10-21-2023

Text augmentation for semantic frame induction and parsing

Saba Anwar
Universität Hamburg

Artem Shelmanov
Mohamed Bin Zayed University of Artificial Intelligence

Nikolay Arefyev
Universitetet i Oslo

Alexander Panchenko
Skolkovo Institute of Science and Technology

Chris Biemann
Universität Hamburg

Language Resources and Evaluation
https://doi.org/10.1007/s10579-023-09679-8

ORIGINAL PAPER

Text augmentation for semantic frame induction and parsing

Saba Anwar[1] · Artem Shelmanov[2] · Nikolay Arefyev[3] · Alexander Panchenko[4,5] · Chris Biemann[1]

Accepted: 28 June 2023

Abstract

Semantic frames are formal structures describing situations, actions or events, e.g., *Commerce buy*, *Kidnapping*, or *Exchange*. Each frame provides a set of frame elements or semantic roles corresponding to participants of the situation and lexical units (LUs)—words and phrases that can evoke this particular frame in texts. For example, for the frame *Kidnapping*, two key roles are *Perpetrator* and the *Victim*, and this frame can be evoked with lexical units *abduct, kidnap*, or *snatcher*. While formally sound, the scarce availability of semantic frame resources and their limited lexical coverage hinders the wider adoption of frame semantics across languages and domains. To tackle this problem, firstly, we propose a method that takes as input a few frame-annotated sentences and generates alternative lexical realizations of lexical units and semantic roles matching the original frame definition. Secondly, we show that the obtained synthetically generated semantic frame annotated examples help to improve the quality of frame-semantic parsing. To evaluate our proposed approach, we decompose our work into two parts. In the first part of text augmentation for LUs and roles, we experiment with various types of models such as distributional thesauri, non-contextualized word embeddings (word2vec, fastText, GloVe), and Transformer-based contextualized models, such as BERT or XLNet. We perform the intrinsic evaluation of these induced lexical substitutes using FrameNet gold annotations. Models based

Nikolay Arefyev: Work done while at the Lomonosov Moscow State University.

✉ Saba Anwar
saba.anwar@studium.uni-hamburg.de

✉ Chris Biemann
chris.biemann@uni-hamburg.de

1 Universität Hamburg, Hamburg, Germany

2 Mohamed bin Zayed University of Artificial Intelligence, Abu Dhabi, United Arab Emirates

3 University of Oslo, Oslo, Norway

4 Skolkovo Institute of Science and Technology, Moscow, Russia

5 Artificial Intelligence Research Institute, Moscow, Russia

Published online: 21 October 2023

on Transformers show overall superior performance, however, they do not always outperform simpler models (based on static embeddings) unless information about the target word is suitably injected. However, we observe that non-contextualized models also show comparable performance on the task of LU expansion. We also show that combining substitutes of individual models can significantly improve the quality of final substitutes. Because intrinsic evaluation scores are highly dependent on the gold dataset and the frame preservation, and cannot be ensured by an automatic evaluation mechanism because of the incompleteness of gold datasets, we also carried out experiments with manual evaluation on sample datasets to further analyze the usefulness of our approach. The results show that the manual evaluation framework significantly outperforms automatic evaluation for lexical substitution. For extrinsic evaluation, the second part of this work assesses the utility of these lexical substitutes for the improvement of frame-semantic parsing. We took a small set of frame-annotated sentences and augmented them by replacing corresponding target words with their closest substitutes, obtained from best-performing models. Our extensive experiments on the original and augmented set of annotations with two semantic parsers show that our method is effective for improving the downstream parsing task by training set augmentation, as well as for quickly building FrameNet-like resources for new languages or subject domains.

Keywords FrameNet · Semantic-frame induction · Semantic-frame parser · Lexical substitution · BERT · XLNet

1 Introduction

Data augmentation refers to techniques used to enlarge human-authored datasets by automatically generating more additional instances that are similar to the original data. In natural language processing (NLP), the augmentation of text is a challenging task because of the discrete, symbolic nature of text data. However, despite the challenges, it provides a way to improve machine learning models in situations where human-annotated data is scarce (Şahin, 2022). In this work, we demonstrate how text augmentation by the means of lexical substitution can be used to enrich representations of semantic frames.

A semantic frame is a linguistic structure used to formally describe the meaning of a situation, action or event (Fillmore, 1982). A frame annotation for a sentence provides (i) a set of *target* words that evoke frames in this sentence, (ii) the respective *frame* for each of the targets and (iii) a set of *arguments* for each of the frames in the sentence. An example sentence is given in Fig. 1 along with two frame annotations taken from FrameNet (Baker et al., 1998)—a widely-used publicly available resource of frame annotations. The example sentence contains two targets: help, which evokes the frame 'Assistance' and hope, which evokes the frame 'Desiring'. The corresponding entries help.v and hope.v with target word(s) lemmas and a part-of-speech tag are called *lexical units* (LUs) or *frame evoking elements (FEE)* in FrameNet. The arguments represent *semantic roles* or *frame elements* (FEs) that act as participants of the situation described by the frame.

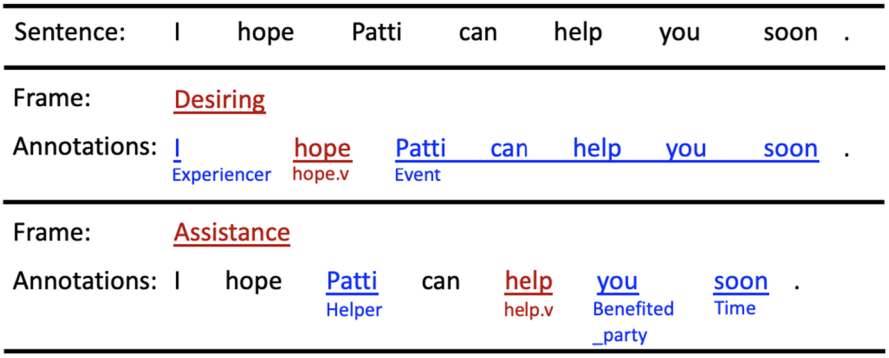

Fig. 1 An example sentence with its color-encoded frame annotations taken from FrameNet. The Red color indicates the lexical unit, and the Blue color indicates the semantic roles. (Color figure online)

Semantic frames have been used in a wide range of applications, such as question answering (Shen & Lapata, 2007; Berant & Liang, 2014; Khashabi et al., 2018), machine translation (Gao & Vogel, 2011; Zhai et al., 2013), and semantic role labeling (Do et al., 2017; Swayamdipta et al., 2018). However, their impact is restricted by the limited availability of annotated resources. Although there are some publicly available resources like FrameNet (Baker et al., 1998) and PropBank (Palmer et al., 2005), yet for many languages and domains, no specialized resources exist. Besides, due to the inherent vagueness of frame definitions, the annotation task is challenging and requires well-trained annotators or very complex crowd-sourcing setups (Fossati et al., 2013).

In this work, we suggest a different approach to the problem: augmenting the FrameNet resource automatically by generating more synthetic examples of existing frame annotations in context via lexical substitution. This way, we are obtaining additional lexical representations of semantic frames (i.e. synonyms of words describing semantic frames). The goal of lexical substitution (McCarthy & Navigli, 2009) is to replace a given word in a particular context with other words, which are semantically similar or related to the original word. The concept is similar to set expansion in its nature; set expansion refers to expanding a small set of seed entities into a larger set by acquiring new entities that belong to the same semantic class (Wang & Cohen, 2007). We consider that given a small set of seed sentences with their frame annotations, we can expand these annotations (a set of seed sentences) by substituting the *targets* and *arguments* of those sentences and aggregating possible substitutions into an induced semantic-frame resource. Table 1 shows one such induced example. To generate these substitutes, we experimented with non-contextualized word embeddings, i.e. fastText (Bojanowski et al., 2017), GloVe (Pennington et al., 2014), and word2vec (Mikolov et al., 2013); distributional thesauri from JoBimText (Biemann & Riedl, 2013); and compared their results to pre-trained Transformer-based contextualized models such as BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019). To complete the comparison, we also include the lexical substitution model of Melamud et al. (2015) that uses dependency-based word and context embeddings and produces context-sensitive lexical substitutes.

Table 1 Lexical representations of the **Assistance** FrameNet frame are retrieved using lexical substitutes from a single seed sentence with the BERT model

Seed sentence: I hope **Patti**$_{\text{Helper}}$ can **help**$_{\text{Assistance}}$ **you**$_{\text{Benefited\_party}}$ **soon**$_{\text{Time}}$.

Substitutes for help: assist, aid
Substitutes for Helper: she, i, he, you, we, someone, they, it, lori, hannah, paul, sarah, melanie, pam, riley
Substitutes for Benefited_party: me, him, folk, her, everyone, people
Substitutes for Time: tomorrow, now, shortly, sooner, tonight, today, later

The words corresponding semantic roles and lexical units are highlighted. Underscored are names of frames and roles

To generate substitutes, we solve two sub-tasks:

- *Lexical unit expansion:* Given a sentence and its *target* word, the task is to generate the frame-meaning-preserving substitutes for this word. This *target* word can be a verb or a noun. The gold substitutes are lexical items specified by FrameNet. We aim at mining their synonyms fitting the semantics of the original FrameNet frame definition.
- *Semantic role expansion:* Given a sentence and an *argument*, the task is to generate frame-meaning-preserving substitutes for this argument. The gold substitutes are concrete realizations of frame in text. We aim at mining their synonyms and other realizations of role-fitting semantics given in the original FrameNet role definition.

Table 1 presents top substitutes produced by BERT for each highlighted word. These substitutes can replace the highlighted words of the seed sentence to generate new sentences. This leads to augmenting the original set of sentences without manual annotations. To assess the quality of these substitutes and their effectiveness for further augmentation of semantic frame expansion, we performed three types of evaluation:

1. Intrinsic evaluation: we evaluate the quality of the substitutes by comparing them to the gold standard FrameNet lexicon, while the performance is reported in terms of precision.
2. Manual evaluation: for a small dataset, we evaluate the quality of the substitutes using human intuition, as the gold standard dataset can be incomplete.
3. Extrinsic evaluation: we conduct an extensive empirical study using the semantic parsers of Swayamdipta et al. (2017) and Shi and Lin (2019). We compare the performance of these parsers on a number of small seed datasets and their augmented versions.

The main contributions of our work are:

- A *one-shot method* for inducing frame-semantic structures using lexical substitution on frame-annotated sentences.
- A *comparative evaluation* for various models including simple non-contextualized word embeddings and Transformer-based models for lexical substitution on the ground truth from FrameNet.
- We show that *combining* the output of individual models can substantially improve the quality of final substitutes in contrast to their individual performance.

– A *manual evaluation* assessment of substitutes to compare it to automatic evaluation with FrameNet gold dataset.
– We *empirically demonstrate* that the dataset augmentation procedure based on the word substitution is improving the performance of frame-semantic parsers. For both parsers Swayamdipta et al. (2017) and Shi and Lin (2019), we see statistically significant improvements in argument identification performance.

The code and datasets are made available online for better reproducibility of our results.[1]

The remainder of this article is organized as follows: Sect. 2 provides an overview of related work for the semantic frame induction task. Section 3 describes the models used for lexical substitution. Section 4 describes the lexical substitution for lexical units and the semantic role expansion experiments. Section 5 describes frame-semantic parsing experiments. Finally, Sect. 6 and 7 conclude the overall findings of this work and discuss the possible future directions.

2 Related work

Many data-driven approaches to frame-semantic parsing that take advantage of annotated resources, such as FrameNet, have been proposed in the literature (Das et al., 2010; Oepen et al., 2016; Yang & Mitchell, 2017; Peng et al., 2018), with SEMAFOR (Das et al., 2014) being the most widely-known system for extracting complete frame structures including target identification, frame identification, argument identification, and argument labeling. Some works focus only on a single parsing step, e.g. frame identification (Hermann et al., 2014; Hartmann et al., 2017); Sikos & Padó 2019, argument labeling with frame identification (Swayamdipta et al., 2017; Yang & Mitchell, 2017), or just argument labeling (Kshirsagar et al., 2015; Roth & Lapata, 2015; Swayamdipta et al., 2018), which can be considered as very similar to PropBank-style (Palmer et al., 2005) semantic role labeling, albeit more challenging because of the high granularity of semantic roles for frames. FrameNet-like resources are available only for very few languages and cover only a few domains. In this article, we venture into the more challenging problem of training a model for frame parsing on merely a very small amount of annotated data. This is similar to the idea of Pennacchiotti et al. (2008), which investigates the utility of semantic spaces and WordNet-based methods to automatically induce new lexical units, evaluating on FrameNet. Resource-scarceness is the typical case here, as some NLP applications might require frames not covered by FrameNet, the granularity of available frames might not match the task, or the parser shall be constructed for a low-resource language.

