Semantic Frame Induction as a Community Detection Problem

Eugénio Ribeiro[1,2,5](✉), Andreia Sofia Teixeira[1,3,4], Ricardo Ribeiro[1,5], and David Martins de Matos[1,2]

[1] INESC-ID, Lisbon, Portugal
eugenio.ribeiro@inesc-id.pt
[2] Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal
[3] Center for Social and Biomedical Complexity, School of Informatics, Computing and Engineering, Indiana University, Bloomington, IN, USA
[4] Indiana University Network Science Institute (IUNI), Indiana University, Bloomington, IN, USA
[5] Instituto Universitário de Lisboa (ISCTE-IUL), Lisbon, Portugal

Abstract. Resources such as FrameNet provide semantic information that is important for multiple tasks. However, they are expensive to build and, consequently, are unavailable for many languages and domains. Thus, approaches able to induce semantic frames in an unsupervised manner are highly valuable. In this paper we approach that task from a network perspective as a community detection problem that targets the identification of groups of verb instances that evoke the same semantic frame. To do so, we apply a graph-clustering algorithm to a graph with contextualized representations of verb instances as nodes connected by an edge if the distance between them is below a threshold that defines the granularity of the induced frames. By applying this approach to the benchmark dataset defined in the context of the SemEval shared task we outperformed all the previous approaches to the task.

Keywords: Semantic frames · Contextualized representations · Community detection · Graph clustering

1 Introduction

A word may have different senses depending on the context in which it appears. Thus, in order to understand its meaning, we must analyze that context and identify the semantic frame that is being evoked [12]. Consequently, sets of frame definitions and annotated datasets that map text into the semantic frames it evokes are important resources for multiple Natural Language Processing (NLP) tasks [1,10,23]. Among such resources, the most prominent is FrameNet [5], providing a set of more than 1,200 generic semantic frames, as well as over 200,000 annotated sentences in English. However, this kind of resource is expensive and time-consuming to build, since both the definition of the frames and the annotation of sentences require expertise in the underlying knowledge. Furthermore,

H. Cherifi et al. (Eds.): COMPLEX NETWORKS 2019, SCI 881, pp. 274–285, 2020.
https://doi.org/10.1007/978-3-030-36687-2_23

it is difficult to decide both the granularity and the domains to consider while defining the frames. Thus, such resources only exist for a reduced amount of languages [8] and even English lacks domain-specific resources in multiple domains.

An approach to alleviate the effort in the process of building semantic frame resources is to induce the frames evoked by a collection of documents using unsupervised approaches. However, most research on this subject focused on arguments and the induction of their semantic roles [16,25,26] or on the induction of semantic frames from verbs with two arguments [18,28]. To address this issue and define a benchmark for future research, a shared task was proposed in the context of SemEval 2019 [21]. This task focused on the unsupervised induction of FrameNet-like frames through the grouping of verbs and their arguments according to the requirements of three different subtasks. The first of those subtasks focused on clustering instances of verbs according to the semantic frame they evoke while the others focused on clustering the arguments of those verbs, both according to the frame-specific slots they fill and their semantic role.

In this paper we approach the first subtask from a network perspective. First, we generate a network in which the nodes correspond to contextualized representations of each verb instance. Then, we create edges between two nodes if the distance between them is lower than a certain threshold which controls the granularity of the induced frames. Finally, we apply a graph-clustering approach to identify communities of nodes that evoke the same frame.

In the remainder of the paper, we start by providing an overview of previous approaches to the task, in Sect. 2. Then, in Sect. 3, we describe our induction approach. Section 4 describes our experimental setup. The results of our experiments are presented and discussed in Sect. 5. Finally, Sect. 6 summarizes the conclusions of our work and provides pointers for future work.

2 Related Work

Before the shared task in the context of SemEval 2019, there were already some approaches to unsupervised semantic frame induction. For instance, LDA-Frames [18] relied on topic modeling and, more specifically, on Latent Dirichlet Allocation (LDA) [7], to jointly induce semantic frames and their frame-specific semantic roles. On the other hand, Ustalov et al. [28] approached the induction of frames through the triclustering of Subject-Verb-Object (SVO) triples using the Watset fuzzy graph-clustering algorithm [27], which induces word-sense information in the graph before clustering. However, although these approaches are able to induce semantic frames, they can only be applied to verb instances with certain characteristics, such as a fixed number of arguments.

