Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 3509–3519
Marseille, 11–16 May 2020

Semi-supervised Deep Embedded Clustering with Anomaly Detection for Semantic Frame Induction

Zheng-Xin Yong[1], Tiago Timponi Torrent[2]
[1]Minerva Schools at Keck Graduate Institute, San Francisco, CA - United States
[2]Federal University of Juiz de Fora - FrameNet Brasil, Juiz de Fora, MG - Brazil
zhengxin.yong@minerva.kgi.edu, tiago.torrent@ufjf.edu.br

Abstract

Although FrameNet is recognized as one of the most fine-grained lexical databases, its coverage of lexical units is still limited. To tackle this issue, we propose a two-step frame induction process: for a set of lexical units not yet present in Berkeley FrameNet data release 1.7, first remove those that cannot fit into any existing semantic frame in FrameNet; then, assign the remaining lexical units to their correct frames. We also present the *Semi-supervised Deep Embedded Clustering with Anomaly Detection* (SDEC-AD) model—an algorithm that maps high-dimensional contextualized vector representations of lexical units to a low-dimensional latent space for better frame prediction and uses reconstruction error to identify lexical units that cannot evoke frames in FrameNet. SDEC-AD outperforms the state-of-the-art methods in both steps of the frame induction process. Empirical results also show that definitions provide contextual information for representing and characterizing the frame membership of lexical units.

Keywords: Semantic Frame Induction, Deep Embedded Clustering, Semi-supervised Learning, Anomaly Detection

1. Introduction

A semantic frame is a conceptual structure that models a type of situation, entity, or event (Ruppenhofer et al., 2006). Frame semantics is useful for inference-based natural language processing tasks such as question answering (Shen and Lapata, 2007), text summarization (Han et al., 2016), and information extraction (Moschitti et al., 2003; Barzdins, 2014). A widely-used frame semantic resource is Berkeley FrameNet (Baker et al., 1998), whose current release 1.7 (BFN 1.7) contains 1224 hierarchically-related semantic frames and 13,669 lexical units (LUs). An LU is a pair of word lemma and the semantic frame it evokes. For example, the LU $abandon.v$ falls under the $Abandonment$ frame, which describes an agent leaving behind an object and rendering such object no longer within one's control.

The lexical coverage of BFN 1.7 is low compared to other semantic lexical resources such as WordNet (Miller, 1995), which contains 210,000 entries, as FrameNet is built entirely manually by linguistic experts. Expanding FrameNet automatically is challenging because of the high number and uneven granularity of semantic frames (Rastogi and Van Durme, 2014), and polysemous lemmas such as $shoot.v$, which can be assigned to multiple frames such as Hit_target and $Ingest\_substance$. The NOTR-LU (lexical unit with no training data) coverage gap (Palmer and Sporleder, 2010), where 24% of LUs in BFN 1.7 lack example sentences, further complicates the challenge as the state-of-the-art frame induction methods (Anwar et al., 2019; Arefyev et al., 2019; Ribeiro et al., 2019) require example sentences featuring the LUs to create vector representations in the semantic space and induce frames.

At the same time, current research (Pennacchiotti et al., 2008; Johansson, 2014; Pavlick et al., 2015; Materna, 2012; Rastogi and Van Durme, 2014; Ustalov et al., 2018) assumes that an *unknown LU*—a generic lexical unit not existing in the FrameNet database—can be characterized by one or more frames existing in BFN 1.7. This assumption is unrealistic given the limited coverage of semantic frames (Palmer and Sporleder, 2010). Rastogi and Van Durme (2014) mention that one of the missing frames is $Programming$, which would contain LUs such as $code.v$ and $program.v$, and which would feature frame elements for the programmer, the programming language, and the purpose of the program. Current models would have assigned $code.v$ to other semantically related frames in BFN 1.7 such as $Creating$, which is problematic because LUs in those frames exhibit different lexicographic behaviors. For instance, in the $Creating$ frame, the programming language is the frame element INSTRUMENT. Its valence pattern will be "LU - [CREATED_ENTITY] -

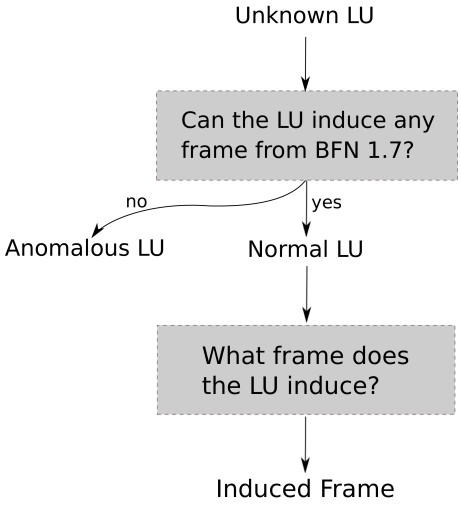


Figure 1: Two-step frame induction workflow of an *unknown LU*, which is a lexical unit (LU) not yet present in the Berkeley FrameNet data release 1.7 (BFN 1.7). An *anomalous LU* is an LU that cannot be assigned to any semantic frame in BFN 1.7, whereas a *normal LU* is the exact opposite: it can induce a frame in BFN 1.7.

[in INSTRUMENT]", which is exemplified by the sentence "I coded [Facebook]$_{\text{CREATED\_ENTITY}}$ [in Python]$_{\text{INSTRUMENT}}$". However, none of the lexical units under the *Creating* frame share the same syntactic realization. The closest is "LU - [CREATED_ENTITY] - [with INSTRUMENT]".

In a departure from previous work on frame induction (see Section 2.1 for the review of previous work), we propose a two-step process to assign an unknown LU to its correct frame (see Figure 1). First, we decide whether any existing frame in BFN 1.7 can characterize the sense of an unknown LU. We cast this step as anomaly detection, where we refer to an LU that does not belong to any semantic frame in BFN 1.7 as an *anomalous LU*. The converse is a *normal LU* that can be assigned to a frame in BFN 1.7. The subsequent step is to use the sense information to assign the normal LU to its frame.

The experimental results demonstrate that by mapping the high-dimensional contextualized representations of normal LUs to a low-dimensional latent space and learning to reconstruct the representations, our *Semi-supervised Deep Embedded Clustering with Anomaly Detection* (SDEC-AD) model outperforms all baseline models in filtering out anomalous LUs and inducing frames for normal LUs. We also show that we can generate embeddings to represent LUs that lack annotated sentences, which addresses the NOTR-LU coverage gap that hinders frame induction.