Several unsupervised semantic frame induction methods have been proposed in the literature. They extract clusters of words from the text, which are then dubbed as semantic frames. These methods are based on hard or probabilistic (soft) clustering of input, commonly represented in the form of dependency trees. Lang and Lapata (2010) perform clustering of verb arguments based on syntactic dependencies. A latent variable probabilistic model is used in Modi et al. (2012) and Titov and Klementiev

[1] https://github.com/uhh-lt/frame-induction-and-parsing.

(2012). Materna (2012, 2013) also cluster subject-verb-object (SVO) triples with a similar model based on LDA (Blei et al., 2003). Kawahara et al. (2014) apply the Chinese Restaurant Process clustering to a collection of verbal predicates and their argument instances. Ustalov et al. (2018) use tri-clustering on SVO triples to jointly induce both lexical units and their arguments. The downside of unsupervised frame induction is the lack of control over the semantics of obtained word clusters and frame granularity. Due to this, such methods are not widely applied.

Our approach conceptually differs from these frame induction methods. We consider the effort of labeling one or a few sentences with frames as tolerable. This enables us to guide the construction of the FrameNet resource with the desired properties. Our experiments show that this minimal supervision can be used to produce the majority of LUs of semantic frames defined in FrameNet and generate meaningful semantic roles. However, since our method uses some training data, it is *not directly comparable* to these completely unsupervised approaches.

There are few recent works that use pre-trained language models for lexical substitution. Our method takes a direct motivation from the works of Amrami and Goldberg (2018) and Arefyev et al. (2019a). Amrami and Goldberg (2018) suggest predicting substitute vectors for target words using pre-trained ELMo (Peters et al., 2018) and dynamic symmetric patterns. Arefyev et al. (2019a) use the same idea of substitute vectors for the SemEval 2019 (QasemiZadeh et al., 2019) frame induction task, but replace ELMo with BERT (Devlin et al., 2019) for improved performance. Zhou et al. (2019) propose a method for lexical substitution with BERT. A more recent work by Arefyev et al. (2020) shows that injecting the information about target word into state-of-the-art language models can significantly improve their performance for lexical substitution. The re-surge of lexical substitution arises from the fact that it has a wide range of applications in NLP tasks such as word sense induction (Amrami & Goldberg, 2018; Arefyev et al., 2019b, 2020), paraphrasing or text simplification (Kriz et al., 2018; Lee & Yeung, 2019). It is also used for quality assessment of semantic distributional models (Buljan et al., 2018). We are—to our knowledge—the first to employ lexical substitution for the expansion of semantic-frame resources and the first to show that it improves the performance of frame parsers. This work is a direct extension of our previous preliminary work (Anwar et al., 2020) with more advanced lexical substitution methods from Arefyev et al. (2020) and with experiments on the frame-semantic parsing task for extrinsic evaluation of the proposed approach.

3 Inducing lexical representations of frames

We experiment with two groups of lexical substitution models: non-contextualized and contextualized models. Regarding non-contextualized models, we report experiments with static embeddings from word2vec (Mikolov et al., 2013), GloVe (Pennington et al., 2014) and fastText (Bojanowski et al., 2017); further, we utilize distributional thesauri constructed with JoBimText (Biemann & Riedl, 2013).

For contextualized models, we use two pre-trained Transformer-based models BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019), as well as the lexical substitution model of Melamud et al. (2015).

3.1 Non-contextualized models

In this section, we describe common approaches to represent meaning of individual words independently of their context.

3.1.1 Non-contextualized word embeddings

Non-contextualized word embeddings are vector representations of words constructed in a way that words occurring in similar contexts are expected to have similar vectors. To produce substitutes for a target word, we take 200 nearest neighbors of the target word according to the cosine similarity measure between non-contextualized embeddings. We use the following pre-trained embeddings: fastText trained on the Common Crawl corpus,[2] GloVe trained on the Common Crawl corpus,[3] word2vec trained on Google News.[4] The embeddings from all of these models have the dimensionality of 300.

3.1.2 Distributional thesauri

In contrast to the standard word embeddings, distributional thesauri (DT) can capture word similarities using simple n-gram context features and more complex linguistic context features (Lin, 1998), e.g. dependency relations. Grammatical features provide a more refined set of similar terms as compared to bag-of-words-based word embeddings, but their representations are sparser. JoBimText (Biemann & Riedl, 2013) is a framework that offers many DTs constructed using various corpora. Context features for each word are ranked using the lexicographer's mutual information (LMI) score (Kilgarriff et al., 2004) and used to compute word similarity by feature overlap. We extract 200 nearest neighbors for the target word. In the experiments, we use two JoBimText DTs: (i) DT built on Wikipedia with *n*-grams as contexts and (ii) DT built on the combination of Wikipedia, Gigaword (Parker et al., 2009), ukWaC (Ferraresi et al., 2008), and LCC (Goldhahn et al., 2012) (59 GB in total) using dependency relations as context.

3.2 Contextualized models

While non-contextualized models are computationally effective, they cannot handle polysemous words. This drawback is addressed by context-aware models that can produce different word representations depending on the context. Therefore, they can also be used to generate different substitutes for a target word depending on its context.

[2] http://fasttext.cc.

[3] https://github.com/stanfordnlp/GloVe.

[4] https://code.google.com/p/word2vec.

3.2.1 Melamud's lexical substitution model

The method proposed by Melamud et al. (2015) uses syntax-based skip-gram embeddings of Levy and Goldberg (2014) for a word and its context to produce context-sensitive lexical substitutes, where the context of a word is represented using its dependency relations. We use the embeddings from Melamud et al. (2015), which were trained on the ukWaC (Ferraresi et al., 2008) corpus. To find dependency relations, we use the Stanford Parser (Chen & Manning, 2014) (version 4.0.0) and collapse dependencies that include prepositions. Top k substitutes are produced only when both the target word and some of its context words are present in the vocabulary of the model.

The following cosine-similarity-based measures are proposed in Melamud et al. (2015) to compute suitability of a substitute s for a given target word t in a given context C:

$$add = \frac{cos(s,t) + \sum_{c \in C} cos(s,c)}{|C| + 1}, \tag{1}$$

$$balAdd = \frac{|C| \cdot cos(s,t) + \sum_{c \in C} cos(s,c)}{2 \cdot |C|}, \tag{2}$$

$$mult = \sqrt[|C|+1]{\frac{cos(s,t)}{2} \cdot \prod_{c \in C} cos(s,c)}, \tag{3}$$

$$balMult = \sqrt[2 \cdot |C|]{\frac{(cos(s,t)+1)^{|C|}}{2} \cdot \prod_{c \in C} \frac{cos(s,c)+1}{2}}. \tag{4}$$

Two of these measures (mult and balMult) use the geometric mean to produce high scores when the target word and the context words are all similar to a substitute word, whereas the other two (add and balAdd) use arithmetic mean to achieve high scores, even if some of them are not similar. The balAdd and balMult measures emphasize more on the similarity of substitutes to the target word.

3.2.2 Pre-trained transformer-based models

Transformer-based models pre-trained on various language modelling objectives can predict the distribution of substitutes for a target word in a given context. In this work, we use two Transformer-based models: BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019). The BERT model is the encoder part of the Transformer model (Vaswani et al., 2017) that was pre-trained with the masked language modeling (MLM) objective. In a nutshell, some randomly selected tokens in the training corpus are replaced by a special [MASK] token, and the objective is to restore those tokens based on the remaining left and right context. The XLNet model is an autoregressive language model pre-trained with the permutation language modelling (PLM) objective. In this objective, a new random permutation of tokens is generated for each example in each epoch. The model learns to predict each word given all preceding words in this permutation. Simple autoregressive models learn to predict words one by one either

from left to right, or vice versa. In contrast, XLNet learns to predict words in any order and to take advantage of both left and right context of each word. Pre-trained Transformer-based models have effectively outperformed the previous state of the art in many downstream tasks. For our experiments, we use *BERT large-cased* and *XLNet large-cased* as implemented in the HuggingFace library (Wolf et al., 2019).

There are several ways to provide information about the target word to the Transformer-based substitution models. We use three options: keeping the original target word in place, using dynamic patterns, and combining conditional probabilities of substitutes given the context and the target word.

Original input: Although BERT and XLNet were both trained to guess a word they do not observe from its context, a substitution model can produce better substitutes if it sees not only the context but also has some information about the target word (Arefyev et al., 2020). The simplest method to introduce the information about the target word is just feed the original example without any masking and get predictions for the position of the first subword of the target word. In case of XLNet, we generate an attention mask resulting in all tokens attending to all other tokens in the content stream. Even though the contextualized embedding for the target position comes from the query stream, it still depends on the target word indirectly through the contextualized embeddings of all other tokens.

Dynamic patterns: Amrami and Goldberg (2018, 2019) use dynamic patterns to inject information about the target word (T) when generating substitutes with ELMo and BERT for the word sense induction task. These patterns are similar to the Hearst patterns (Roller et al., 2018) and are used to replace the target word (T) with some coordinate structure (e.g. "T and -") to extract better substitutes. For example, the sentence "Rob **sold** his car to Miller", after applying the pattern "T and -" to the target word "sold" will be transformed to "Rob **sold and -** his car to Miller". Now the substitutes will be generated for the token "-" instead of the original target word "sold". Arefyev et al. (2019a, 2019b, 2020) also use these patterns to generate substitutes for solving the lexical frame induction and the word sense induction tasks. For our experiments with BERT, we try patterns "T and -" and "T and T" (where the target word is duplicated).

+embs: Arefyev et al. (2020) proposed a method that combines the probability of a potential substitute occurring in a given context $P(s|C)$ with the probability reflecting distributional similarity of this substitute to the target word $P(s|T)$:

$$P(s|C,T) \propto \frac{P(s|C)P(s|T)}{P(s)^{\beta}}. \tag{5}$$

The probability $P(s|C)$ is directly estimated by a language model, while $P(s|T)$ is calculated by applying the temperature softmax over the inner product of their non-contextualized embeddings:

$$P(s|T) \propto exp\left(\frac{\mathbf{v}_s, \mathbf{v}_T}{\tau}\right). \tag{6}$$

where, $\mathbf{v}_s$ and $\mathbf{v}_T$ are embeddings of the corresponding words, and τ is the temperature hyperparameter used to balance between closeness of substitutes to the target word and their fitness to the given context. The hyperparameter β can be tuned to promote or penalize frequent words as substitutes. Prior word probabilities $P(s)$ are obtained from the wordfreq library.[5] The optimal values of τ and β are selected using the development dataset. In the experiments with XLNet, we selected these values for lexical unit expansion and semantic roles expansion separately.

3.3 Combination of models

To combine the advantages of several models, we also ensemble the predictions of the best-performing models. Since different models produce scores that are not directly comparable, we consider only substitute ranks, i.e. their positions after ordering substitutes according to their scores obtained from each model. We compute the combined rank as:

$$Combined\ Rank(w) = \frac{1}{L}\sum_{i=1}^{L} rank_i(w). \tag{7}$$

where $rank_i(w)$ is the rank of w among the substitutes predicted by the i-th model if it is predicted by the i-th model, and 1000 otherwise. Each model predicts at most 200 substitutes and the value 1000 is used to penalize the combined rank of a substitute that is not predicted by all models in the ensemble. The goal of this penalization is to rank words that are predicted by N models higher than words that are predicted by $N-1$ models.

4 Intrinsic evaluation: augmenting lexical descriptions in FrameNet

In this section, we show how lexical substitution can be used to fill the gaps in a lexical resource. Namely, given a partially completed descriptions of lexical-semantic frames from the FrameNet resource, one can reconstruct the missing semantic roles and lexical units using our approach.

4.1 Experimental setup

4.1.1 Datasets

We use FrameNet (Baker et al., 1998) version 1.7 to generate our evaluation datasets. The combined data from fulltext and exemplars annotations of FrameNet contains around 170k sentences with 1014 frames, 7828 types of semantic roles, and 10,340 unique lexical units. Table 2 describes more characteristics of these datasets. The datasets for evaluation were derived automatically. Semantic roles and lexical units can consist of single or multiple tokens. For this work, we have only considered single-token substitution.

[5] https://pypi.org/project/wordfreq.