Since we are approaching one of the subtasks defined in the context of SemEval 2019s Task 2, the most important approaches to describe in this section are those which competed in that subtask. Arefyev et al. [3] achieved the highest performance in the competition using a two-step agglomerative clustering approach. First, it generates a small set of large clusters containing instances of verbs which have at least one sense that evokes the same frame. Then, the verb instances of each cluster are clustered again to distinguish the different frames that are evoked according to the different senses. In both steps, the generation of the representations of the instances relies on BERT [11]. Nonetheless, while the first step relies on the contextualized representation given by an empirically selected layer of the model, the second step uses BERT as a language model to generate possible context words that provide cues for the sense of the verb instance. To do so, multiple Hearst-like patterns [15] are applied to the sentence in which the verb instance occurs and the context words correspond to those generated to fill the slots in the patterns. The representation of the instance is then given by a tf-idf-weighted average of the representations of the most probable context words. The number of clusters in the first step was obtained by performing grid search while clustering the development and test data together. The selected value corresponds to that which led to maximum performance on the development data. In the second step, clusters with less than 20 instances or containing specific undisclosed verbs were left intact. In the remainder, the number of clusters was selected to maximize the silhouette score.

Anwar et al. [2] used a more simplistic approach based on the agglomerative clustering of contextualized representations of the verb instances. The number of clusters was defined empirically. In the system submitted for participation in the competition, the contextualized representations were obtained by concatenating the context-free representation of the verb instance obtained using Word2Vec [19] with the tf-idf-weighted average of the representations of the remaining words in the sentence. However, in a post-evaluation experiment, better results were achieved using the mean of contextualized representations generated by ELMo [20].

Finally, Ribeiro et al. [22] also relied on contextualized representations of the verb instances, but used a graph-based approach. They experimented with both the sum of the representations generated by ELMo [20] and those generated by the last layer of the BERT model [11]. Better results were achieved with the former. The contextualized representations are used as the nodes in a graph and connected by a distance-weighted edge if the cosine distance between them is below a threshold based on a function of the mean and standard deviation of the pairwise distances between the nodes. Finally, the Chinese Whispers [6] algorithm is applied to the graph to identify communities of nodes that evoke the same frame. Although the high performance achieved on the development data did not generalize to the test data, this simple approach has the potential to achieve higher results with some modifications. Thus, the work described in this paper is based on this approach

3 Semantic Frame Induction Approach

In general, our approach, summarized in Algorithm 1, is very similar to the one used by Ribeiro et al. [22] in the context of the SemEval shared task. It starts by generating a contextualized representation of each verb instance. These representations are then used as the nodes in a network or graph in which each pair of nodes is connected through an edge if the distance between them is below a certain threshold. Finally, the Chinese Whispers algorithm is applied to the graph to identify communities of verb instances that evoke the same frame. However, it has some key modifications that improve its performance.

Algorithm 1. Frame Induction Approach

Input: S // The set of sentences
Input: T // The set of head tokens to cluster
Input: EMBED // The approach for generating contextualized representations
Input: d // The neighboring threshold
Output: C // The set of clusters
$V \leftarrow \{\text{EMBED}(S_t, t) : t \in T\}$
$D \leftarrow \{1 - \cos(\theta_{v,v'}) : (v, v') \in V^2, v \neq v'\}$ // $\theta_{v,v'}$ is the angle between v and v'
$W \leftarrow \{1 - D_{v,v'} : (v, v') \in V^2, v \neq v'\}$ // The weights of the edges
$E \leftarrow \{(v, v', W_{v,v'}) : (v, v') \in V^2, v \neq v', D_{v,v'} < d\}$
$G \leftarrow (V, E)$
$C \leftarrow \text{CHINESEWHISPERS}(G)$
return C

Starting with the representation of verb instances, the use of contextualized word representations in all of the approaches that competed in the SemEval shared task proves their importance for distinguishing different word senses, which evoke different frames. Ribeiro et al. [22] experimented with representations generated by both ELMo and BERT and achieved better results using the former. Furthermore, in their experiments, Arefyev et al. [3] noticed that BERT tends to generate representations of the different forms of the same lexeme which are distant in terms of the typically used euclidean and cosine distances. They tried to identify a distance metric that was appropriate for correlating such representations, but were unsuccessful. Thus, although it is not the current state-of-the-art approach for generating contextualized word representations, we rely on ELMo in our approach. The generated representations include a context-free representation and context information at two levels. According to the experiments performed by the authors of ELMo, the first level is typically related to the syntactic context, while the second is typically related to the semantic context. In addition to the combination of all information, we also explore the use of each level independently. This way, we are able to assess which information is actually important for the task.

