results for SVC, MLP and CNN) the accuracy-results are generally lower although the number of training samples is comparable (see Table 2 and 4). We therefore connect this effect mainly to the different sizes of training corpora that were used for the Word2Vec training. For example, the historical corpus of 1989 is many times smaller than the corpus of 2019, which can lead to averagely worse contextualized entities after Word2Vec training. We conclude from this, that if the contextualization quality decreases, this also hurts classification accuracy.

4.5 Identification of a Semantic Subspace

In the previous sections we have shown that latent information about the type of a DDA in the embedding space probably exists and in addition that the strength of this information decreases with increasing distance. In this section we want to investigate and extract the latent semantic space which probably contains a particular semantic information. In this context we first describe our evaluation dataset followed by the evaluation implementation. Finally, we present the results followed by an interpretation and discussion.

Evaluation Dataset: For the evaluation we use the embedding space and the CTD associations (33541 inducing and 18664 therapeutic) from the previous experiments. In total, our entity set contains for this experiment ~ 4700 individual drug and disease entities for this experiment.

Evaluation Implementation: For our evaluation we implement the approach as described in section 3.3. Here we perform the approach separately for "treats" and "induces" associations and determine 10% of all entities (~ 470) for the different semantic vectors. The remaining rest of the entities and their associations are used exclusively for testing the subspaces. We set the parameter λ to the value 0.0001. With this parameter we can regulate the number of training entities and the MSE: Lower parameter value results in fewer training entities but also in a lower MSE. Afterward, we used the same Scikit's[11] implementation of LAR with Cross-Validation and in Lasso-Mode as used in [30] for training. Here, we performed a 10-fold cross-validation with 200 iterations. Other parameters were not set for the LAR training. For the therapeutic subspace a total of 114 of the 199 dimensions were determined. Setting the standard-deviation parameter α for the therapeutic subspace to 0.185, filtered out 15 more dimensions and led to minor improvements in our evaluation. For the “induces” subspace no improvement could be achieved with a certain α parameter.

Similarity-Measures: With an increasing number of similar contexts the cosine similarity between two words [31] as well as for pharmaceutical associations like DDAs [17] increases. Whereby the cosine-similarity usually ranges between 0 and 1 for word pairs [31] as well as for DDAs [17]. A value of 0 means that the two vectors are orthogonal to each other and a value of 1 means that they lie exactly on each other. It is important to know that for a cosine similarity values of < 0.3, between a drug and disease embedding, a DDA is highly unlikely [17] or in short: A drug and a disease have probably no common contexts. Therefore, we limit our investigation to similarity values between [0.3, 1.0]. In 0.1 steps we determine all DDAs of a respective interval using the k-NDNs approach that was already described in our previous experiments. An exception is the interval of [0.8, 1.0]. Here we had to use a larger interval in order to identify enough DDAs (> 50) that are also present in our CTD dataset since only few (CTD) associations exist (< 5) with a similarity of

[11] https://scikit-learn.org

$>=$ 0.9. Afterwards, we measured the proportion of therapeutic and induced associations in each interval.

Results and Result Interpretation: In Figure 3 we present the results only for the therapeutic subspace, as we were not able to extract an "induces" subspace with the desired properties despite different parameter settings and y_s vectors. We ascribe this mainly to two reasons: 1) The method we adapted is not sensitive enough to detect such a latent semantic subspace and 2) the therapeutic semantic is the dominant semantic in the original space and thus overlays the induces-associations.

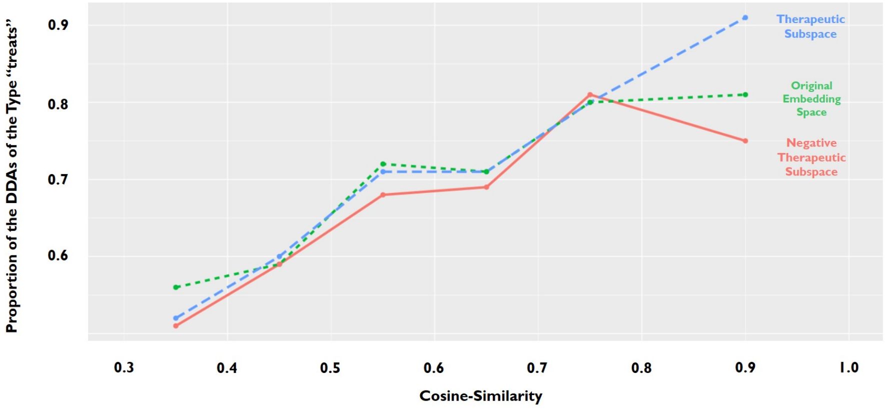

Fig. 3 Proportion of therapeutic DDAs in the therapeutic subspace compared to the original embedding space and to the negative therapeutic subspace (All filtered out dimensions). Points show the average values of each interval.