Our contributions are three-fold:

1. we propose representing LUs that lack example sentences using their definitions so they can participate in the two-step frame induction workflow,
2. our autoencoder-based model (SDEC-AD) is the first algorithm that detects LUs that cannot be classified into any frame in the FrameNet database, preserving the consistency of the inventory of frames and LUs,
3. we are the first to apply the deep-embedded clustering algorithm to induce semantic frames, achieving state-of-the-art performance.

2. Related Work

2.1. Semantic Frame Induction

Semantic frame induction is the task of labeling an unknown LU with a correct frame. Some (Pennacchiotti et al. (2008);Tonelli and Pianta (2009);Green et al. (2004)) rely on additional semantic information from other complementary lexical resources such as WordNet (Miller, 1995) to induce frames. Pavlick et al. (2015) and Rastogi and Van Durme (2014) use The Paraphrase Database (PPDB) (Ganitkevitch et al., 2013) to paraphrase frame-annotated sentences and assign the paraphrased LUs to their frames. On the other hand, instead of augmenting FrameNet with other lexical resources, Materna (2012), Modi et al. (2012), and Ustalov et al. (2018) use semantic role information to induce frames for verb lemmas.

Recognizing that unsupervised methods can induce frames better for unseen data than supervised methods, QasemiZadeh et al. (2019) aimed to benchmark unsupervised systems that assign verb lemmas to their semantic frames without using any explicit semantic annotation. They presented the task of Unsupervised Lexical Frame Induction in the International Workshop on Semantic Evaluation in 2019, and all the top three performing systems (Arefyev et al., 2019; Anwar et al., 2019; Ribeiro et al., 2019) employed a distributional approach: the systems cluster verb lemmas, which are represented as contextualized vectors computed over their example sentences by pretrained language models, in a semantic space.

The best system (Arefyev et al., 2019) implemented a two-phase hierarchical agglomerative clustering method. They clustered the BERT representations (Devlin et al., 2019) of lemmas and then split each cluster into two by further clustering the lemmas' substitutes, which are generated using Hearst-like patterns. The other two models used different clustering algorithms and representations of verb lemmas: Anwar et al. (2019) used the hierarchical agglomerative clustering method and represented the lemmas with a combination of sentence and word ELMo representations (Peters et al., 2018), whereas Ribeiro et al. (2019) employed a graph clustering algorithm known as Chinese Whispers to the BERT representations.

We are uncertain whether the three systems can reproduce frame induction success with the LUs in BFN 1.7, as the LUs in BFN 1.7 have parts of speech other than verb, and example sentences for many LUs are unavailable. Moreover, all three models directly cluster on the high-dimensional vector representations of LUs, and all report sensitivity to the choice of hyperparameters, particularly the number of clusters. In this paper, we fix the number of clusters as the number of existing frames in BFN 1.7, and we compare the state-of-the-art frame induction models with a deep clustering algorithm that uses a low-dimensional latent space to produce better clusters of high-dimensional data points.

2.2. Deep Clustering

Deep clustering is a type of clustering that uses a deep neural network to learn dense feature representations that favor a clustering task. When the dimensionality of the input feature space is very high, similarity metrics used by traditional clustering algorithms such as k-means and hierarchical methods become unreliable, which renders the direct clustering of the input embeddings ineffective (Guo et al., 2017). In contrast, deep clustering algorithms learn the representations in a low dimensional, clustering-friendly feature space. Xie et al. (2016) put forth the deep embedded clustering (DEC) algorithm—one of the most representative methods for deep clustering—that jointly learns the feature representations and the clustering assignments. To improve DEC's clustering performance, Ren et al. (2019) propose a semi-supervised version of DEC (SDEC), which incorporates pairwise constraints in the feature learning process such that data points in the same cluster become closer, and the incorrect cluster assignments can be adjusted by the existing information about the data.

We propose applying SDEC to the frame induction task to overcome the "curse of dimensionality" (Bellman, 1966) of the high-dimensional contextualized representations of LUs. We also augment SDEC to detect LUs that cannot fit into any frame in BFN 1.7 as SDEC uses an autoencoder

structure that excels at anomaly detection (see Section 2.3)

2.3. Anomaly Detection using Autoencoder

An autoencoder is an unsupervised learning algorithm that learns to reconstruct its input using a deep neural network. Its network structure consists of an encoder and a decoder. The encoder maps the original input vector to a hidden representation lower in dimensionality, and the decoder maps the hidden representation back to the original input space. The difference between the original input vector and the reconstructed vector is known as the reconstruction error. An autoencoder learns to minimize this reconstruction error such that the autoencoder approximates an identity function (Ng, 2010).

After the autoencoder is trained to reconstruct data without anomalies, the reconstruction error for anomalies is high (An and Cho, 2015), which enables anomaly detection. Autoencoders have been applied across the natural language processing domain to detect anomalies, such as web attacks (Vartouni et al., 2018), SMS spams (Al Moubayed et al., 2016) and even novel sport ideas (Mei et al., 2018). In our task, the anomalies are the LUs that cannot evoke any semantic frame in BFN 1.7.

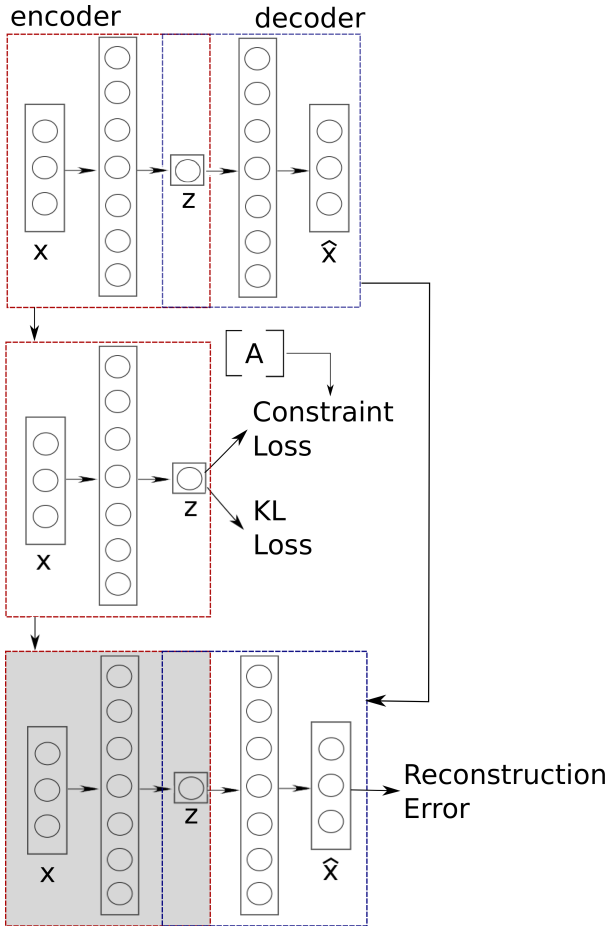


Figure 2: Framework of our proposed method (SDEC-AD).