Table 2 Statistics of evaluation datasets for verb lexical units, noun lexical units, and semantic role expansion tasks derived from FrameNet-1.7

Characteristics	Verb lexical unit	Noun lexical unit	Semantic roles
Number of frames	644	650	1014
Number of LUs/roles	3894	4226	7828
Total number of annotations	82,410	78,384	405,588
Annotations with single-token LU/role	79,584	76,229	191,251
Number of sentences	77,119	67,817	167,636
Number of sentences per frame			
Mean	127	118	190
Std	311	285	396
Min	1	1	1
25%	21	10	27
50%	52	34	74
75%	119	93	181
Max	5976	3101	6394
Sentence length in words			
Mean	23	25	24
Std	11	11	11
Min	1	1	1
25%	15	18	16
50%	22	24	23
75%	29	31	30
Max	250	250	250

Single-token lexical unit and semantic role expansion: In order to create evaluation data for the LU expansion tasks, for each sentence containing an annotated LU we consider other LUs of the corresponding semantic frame as gold substitutes. We keep only LUs marked as *verbs* and *nouns* in FrameNet. FrameNet annotations contain 10 different types of lexical units based on their part-of-speech tags, but verbs and nouns cover about 79% of annotations. We created two separate datasets for the *verb lexical unit expansion* task and the *noun lexical unit expansion* task. To construct the evaluation dataset for the *semantic role expansion* task, for each of the sentences that contain an annotation of a given semantic role we consider all the single-word annotations from the rest of the corpus marked with the same role and related to the same frame as the gold substitutes.

An example frame with its full set of lexical units and two semantic roles is shown in Fig. 2. It contains some example sentences annotated with lexical units and semantic roles. To illustrate how we generated the evaluation datasets Table 3 shows evaluation data that could have been generated based on annotations in Fig. 2. However, the final datasets for experiments have been generated using all data from fulltext and exemeplars annotations of the FrameNet resource, but not example sentences from the frame description files. The resulting datasets contain 79,584 records for verb LUs,

Arrest

Definition:

Authorities charge a Suspect, who is under suspicion of having committed a crime (the Charges), and take him/her into custody.

The police ARRESTED Harry on charges of manslaughter.

FEs:

Authorities [Auth]

The Authorities charge the Suspect with committing a crime, and take him/her into custody.
They ARRESTED Harry for shop lifting.

Suspect [Susp]

The Suspect is taken into custody, under suspicion of having committed a crime.
The police placed Jhon under house ARREST.
She was ARRESTED with the Minister of Finance, Javier Soledad.

...
(we omit other FEs to keep this example short)

Lexical Units:

apprehend.v, apprehension.n, arrest.n, arrest.v, book.v, bust.n, bust.v, collar.v, cop.v, nab.v, summons.v

Fig. 2 The frame *Arrest* from FrameNet simplified for illustrative purposes. It contains a frame definition and an example sentence, as well as names, descriptions and examples for a few semantic roles (FEs), and finally, a set of lexical units associated with this frame

Table 3 Evaluation data generated from the FrameNet descriptions shown in Fig. 2

Sentence with Target Word	Gold Set Substitutes
Verb Lexical Unit Expansion	
The police arrested Harry on charges of manslaughter . They arrested Harry for shop lifting . She was arrested with the Minister of Finance, Javier Soledad .	apprehend, book, bust, collar, cop, nab, summons
Noun Lexical Unit Expansion	
The police placed Jhon under house arrest .	apprehension, bust
Semantic Role Expansion	
The police arrested Harry on charges of manslaughter .	She, Jhon
The police placed Jhon under house arrest .	She, Harry
She was arrested with the Minister of Finance, Javier Soledad .	Harry, Jhon
They arrested Harry for shop lifting .	-

Since we consider only single-token words, the gold set substitutes in the last row is empty

76,229 for noun LUs, and 191,252 records for role expansion. We use 10% of examples as the development sets for tuning the hyperparameters τ and β of BERT+embs and XLNet+embs.

Hyperparameters for +embs: Following Arefyev et al. (2020), we set the default values of the hyperparameters $\beta = 1$ and $\tau = 0.1$. For XLNet+embs we additionally selected the optimal values of these hyperparameters based on the development subsets of all three datasets. The following values were selected: $\beta = 0.5$ and $\tau = 0.05$ for the verb lexical unit expansion task, $\beta = -1$ and $\tau = 0.07$ for the noun lexical unit

expansion task, and $\beta = -0.5$ and $\tau = 0.15$ for the semantic role expansion task. Negative values of β mean that it is beneficial to promote more frequent words as substitutes for the latter two tasks.

4.1.2 Evaluation measures

To evaluate the quality of generated substitutes, we use the standard ranking metric precision at k ($p@k$), where k represents the number of the highest ranked substitutes to be considered. While $p@k$ measures the correctness of the first k substitutes, to evaluate the quality of the entire list of generated substitutes, we use mean average precision at level k ($MAP@k$):

$$MAP@k = \frac{1}{N}\sum_{i=1}^{N} AP^i@k, \tag{8}$$

where

$$AP^i@k = \frac{1}{min(k, R^i)}\sum_{l=1}^{k} r_l^i \cdot p^i@l.$$

Here, N is the total number of examples in the dataset; R^i is a number of possible correct answers for the i-th example; r_l^i equals 1 if the l-th predicted substitute for the i-th example is correct and 0 otherwise. We present $p@k$ at levels: 1, 5, as well as $MAP@50$. Sometimes the post-processing procedure leads to fewer than k substitutes generated. We consider absence of a substitute for a position as a wrong answer of the model.

4.1.3 Text pre-processing

For non-contextualized embeddings, we tried generating substitutes for the target word both with and without lemmatization and found that lemmatizing the target word has no positive effect on the model performance. We assume that the grammmatical form of the target word contains some information about its context, and this can help generating better substitutes by the models that do not have direct access to the context. Thus, we do not use lemmatization for this kind of models. For DTs, lemmatization produced better results, mainly because corpora were lemmatized before building the DTs. Therefore, we employ lemmatization in case of DT models. For all contextualized models, the context is tokenized on whitespace and we do not apply any pre-processing to the original sentences extracted from the FrameNet-annotated corpus.

For BERT, we remove all subwords of the target word except for the first subword during pre-processing. This results in better substitutes generated, see Table 4 for an illustration of this effect. For XLNet, we follow Arefyev et al. (2020) and prepend a fixed prompt, i.e., a 'warming up' text fragment, ending with the end-of-document token as the initial context to each example.

Table 4 Pre-processing target words with multiple subwords

Sentence	Work advances at Iranian Uranium **Enrichment** Site .
Tokens	Work, advances, at, Iranian, U, ##rani, ##um, En, ##rich, ##ment, Site, .
Substitutes	En, en, Em, Ex, E, In, Re, Eva, Un, Up
Modified tokens	Work, advances, at, Iranian, U, ##rani, ##um, En, Site, .
Substitutes	Test, Production, Mining, Project, Testing, Research, Development, Mine, Launch, Plant

The target word is **bold**. In the last row, the model generates better substitutes when subwords *##rich*, *##ment* are removed

4.1.4 Post-processing of substitutes

Lexical substitutes can contain noisy tokens, such as numbers, individual symbols, model specific special tokens, e.g. `[UNK]`, or sub-words marked with the ## prefix. In post-processing, we remove all such non-words from the list of generated substitutes. Substitutes often contain different forms of the same word, especially when static word embeddings are employed. Therefore, we lemmatize the generated substitutes using the Pattern library (Smedt & Daelemans, 2012) and remove duplicated lemmas. For the verb lexical unit expansion task, we drop all substitutes that are not verbs. For this purpose, we use a dictionary of verbs composed of verb lexicons taken from Pattern, WordNet (Miller, 1995), and FreeLing (Padró & Stanilovsky, 2012). For the noun lexical unit expansion task, we remove stopwords, apply POS tagging, and retain only nouns in the final output.

4.1.5 Combination of models

For model combinations, we consider the best-performing individual models according to the mean average precision from the following categories: non-contextualized embeddings (nc-emb), distributional thesaurus based models (DT), the contextualized models of Melamud et al. (2015), contextualized models based on pre-trained Transformer-based models, and the Transformer models with the embeddings of target words (+embs).

4.2 Results

4.2.1 Lexical unit expansion task

The results for the lexical unit expansion tasks are presented in Table 5.

Verb lexical unit: In the verb lexical unit expansion task the best performance among non-contextualized models was achieved by fastText ($p@1 = 0.388$ and $MAP@50 = 0.156$), closely followed by word2vec ($p@1 = 0.388$ and $MAP@50 = 0.151$). The DTs considered in our experiments perform worse than the embedding-based models word2vec and fastText. Among Melamud's models, the best performance was achieved by the balAdd (Melamud et al., 2015) with $p@1 = 0.393$ and $MAP@50 = 0.156$, whereas the balMult performed slightly worse because balAdd can produce substitutes even if the context has no similarity to the substitute, which is only useful for

Table 5 Evaluation of lexical substitutes for lexical unit and semantic role expansion

Model	Verb Lexical Unit Expansion			Noun Lexical Unit Expansion			Semantic Role Expansion		
	p@1	p@5	MAP@50	p@1	p@5	MAP@50	p@1	p@5	MAP@50
Upperbound	1.000	0.904	1.000	1.000	0.918	1.000	1.000	0.971	1.000
Non-contextualized models									
GloVe	0.373	0.254	0.131	0.357	0.267	0.109	0.303	0.251	0.070
fastText	**0.388**	**0.286**	**0.156**	0.193	0.148	0.056	0.184	0.135	0.029
word2vec	**0.388**	0.273	0.151	0.278	0.186	0.068	0.319	0.223	0.051
DT wiki	0.313	0.208	0.106	0.388	0.294	0.137	**0.368**	**0.281**	**0.091**
DT 59g	0.353	0.255	0.141	**0.398**	**0.307**	**0.145**	0.350	0.273	0.086
Contextualized models: Melamud)									
Melamud add	0.326	0.234	0.125	0.299	0.217	0.089	0.369	0.278	0.070
Melamud balAdd	**0.393**	**0.281**	**0.156**	**0.328**	**0.240**	**0.099**	**0.378**	**0.284**	**0.072**
Melamud mult	0.279	0.206	0.107	0.261	0.195	0.078	0.355	0.265	0.066
Melamud balMult	0.390	0.279	0.154	0.324	0.238	**0.099**	0.375	0.277	0.070
Contextualized Transformer-based models									
BERT	0.386	0.264	0.137	0.403	0.289	0.114	0.472	0.388	0.118
XLNet	0.352	0.260	0.137	0.386	0.293	0.122	0.513	0.421	0.144
BERT [Tand-]	0.199	0.158	0.069	0.204	0.162	0.059	0.38	0.3	0.084
BERT [TandT]	0.407	0.277	0.139	0.401	0.288	0.114	0.446	0.369	0.112
BERT+embs	0.442	0.299	0.162	0.44	0.323	0.136	0.444	0.381	0.123
XLNet+embs	0.487	0.333	0.189	0.486	0.363	0.167	0.522	0.449	0.159
XLNet+embs (optimal)	**0.504**	**0.347**	**0.199**	**0.499**	**0.371**	**0.171**	**0.542**	**0.461**	**0.161**
Combined models									
nc-emb + DT	0.435	0.313	0.179	0.431	0.323	0.147	0.388	0.312	0.102
Melamud balAdd + DT	0.452	0.325	0.184	0.448	0.343	0.160	0.409	0.358	0.108
Melamud balAdd + nc-emb	0.440	0.323	0.184	0.432	0.326	0.142	0.472	0.378	0.110
XLNet + nc-emb	0.472	0.321	0.179	0.477	0.347	0.146	0.526	0.398	0.131
XLNet + DT	0.461	0.312	0.163	0.495	0.367	0.166	0.501	0.411	0.139
XLNet + Melamud balAdd	0.476	0.326	0.182	0.496	0.368	0.159	**0.574**	**0.482**	0.148
XLNet+embs + nc-emb	0.486	0.331	0.187	0.486	0.354	0.155	0.507	0.387	0.128
XLNet+embs + DT	0.491	0.332	0.180	**0.513**	0.380	0.179	0.484	0.407	0.141
XLNet+embs + Melamud balAdd	0.494	0.338	0.191	0.506	0.375	0.170	0.563	0.480	0.151
Melamud balAdd + nc-emb + DT	0.465	0.337	0.195	0.460	0.352	0.165	0.489	0.391	0.127
XLNet + nc-emb + DT	0.487	0.334	0.194	0.496	0.370	0.171	0.489	0.410	0.141
XLNet + Melamud balAdd + nc-emb	0.492	0.336	0.198	0.503	0.371	0.168	0.559	0.459	0.152
XLNet + Melamud balAdd + DT	**0.496**	**0.342**	0.199	0.512	0.384	0.186	0.551	0.467	**0.156**
XLNet+embs + nc-emb + DT	0.482	0.336	0.195	0.495	0.369	0.174	0.462	0.395	0.139
XLNet+embs + Melamud balAdd + DT	0.495	**0.342**	**0.201**	0.507	**0.385**	**0.189**	0.536	0.457	**0.156**

Here, *+embs* refers to the default values of the hyperparameters τ and β, whereas *+embs (optimal)* refers to the model with these hyperparameters selected on the development sets. The best result in each block is in **bold**. The best result in the table is **underlined**. For combined models, one best model is taken from each category, for the verbs lexical unit expansion task, DT is DT 59G, nc-embs is fastText; for the nouns lexical unit expansion task, DT is DT 59G, nc-embs is GloVe; for semantic role expansion task, DT is DT wiki, nc-embs is GloVe. The Upperbound results present the maximum possible score if all predictions are considered correct. For this, substitutes were taken randomly from the gold dataset

monosemous words. Even though simple BERT and XLNet models (without masking) performed comparably, they could not outperform fastText and word2vec. However, a close examination of some examples shows that contextualized models do make a difference when the target word is polysemous, see Table 6.