On the other side, as shown in Figure 3, the therapeutic subspace has the desired properties: With high cosine similarity values, the probability of therapeutic-associations increases faster compared to the original embedding space. At lower values, the probability for a therapeutic association decreases faster. For higher similarity values (≥ 0.8) we can retrieve 10% more therapeutic associations. For comparison, we also show the subspace which is created by the non-projected dimensions (100 in total) and which we call the "negative" therapeutic subspace here. As can be seen, the proportion of therapeutic associations is lowest in almost all intervals. Therefore, we assume that the dimensions in which the therapeutic semantics are less represented, have actually been removed.

Comparing both of the associations in the original embedding space, the proportion of therapeutic DDAs increases with increasing similarity much stronger than in comparison to the induced associations. This strong imbalance is all the more surprising when one considers that our CTD dataset contains almost twice as many *induces* associations as *therapeutic* ones. We conclude from this result that therapeutic associations dominate in medical publications and are much better described. This probably also affects also the semantic quality of our y_s-vectors and probably the desired proportionality is better reflected for threats-associations. This quality-difference also seems to be reflected in the number of publications that can be found in PubMed for the terms "treats" and "induces". For the term "treats", we get almost 10 million results, whereas for "induces" we get only 3 million results.

5 Conclusion and Future Work

In this paper, we addressed the central question of finding interpretable semantics from Neural Embedding Models (NEMs) to disambiguate associations such as “treats” and “induces” between pharmaceutical entities. To do so, we explored the use of deep learning models to learn a latent feature space from the NEMs and performed an in-depth analysis of the performance of the models. We found that the deep learning models are stable in the sense that they learned a latent feature space that successfully delivers an accuracy of up to 80%. Moreover, we proposed Hybrid Deep-Learning models that incorporate domain knowledge. With our retrospective analysis, we showed that these models are robust and lead to improved performance for real DDA predictions.

In our last section we extracted and investigated a latent semantic subspace from the original embedding space using a feature selection approach adapted for DDAs. We were able to identify a therapeutic subspace in which DDAs of the type "treats" become more probable with increasing cosine-similarity. Therefore, we assume that such a subspace is generally better interpretable in regard to therapeutic associations.

References

1. Gefen, D., Miller, J., Armstrong, J. K., Cornelius, F. H., Robertson, N., Smith-McLallen, A., & Taylor, J. A. (2018). Identifying patterns in medical records through latent semantic analysis. Communications of the ACM, 61(6), 72-77.
2. Chiu, B., Crichton, G., Korhonen, A., & Pyysalo, S. (2016, August). How to train good word embeddings for biomedical NLP. In Proceedings of the 15th workshop on biomedical natural language processing (pp. 166-174).
3. Chiang, A. P., & Butte, A. J. (2009). Systematic evaluation of drug–disease relationships to identify leads for novel drug uses. Clinical Pharmacology & Therapeutics, 86(5), 507-510.
4. Herskovic, J. R., Tanaka, L. Y., Hersh, W., & Bernstam, E. V. (2007). A day in the life of PubMed: analysis of a typical day's query log. Journal of the American Medical Informatics Association, 14(2), 212-220.
5. Larsen, P. O., & Von Ins, M. (2010). The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index. Scientometrics, 84(3), 575-603.
6. Baroni, M., Dinu, G., & Kruszewski, G. (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 (Volume 1: Long Papers) (Vol. 1, pp. 238-247).
7. Gottlieb, A., Stein, G. Y., Ruppin, E., & Sharan, R. (2011). PREDICT: a method for inferring novel drug indications with application to personalized medicine. Molecular systems biology, 7(1), 496.