3. Proposed Method

Consider the problem of assigning a set of LUs that do not exist in BFN 1.7 to k frames. Here, k is the number of frames in BFN 1.7, and the LUs are embedded by language models in the vector space X. First, we remove the subset of anomalous LUs that cannot be described by any existing frame in BFN 1.7. Next, we group the n remaining normal LUs $\{x_i \in X\}_{i=1}^{n}$ into k clusters where each cluster is represented by a centroid $\{\mu_j\}_{j=1}^{k}$ and corresponds to a semantic frame. This is a hard clustering task—each LU with a unique identifier is only assigned to one frame. Hereafter, normal LUs refer to both unknown LUs that can induce semantic frames from BFN 1.7 or LUs that already exist in BFN 1.7, unless otherwise specified.

We propose the Semi-supervised Deep Embedded Clustering with Anomaly Detection (SDEC-AD) model that jointly identifies anomalous LUs and assigns normal LUs to their frames [1]. For n normal LUs embedded in the vector space X, SDEC-AD uses an autoencoder to represent the semantic features of normal LUs in a low-dimensional latent space Z using a non-linear transformation $f_\theta : X \rightarrow Z$ (where θ are the learnable parameters of the encoder). After embedding the normal LUs in the latent space Z, SDEC-AD uses frame information about the LUs existing in BFN 1.7 to cluster the normal LUs (including the unknown LUs), which is the semi-supervised frame induction process. SDEC-AD also learns to reconstruct normal LUs from the latent space X to its original vector space X by minimizing their reconstruction error. Since SDEC-AD is not exposed to anomalous LUs during the training phase, SDEC-AD can identify anomalous LUs that have high reconstruction error.

Figure 2 illustrates that SDEC-AD has a deep encoder-decoder architecture. First, the encoder layers learn to encode the input embedding x into a low-dimensional latent representation z, and the decoder layers learn to reconstruct z back to the original embedding (top part of Figure 2). The reconstructed embedding is $\hat{x}$. The encoder layers are further trained to cluster the latent representations under the supervision of the pairwise constraints A (center part of Figure 2). The number of clusters is the number of lexical frames in BFN 1.7. By minimizing the Kullback-Leibler divergence clustering loss (KL loss) and the constraint loss, SDEC-AD jointly learns to represent and cluster x in the latent space. Finally, the decoder layers are retrained to reconstruct $\hat{x}$ from z by minimizing the reconstruction error (bottom part of Figure 2). The encoder layers (grey area), including the latent hidden representations, are frozen to prevent updating their parameters. Notice that we retrain the decoder layers of SDEC-AD (lower part of Figure 2) after SDEC-AD learns to embed and cluster normal LUs in the latent space Z (center part of Figure 2). The reason is that after the encoder layers learn to embed semantic features of normal LUs in the latent space Z (Xie et al., 2016), the reconstruction error of anomalous LUs becomes more distinguishable than that of normal LUs.

3.1. Parameter Initialization

We initialize SDEC-AD with fully-connected stacked autoencoders—each layer is a denoising auto-encoder

[1] https://github.com/yongzx/Semi-supervised-Deep-Embedded-Clustering-with-Anomaly-Detection-for-Semantic-Frame-Induction

trained to reconstruct the previous layer's output after random corruption. The structure of the stacked autoencoders is d-7500-1000-7500-d, where d is the dimension of the vector representations of LUs. We use the same parameter settings and nonlinear activation functions as SDEC (Ren et al., 2019).

The encoder in SDEC-AD receives the vector representations of LUs $\{x_i \in X\}_{i=1}^n$ as input and returns their feature representations $\{z_i \in Z\}_{i=1}^n$ in the latent space Z as outputs. We then employ supervised initialization (See section 3.2) to obtain k initial centroids $\{\mu_j\}_{j=1}^k$ in space Z, where k is the number of frames in BFN 1.7.

3.2. Supervised Initialization

We apply the "exploit and explore" mechanism (Lemaire et al., 2015) by first exploiting the information about the frame labels of normal LUs to initialize the centroids before we start clustering the normal LUs. We define each centroid $\{\mu_j\}_{j=1}^k$ as the average of the vectors (in the latent space Z) of normal LUs that share the same semantic frame. Note that we only use the normal LUs already present in BFN 1.7 to initialize the centroids.

3.3. Clustering with KL Divergence and Pairwise Constraints

We use the objective function of SDEC (Ren et al., 2019) to train SDEC-AD on clustering the latent feature representations of normal LUs $\{z_i \in Z\}_{i=1}^n$. The objective function is

$$L = L_u + \lambda L_s \tag{1}$$

where L_u is the unsupervised Kullback-Leibler (KL) divergence clustering loss, L_s is the semi-supervised constraint loss, and λ is the parameter that controls the degree of supervision.

SDEC-AD learns the latent representations that favor the clustering of normal LUs by minimizing L_u. It treats the centroids $\{\mu_j\}_{j=1}^k$ as trainable weights and assigns each embedded latent point z_i to a soft label q_i by Student's t-distribution:

$$q_{ij} = \frac{(1 + \left\|z_i - \mu_j\right\|^2)^{-1}}{\sum_{j'}(1 + \left\|z_i - \mu_{j'}\right\|^2)^{-1}} \tag{2}$$

where q_{ij} represents the probability of z_i belonging to cluster j. L_u is then defined as

$$L_u = KL(P \parallel Q) = \sum_i \sum_j p_{ij} \log(\frac{p_{ij}}{q_{ij}}) \tag{3}$$

where Q is the soft assignment and P is the target distribution, defined as

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_{j'}(q_{ij}^2 / \sum_i q_{ij'})} \tag{4}$$

At the same time, SDEC-AD minimizes the semi-supervised constraint loss L_s to move normal LUs with the same frames closer and normal LUs with different frames more apart. To calculate L_s, SDEC-AD requires a matrix A that describes must-link and cannot-link pairwise constraints of LUs. The matrix A is defined as:

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{n1} \\ a_{12} & a_{22} & \dots & a_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \tag{5}$$

For must-link constraints, when two LUs x_i and x_k share the same frame, $a_{ik} = 1$. On the other hand, when the two LUs are in two different frames, they satisfy the cannot-link constraint, so $a_{ik} = -1$. The constraint loss L_s is then defined as:

$$L_s = \frac{1}{n} \sum_{i=1}^n \sum_{k=1}^n a_{ik} \|z_i - z_k\|^2 \tag{6}$$

We only consider must-link and cannot-link constraints for normal LUs that exist in BFN 1.7; in other words, SDEC-AD is trained in a semi-supervised manner where it uses the frame label information for normal LUs in BFN 1.7 to cluster unknown normal LUs.