To generate the contextualized representation of multi-word verb instances, we use a dependency parser to identify the head word and use the corresponding representation, since it contains information from the other words.

In our approach, the contextualized representations of the verb instances are used as the nodes of a graph. To generate the edges, the first step is to calculate the pairwise distance between those representations. We use the cosine distance since it is bounded and the magnitude of word vectors is typically related to the number of occurrences. Thus, the angle between the vectors is a better indicator of similarity. Furthermore, the euclidean distance has issues in spaces with high dimensionality. Still, we performed preliminary experiments to confirm that using the cosine distance leads to better results than the euclidean distance.

Each pair of nodes in the graph is connected through an edge if the distance between them is below a certain threshold. The definition of this threshold is particularly important, since it controls the granularity of the induced frames. Having control over this granularity is important, since it allows us to induce more specific or more abstract frames, both of which are relevant in different scenarios. Furthermore, this control allows us to define granularity in a small set of instances and then induce frames with a similar granularity in a different set. The latter was the main issue of Ribeiro et al.'s [22] approach at SemEval, whose performance on the development set did not generalize to the test set. That happened since the threshold was selected using a function of the statistics of the distribution of pairwise distances, which vary according to the contexts covered by the datasets and the number of instances. Consequently, applying the same function on the development and test sets led to the generation of frames with different granularity. We fix this issue by defining the threshold through grid search on the development set and then using the same fixed threshold across sets.

Another difference of our approach is the weighting of the edges. While Ribeiro et al. [22] attributed a weight corresponding to the distance between the nodes, we weight the edges using the cosine similarity. This is more appropriate, since the Chinese Whispers [6] algorithm that we use to identify the communities of nodes that evoke the same frame attributes more importance to edges with higher weight. Chinese Whispers is a simple but effective graph-clustering algorithm based on the idea that nodes that broadcast the same message to their neighbors should be aggregated. It starts by attributing each node to a different cluster. Then, in each iteration, the nodes are processed in random order and are attributed to the cluster with highest sum of edge weights in their neighborhood. This process is repeated until there are no changes or the maximum number of iterations is reached. Chinese Whispers is appropriate for this task since it identifies the number of cluster on its own, is able to handle clusters of different sizes, and scales well to large graphs. Furthermore, it typically outperforms other clustering approaches on NLP tasks.

4 Experimental Setup

In this section we describe our experimental setup in terms of data, evaluation approach, and implementation details.

4.1 Dataset

In our experiments, we used the same dataset used in the context of SemEval 2019s Task 2. This dataset consists of sentences extracted from the Penn Treebank 3.0 [17] and annotated with FrameNet frames. Since we are focusing on clustering verb instances into semantic frame heads, we are not interested in the annotations of the arguments. The development set consists of 600 verb instances extracted from 588 sentences and annotated with 41 different frames. The test set consists of 4,620 verb instances extracted from 3,346 sentences and annotated with 149 different frames. Additionally, all the sentences are annotated with morphosyntactic information in the CoNLL-U format [9].

4.2 Evaluation Approach

For direct comparison with the approaches that competed in SemEval's task, we evaluate our approach using the same metrics used on the task: Purity F_1, which is the harmonic mean of purity and inverse-purity [24], and BCubed F_1, which is the harmonic mean of BCubed Precision and BCubed Recall [4]. While the first focuses on the quality of each cluster independently, the latter focuses on the distribution of instances of the same category across the clusters. Additionally, we report the number of induced clusters. Since the Chinese Whispers algorithm is not deterministic, the values we report for these metrics refer to the mean and standard deviation over 30 runs.

Since we are approaching the problem from a network-based perspective, we also report the number of edges, the diameter, and the clustering coefficient of the network corresponding to the neighboring threshold with highest performance in each scenario.