Applying the dynamic patterns helped to improve the performance of BERT. While BERT with the pattern "T and -" is substantially worse than just using the vanilla BERT model without masking, the second pattern "T and T" yields the best results for the BERT models. These experiments confirm that such dynamic patterns can help to better capture the semantics of a target word and produce better substitutes. BERT with the pattern "T and T " outperforms all other models in terms of precision at $k = 1$, but distinctively falls behind fastText, balAdd, and balMult (Melamud et al., 2015) on the higher levels of precision and in terms of the MAP score.

Using the +embs method proved to be a better approach to target word injection compared to the dynamic patterns. With this approach both BERT and XLNet have outperformed all other models. For XLNet, using +embs with the optimal hyperparameters has achieved the best performance overall with $p@1 = 0.504$ and $MAP@50 = 0.199$. Even the model with the default hyperparameters has obtained better performance than all other models ($p@1 = 0.487$ and $MAP@50 = 0.189$).

For combined models we considered: (1) *fastText* as nc-emb, (2) *DT 59g* as DT, (3) *balAdd* for Melamud et al. (2015), (4) *XLNet*, and (5) XLNet+embs with optimal hyperparameters. Combining substitutes predicted by individual models has a mix effect and the combined scores are sensitive to the individual performance of participating models of the combination. Overall, the highest MAP score is achieved by combining XLNet+embs with balAdd (Melamud et al., 2015) and DT ($MAP = 0.201$). For the combinations that are based on XLNet+embs, the precision scores slightly decrease in comparison to its individual performance. But, all other combinations obtain higher precision scores than their individual counterparts. Especially the tri-model combinations based on simple XLNet model closely matched the performance of the best model of XLNet+embs ($MAP = 0.199$ and $MAP = 0.198$).

Table 6 contains example sentences with highlighted target words and top 10 substitutes generated by all models (along with the ground truth FrameNet annotations). The first example presents a LU that is associated with only one frame and is unambiguous, all models have produced many matching substitutes. The other two examples present the fact that LUs contain various senses, leading to multiple associated frames to it. Non-contextualized models except GloVe and fastText have predicted at least one valid substitute for the first frame *Departing*, but most of them failed to produce any substitutes in top 10 for the *Causation* frame. But BERT and XLNet have successfully generated several matching substitutes for both cases. Particularly XLNet has predicted more matching substitutes than BERT.

Noun lexical unit: Among non-contextualized models, the best performance was achieved by DT 59g ($p@1 = 0.398$ and $MAP@50 = 0.145$). Unlike verbs for which the embedding-based models outperformed DTs, in case of nouns DTs perform better than embeddings. The contextualized models of Melamud et al. (2015) also have significantly lower performance compared to DTs. BERT and XLNet perform comparably to DTs. Dynamic patterns do not improve BERT's performance, but the +embs method improves the results significantly for both BERT and XLNet models. The XLNet+embs model with the optimal hyperparameters has achieved the best performance with $p@1 = 0.499$ and $MAP@50 = 0.171$.

For combining the following models were taken: (1) *GloVe* as nc-emb, (2) *DT 59g* as DT, (3) *balAdd* for Melamud et al. (2015), (4) *XLNet* and (5) XLNet+embs with the optimal hyperparameters. The highest MAP score was again achieved by combining XLNet+embs with balAdd (Melamud et al., 2015) and DT ($MAP = 0.189$). Unlike verbs, for nouns proper model combinations improve the results compared to the best individual model. Since the embedding-based models perform worst for nouns, the combinations with nc-emb have lowest MAP scores among all other combinations.

As mentioned in Sect. 4.1.4, we use the pattern library for lemmatization and POS tagging for nouns. To investigate the effect of POS tagging on model performance we

Table 6 Examples for the verb lexical unit expansion task

Frame: Statement
Seed sentence: The report **stated**$_{\text{state}}$, however , that some problems needed to be solved , principally that of lack of encouragement of cadres and individuals to exercise their democratic right of freedom of expression .
GloVe: explain, note, agree, acknowledge, mention, conclude, argue, discuss, suggest, indicate **fastText:** note, explain, indicate, reiterate, opine, mention, re-state, assert, aver, acknowledge **word2vec:** comment, note, assert, remark, explain, indicate, emphasize, stress, intimate, say **DT wiki:** initial, exact, underlie, require, continue, increase, allege, own, observe, expect **DT 59g:** continue, seem, exact, increase, underlie, initial, profess, purport, mark, general **Melamud add:** aver, assert, argue, indicate, say, contend, note, reiterate, concede, suggest **Melamud balAdd:** indicate, stipulate, assert, reiterate, say, note, aver, re-state, restate, argue **Melamud mult:** argue, aver, assert, contend, say, concede, indicate, reiterate, note, suggest **Melamud balMult:** indicate, stipulate, assert, say, aver, reiterate, note, argue, re-state, emphasize **BERT:** find, conclude, say, note, declare, indicate, cite, identify, express, warn **XLNet:** note, conclude, say, find, acknowledge, stress, add, indicate, recognize, emphasize **XLNet+embs:** note, indicate, say, acknowledge, mention, assert, comment, stipulate, suggest, declare **FrameNet gold:** proclaim, mention, claim, detail, profess, tell, caution, allow, propose, comment, preach, reaffirm, avow, challenge, recount, reiterate, pronounce, relate, remark, report, say, speak, state, allege, suggest, conjecture, talk, write, contend, venture, declare, add, hazard, pout, announce, exclaim, smirk, address, confirm, explain, assert, gloat, acknowledge, insist, maintain, note, observe, aver, refute, attest, describe
Frame: Departing
Seed sentence: We hurried down the village street and found , as we had expected , that the inspector was just **leaving**$_{\text{leave}}$ his lodgings .
GloVe: return, back, left, rest, stay, while, arrive, out, off, stick **fastText:** left, abandon, return, rejoin, exit, arrive, reenter, re-enter, enter, depart **word2vec:** left, return, depart, exit, enter, abandon, arrive, quit, rejoin, reenter **DT wiki:** demise, body, action, bid, plight, fate, change, lap, row, sonnet **DT 59g:** see, go, get, back, long, body, will, cool, muse, reshuffle **Melamud add:** resign, vacate, abandon, quit, bide, desert, shun, outgrow, neglect, retire **Melamud balAdd:** abandon, vacate, quit, return, left, depart, re-join, enter, rejoin, re-enter **Melamud mult:** resign, vacate, bide, quit, abandon, contemplate, shun, neglect, outgrow, desert **Melamud balMult:** abandon, vacate, quit, return, left, depart, resign, enter, re-enter, desert **BERT:** enter, depart, exit, change, left, reach, turn, take, putt, make **XLNet:** enter, approach, reach, depart, clean, finish, find, pass, check, change **XLNet+embs:** depart, left, enter, quit, abandon, flee, return, finish, arrive, reach **FrameNet gold:** escape, vamoose, vanish, skedaddle, depart, exit, decamp, leave, emerge, disappear
Frame: Causation
Seed sentence: Mysteriously , the Anasazi vanished from the valley around a.d. 1150 , **leaving**$_{\text{leave}}$ it to be repopulated by the Southern Paiutes , another hunter-gatherer tribe .
GloVe: return, back, left, rest, stay, while, arrive, out, off, stick **fastText:** left, abandon, return, rejoin, exit, arrive, reenter, re-enter, enter, depart **word2vec:** left, return, depart, exit, enter, abandon, arrive, quit, rejoin, reenter **DT wiki:** demise, body, action, bid, plight, fate, change, lap, row, sonnet **DT 59g:** see, go, get, back, long, body, will, cool, muse, reshuffle **Melamud add:** chance, obliterate, abandon, return, rejoin, leg, quit, desecrate, bequeath, replace **Melamud balAdd:** abandon, return, rejoin, chance, obliterate, re-join, bequeath, left, desecrate, prefer **Melamud mult:** chance, obliterate, return, quit, abandon, leg, rejoin, replace, reassemble, desecrate **Melamud balMult:** abandon, chance, return, obliterate, rejoin, bequeath, left, re-join, desecrate, prefer **BERT:** left, allow, abandon, cause, putt, join, place, free, lead, help **XLNet:** allow, cause, force, prompt, lead, enable, send, abandon, permit, reveal **XLNet+embs:** depart, left, abandon, allow, cause, return, send, force, flee, quit **FrameNet gold:** cause, leave, mean, render, wreak, bring, dictate, sway, force, make, precipitate, send, raise, motivate, induce, put, see

Matches with the gold substitutes are in green. XLNet+embs is the best model with optimal hyperparameters

compare the results produced with the pattern library and another library lemminflect.[6] The former only returns the most suitable POS tag, while the latter returns all possible POS tags, which may work better for words that can have multiple tags. Table 24 shows the results with lemminflect used for lemmatization and POS tagger. We can see that all embedding-based models have improved significantly. This shows that

[6] https://github.com/bjascob/LemmInflect.

correct POS tagging is crucial for nouns, otherwise you may drop good candidates in the final output.

4.2.2 Semantic role expansion task

The evaluation results for the semantic role expansion task are presented in Table 5. For role expansion experiments, the non-contextualized models and Melamud et al. (2015) models are outperformed by BERT and XLNet with a significant margin with $p@1 = 0.471$ and $MAP@50 = 0.118$ for BERT and $p@1 = 0.513$ and $MAP@50 = 0.144$ for XLNet. The DTs performed substantially better than word embedding models and also comparably to the models of Melamud et al. (2015). A better score is achieved by the DT trained on Wikipedia. But the performance of static word embeddings has dropped, especially the performance of fastText is worst compared to all models, in contrast to the previous experiment where it was found to be the best model. In contrast to previous experiments, the performance of the Melamud et al. (2015) models is also dropped significantly in comparison to BERT and XLNet.

XLNet has performed better than BERT in all settings with $p@1 = 0.513$ and $MAP@50 = 0.144$ for the simple model without masking and with $p@1 = 0.522$ and $MAP@50 = 0.159$ with +embs method and default hyperparameters. Whereas selecting optimal hyperparameters further improved its performance with $p@1 = 0.542$ and $MAP@50 = 0.161$ making it the best overall model. The dynamic patterns did not help to improve the performance of the BERT model for this particular task, most probably because these patterns are not suitable for the semantic role extraction task. Although without +embs method, BERT and XLNet were outperformed by several non-contextualized models in the task of LU expansion, in this experiment, they obtained superior performance compared to all these models. This fact reflects the importance of the context for making reasonable substitutions of words that bear semantic roles. Another reason lies in the fact that their fixed size vocabulary covers more frequent words like verbs than nouns for role arguments.

Combining substitutes predicted by multiple models helps to substantially improve the scores for those which performed worst as single, but shows mixed effect for combinations where one model was significantly better than others. For semantic role expansion task, models we considered are (1) *GloVe* as nc-emb, (2) *DT wiki* as DT, (3) *balAdd* for Melamud et al. (2015), (4) *XLNet* and (5) XLNet+embs with optimal hyperparameters. The highest MAP score was achieved by combining XLNet+embs with balAdd (Melamud et al., 2015) and DT, but highest precision was achieved by combination of balAdd with XLNet ($p@1 = 0.574$) and XLNet+embs ($p@1 = 0.563$). Both of these combinations got best precision scores for smaller values of k in comparison to the single best model of XLNet+embs with optimal hyperparameters, which scored highest MAP score ($MAP = 0.161$). Overall, with $p@1$ approaching 55% and $p@5$ approaching 47% and given that our gold standard is necessarily incomplete, this paves the way to fully-automatic expansion for semantic role resources.

Table 7 contains three example sentences with highlighted arguments for semantic roles and top 10 substitutes generated by all models (along with the ground truth FrameNet annotations). The first example demonstrates several valid matching substitutes, because *vehicle* is the most common sense of "car". Whereas, the other two

examples present an argument "bank" with multiple associated semantic roles. Again, BERT and XLNet were able to distinguish both senses of "bank" and produced several valid substitutes.

For roles, we also produce results with stopwords removal to see how it affects the performance. The results are reported in Table 25. In comparison of these scores to Table 5, we can see that for non-contextualized models and the models of Melamud et al. (2015), there is no meaningful difference in scores, which suggests that these models actually rarely produce such words in their output. For Transformer-based models, results have improved substantially. Since these models predict a word given on its context, there is a high likelihood that based on the position of words and their context, some bad candidate words are produced. Since these models have further improved, this has a slightly negative effect on the combinations, and the difference in their scores from individual models is increased. Overall, XLNet+embs model yields the highest scores ($p@1 = 0.581$ and $MAP@50 = 0.176$).