3.4. Retraining of Decoders for Anomaly Detection

After training the encoder of SDEC-AD to embed and cluster normal LUs in the latent space Z, we freeze the encoder and train the decoder to map the latent representations of LUs $\{z_i \in Z\}_{i=1}^n$ back to their original representations $\{x_i \in X\}_{i=1}^n$. The objective function here is the squared reconstruction error, defined as

$$L_\theta(x_i; \hat{x}_i) = \frac{1}{n} \sum_{i=1}^n (\hat{x}_i - x_i)^2 \tag{7}$$

where x_i is the original representation and $\hat{x}_i$ is the reconstructed representation of a normal LU. The error represents the semantic-wise difference between the original and reconstructed LU. Minimizing the reconstruction error of normal LUs makes SDEC-AD learn to capture and reconstruct semantic features of normal LUs. Therefore, the reconstruction error of normal LUs is lower than that of anomalous LUs because SDEC-AD "recognizes" normal LUs. A threshold τ can be chosen such that when the reconstruction error of an LU is above τ, the LU is considered anomalous.

3.5. Optimization

We use the stochastic gradient descent (SGD) and backpropagation to optimize both the centroids $\{\mu_j\}_{j=1}^k$ and the parameters of the encoder and decoder layers of SDEC-AD. During backpropagation, SDEC-AD passes the gradient $\frac{\partial L}{\partial z_i}$ down the encoder layers to update their parameters θ, which are used to perform the nonlinear transformation f_θ. At the same time, the gradient $\frac{\partial L}{\partial \mu_i}$ updates the centroids $\{\mu_j\}_{j=1}^k$. Similarly, the gradient $\frac{\partial L}{\partial \hat{x}_i}$ is used to update the parameters of the decoder layers so the decoder learns to reconstruct LUs $\{\hat{x}_i \in X\}_{i=1}^n$ from their latent representations $\{z_i \in Z\}_{i=1}^n$.

Lexical Units in Berkeley FrameNet 1.7 (Normal)	
Abandonment	abandon.v, abandoned.a, abandonment.n, forget.v, ...
Abounding_with	adorned.a, asphalted.a, bedecked.a, bejewelled.a, ...
Absorb_heat	bake.v, barbecue.v, blanch.v, boil.v, ...
Unknown Lexical Units in FrameNet+ (Normal)	
Abandonment	abdicate.v, discard.v, discontinue.v, drop.v, ...
Abounding_with	accent.v, border.v, clutter.v, cluttered.a, ...
Absorb_heat	-
Unknown Lexical Units from WordNet (Anomalous)	
-	piscivorous.s.01, radical.a.05, nut-bearing.s.01, caulescent.a.01, ...

Table 1: Examples of frame labels and their associated lexical units in the frame induction datasets. A dash (-) is used to indicate the absence of frame labels or lexical units. *Unknown lexical units* are lexical units not existing in FrameNet 1.7. *Normal lexical units* refer to lexical units that can be assigned to a semantic frame in FrameNet 1.7, whereas *anomalous lexical units* cannot.

4. Experimental Setup

4.1. Datasets

Similarly to Pennacchiotti et al. (2008), our gold standard for the induction task is the BFN 1.7 database.[2] We retrieved 19770 unknown normal LUs from FrameNet+ (FN+) database (Pavlick et al., 2015) after we preprocessed it by removing frames such as $Make\_possible\_to\_do$ and $Containment\_relation$ that do not exist in BFN 1.7. Every unknown normal LU has an example sentence and can be assigned to an existing frame in BFN 1.7.

To assess the ability of the models in discriminating anomalous LUs, we hand-picked 300 WordNet synsets according to the following criteria. First, we removed the lexical entries in WordNet (Miller, 1995) that already exist in Berkeley FrameNet 1.7. Subsequently, we selected 300 lexical units which cannot be represented by any semantic frame in Berkeley FrameNet 1.7 (Baker et al., 1998) based on the frame definitions and the lexicographic properties of the existing lexical units in the frames. 218 of the anomalous lexical units are adjective satellites and the rest are adjectives. 90 of the chosen lexical units have example phrases and/or sentences.[3].

4.2. Implementations

4.2.1. Vector Representations of Lexical Units

We use the BERT (bert-base-uncased) (Devlin et al., 2019) and ELMo (Peters et al., 2018) language models to generate the contextualized word embeddings to represent LUs. Both BERT and ELMo embeddings can capture the semantic context and achieve state-of-the-art result in inducing frames (Arefyev et al., 2019; Anwar et al., 2019; Ribeiro et al., 2019).

We experimented with representing LUs from BFN 1.7 with and without information from their definitions. Without including the definition, the vector representation of an LU is its contextualized word embedding based on its example sentences. To infuse information from the definition into the LU representation, we first remove the stopwords from the definition, and create the *definition embedding* by averaging embeddings of all word tokens in the definition. Finally, we represent the LU by adding the definition embedding to the contextualized representation generated from its example sentences. If an LU lacks example sentences, its representation is the definition embedding. We evaluate whether definition-infused contextualized representations improve frame induction using LUs from BFN 1.7 that have at least one example sentence.

Since results (see Section 5.1) confirm that definitions improve semantic frame induction, we represent all LUs (including WordNet synsets) with their definitions and example sentences (if exist) for the frame induction (Section 5.2) and anomaly detection (Section 5.3) experiments. The exceptions are the LUs from FN+ as they lack definitions, so only their example sentences are used to generate the contextualized representations.

4.2.2. Evaluation of Proposed Model

We evaluate the frame-induction performance of SDEC-AD on two datasets: LUs that only come from BFN 1.7 and LUs that come from both BFN 1.7 and FN+. As mentioned in Section 3.3, the pairwise constraint matrix A is created using the existing LUs in BFN 1.7. We independently run SDEC-AD 20 times and report the average results.