In addition to that of the approaches that competed in SemEval's task, we also compare the performance of our approach with a baseline that consists of generating one cluster per verb.

4.3 Implementation Details

To obtain the contextualized representation of the verb instances we used the ELMo model provided by the AllenNLP package [13] to generate the contextualized embeddings for every sentence in the dataset and then selected the representations of the head token of each instance. The representation of each verb instance is then given by three vectors of dimensionality 1,024, corresponding to the context-free representation of the head token and the two levels of context information. We experimented both with each vector independently, as well as their combination. To combine the vectors we used their sum, since it represents the variation of the context-free representation according to the context.

To apply the Chinese Whispers algorithm, we relied on Ustalov's [29] implementation in Python, which requires the graph to be built using the NetworkX package [14]. We did not use weight regularization and performed a maximum of 20 iterations.

Finally, to obtain the syntactic dependencies used to determine the head token of multi-word verbs, we used the annotations provided with the dataset, which were obtained automatically using a dependency parser.

5 Results and Discussion

Before starting the discussion, it is important to make some remarks regarding the presentation of the results. First, although the cosine distance varies in the interval [0, 2], for readability, we only plot the results in the interval [0, 1], since for neighboring thresholds above that value the verb instances are always grouped into a single cluster. Furthermore, we do not include the value of the graph diameter in our tables, since the graph corresponding to the threshold that leads to higher performance in each scenario is never connected. Thus, the diameter is always infinite.

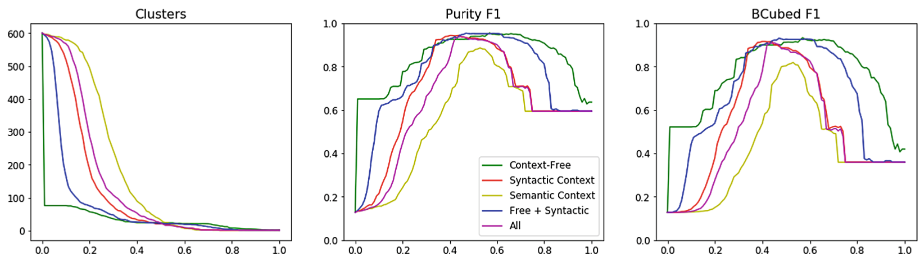

Fig. 1. Results on the development data using the different levels of ELMo representations. The xx axis refers to the neighboring threshold used to create the edges.

Table 1. Results on the development data using the different levels of ELMo representations. d refers to the neighboring threshold. CC refers to the clustering coefficient.

	d	Edges	CC	Clusters	Purity F_1	BCubed F_1
Context-Free	0.57	39,441	0.97	22.03 ± 0.31	95.57 ± 0.58	93.35 ± 0.71
Syntactic Context	0.41	27,201	0.80	30.93 ± 0.25	94.32 ± 0.16	91.65 ± 0.20
Semantic Context	0.53	20,660	0.67	24.47 ± 0.52	88.64 ± 0.33	81.92 ± 0.54
Free + Syntactic	0.47	37,913	0.94	22.97 ± 0.41	$\mathbf{95.83 \pm 0.28}$	$\mathbf{93.66 \pm 0.32}$
All	0.45	21,448	0.73	34.73 ± 0.51	93.93 ± 0.30	91.04 ± 0.59

Starting with the information provided by the multiple levels included in ELMo representations, in Fig. 1 and the first block of Table 1, we can see that, independently, the context-free representation is the most informative of the three and the most robust to changes in the threshold, with a wide interval with reduced decrease in performance around the threshold with highest performance. The initial drop in the number of clusters is due to its lack of context information, which makes all the instances of the same verb become connected as soon as the threshold is higher than zero.

The lower performance of the levels that provide context information on their own was expected, since they represent changes in the word sense of the verb according to the context, but lack information regarding the verb itself. Surprisingly, the level that typically captures the semantic context leads to worse performance than that which captures syntactic context and even harms performance in combination with the other levels. However, this can be explained by the fact that the ELMo model was trained for a specific task and, consequently, the semantic context is overfit to that task. On the other hand, the syntactic context is more generic and, since the sense of a verb can be related to the syntactic tree in which it occurs, it provides important information for the task.