4.2.3 Effect of gold set size

The results reported in Table 5 are generated using whole datasets, without doing any filtering on the size of the gold sets. We have reported MAP at $k = 50$. but there are many instances in these datasets where the size of the gold set is really small. The average size is 22 for verbs, 27 for nouns, and 73 for roles. Whereas the minimum size is 1 for all three. For smaller sets, it is really hard to predict the candidates, especially if the gold members consist of rare words. In a number of situations, even if these members are produced by the model, they may not be ranked higher in the list of potential substitutes. Figure 3 shows the performance of the XLNet+embs model for all three datasets, where they were filtered against the minimum number of values in gold sets. We use a minimum size of 5, 10, and 15. It shows that precision at all values of k increases if we filter smaller sets, but not by a large margin. This suggests that the ranking of candidates needs to be further investigated.

4.3 Examples of induced lexical semantic frame representations

This section contains a qualitative analysis of lexical expansion examples of few semantic frames for all lexical substitution models, along with the ground truth from FrameNet. Each example sentence represents a specific frame and a single target word labeled either as a lexical unit or a semantic role. For each model, top 10 final substitutes are given. Examples of semantic roles expansion are presented in Table 7. Examples of lexical units expansions are presented in Table 6. Each table contains examples of ambiguous and unambiguous words to compare the substitutes in each use case.

In summary, it is evident that for non-ambiguous words, most models produce several valid substitutes, but for ambiguous polysemous words most of the non-contextualized models either were unable to produce any valid substitutes or they produced a few good substitutes for one sense only. In contrast, contextualized models produce valid substitutes in most situations. A deeper analysis of these examples

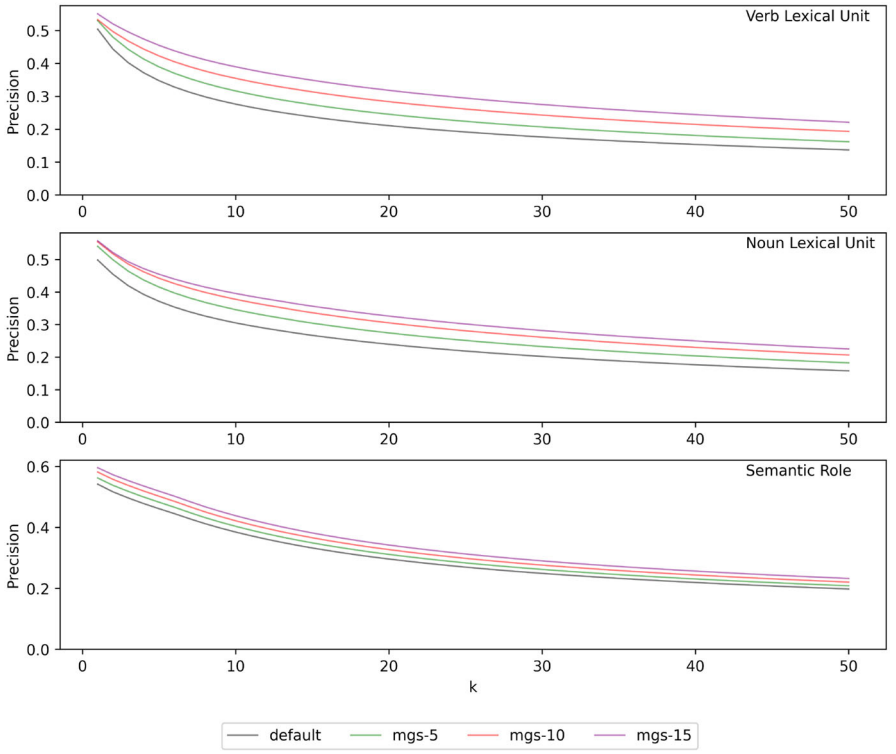

Fig. 3 Precision@k curve for XLNet+embs (optimal) model for all three datasets of verbs, nouns, and roles. Here, *mgs* means minimum gold set size

provides some key insights into the intrinsic evaluation framework. Like, it can be noted that some substitutes may seem to be semantically valid but may not be present in the FrameNet lexicon and hence not marked as true. Similarly, a substitute can actually be a wrong fit, although it is present in the FrameNet lexicon, because it may change the meaning of the sentence or make it grammatically incorrect. We will dive deeper into this issue in the following section.

4.4 Manual evaluation of lexical substitutes

4.4.1 Problems with automatic evaluation of lexical substitutes

As discussed in Sect. 4.3, automatic evaluation of lexical substitutes using the current gold datasets faces two problems related to FrameNet coverage and semantics of substitute and its context.

Scenario A—substitute fits the context, but is not present in the gold dataset: A substitute may be a good candidate to replace the target word within the context, but is considered as wrong because of not being present in the gold dataset. FrameNet pro-

Table 7 Examples for the semantic role expansion task

Frame: Vehicle
Seed sentence: The paper 's local administrator , Maria Luz Lopez , was shot dead , and her mother wounded , while her **car$_{\text{Vehicle}}$** was stopped for a red light.
GloVe: vehicle, automobile, truck, auto, drive, dealership, rental, motorcycle, motorbike, suv **fastText:** vehicle, automobile, car–and, car.but, car.it, car.so, car.now, car., car.when, car.the **word2vec:** vehicle, suv, minivan, truck, ford_focu, honda_civic, jeep, pickup_truck, toyota_camry, scooter **DT wiki:** vehicle, automobile, truck, sedan, bus, chassi, carriage, limousine, wagon, jeep **DT 59g:** vehicle, truck, automobile, sedan, jeep, suv, minivan, bus, limousine, wagon **Melamud add:** vehicle, bike, minivan, land-rover, limo, bicycle, lorry, motor-car, scooter, horsebox **Melamud balAdd:** vehicle, bike, minivan, horsebox, land-rover, automobile, bicycle, scooter, limo, 2cv **Melamud mult:** vehicle, bike, limo, lorry, bicycle, land-rover, motor-car, minivan, pushbike, scooter **Melamud balMult:** vehicle, bike, minivan, land-rover, bicycle, limo, horsebox, scooter, motor-car, automobile **BERT:** vehicle, cruiser, sedan, family, house, traffic, automobile, truck, mother, head **XLNet:** vehicle, van, bus, suv, taxi, truck, convoy, family, driver, minivan **XLNet+embs:** vehicle, van, truck, bus, automobile, suv, taxi, minivan, sedan, driver **FrameNet gold:** helicopter, airplane, ship, vessel, subway, boat, vehicle, stryker, tank, truck, aircraft, bike, bus, car, train, plane, cab, carriage, automobile, buse, ferry, tram, sedan, taxi, tricycle, submarine, yacht, aeroplane, chopper
Frame: Part_orientational
Seed sentence: Repton was an Anglo-Saxon town, on the south **bank$_{\text{Part}}$** of the River Trent, and was at one time a chief city of the Kingdom of Mercia.
GloVe: draft, financial, credit, lender, loan, lend, cash, mortgage, banker, finance **fastText:** bank.the, bank.it, bank.thi, bank.so, bank., bank.he, draft, bank.but, bank.in, bank.a **word2vec:** draft, lender, banker, depositor, mortgage_lender, bofa, citibank, branch, bankā, kaupthing_iceland **DT wiki:** shore, company, draft, lender, embankment, shoreline, floodplain, slope, coast, riverbank **DT 59g:** lender, company, insurer, draft, brokerage, firm, issuer, institution, thrift, steelmaker **Melamud add:** sea-coast, seacoast, dnieper, rhin, scheldt, acclivity, vistula, shore, copperbelt, isthmu **Melamud balAdd:** riksbank, bundesbank, aib, seacoast, southbank, westbank, ebrd, pariba, sea-coast, declivity **Melamud mult:** sea-coast, rhin, dnieper, seacoast, scheldt, isthmu, shore, acclivity, coast, bight **Melamud balMult:** seacoast, sea-coast, riksbank, southbank, acclivity, declivity, bundesbank, dnieper, scheldt, vistula **BERT:** side, shore, river, west, east, hand, branch, corner, fork, bend **XLNet:** side, shore, branch, reach, coast, end, edge, course, part, bend **XLNet+embs:** side, shore, draft, branch, coast, end, river, reach, banker, edge **FrameNet gold:** bottom, rear, north, north-south, northwest, west, side, territory, western, end, south, aquifer, back, left, window, top, heart, face, dynasty, tip, front, coast, southern, northernmost, northern, part, eastern, aegean, base, peak, area, portion, island, edge, sliver, strip, region, east, bank, fork, aisle, wall, shore, feet, leg, paw, quarter, wing, femora, half, halve, reach, slope, sea-board, borderland, ring, step, drawer, lip, realm, claw, border, ridge, foot, summit, door, gate, apse, façade, hemisphere, boundary, section, entrance, province, point, apex, corner, axle, page, pocket, seat, stair, underbelly, crest, layer, floor, button, shelf, flank, frontier, peninsula, hill, underside, coastline, spoiler, tailcone, panel, wheel
Frame: Abounding_with
Seed sentence: For their sledging trick, they love a steep, snow covered **bank$_{\text{Location}}$** and will lie on the top, facing downhill, then tuck up their front paws so that they slide along upon their chests.
GloVe: draft, financial, credit, lender, loan, lend, cash, mortgage, banker, finance **fastText:** bank.the, bank.it, bank.thi, bank.so, bank., bank.he, draft, bank.but, bank.in, bank.a **word2vec:** draft, lender, banker, depositor, mortgage_lender, bofa, citibank, branch, bankā, kaupthing_iceland **DT wiki:** shore, company, draft, lender, embankment, shoreline, floodplain, slope, coast, riverbank **DT 59g:** lender, company, insurer, draft, brokerage, firm, issuer, institution, thrift, steelmaker **Melamud add:** tepee, sandbar, sundeck, phonebox, sandpit, haystack, walkway, flowerb, henhouse, declivity **Melamud balAdd:** cahoot, flexaccount, hsbc, citibank, natwest, draft, tsb, banker, declivity, barclay **Melamud mult:** tepee, sandpit, sandbar, sundeck, phonebox, walkway, flowerb, lampshade, haystack, henhouse **Melamud balMult:** sandbar, flexaccount, declivity, cahoot, banker, natwest, draft, haystack, acclivity, hsbc **BERT:** slope, hill, ditch, mountain, river, ridge, hillside, stream, steep, tier **XLNet:** slope, hill, surface, trail, area, spot, hillside, path, embankment, grind **XLNet+embs:** slope, hill, surface, embankment, area, trail, spot, wall, hillside, mountain **FrameNet gold:** ringer, it, kitchen, hill, equipment, island, street, nut, place, which, plimsoll, paper, bread, roll, egg, scone, tin, salmon, dish, potatoe, kavo, hillside, fiord, sea, pottery, cuff-link, porcelain, bowl, room, somethe, that, pocket, hand, gorget, finger, office, bookshelve, stall, animal, bird, mushroom, olive, folder, fish, pepper, pension, panel, door, donut, stoneware, tile, window, eye, veal, walnut, i, jeep, collection, frame, mirror, everythe, bedroom, barge, easel, desk, arbour, bank, bar, cinema, appearance, raspberry, ful, glass, mug, tankard, river, goblet, pew, skin, ceil, bookcase, figure, face, plaster, wall, wood, buse, fishing-boat, sign, poplar, curtain, promenade, avenue, pasture, land, another, weapon, bottle, ditch, everywhere, meadow, pasta, depression, church, sandbag, sofa, bubble, car, countryside, closet, hallway, pond, train, road, home, accommodation, dwelling, fireplace, floor, roof, corridor, uniform, bed, oak, bath, dump, nylon, chalet, balcony, machinery, reef, overhead, belt, path, roadway, area, courtyard, terrace, entrance, character, liverpool, toenail, shaft, object, neck, fingerboard, they, unit, table, pot, fingernail, moccasin, tray, goldie, peach, inn, ingushetia, sidewalk, mast, nail, floorboard, rail, plywood, launch, cabin-top, toy, she, anglo-saxon

Green color indicates matches with the gold annotations. Here, XLNet+embs is the best model with optimal hyperparameters

Table 8 Examples of substitutes for **scenario A**, the substitutes that are marked as right by the annotator, but not present in the gold dataset are highlighted

Scenario A: substitute fits the context, but is not present in the gold dataset	
Frame: Coming_to_believe, Target word: **Lexical Unit**	
Seed sentence:	I felt angry and humiliated at the time of the argument but I can **guess** at how he must have felt .
Substitutes	suppose , imagine , think , speculate, assume , conjecture , bet, understand , presume , wonder
Frame: Type, Target Word: **Lexical Unit**	
Seed sentence:	The allocation of the principal services provided by these different **types** of local authority is illustrated on page four .
Substitutes	kinds, sorts, forms, categories , varieties, levels, styles , models , modes , classes
Frame: Affirm_or_deny, Target Word: **Semantic Role**	
Seed sentence:	**Tajikstan** denies it is trying to export uranium to anyone .
Substitutes	Turkmenistan , Uzbekistan , Kazakhstan , Iran, Russia , Pakistan, Azerbaijan , Kyrgyzstan , Ukraine , Tajik
Frame: Getting, Target Word: **Semantic Role**	
Seed sentence:	At Goodwill **she** gained in self - confidence , in her vision of her future and in the job skills she needed to find and keep a good job .
Substitutes	i, we, he, her, they, sarah , lisa , you, mary , susan

vides a gold dataset of all lexical units for each frame. Normally, we can expect that if these lexical units are verb predicates, then probably the set would be more complete than those of noun predicates. As in the case of noun predicates, they usually do not have a strictly closed set of suitable variants and thus it is highly likely that a given substitute may not be covered in the gold dataset (which also happens with verbs, albeit to a lesser extent since the number of verbs is generally lower than the number of nouns). Similarly, this is aggravated in the case of semantic roles, as theoretically the roles can have countless valid arguments and that works for semantic parsing but becomes an issue for augmentation tasks if a gold dataset shall be used for evaluation. As for our experiments, the gold dataset for roles is extracted from the available sentence annotation, it cannot be considered as a proper gold set for evaluation. See Table 8 for more explanation, for the given sentences and the list of substitutes, there are multiple correct answers that are not present in the gold dataset.