For anomaly detection, we train SDEC-AD only with LUs existing in BFN 1.7, and we measure its ability to discriminate anomalous LUs from normal LUs on the combined dataset of normal LUs from BFN 1.7, unknown normal LUs from FN+, and anomalous WordNet synsets.

4.2.3. Baselines

To evaluate the effectiveness of our SDEC-AD model, we apply the winning models benchmarked in SemEval-2019 Task 2: Unsupervised Lexical Frame Induction (QasemiZadeh et al., 2019) to our frame induction task. Their frame induction method details are as follows:

[2]We accessed Berkeley FrameNet data release 1.7 (BFN 1.7) using the NLTK FrameNet API (Schneider and Wooters, 2017).

[3]The dataset of *anomalous lexical units*—lexical units that cannot be assigned to any frame in Berkeley FrameNet data release 1.7—is uploaded to the LRE Map repository.

	SDEC-AD		Ribeiro et al. (2019)		Anwar et al. (2019)		Arefyev et al. (2019)	
	BERT	ELMo	BERT	ELMo	BERT	ELMo	BERT	ELMo
With Definitions	**0.801**	**0.754**	**0.232**	**0.376**	**0.215**	**0.217**	0.207	0.193
Without Definitions	0.797	0.744	0.180	0.339	0.209	0.211	**0.228**	**0.195**

Table 2: Clustering results of lexical units from Berkeley FrameNet data release 1.7 measured by the harmonic mean of Purity and inverse-Purity.

	SDEC-AD		Ribeiro et al. (2019)		Anwar et al. (2019)		Arefyev et al. (2019)	
	BERT	ELMo	BERT	ELMo	BERT	ELMo	BERT	ELMo
With Definitions	**0.693**	**0.632**	**0.169**	**0.262**	**0.194**	**0.195**	0.158	0.137
Without Definitions	0.688	0.620	0.130	0.238	0.186	0.189	**0.168**	**0.140**

Table 3: Clustering results of lexical units from Berkeley FrameNet data release 1.7 measured by the harmonic mean of BCubed precision and recall.

- Ribeiro et al. (2019): The authors treat LUs as vertices and connect them to the neighboring LUs through edges weighted by cosine distances. They use a threshold (based on the mean and standard deviation of the pairwise distance distribution) to determine the density of the graph, and they apply the Chinese Whispers algorithm to obtain the clusters of LUs.
- Anwar et al. (2019): The authors run the agglomerative clustering algorithm with Manhattan affinity and single linkage on the LU representations. Differing from the original implementation, we fix the number of clusters, which is a hyperparameter, as the number of unique frames in the dataset.
- Arefyev et al. (2019): First, the authors run an agglomerative clustering algorithm with cosine affinity and average linkage on the LU embeddings. They perform a grid search to find the optimal number of clusters. Then, they use language models such as BERT and ELMo to generate substitutes for the LUs using the example sentences, and they build TFIDF BoW vectors for the substitutes of each cluster. Finally, they use the agglomerative clustering algorithm to split each cluster of LUs into two clusters of their substitutes.

In addition to using the three models as baselines for frame induction for normal LUs, we adapt the three models to the anomaly detection task using the distance-based approach: an LU is considered anomalous when its distance—measured by different distance metrics used by different models—to the closest clusters (in the case of agglomerative clustering) or the closest LU (in the case of graph clustering) is above a certain threshold value τ (Satari et al., 2019; Akoglu et al., 2015). We use a random classifier as a baseline model for the anomaly detection task to simulate the process of randomly classifying an LU as anomalous or normal. The probability of random classification is 1%, which is the ratio of our hand-picked anomalous LUs to the unknown LUs.

We assess the baseline models' performance in frame induction and anomaly detection similarly for the evaluation of SDEC-AD (see Section 4.2.2).

4.3. Evaluation Metrics

Similarly to QasemiZadeh et al. (2019), we report the models' performance in identifying LUs that evoke the same frame using two measures for evaluating text clustering techniques: the harmonic mean of Purity and inverse-Purity (PIF) and the harmonic mean of BCubed precision and recall (BCF).

We frame the task of distinguishing anomalous LUs as an anomaly detection task, and we use the area under the receiver operating characteristic curve (AUC ROC) and the area under the precision-recall curve (AUC PRC) as the performance metrics. Since SDEC-AD and the baseline models produce anomaly scores—the reconstruction error and distances between LUs or clusters—instead of binary labels, and we do not know the best anomaly threshold τ, AUC ROC and AUC PRC are desirable metrics as they are threshold-invariant: they measure the quality of a model's predictions of anomalous LUs irrespective of the anomaly threshold τ (Chen et al., 2016).

5. Results and Discussion

Our experiments differ from the SemEval-2019 Task 2 (QasemiZadeh et al., 2019)—where the baseline models (Ribeiro et al., 2019; Anwar et al., 2019; Arefyev et al., 2019) are initially designated for—by two aspects: (1) the LUs in our dataset (see Section 4.1) are not only restricted to verb lemmas, and (2) SemEval-2019 Task 2 did not require the models to identify anomalous LUs. Note that the performance results of baseline models shown in Table 2, Table 3, Table 4 and Figure 3 originate from our empirical study and not from their reported figures in SemEval-2019 Task 2.

5.1. Effects of Definitions on Representations of Lexical Units

When example sentences and definitions form the hybrid contextualized representations of lexical units (LUs), more LUs with the same semantic frame are clustered together (see Table 2 and Table 3). The reason is that the definition contains information enough to aid in the identication of a concept or a semantic frame (Orfan and Allen, 2013; Spiliopoulou and Hovy, 2019), so the additional semantic context helps disambiguate polysemous lemmas better.

On the other hand, we obtain an opposite effect from Arefyel et al.'s (2019) model, which involves clustering the embeddings of LUs and subsequently the TFIDF BoW of the substitutes of LUs (see Section 4.2.3). A possible explanation for the contradictory result is that the first clustering step has returned clusters of LUs which are refined enough and where many clusters already correspond to frames in a one-to-one manner, so further splitting each cluster into two worsens the clustering result.

An advantage of using definitions to represent LUs is that we can now create contextualized representations for LUs that lack example sentences for frame induction, which is a significant breakthrough given that many unknown LUs lack annotations. Even in BFN 1.7, which is the biggest frame semantics database in terms of annotation, 39% LUs lack lexicographic annotations, and 24% LUs lack example sentences.