8. Zhang, W., Yue, X., Chen, Y., Lin, W., Li, B., Liu, F., & Li, X. (2017, November). Predicting drug-disease associations based on the known association bipartite network. In 2017 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) (pp. 503-509). IEEE.
9. Tshitoyan, V., Dagdelen, J., Weston, L., Dunn, A., Rong, Z., Kononova, O., & Jain, A. (2019). Unsupervised word embeddings capture latent knowledge from materials science literature. Nature, 571(7763), 95.
10. Agarwal, P., & Searls, D. B. (2009). Can literature analysis identify innovation drivers in drug discovery? Nature Reviews Drug Discovery, 8(11), 865.
11. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.
12. Dudley, J. T., Deshpande, T., & Butte, A. J. (2011). Exploiting drug–disease relationships for computational drug repositioning. Briefings in bioinformatics, 12(4), 303-311.
13. Lev, G., Klein, B., & Wolf, L. (2015, June). In defense of word embedding for generic text representation. In International Conference on Applications of Natural Language to Information Systems (pp. 35-50). Springer, Cham
14. Agarwal, P., & Searls, D. B. (2009). Can literature analysis identify innovation drivers in drug discovery? Nature Reviews Drug Discovery, 8(11), 865.
15. Keiser, M. J., Setola, V., Irwin, J. J., Laggner, C., Abbas, A. I., Hufeisen, S. J., & Whaley, R. (2009). Predicting new molecular targets for known drugs. Nature, 462(7270), 175.
16. Lotfi Shahreza, M., Ghadiri, N., Mousavi, S. R., Varshosaz, J., & Green, J. R. (2017). A review of network-based approaches to drug repositioning. Briefings in bioinformatics, bbx017.
17. Wawrzinek, J., & Balke, W. T. (2018, November). Measuring the Semantic World–How to Map Meaning to High-Dimensional Entity Clusters in PubMed? In International Conference on Asian Digital Libraries (pp. 15-27). Springer, Cham.
18. Hill, F., Reichart, R., & Korhonen, A. (2015). Simlex-999: Evaluating semantic models with (genuine) similarity estimation. Computational Linguistics, 41(4), 665-695.
19. Rinaldi, F., Clematide, S., & Hafner, S. (2012, April). Ranking of CTD articles and interactions using the OntoGene pipeline. In Proceedings of the 2012 BioCreative Workshop.
20. Jensen, L. J., Saric, J., & Bork, P. (2006). Literature mining for the biologist: from information retrieval to biological discovery. Nature reviews genetics, 7(2), 119.
21. Mikolov, T., Yih, W. T., & Zweig, G. (2013). Linguistic regularities in continuous space word representations. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 746-751).
22. Goodfellow, I. et al. 2016. Deep Learning–book. MIT Press. 521, 7553 (2016), 800.
23. Kingma, D.P. and Ba, J. 2014. Adam: A Method for Stochastic Optimization. CoRR. abs/1412.6980, (2014).
24. Hinton, G.E. et al. 2012. Improving neural networks by preventing co-adaptation of feature detectors.
25. Patrick, M. T., Raja, K., Miller, K., Sotzen, J., Gudjonsson, J. E., Elder, J. T., & Tsoi, L. C. (2019). Drug Repurposing Prediction for Immune-Mediated Cutaneous Diseases using a Word-Embedding–Based Machine Learning Approach. Journal of Investigative Dermatology, 139(3), 683-691
26. Rothe, S. et al. 2016. Ultradense Word Embeddings by Orthogonal Transformation. Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (San Diego, California, Jun. 2016), 767–777.
27. Jha, K., Wang, Y., Xun, G., & Zhang, A. (2018, November). Interpretable Word Embeddings for Medical Domain. In 2018 IEEE International Conference on Data Mining (ICDM) (pp. 1061-1066). IEEE.
28. Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68(1), 49-67.
29. Efron, B., Hastie, T., Johnstone, I., & Tibshirani, R. (2004). Least angle regression. The Annals of statistics, 32(2), 407-499.

30. Wawrzinek, J., Pinto, J. M. G., & Balke, W. T. Mining Semantic Subspaces to Express Discipline-Specific Similarities. In 2020 ACM/IEEE Joint Conference on Digital Libraries (JCDL). IEEE.
31. Schakel, A. M., & Wilson, B. J. (2015). Measuring word significance using distributed representations of words. arXiv preprint arXiv:1508.02297.
32. Wawrzinek, J., Pinto, J. M. G., Wiehr, O., & Balke, W. T. (2020, September). Semantic Disambiguation of Embedded Drug-Disease Associations using Semantically Enriched Deep-Learning Approaches. In International Conference on Database Systems for Advanced Applications (Forthcoming). Springer, Cham.