Scenario B—substitute is present in the gold dataset, but does not fit the context: A given substitute is present in the gold dataset, but may not fit the given context or change the context meaning altogether. See Table 9, for examples of such substitutes. The second example describes a situation where a body part is involved, but not all body parts can be folded as per the context. In the last example, the target word is a semantic role of type *Speaker*, which can be a pronoun or a person's name. But not all pronouns can fit this context.

Table 9 Examples of substitutes for **scenario B**, the substitutes that are present in the gold dataset but do not fit the context are highlighted

Scenario B: substitute is present in the gold dataset, but does not fit the context	
Frame: Arriving, Target word: **Lexical Unit**	
Seed sentence:	Daniel **approched** her from behind and unwound her arms and substituted his own .
Substitutes	contacted, reached, entered, attacked, confronted, pursued, touched, grabbed, embraced, assaulted
Frame: Body_part, Target word: **Lexical Unit**	
Seed sentence:	His thin companion folded his **limbs** like an insect as he sat down .
Substitutes	legs, arms, hands, bodies, wings, feet, fingers, muscles, shoulders, thighs
Frame: Body_parts, Target word: **Semantic Role**	
Seed sentence:	‘ What the hell has Marissa got to do with any of this ? ” he demanded brusquely , frowning down at the flushed **cheeks** of the girl sitting on the sofa .
Substitutes	faces, lips, skins, eyes, foreheads, features, hairs, breasts, noses, bodies
Frame: Communication_manner, Target word: **Semantic Role**	
Seed sentence:	At Goodwill **she** gained in self - confidence , in her vision of her future and in the job skills she needed to find and keep a good job .
Substitutes	i, we, he, her, they, sarah, lisa, you, mary, susan

Table 10 Examples with their target words highlighted and the list of top 10 final substitutes

Frame: Body_parts, Target word: **Lexical Unit**	
Description:	This frame covers words for Body_part(s) (BP) belonging to a Possessor (Poss), which may be characterized by a Descriptor (Desc). The location of the BP may be identified in terms of its Attachment or its Orientational_location. A Subregion of a BP may also be indicated.
Seed sentence:	The arms are covered in **skin** .
Substitutes	flesh, hair, blood, fur, leather, bone, body, muscle, feather, cloth
does NOT fit context	leather, bone, body, muscle, feather
fit context NOT frame	blood, fur, cloth
fit context AND frame	flesh, hair
match FrameNet gold	flesh, hair, body
Frame: Getting, Target word: **Semantic Role**	
Description:	A Recipient starts off without the Theme in their possession, and then comes to possess it. Although the Source from which the Theme came is logically necessary, the Recipient and its changing relationship to the Theme is profiled.
Semantic Role Description	**Recipient**: The Recipient indicates the entity that ends up in possession of the Theme.
Seed sentence:	At Goodwill **she** gained in self - confidence , in her vision of her future and in the job skills she needed to find and keep a good job .
Substitutes	i, we, he, her, they, sarah, lisa, you, mary, susan
does NOT fit context	i, we, he, her, they, you
fit context NOT frame	-
fit context AND frame	sarah, lisa, mary, susan
match FrameNet gold	i, we, he, they, you

Against each use-case, given is the list of substitutes marked as true by the annotator. For each seed sentence, its target word is color-highlighted

Table 11 Statistics of three datasets for manual evaluation sampled randomly from datasets used in automatic evaluation

Characteristics	Verb lexical unit	Noun lexical unit	Semantic roles
Number of frames	41	43	44
Number of lexical units/roles	50	50	47
Number of annotations	50	50	50
Number of sentences	50	50	50

4.4.2 Evaluation framework

To manually analyse the appropriateness of a given substitute in the scenarios discussed in Sect. 4.4.1, we define the following rules to evaluate:

- It does *not fit the context*. The objective is to maintain the sentence's meaning. This use-case will also drop the substitute, which can make the sentence grammatically incorrect.
- It does *fit the context, not the frame*. The sentence is still meaningful but does not preserve the frame meaning. We use the formal descriptions of frames to decide if a sentence represents the frame. Additionally, for semantic roles, we also consider the semantic role definition. Because frame description alone is not sufficient to evaluate semantic roles.
- It does *fit both the context and the frame*. The ideal scenario to replace the target word would be these substitutes, as the main motivation of this work is to preserve the original frame.

Table 10 contains examples for each use case. It also includes the list of substitutes matched with the gold dataset.

4.4.3 Datasets and substitution model

We randomly sampled 50 annotations for each type of target word (verb, noun, and semantic role). Table 11 shows statistics of these datasets. Each annotated instance is evaluated for top 10 substitutes. In summary, the annotator has to evaluate 500 substitutes for each dataset. For the substitution model, we choose the best-performing single model i.e. *XLNet+embs*.

4.4.4 Results

Table 12 shows the results for all datasets against each evaluation use-case. For all datasets, there is a significant improvement in the use-case where the substitute *fits both the context and the frame*, and precision has improved consistently for all values of k. Precision values for the use-case of *does NOT fit context* support our first problem for automatic evaluation, that even though the substitute can be present in the gold dataset, it does not fit the context and hence should be ignored. For example, in

Table 12 Manual evaluation of lexical substitutes for sampled datasets of 500 annotations (50 contexts, 10 substitutions)

Evaluation	Verb Lexical Unit Expansion				Noun Lexical Unit Expansion				Semantic Role Expansion			
	p@1	p@3	p@5	p@10	p@1	p@3	p@5	p@10	p@1	p@3	p@5	p@10
does NOT fit context	0.180	0.200	0.244	0.316	0.140	0.140	0.176	0.222	0.240	0.240	0.248	0.292
fit context NOT frame	0.280	0.253	0.252	0.292	0.180	0.247	0.276	0.324	0.080	0.113	0.132	0.142
fit context AND frame	**0.520**	**0.533**	**0.496**	**0.372**	**0.660**	**0.607**	**0.544**	**0.450**	**0.660**	**0.640**	**0.620**	**0.568**
match FrameNet gold	0.360	0.360	0.328	0.230	0.500	0.453	0.372	0.286	0.480	0.427	0.420	0.340

The substitutes were produced using *XLNet+embs* model with optimal hyperparameters

Table 10, the substitute *body* does not make sense to replace the original word *skin*, but it is present in the gold dataset. For those substitutes that fit the context, there can still be some scenarios where they do not preserve the frame. For example, see Table 10, the substitutes *blood*, *fur*, *cloth* do fit the context very well, but since they do not maintain the frame for not being a body part, they cannot be accepted as correct. Not surprisingly, the numerical scores are higher than in the automatic evaluation as the manual judgements are not prone to incompleteness of lexical-semantic resources.

5 Extrinsic evaluation: frame-semantic parsing with lexically expanded FrameNet

To evaluate the quality of automatically constructed frame structures, we conducted extensive experiments using two frame-semantic parsers. Our goal was to determine whether these induced frame structures could improve parsing performance in situations where annotated data is scarce. We select a small sample from the FrameNet dataset with original annotations as a seed dataset. Then we augment it by incorporating new sentences constructed using our lexical substitution approach, while keeping the annotations same, which results in a larger training dataset. We compare the performance of the parsers trained on the augmented dataset and on the seed dataset. We do not change the test and development (dev) sets.

5.1 Experimental setup

In this section, we describe the choice of models for lexical substitution, details of the semantic parsers used in our experiments, and the procedure for the construction of training datasets, including the pre and post-processing steps.

5.1.1 Lexical substitution models

For extrinsic evaluation, we select two substitution models with the best performance in the intrinsic evaluation (Table 5). The first model is *XLNet+embs* with optimal hyperparameters, which demonstrates the best results for both tasks of the lexical unit expansion and the semantic role expansion. The second model is *BERT* without dynamic patterns and the +embs method extension. We choose to use the standard *BERT* model without any extensions in order to determine whether performance dif-

ferences observed in intrinsic evaluation of these two models would also be reflected in a presumably less sensitive extrinsic evaluation.

5.1.2 Frame-semantic parsers

We conduct experiments with: (1) open-SESAME (SEmi-markov Softmax-margin ArguMEnt)—a neural network-based frame-semantic parser by Swayamdipta et al. (2017), and (2) a BERT-based parser for relation extraction and semantic role labeling inspired by Shi and Lin (2019).

Open-SESAME parser (Swayamdipta et al., 2017): The Open-SESAME parser decomposes the task of frame-semantic parsing into three sub-tasks and implements an independently trained model for each sub-task: (1) ArgId model: to identify and label semantic arguments, (2) FrameId: to identify frames using gold targets, and (3) TargetId: to identify target predicates using lexical units of FrameNet (this model is not discussed in the original publication). The objective of the argument identification model is to identify argument spans and their labels. It uses a softmax-margin segmental recurrent neural network as a baseline syntax-free model and adds several modifications to further improve the performance. In particular, it adds some sort of syntax information and syntactic scaffolding. For our experiments, we only used the baseline syntax-free model. The model accepts as input a sentence in form of a token sequence, token part-of-speech tags, a target span, and an associated lexical unit with its frame and outputs a list of possible labeled segments with their start and end positions in the input sentence. The labels are either semantic roles or "null". The ArgId model only handles non-overlapping segments, and segmentation is only produced for the input frame and its target. The maximum length of a span to be considered can be specified as a hyperparameter. The frame identification model is a syntax-free bidirectional LSTM that takes the same input as the argument identification model except the frame and identifies the frame evoked by the target. It can not predict frames for targets that are not present in the FrameNet lexicon. The target identification model is also based on bidirectional LSTM. It takes as input a sequence of tokens from a given sentence, their part-of-speech tags, and lemmas and for each token, it outputs a binary label indicating whether it is a target or not. The list of possible targets is available through the FrameNet lexicon of lexical units. In our experiments, we use the official, publicly available implementation of the Open-SESAME parser.[7]

BERT parser (Shi & Lin, 2019): The BERT-based parser semantic parser is originally designed for PropBank-style arguments and unlike open-SESAME, it does not perform the target and sense (frame) identification as separate independent tasks. For argument identification and labeling, it can perform sense disambiguation for targets before argument identification (end-to-end). Therefore, it can work with only sentence and the target predicate while keeping the target frame as optional input. For the target sense disambiguation task, it takes a sentence as input and formulates the task as a sequence labeling problem, where each token is assigned a label. The target token is assigned the sense (frame) label and all remaining tokens are assigned either label

[7] https://github.com/swabhs/open-sesame.

'X' (non-target tokens) or 'O' (sub-tokens of any non-target token). This sequence of tokens is passed through the BERT encoder to obtain contextualized embeddings. The predicate tokens are distinguished by concatenating these contextual embedding to 'predicate indicator' embeddings before making a final prediction using a one-hidden-layer multi layer perceptron (MLP). For argument identification and labeling task, it takes as input a pair of sentence and its target predicate, arguments spans are predicted as BIO (Beginning, Inside, Outside) labels for all tokens. The target predicate is paired with the sentence and passed through the BERT encoder to make the sentence embeddings target-aware. These contextualized sentence embeddings are concatenated with 'predicate indicator' embeddings and passed to one-layer BiLSTM to obtain hidden states of each token to make the final prediction. The hidden state of the predicate token is concatenated to the hidden state of each token and passed to the MLP to get the probability distribution over the label set. We use the implementation provided by the AllenNLP library[8] and conduct experiments both with and without gold frames.