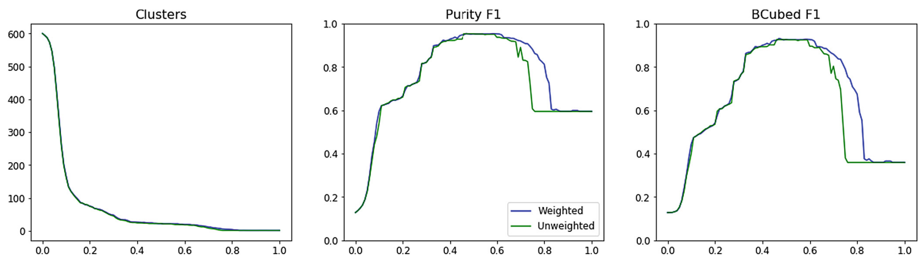

Fig. 2. Results on the development data according to the weighting of the edges. The xx axis refers to the neighboring threshold used to create the edges.

As shown in the second block of Table 1, the highest performance is achieved when using the combination of the context-free representation and the syntactic context. Still, the average increase in BCubed F_1 in relation to when using the context-free representation on its own is of just 0.33% points, which suggests that the context information is only able to disambiguate a reduced amount of specific cases. However, the threshold that leads to the highest performance in the combination is lower. This means that the graph has less edges and consequently, is less connected. Still, the number of clusters, around 23, is nearly half of the number of frames in the gold standard, 41, which means that the graph should be even less connected. Since the performance decreases for lower thresholds, this suggests that either the representations or the distance metric are unable to capture all the information required to group the instances in FrameNet-like frames.

Table 2. Results on the development data according to the weighting of the edges. *d* refers to the neighboring threshold. *CC* refers to the clustering coefficient.

	d	Edges	CC	Clusters	Purity F_1	BCubed F_1
Weighted	0.47	37,913	0.94	22.97 ± 0.41	$\mathbf{95.83 \pm 0.28}$	$\mathbf{93.66 \pm 0.32}$
Unweighted	0.46	37,415	0.93	22.90 ± 0.30	95.77 ± 0.16	93.56 ± 0.31

Regarding the weighting of the edges, the results in Table 2 show that the difference in average top performance is of just 0.06 and 0.10% points in terms of Purity F_1 and BCubed F_1, respectively. This suggests that the presence of the edges is more important for the approach than their weight. Still, in Fig. 2 we can see that using weighted edges increases the robustness of the approach to changes in the neighboring threshold.

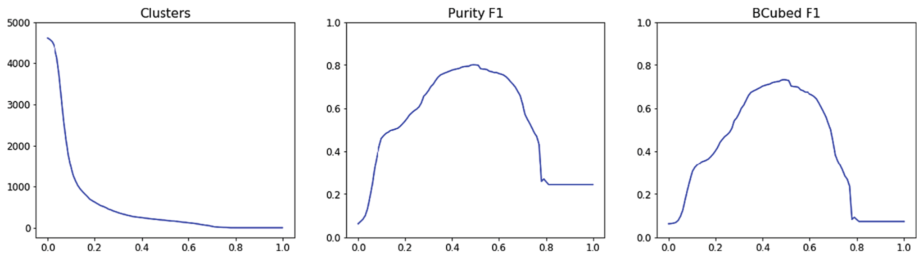

Fig. 3. Results on the test data. The xx axis refers to the neighboring threshold used to create the edges.

Table 3. Results on the test data. d refers to the neighboring threshold. CC refers to the clustering coefficient.

	d	Edges	CC	Clusters	Purity F_1	BCubed F_1
Dev. Threshold	0.47	347,202	0.91	196.63 ± 1.68	79.97 ± 0.21	73.07 ± 0.25
Best Threshold	0.49	364,829	0.91	186.33 ± 0.98	$\mathbf{80.26 \pm 0.17}$	$\mathbf{73.43 \pm 0.19}$

Figure 3 shows the results achieved when applying the same approach to the test data. Although the performance is lower, we can observe patterns similar to those observed on the development data. The only difference is that there is a more pronounced performance drop immediately after the threshold that leads to highest performance. Nonetheless, as shown in Table 3, the threshold selected on development data, 0.47, is lower and very close to the best threshold on test data, 0.49. This shows that our grid-search approach to define the threshold generalizes well. Still, the average performance loss in relation to when using the best threshold is of 0.29 and 0.36% points in terms of Purity F_1 and BCubed F_1, respectively. It is interesting to observe that, contrarily to what happened on development data, the approach overestimates the number of clusters. However, this can be explained by the fact that the test data includes more instances of different verbs that evoke the same frame. Once again, this suggests that either the representations or the distance metric are unable to capture all the required information.