5.2. Frame Induction

Table 4 demonstrates that, when LUs are represented by BERT embeddings of their definitions and example sentences (if they exist), the SDEC-AD model outperforms the baseline models in assigning frames to normal LUs that either exist in or absent from BFN 1.7. High PIF and BCF scores indicate that SDEC-AD produces homogeneous and completed clusters of LUs—each cluster is representative of a semantic frame because LUs that share the same frame are grouped in a larger cluster instead of multiple smaller separate clusters. Since the clustering experiment included normal LUs that lack example sentences and are represented by their definition embeddings, the result suggests that SDEC-AD can overcome the NOTR-LU coverage gap and predict frames for these LUs by using the contextual information from definitions. The better performance of SDEC-AD comes from incorporating prior knowledge about the known LUs in BFN 1.7 as pairwise information and representing LUs at a lower dimension, which makes the distance between two LUs with different frames more distinguishable.

SDEC-AD clusters LUs better when we use BERT to generate representations of LUs. This observation suggests that BERT representations undergo less semantic information loss than its counterpart ELMo when the high-dimensional contextualized embeddings are mapped to the latent space. Our experiments also reveal that the effect of language models (for generating contextualized representations of LUs) on the frame induction performance depends on the clustering model used. For instance, Ribeiro et al.'s (2019) model, which uses the Chinese Whispers graph-clustering algorithm, returns more homogenous clusters of ELMo representations of LUs, but the other two baseline models that use the agglomerative clustering method yield the opposite outcome.

The baseline models perform well in assigning LUs to their correct frames in the SemEval-2019 Task 2 (see Table 5) but poorly in our experiment (see Table 4). The large margin in performance suggests that the baseline models are not capable of inducing frames when the LUs are not only restricted to verb lemmas and when the number of semantic frames is large—there are only 149 frames in the dataset of SemEval-2019 Task 2 (QasemiZadeh et al., 2019) but there are 1073 lexical frames in BFN 1.7[4]. This can be correlated to the fact that verbal valence patterns—that is, the patterns extracted from the metadata associated to example sentences through annotation—in FrameNet are more informative than those presented by LUs with a different POS (nouns, adjectives, and adverbs) (Peron-Corrêa et al., 2016). In our experiment, we also observe that three baseline models are biased to assign the LUs to frames with five or more existing LUs. One potential reason for the poor performance of baseline models is the "curse of dimensionality" (Bellman, 1966). At the high-dimensional vector space, the vectors of LUs within the same frame are much more spread out. Therefore, the more LUs a frame contains, the more likely that a new LU is assigned to the denser frame. We want to point out that, since the SemEval-2019 Task 2 dataset (QasemiZadeh et al., 2019) is no longer freely available, we could not conduct further performance comparison of SDEC-AD with baselines on the SemEval-2019 frame induction task. This is indicated by the missing result for SDEC-AD in Table 5.

5.3. Anomaly Detection

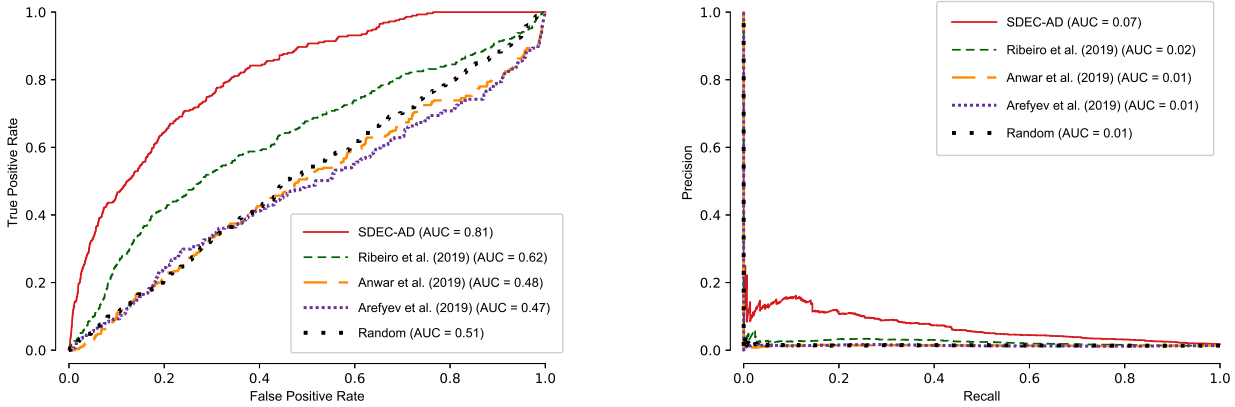
AUC ROC and AUC PRC measure models' performance—precision, recall, specificity, and specificity—aggregated across all possible anomaly score thresholds. The higher the AUC ROC and AUC PRC scores, the better the model in identifying the anomalies. Figure 3 shows that SDEC-AD is the best model in separating anomalous LUs from normal LUs. Anwar et al.'s (2019) and Arefyev et al.'s (2019) models obtain an AUC ROC score of about 0.5, which indicates that both models cannot discriminate between anomalous and normal LUs as they are no better than a random classifier. While the AUC ROC of Ribeiro et al.'s (2019) model is greater than that of a random classifier, the AUC PRC of the two baseline models is close, signifying that Ribeiro et al.'s (2019) model cannot discriminate between anomalous and normal LUs in an imbalanced dataset. Ribeiro et al.'s (2019) model is biased towards predicting anomalous LUs as normal, so it achieves a good false positive rate in the ROC curve but attains very low precision in the PR curve.

The baseline clustering models cannot discriminate anomalous LUs from normal LUs using the distance metrics for two reasons. First, the high dimensional semantic spaces of BERT or ELMo representations render the distance metrics meaningless in assessing the dissimilarity of LUs (Assent, 2012). Second, the published models are suitable for

[4]We count the number of lexical frames—semantic frames that contain lexical units—in BFN 1.7 accessed through the NLTK FrameNet API (Schneider and Wooters, 2017)

Models	LM	BFN 1.7		BFN 1.7 & FN+	
		PiF	BcF	PiF	BcF
SDEC-AD	**BERT**	**0.786**	**0.661**	**0.425**	**0.343**
	ELMo	0.739	0.603	0.421	0.336
Ribeiro et al. (2019)	BERT	0.186	0.134	0.076	0.046
	ELMo	0.280	0.186	0.132	0.075
Anwar et al. (2019)	BERT	0.180	0.162	0.151	0.133
	ELMo	0.178	0.159	0.095	0.071
Arefyev et al. (2019)	BERT	0.200	0.143	0.200	0.143
	ELMo	0.178	0.122	0.178	0.122

Table 4: Clustering results of four clustering models and two language models (LM) on lexical units from Berkeley FrameNet data release 1.7 (BFN 1.7) and lexical units from both BFN 1.7 and FN+ database (BFN 1.7 & FN+). Our proposed method (SDEC-AD) outperforms baseline models in clustering together lexical units that share the same semantic frames.