Note that the open-SESAME parser does not leverage pre-training, but it uses syntax information (part-of-speech tags) for parsing. In contrast, a BERT-based parser (Shi & Lin, 2019) takes advantage of pre-training, while avoids using any syntax information. We consider it is interesting to investigate the effect of lexical expansion for such two conceptually different semantic parsers.

5.1.3 Seed datasets

We use scripts from the open-SESAME parser Swayamdipta et al. (2017) to split full-text annotations of FrameNet-1.7 into train, test, and dev splits. The test set is similar to previous studies (Das et al., 2014). It contains 16 documents, while 8 documents are used for the dev set. The statistics for all three splits are given in Table 13. For our experiments, we generate two sets of splits: (a) with verbs as lexical units; (b) with nouns as lexical units. To do comparative experiments after lexical expansion, all other train datasets were sampled from the train set of these two datasets while keeping their respective test and dev sets same. Seed training datasets were constructed by randomly sampling one frame annotation per sentence. This strategy provides the train dataset for verbs with 2746 annotations in total with 7 annotations per frame on average, and for nouns, it provides 9293 annotations in total with 8 annotations per frame on average.

5.1.4 Dataset expansions

Each annotation of the seed dataset was augmented using three types of words simultaneously. The first two types are based on FrameNet annotations of the sentence tokens:

- *lexical unit:* a single-token lexical-unit, that can be either verb or a noun
- *role:* all single-token roles.

[8] https://docs.allennlp.org/models/main/models/structured_prediction/models/srl_bert/.

Table 13 Statistics for data splits for FrameNet-1.7 fulltext annotations and the seed datasets

Data split	# of annotations	# of sentences	# of frames	# of lexical units
FN-1.7 Fulltext Swayamdipta et al. (2017)				
Train	19,391	3353	753	2996
Test	6714	1247	574	1678
Dev	2272	326	368	785
Verbs				
Train	5739	2746	428	923
Test	1904	922	314	485
Dev	686	292	190	271
AnnotationPerSentence-Verbs				
Train	2746	2746	362	684
Nouns				
Train	9293	2996	464	1536
Test	2981	1003	311	847
Dev	1063	293	198	377
AnnotationPerSentence-Nouns				
Train	2996	2996	354	926

The third word type is based on a POS tag of the sentence tokens:

– *noun:* any word that is a noun or a part of a noun phrase but is neither a lexical unit or a single-token role. The reason to select such nouns for expansion comes from the semantics of roles, as major portion of a sentence is usually covered with semantic roles, which can be mostly multi-token and this ends up with a very few words to be substituted as a single-token roles. This configuration will substitute all noun tokens except those already been substituted as roles. To determine whether a word is a noun, we used predicted part-of-speech tags generated during the pre-processing phase of the parser. We augmented only a fraction of sentence tokens as nouns. For this purpose, we experimented with values in the range of [10, 30, and 50]% of sentence tokens.

For all train datasets, each annotation of the seed dataset was augmented with two more annotations ($k = 2$) unless mentioned otherwise, to get an approximately three times larger augmented training dataset. See Tables 14 and 15 for statistics of the augmented train datasets under various configurations using BERT as a substitution model. We constructed datasets with expansion of only single word types like either **lexical unit** or **role** or **noun** and then with all combined. For all three word types, the order of expansion is always the lexical unit, followed by role and then noun. This ensures that one specific word is augmented only once even if it belongs to multiple word types.

Table 14 Statistics of the train datasets augmented from seed dataset AnnotationPerSentence-Verbs, for different configurations

Expansion Configurations	Annotations	LUs	Avg. per Frame	Avg. per LU
augmented- lexical unit	7100	1026	20	7
augmented- roles	5734	684	16	8
augmented- nouns-10pc	8162	684	23	12
augmented- nouns-30pc	8162	684	23	12
augmented- nouns-50pc	8162	684	23	12
augmented- lexical unit-roles-nouns-10pc	8238	1026	23	8
augmented- lexical unit-roles-nouns-30pc	8238	1026	23	8
augmented- lexical unit-roles-nouns-50pc	8238	1026	23	8

Overall 362 frames were involved. Base model is BERT

Table 15 Statistics of the train datasets augmented from the seed dataset AnnotationPerSentence-Nouns, for different configurations

Expansion Configurations	Annotations	LUs	Avg. per Frame	Avg. per LU
augmented- lexical unit	7910	1492	22	5
augmented- roles	5094	926	14	5
augmented- nouns-10pc	8848	926	25	10
augmented- nouns-30pc	8848	926	25	10
augmented- nouns-50pc	8848	926	25	10
augmented- lexical unit-roles-nouns-10pc	8967	1492	25	6
augmented- lexical unit-roles-nouns-30pc	8967	1492	25	6
augmented- lexical unit-roles-nouns-50pc	8967	1492	25	6

Overall 354 frames were involved. Base model is BERT

5.1.5 Post-processing

The list of substitutes produced by the lexical substitution model was post-processed before the final augmentation. Some of these post-processing steps are common to all word types such as removal of noisy words, duplicates, and seed words. While the specific ones for each word type are as follows:

- *lexical unit:* substitutes for lexical-unit were filtered as per their gold annotations (frame parser can not predict a frame for a target not present in the FrameNet lexicon). Final substitutes were lemmatized and then inflected to match the tense form of the substituted lexical unit. We use the lemminflect[9] library as an inflection engine.
- *role:* substitutes for roles were also filtered as per their gold annotations and also for a basic list of stop-words.
- *noun:* substitutes for nouns were filtered for a basic list of stop-words including digits and the minimum length of two characters. Final filtering was done based on part-of-speech tags to retain only nouns. The final list was lemmatized and inflected to match the singular or plural form of the substituted noun. For lemmatization and part-of-speech tagging, we used the NLTK library.

After substituting all target words, the augmented sentence was again parsed for part-of-speech tags. Tables 14 and 15 provides the total number of annotations for

[9] https://pypi.org/project/lemminflect/.

Table 16 Examples of expansions using the configuration of lexical unit-roles-nouns-50pc for XLNet+embs and BERT as lexical substitution models

BERT	XLNet+embs
The recent wildfires in Los Angeles , California , have burned $_{\text{Fire\_burning}}$ more than 42,000 acres (17,000 hectares) of land .	
The recent fires in Las La , ca , have blazed$_{\text{Fire_burning}}$ more than 42,000 areas (17,000 has) of property .	The recent fires in San County , ca , have blazed $_{\text{Fire\_burning}}$ more than 42,000 hectares (17,000 acres) of forest .
The recent flames in La Pictures , La , have blazed $_{\text{Fire\_burning}}$ more than 42,000 homes (17,000 kilometres) of farmland .	The recent flames in Las California , Calif , have blazed $_{\text{Fire\_burning}}$ more than 42,000 miles (17,000 kms) of property .
Nearly 1,000 homes were destroyed $_{\text{Destroying}}$, and more than 10,000 residents were evacuated .	
Nearly 1,000 houses were leveled $_{\text{Destroying}}$, and more than 10,000 homes were evacuated .	Nearly 1,000 houses were demolished $_{\text{Destroying}}$, and more than 10,000 people were evacuated .
Nearly 1,000 dwellings were devastated $_{\text{Destroying}}$, and more than 10,000 families were evacuated .	Nearly 1,000 buildings were ruined $_{\text{Destroying}}$, and more than 10,000 families were evacuated .
Officials now report $_{\text{Statement}}$ that firefighters are making some progress .	
Police have write $_{\text{Statement}}$ that people are making some work .	Authorities today announce $_{\text{Statement}}$ that rescuers are making some success .
They do confirm $_{\text{Statement}}$ that flames are making some move .	They finally claim $_{\text{Statement}}$ that fires are making some advance .

The shaded row indicates the seed sentence from the *AnnotationPerSentence-Verbs* dataset

different configurations for both seed datasets of verbs and nouns. As expected, the datasets augmented with just lexical units and roles are the smallest, because for lexical units, the list of final substitutes can be empty if no substitute matches with the gold set of the annotated frame, and for roles, single-token roles may not be present in a sentence. The datasets where all nouns were augmented are larger in comparison to all other configurations. A few examples of sentences taken from *AnnotationPerSentence-Verbs*, and augmented using one of these configurations are given in Table 16.

5.2 Examples of augmented sentences

Table 16 shows few examples of augmentation results along with original seed sentences. Here, the seed dataset was *AnnotationPerSentence-Verbs*. Each sentence is highlighted for all three word types, i.e. target words and phrases for their corresponding word types, which are lexical unit, roles and nouns. As mentioned previously, only single-token roles are augmented. We do augmentations for the seed sentence using two top substitutes from the final list after post-processing steps. These augmented examples were produced using the configuration of lexical unit-roles-nouns-50pc. In some cases, the quality of substitutes for roles and nouns is less reliable as per the overall semantics of the sentence, especially for roles as their gold dataset is limited to FrameNet annotations, and unlike lexical units elements of these gold datasets can be semantically very different from each other. For example, predicting pronouns to substitute nouns. But substitutes for lexical units and nouns are plausible in most cases and preserve the meaning of the sentence.

Table 17 The performance of Swayamdipta et al. (2017) the frame-semantic parser for the **target identification** model in terms of the F_1 score: **TargetId – Verbs**

Train dataset	**mean**	**std**	**p-value**	**mean**	**std**	**p-value**
Seed dataset: AnnotationPerSentence-Verbs	60.67	2.87	–	60.67	2.87	–
Expansions at k=2		**BERT**			**XLNet+embs**	
augmented- lexical unit	59.16	2.90	p>0.01	57.63	3.37	p>0.01
augmented- roles	58.99	2.58	p>0.01	61.05	3.62	p>0.01
augmented- nouns-10pc	59.83	1.71	p>0.01	60.23	2.84	p>0.01
augmented- nouns-30pc	59.74	2.29	p>0.01	59.77	2.01	p>0.01
augmented- nouns-50pc	60.26	2.57	p>0.01	**61.67**	3.71	p>0.01
augmented- lexical unit-roles-nouns-10pc	**60.47**	2.62	p>0.01	61.09	2.30	p>0.01
augmented- lexical unit-roles-nouns-30pc	59.44	2.91	p>0.01	58.97	4.05	p>0.01
augmented- lexical unit-roles-nouns-50pc	57.31	1.83	p>0.01	61.23	2.74	p>0.01

5.3 Results with the ppen-SESAME parser

Hyperparameters: Optimal hyperparameters for the argument identification model are presented in Swayamdipta et al. (2017). However, the hyperparameters for the frame and target identification model are omitted. In our experiments, we used the default values for everything defined in the source code of the parser, except for the maximum number of epochs. For target and frame identification, we use 100 epochs with an early stopping patience of 25 epochs. For argument identification, we use 10 epochs with an early stopping patience of 3 epochs. We use these default values to get the total number of training steps for seed datasets. The augmented datasets are three-time larger than the seed datasets; we used the same number of training steps for them as per the corresponding seed dataset and model to keep the training time similar for all of them. For the seed datasets of *AnnotationPerSentence-Verbs*, this would give 274,600 steps for the target and frame identification models and 27,460 steps for the argument identification models. For the seed datasets of *AnnotationPerSentence-Nouns*, this would give 299,600 steps for the target and frame identification models and 29,960 steps for the argument identification models. This will reduce the bias in model performance because of the larger size and more training iterations. The final model was selected as per the best F_1 score on the dev dataset during training. To compensate for variance in model performance due to random weight initialization, all experiments were run 10 times and their mean and standard deviation on F_1 score is reported for both BERT and XLNet+embs. In addition to these measures, we also calculate p values for the paired student t-test to determine the statistical significance of performance differences between models trained on augmented datasets and models trained on the seed datasets. The null hypothesis assumes that both models performed similarly and any difference in their mean performance is not supported statistically. P values are compared against 99% confidence; a p value below 0.01 supports the alternative hypothesis and indicates that both models performed differently and this difference is statistically significant.