Table 4. Comparison with previous approaches in terms of performance on the test data.

	Purity F_1	BCubed F_1
Baseline	73.78	65.35
Ribeiro et al. [22]	75.25	65.32
Anwar et al. [2]	76.68	68.10
Arefyev et al. [3]	78.15	70.70
Our Approach (Dev. Threshold)	**79.97**	**73.07**

Finally, Table 4 compares the results of our approach with those of the systems that competed in the SemEval shared task. First of all, it is important to refer that while Ribeiro et al.'s [22] approach, on which ours is based, performed worse than the one-frame-per-verb baseline, our surpasses it by 4.37% points in terms of Purity F_1 and 7.72% points in terms of BCubed F_1. This shows the importance of discarding the semantic context provided in the ELMo representations and, most importantly, of identifying a neighboring threshold that allows the approach to generalize. Furthermore, our approach also outperforms the more complex approach by Arefyev et al. [3] by 2.37% points in terms of BCubed F_1. Consequently, it achieves the current state-of-the-art performance on the task.

6 Conclusions

In this paper we have approached semantic frame induction as a community detection problem by applying the Chinese Whispers graph-clustering algorithm to a network with contextualized representations of verb instances as nodes connected by an edge if the cosine distance between them is below a threshold that defines the granularity of the induced frames.

We have shown that the best performance is achieved when using verb instance representations given by the combination of the context-free and syntactical context levels of ELMo representations. The semantic context level impairs the performance since it is overfit to the task on which the model was trained.

We have also observed that weighting the edges with the cosine similarity between the nodes improves the robustness to changes in the neighboring threshold.

We have performed our experiments on the benchmark dataset defined in the context of SemEval 2019s Task 2, which allows us to compare our results with those of previous approaches. In this context, the most important step is to identify the threshold that defines correct granularity according to the gold standard annotations. We did so by performing grid search on the development data and used the same fixed threshold on the test data. This way, we solved the main issue of the approach on which ours was based, which was its lack of generalization ability. In fact, the difference between the best threshold on the development set and that which would lead to the best performance on the test set was of just 0.02.

Using this approach we were able to outperform the more complex approach that won the SemEval shared task by 2.37% points in terms of BCubed F_1. Thus, it achieves the current state-of-the-art performance on the task.

Although we were able to outperform all the previous approaches on the task, the 73.07 BCubed F_1 score achieved on the test data shows that the approach is not able to capture all the information required to induce FrameNet-like frames and that there is still room for improvement. Thus, as future work, we intend to assess the cases that our approach fails to cluster to check whether a different clustering approach or additional features are required, or an adaptation of the contextualized representations is enough. Regarding the latter, it would be interesting to assess whether fine tuning the ELMo representations to the task would make the semantic context level provide relevant information.

Finally, since this approach achieves state-of-the-art performance when inducing semantic frames from verb instances, we intend to assess whether it is also appropriate to induce the semantic roles and the frame-specific slots filled by the arguments of the verbs.

Acknowledgements. This work was supported by Portuguese national funds through Fundação para a Ciência e a Tecnologia (FCT), with reference UID/CEC/50021/2019, and PT2020, project number 39703 (AppRecommender).