(a) Receiver operating characteristic (ROC) curves.

(b) Precision-Recall (PR) curves.

Figure 3: ROC curves (left figure (a)) and PR curves (right figure (b)) of different models in identifying lexical units that cannot be assigned to any semantic frame in FrameNet 1.7.

Models	PiF	BcF
Ribeiro et al. (2019)	75.25	65.32
Anwar et al. (2019)	76.68	68.10
Arefyev et al. (2019)	78.15	70.70
SDEC-AD	(?)	(?)

Table 5: Reported performance of baseline models on clustering verb lemmas that share the same semantic frames in SemEval-2019 Task 2 (QasemiZadeh et al., 2019).

optimizing clustering and frame assignment—as required in the original SemEval-2019 Task 2 (QasemiZadeh et al., 2019)—but not for detecting outliers. In contrast, SDEC-AD only learns the low-dimensional representation of semantic content in BFN 1.7 during its training stage. The high reconstruction error is more indicative of anomalous LUs than the large distances between anomalous LUs and normal LUs in the high-dimensional semantic vector space. The probability distribution of 1% assumed by the random classifier (based on the proportion of anomalous LUs in our dataset) is unrealistically low since many domain-specific lexical units cannot be represented by semantic frames in BFN 1.7 (Venturi et al., 2009; da Costa et al., 2018; Dolbey et al., 2006). Besides, the dataset of anomalous LUs only consists of adjectives and adjective satellites. Further research on the prior distribution of anomalous LUs with different parts-of-speech is therefore warranted.

Even though SDEC-AD performs better than the rest in detecting anomalous lexical units, its ability to identify nouns and verbs that cannot fit into the existing frames of BFN 1.7 is unknown, especially when reconstruction-error-based autoencoders lack the ability to address variability (An and Cho, 2015). Besides, we did not define the exact anomaly threshold τ of the reconstruction error, which controls the ratio of false positives to false negatives, for SDEC-AD in our experiments. In practice, τ should be defined such that SDEC-AD identifies anomalous LUs with

high recall. High recall is more important than high precision in expanding FrameNet because we want to minimize false negatives and avoid the *contamination* of FrameNet—a situation where semantic frames contain anomalous LUs.

6. Conclusion

This paper presents using *Semi-supervised Deep Embedded Clustering with Anomaly Detection*, or SDEC-AD, to learn clustering-friendly representations of lexical units (LUs) for frame assignment. For a set of LUs not yet present in Berkeley FrameNet data release 1.7 (BFN 1.7), SDEC-AD removes the LUs that cannot be characterized by any semantic frame in BFN 1.7 and subsequently labels the remaining LUs with their correct frames. This two-step frame induction process automatically expands the lexical coverage in BFN 1.7 without compromising the within-frame consistency. Empirical studies show that SDEC-AD outperforms state-of-the-art unsupervised frame induction models. Furthermore, we demonstrate that using the definitions of LUs, which are already present in the lexical resource, enable us to better assign LUs, including those that lack example sentences, to their frames. In the future, we will explore representing frames and their spatial relations in the low-dimensional latent space using SDEC-AD and predict frame-to-frame relations.

7. Acknowledgements

This research used the archive and facilities of the Distributed Little Red Hen Lab, co-directed by Francis Steen and Mark Turner, and the FrameNet Brasil Computational Linguistics Lab. The material is based upon work supported by the National Science Foundation under grant 1028381 (2010-2015). The FrameNet Brasil Computational Linguistics Lab is funded by CAPES PROBRAL (grant # 88887.144043/2017-00).

8. Bibliographical References

Akoglu, L., Tong, H., and Koutra, D. (2015). Graph based anomaly detection and description: a survey. *Data mining and knowledge discovery*, 29(3):626–688.

Al Moubayed, N., Breckon, T., Matthews, P., and McGough, A. S. (2016). Sms spam filtering using probabilistic topic modelling and stacked denoising autoencoder. In *International Conference on Artificial Neural Networks*, pages 423–430. Springer.

An, J. and Cho, S. (2015). Variational autoencoder based anomaly detection using reconstruction probability. *Special Lecture on IE*, 2(1).

Anwar, S., Ustalov, D., Arefyev, N., Ponzetto, S. P., Biemann, C., and Panchenko, A. (2019). HHMM at SemEval-2019 task 2: Unsupervised frame induction using contextualized word embeddings. In *NAACL HLT 2019 : The International Workshop on Semantic Evaluation, Proceedings of the Thirteenth Workshop, June 6-June 7, 2019, Minneapolis, Minnesota, USA*, pages 125–129, Stroudsburg, PA. Association for Computational Linguistics, ACL.

Arefyev, N., Sheludko, B., Davletov, A., Kharchev, D., Nevidomsky, A., and Panchenko, A. (2019). Neural GRANNy at SemEval-2019 task 2: A combined approach for better modeling of semantic relationships in semantic frame induction. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 31–38, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Assent, I. (2012). Clustering high dimensional data. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2(4):340–350.

Barzdins, G. (2014). FrameNet CNL: A knowledge representation and information extraction language. In *International Workshop on Controlled Natural Language*, pages 90–101. Springer.

Bellman, R. (1966). Dynamic programming. *Science*, 153(3731):34–37.

Chen, T., Tang, L.-A., Sun, Y., Chen, Z., and Zhang, K. (2016). Entity embedding-based anomaly detection for heterogeneous categorical events. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 1396–1403. AAAI Press.

da Costa, D., Gamonal, M. A., Paiva, V. M. R. L., Marção, N. D., Peron-Corrêa, S. R., de Almeida, V. G., da Silva Matos, E. E., and Torrent, T. T. (2018). Framenet-based modeling of the domains of tourism and sports for the development of a personal travel assistant application. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, pages 6–12.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Dolbey, A., Ellsworth, M., and Scheffczyk, J. (2006). Bioframenet: A domain-specific framenet extension with links to biomedical ontologies. In *KR-MED*, volume 222.

Green, R., Dorr, B. J., and Resnik, P. (2004). Inducing frame semantic verb classes from WordNet and LDOCE. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 375–382, Barcelona, Spain, July.