References

1. Aharon, R.B., Szpektor, I., Dagan, I.: Generating entailment rules from framenet. In: ACL, vol. 2, pp. 241–246 (2010)
2. Anwar, S., Ustalov, D., Arefyev, N., Ponzetto, S.P., Biemann, C., Panchenko, A.: HHMM at SemEval-2019 Task 2: unsupervised frame induction using contextualized word embeddings. In: SemEval, pp. 125–129 (2019)
3. Arefyev, N., Sheludko, B., Davletov, A., Kharchev, D., Nevidomsky, A., Panchenko, A.: Neural GRANNy at SemEval-2019 Task 2: a combined approach for better modeling of semantic relationships in semantic frame induction. In: SemEval, pp. 31–38 (2019)
4. Bagga, A., Baldwin, B.: Algorithms for scoring coreference chains. In: LREC, pp. 563–566 (1998)
5. Baker, C.F., Fillmore, C.J., Lowe, J.B.: The Berkeley FrameNet project. In: ACL/COLING, vol. 1, pp. 86–90 (1998)
6. Biemann, C.: Chinese whispers: an efficient graph clustering algorithm and its application to natural language processing problems. In: Workshop on Graph-based Methods for Natural Language Processing, pp. 73–80 (2006)
7. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**, 993–1022 (2003)
8. Boas, H.C. (ed.): Multilingual FrameNets in Computational Lexicography: Methods and Applications. Mouton de Gruyter, Berlin (2009)
9. Buchholz, S., Marsi, E.: CoNLL-X shared task on multilingual dependency parsing. In: CoNLL, pp. 149–164 (2006)

10. Das, D., Chen, D., Martins, A.F.T., Schneider, N., Smith, N.A.: Frame-semantic parsing. Comput. Linguist. **40**(1), 9–56 (2014)
11. Devlin, J., Chang, M.W., Kenton, L., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT, vol. 1, pp. 4171–4186 (2019)
12. Fillmore, C.J.: Frame semantics and the nature of language. In: Annals of the New York Academy of Sciences (Origins and Evolution of Language and Speech), vol. 280, pp. 20–32 (1976)
13. Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., Liu, N.F., Peters, M., Schmitz, M., Zettlemoyer, L.S.: AllenNLP: a deep semantic natural language processing platform. CoRR abs/1803.07640 (2017). http://arxiv.org/abs/1803.07640
14. Hagberg, A., Schult, D., Swart, P.: NetworkX. GitHub (2004). https://networkx.github.io/
15. Hearst, M.A.: Automatic acquisition of hyponyms from large text corpora. In: COLING, vol. 2, pp. 539–545 (1992)
16. Lang, J., Lapata, M.: Similarity-driven semantic role induction via graph partitioning. Comput. Linguist. **40**(3), 633–670 (2014)
17. Marcus, M., Santorini, B., Marcinkiewicz, M.: Building a large annotated corpus of english: the Penn Treebank. Comput. Linguist. **19**(2), 330–331 (1993)
18. Materna, J.: LDA-frames: an unsupervised approach to generating semantic frames. In: CICLing, pp. 376–387 (2012)
19. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 3111–3119 (2013)
20. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L.: Deep contextualized word representations. In: NAACL-HLT, vol. 1, pp. 2227–2237 (2018)
21. QasemiZadeh, B., Petruck, M.R.L., Stodden, R., Kallmeyer, L., Candito, M.: SemEval-2019 Task 2: unsupervised lexical frame induction. In: SemEval, pp. 16–30 (2019)
22. Ribeiro, E., Mendonça, V., Ribeiro, R., Martins de Matos, D., Sardinha, A., Santos, A.L., Coheur, L.: L2F/INESC-ID at SemEval-2019 Task 2: unsupervised lexical semantic frame induction using contextualized word representations. In: SemEval, pp. 130–136 (2019)
23. Shen, D., Lapata, M.: Using semantic roles to improve question answering. In: EMNLP-CoNLL, pp. 12–21 (2007)
24. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. In: KDD Workshop on Text Mining (2000)
25. Titov, I., Khoddam, E.: Unsupervised induction of semantic roles within a reconstruction-error minimization framework. In: NAACL-HLT, vol. 1, pp. 1–10 (2015)
26. Titov, I., Klementiev, A.: A Bayesian approach to unsupervised semantic role induction. In: EACL, vol. 1, pp. 12–22 (2012)
27. Ustalov, D., Panchenko, A., Biemann, C.: Watset: automatic induction of synsets from a graph of synonyms. In: ACL, vol. 1, pp. 1579–1590 (2017)
28. Ustalov, D., Panchenko, A., Kutuzov, A., Biemann, C., Ponzetto, S.P.: Unsupervised semantic frame induction using triclustering. In: ACL, vol. 2, pp. 55–62 (2018)
29. Ustalov, D., et al.: Chinese Whispers for Python. GitHub (2018). https://github.com/nlpub/chinese-whispers-python/