Guo, X., Gao, L., Liu, X., and Yin, J. (2017). Improved deep embedded clustering with local structure preservation. In *IJCAI*, pages 1753–1759.

Han, X., Lv, T., Hu, Z., Wang, X., and Wang, C. (2016). Text summarization using FrameNet-based semantic graph model. *Scientific Programming*, 2016.

Johansson, R. (2014). Automatic expansion of the Swedish FrameNet lexicon: Comparing and combining lexicon-based and corpus-based methods. *Constructions and Frames*, 6(1):92–113.

Lemaire, V., Ismaili, O. A., and CornuÃ©jols, A. (2015). An initialization scheme for supervized k-means. In *2015 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, July.

Materna, J. (2012). LDA-frames: An unsupervised approach toÂ generating semantic frames. In Alexander Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, pages 376–387, Berlin, Heidelberg. Springer Berlin Heidelberg.

Mei, M., Guo, X., Williams, B. C., Doboli, S., Kenworthy, J. B., Paulus, P. B., and Minai, A. A. (2018). Using semantic clustering and autoencoders for detecting novelty in corpora of short texts. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Modi, A., Titov, I., and Klementiev, A. (2012). Unsupervised induction of frame-semantic representations. In *Proceedings of the NAACL-HLT Workshop on the Induction of Linguistic Structure*, pages 1–7, Montréal, Canada, June. Association for Computational Linguistics.

Moschitti, A., Morarescu, P., and Harabagiu, S. M. (2003). Open-domain information extraction via automatic semantic labeling. In *In Proceedings of FLAIRS*.

Ng, A. (2010). Sparse autoencoder. [Online; accessed 8-September-2019].

Orfan, J. and Allen, J. (2013). Toward learning high-level semantic frames from definitions. In *Proceedings of the Second Annual Conference on Advances in Cognitive Systems ACS*, volume 125, page 134.

Palmer, A. and Sporleder, C. (2010). Evaluating FrameNet-style semantic parsing: the role of coverage gaps in FrameNet. In *Coling 2010: Posters*, pages 928–936, Beijing, China, August. Coling 2010 Organizing Committee.

Pavlick, E., Wolfe, T., Rastogi, P., Callison-Burch, C., Dredze, M., and Van Durme, B. (2015). FrameNet+: Fast paraphrastic tripling of FrameNet. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 408–413, Beijing, China, July. Association for Computational Linguistics.

Pennacchiotti, M., De Cao, D., Basili, R., Croce, D., and Roth, M. (2008). Automatic induction of FrameNet lexical units. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 457–465, Stroudsburg, PA, USA. Association for Computational Linguistics.

Peron-Corrêa, S., Diniz, A., Lara, M., Matos, E., and Torrent, T. (2016). Framenet-based automatic suggestion of translation equivalents. In João Silva, et al., editors, *Computational Processing of the Portuguese Language*, pages 347–352, Cham. Springer International Publishing.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June. Association for Computational Linguistics.

QasemiZadeh, B., Petruck, M. R. L., Stodden, R., Kallmeyer, L., and Candito, M. (2019). SemEval-2019 task 2: Unsupervised lexical frame induction. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 16–30, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Rastogi, P. and Van Durme, B. (2014). Augmenting FrameNet via PPDB. In *Proceedings of the Second Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 1–5, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Ren, Y., Hu, K., Dai, X., Pan, L., Hoi, S. C., and Xu, Z. (2019). Semi-supervised deep embedded clustering. *Neurocomputing*, 325:121 – 130.

Ribeiro, E., Mendonça, V., Ribeiro, R., Martins de Matos, D., Sardinha, A., Santos, A. L., and Coheur, L. (2019). L2F/INESC-ID at SemEval-2019 task 2: Unsupervised lexical semantic frame induction using contextualized word representations. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 130–136, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Ruppenhofer, J., Ellsworth, M., Petruck, M. R., Johnson, C. R., and Scheffczyk, J. (2006). *FrameNet II: Extended Theory and Practice*. International Computer Science Institute, Berkeley, California. Distributed with the FrameNet data.

Satari, S. Z., Di, N. F. M., and Zakaria, R. (2019). Single-linkage method to detect multiple outliers with different outlier scenarios in circular regression model. In *AIP Conference Proceedings*, volume 2059, page 020003. AIP Publishing.

Schneider, N. and Wooters, C. (2017). The NLTK FrameNet API: Designing for discoverability with a rich linguistic resource. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 1–6, Copenhagen, Denmark, September. Association for Computational Linguistics.

Shen, D. and Lapata, M. (2007). Using semantic roles to improve question answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 12–21, Prague, Czech Republic, June. Association for Computational Linguistics.

Spiliopoulou, E. and Hovy, E. (2019). Definition frames: Using definitions for hybrid concept representations. *arXiv preprint arXiv:1909.04793*.

Tonelli, S. and Pianta, E. (2009). A novel approach to mapping FrameNet lexical units to WordNet synsets. In *Proceedings of the Eighth International Conference on Computational Semantics*, IWCS-8 '09, pages 342–345, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ustalov, D., Panchenko, A., Kutuzov, A., Biemann, C., and Ponzetto, S. P. (2018). Unsupervised semantic frame induction using triclustering. *ACL 2018 - 56th Annual*

Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers), 2:55–62.

Vartouni, A. M., Kashi, S. S., and Teshnehlab, M. (2018). An anomaly detection method to detect web attacks using stacked auto-encoder. In *2018 6th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS)*, pages 131–134. IEEE.

Venturi, G., Lenci, A., Montemagni, S., Vecchi, E. M., Sagri, M. T., Tiscornia, D., and Agnoloni, T. (2009). Towards a framenet resource for the legal domain. *LOAIT*, pages 67–76.

Xie, J., Girshick, R., and Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 478–487. JMLR.org.

9. Language Resource References

Baker, Collin F. and Fillmore, Charles J. and Lowe, John B. (1998). *The Berkeley FrameNet Project*. Association for Computational Linguistics, ACL '98/COLING '98.

Ganitkevitch, Juri and Van Durme, Benjamin and Callison-Burch, Chris. (2013). *PPDB: The paraphrase database*.

Miller, George A. (1995). *WordNet: A Lexical Database for English*. ACM.

Pavlick, Ellie and Wolfe, Travis and Rastogi, Pushpendre and Callison-Burch, Chris and Dredze, Mark and Van Durme, Benjamin. (2015). *FrameNet+: Fast Paraphrastic Tripling of FrameNet*. Association for Computational Linguistics.