Tables 17 and 18 summarize results for the target identification model and Tables 19 and 20 summarize the performance of frame identification models for training

Table 18 The performance of Swayamdipta et al. (2017) the frame-semantic parser for the **target identification** model in terms of the F_1 score: **TargetId – Nouns**

Train dataset	mean	std	p-value	mean	std	p-value
Seed dataset: AnnotationPerSentence-Nouns	40.67	3.44	–	40.67	3.44	–
Expansion at k =2	**BERT**			**XLNet+embs**		
augmented- lexical unit	40.51	3.50	$p > 0.01$	39.42	2.98	$p > 0.01$
augmented- roles	39.07	2.17	$p > 0.01$	39.54	1.99	$p > 0.01$
augmented- nouns-10pc	40.93	2.99	$p > 0.01$	39.47	3.53	$p > 0.01$
augmented- nouns-30pc	**41.96**	2.53	$p > 0.01$	39.56	1.54	$p > 0.01$
augmented- nouns-50pc	41.04	1.01	$p > 0.01$	40.70	2.71	$p > 0.01$
augmented- lexical unit-roles-nouns-10pc	41.95	2.06	$p > 0.01$	**41.26**	1.58	$p > 0.01$
augmented- lexical unit-roles-nouns-30pc	40.26	1.27	$p > 0.01$	40.35	1.66	$p > 0.01$
augmented- lexical unit-roles-nouns-50pc	40.72	2.32	$p > 0.01$	39.36	1.77	$p > 0.01$

Table 19 The performance of Swayamdipta et al. (2017) the frame-semantic parser for the **frame identification** model in terms of the F_1 score: **FrameId – Verbs**

Train dataset	mean	std	p-value	mean	std	p-value
Seed dataset: AnnotationPerSentence-Verbs	**75.19**	0.87	–	**75.19**	0.87	–
Expansions at k=2	**BERT**			**XLNet+embs**		
augmented- lexical unit	69.23	0.69	$p>0.01$	70.11	1.06	$p>0.01$
augmented- roles	73.69	0.83	$p>0.01$	**74.09**	1.06	$p>0.01$
augmented- nouns-10pc	73.65	1.28	$p>0.01$	73.75	1.60	$p>0.01$
augmented- nouns-30pc	**74.55**	1.18	$p>0.01$	73.99	1.38	$p>0.01$
augmented- nouns-50pc	74.03	1.12	$p>0.01$	74.04	1.62	$p>0.01$
augmented- lexical unit-roles-nouns-10pc	68.56	0.87	$p>0.01$	69.63	1.18	$p>0.01$
augmented- lexical unit-roles-nouns-30pc	69.40	1.02	$p>0.01$	69.15	1.09	$p>0.01$
augmented- lexical unit-roles-nouns-50pc	67.94	1.42	$p>0.01$	68.82	0.71	$p>0.01$

Table 20 The performance of the frame-semantic parser by Swayamdipta et al. (2017) for the **frame identification** model in terms of the F_1 score: **FrameId – Nouns**

Train dataset	mean	std	p-value	mean	std	p-value
Seed dataset: AnnotationPerSentence-Nouns	88.56	0.78	–	88.56	0.78	–
Expansion at k =2	**BERT**			**XLNet+embs**		
augmented- lexical unit	**89.15**	0.69	$p > 0.01$	88.89	1.32	$p > 0.01$
augmented- roles	87.93	0.59	$p > 0.01$	87.95	0.35	$p > 0.01$
augmented- nouns-10pc	88.39	0.53	$p > 0.01$	87.93	0.63	$p > 0.01$
augmented- nouns-30pc	87.82	0.59	$p > 0.01$	87.98	0.68	$p > 0.01$
augmented- nouns-50pc	88.16	0.59	$p > 0.01$	88.50	0.35	$p > 0.01$
augmented- lexical unit-roles-nouns-10pc	88.74	0.58	$p > 0.01$	89.02	0.44	$p > 0.01$
augmented- lexical unit-roles-nouns-30pc	88.47	0.74	$p > 0.01$	**89.39**	0.53	$p > 0.01$
augmented- lexical unit-roles-nouns-50pc	88.84	0.96	$p > 0.01$	89.38	0.71	$p > 0.01$

datasets reported in Tables 14 and 15 respectively. For *AnnotationPerSentence-Nouns* dataset, the TargetId model managed to improve in multiple settings for BERT, getting the highest gain where 30% of nouns were expanded ($F_1 = 41.96$). For *AnnotationPerSentence-Verbs* dataset, it also scored better in multiple settings, getting the highest gain where 50% of nouns were expanded ($F_1 = 61.67$). However, this

Table 21 The performance of the frame-semantic parser by Swayamdipta et al. (2017) for **argument identification** and labeling in terms of the F_1 score: **ArgId – Verbs**

Train dataset	**mean**	**std**	**p-value**	**mean**	**std**	**p-value**
Seed dataset: AnnotationPerSentence-Verbs	46.82	1.04	–	46.82	1.04	–
Expansions at k=2	**BERT**			**XLNet+embs**		
augmented- lexical unit	47.07	0.93	p>0.01	47.43	0.77	p>0.01
augmented- roles	46.42	0.98	p>0.01	46.56	0.96	p>0.01
augmented- nouns-10pc	48.43	1.00	p<0.01	48.17	1.48	p>0.01
augmented- nouns-30pc	49.24	0.86	p<0.01	**49.47**	0.72	p<0.01
augmented- nouns-50pc	49.08	1.41	p<0.01	48.86	0.94	p<0.01
augmented- lexical unit-roles-nouns-10pc	48.77	1.04	p<0.01	48.52	1.05	p<0.01
augmented- lexical unit-roles-nouns-30pc	**50.00**	1.28	p<0.01	49.28	1.33	p<0.01
augmented- lexical unit-roles-nouns-50pc	49.43	0.80	p<0.01	49.25	0.97	p<0.01

difference in the mean scores of the models is not statistically significant ($p > 0.01$). We assume that the base dataset already contains sufficient examples per target on average and further expansions do not help it, but rather decreased its performance in some cases. The high standard deviation also shows that the original hyperparameters such as learning rate and dropout rate are less optimal for these datasets and need to be tuned before drawing final conclusions. For the FrameId model, the performance did not improve for all datasets augmented from *AnnotationPerSentence-Verbs*. In the case of datasets augmented from *AnnotationPerSentence-Nouns*, it is better in multiple cases for both BERT and XLNet+embs, but not statistically significant. The datasets, where the lexical unit is not augmented, managed to perform better than those where it was augmented. In the latter case F_1 decreased. That is most probably because augmented datasets only added new targets but with the same frame, because no new frame is added to the train, this affects negatively as new targets get just one example of the frame for them. This drop in performance is statistically significant. Contrary to target identification, the standard deviation also remained on the lower side (less than 1.5) for all models, which also hints that hyperparameters are good enough to yield robustness in results.

The results for the ArgId model are reported in Tables 21 and 22. The F_1 of the model on the augmented datasets is improved for many of the configurations. For the verbs dataset, the highest F_1 score is achieved for the dataset where expansion configurations are lexical unit-roles-nouns-30pc for BERT ($F_1 = 50.00$) and nouns-30pc for XLNet+embs ($F_1 = 49.47$). For the nouns dataset, the highest F_1 score is achieved for the dataset where expansion configurations are lexical unit-roles-nouns-50pc for BERT ($F_1 = 65.10$) and lexical unit-roles-nouns-50pc for XLNet+embs ($F_1 = 65.31$). The difference in the performance of models for the augmented datasets is also supported statistically with p values lower than 0.01, particularly in the datasets where all three types of words were augmented. Overall expansion configuration comprising nouns performed better as they got more diversified sentences for training than other configurations.

The negative results for target and frame identification indicate that using data augmentation to generate more training data is not always useful and it depends on the nature of the data and task itself. Since, we sample data per sentence, that is

Table 22 The performance of the frame-semantic parser by Swayamdipta et al. (2017) for **argument identification** and labeling in terms of the F_1 score: **ArgId – Nouns**

Train datset	mean	std	p-value	mean	std	p-value
Seed dataset: AnnotationPerSentence-Nouns	62.48	0.6	–	62.48	0.6	–
Expansion at k =2	**BERT**			**XLNet+embs**		
augmented- lexical unit	63.26	0.95	$p > 0.01$	63.49	0.86	$p < 0.01$
augmented- roles	61.84	1.06	$p > 0.01$	61.37	0.77	$p < 0.01$
augmented- nouns-10pc	62.92	0.81	$p > 0.01$	62.68	0.84	$p > 0.01$
augmented- nouns-30pc	63.49	0.66	$p < 0.01$	63.20	0.78	$p > 0.01$
augmented- nouns-50pc	63.41	0.78	$p < 0.01$	63.50	0.89	$p < 0.01$
augmented- lexical unit-roles-nouns-10pc	64.55	0.56	$p < 0.01$	64.41	0.45	$p < 0.01$
augmented- lexical unit-roles-nouns-30pc	64.69	0.61	$p < 0.01$	64.87	0.71	$p < 0.01$
augmented- lexical unit-roles-nouns-50pc	**65.10**	0.87	$p < 0.01$	**65.31**	0.87	$p < 0.01$

more suitable for arguments identification, as each sentence occurred once in the seed dataset, it does not seems to be a useful strategy for frame and target identification as they already had enough average number of annotations per instance (see Tables 14, 15). But data would have been sampled as per frame and target then augmentations would have helped and that we actually observed in our initial set of experiments. That different sampling strategy for these tasks does benefit from data augmentation during frame parsing. Dementieva et al. (2020) also reported similar findings for the task of propaganda detection. Similar to our choice of different words, Dementieva et al. (2020) augmented nouns, adjectives, adverbs, and verbs using GloVe, fastText, and BERT as substitution models to generate more training sentences. Their experiments with many different settings showed a slight shift in the precision and recall score while the F_1 score did not improve except very slightly in two cases. Another work from Fenogenova (2021) used the fine-tuned mT5 (Xue et al., 2021) model for paraphrasing to generate augmented data for the tasks of sentiment analysis, textual entailment, and question-answering in the Russian language. They also reported similar findings with all three tasks where the performance of the model remained nearly similar with both the original and the augmented training datasets.

5.3.1 Effect of train dataset size over model performance

To further validate the performance of all models against any bias in the seed dataset construction and to see the effect of the seed dataset size on model performance, we trained the two best models on multiple seed datasets. All seed datasets were constructed by randomly sampling the N percentage of training examples from the verbs and nouns dataset. We selected the values of N as 10, 20, 30, 40, 50, and 100%. Each seed dataset was further augmented into two datasets using BERT and XLNet+embs as lexical substitution models. Two best expansion configurations are selected that are lexical unit-roles-nouns-50pc and nouns-50pc. Models trained on the seed datasets use the same number of epochs as discussed in Sect. 5.3. To train each model on augmented datasets, the number of training steps were determined as per the size of their corresponding seed dataset and the model. As per previous experiments,

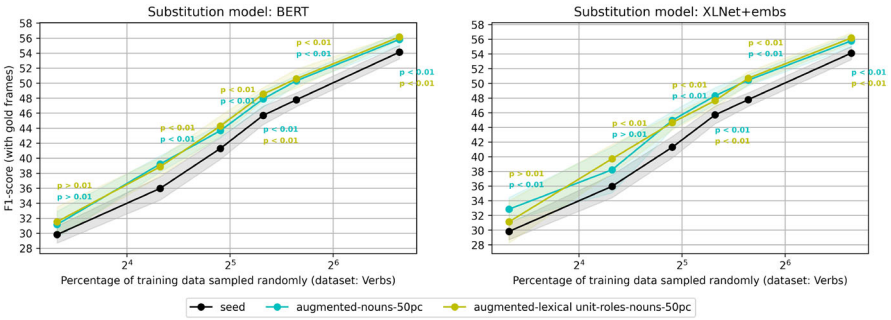

Fig. 4 Evaluation of lexical expansion for the ArgId model over increasing size of the seed training dataset. The shaded region represents the standard deviation based on 10 runs of the model. The x-axis is in log scale. Source dataset: Verbs

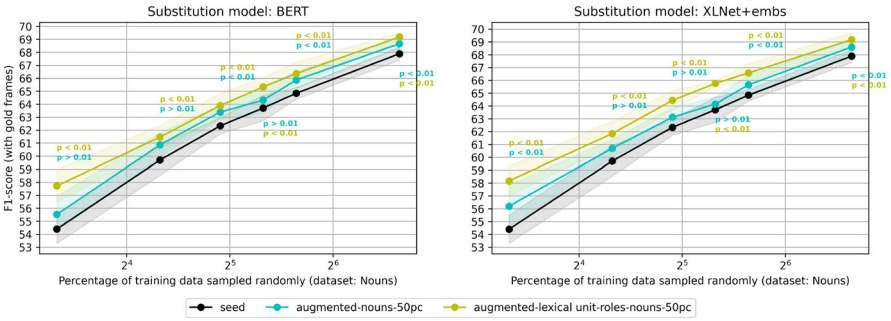

Fig. 5 Evaluation of lexical expansion for the ArgId model over increasing size of the seed training dataset. The shaded region represents standard deviation based on 10 runs of the model. The x-axis is in log scale. Source dataset: Nouns

each experiment was run 10 times with different random seeds to get the mean and standard deviation for the curve.

The learning curves are shown in Figs. 4 and 5. Both augmented datasets have consistently improved the model performance over their seed datasets and on average obtained 2–3% gain in F_1 . For datasets sampled from verbs, the difference in model performance for the seed and augmented datasets remained consistent and statistically significant for sample sizes larger than 10%, and it is true for both models regardless of the expansion configurations. But for datasets sampled from nouns, only the expansion configuration lexical unit-roles-nouns-50pc shows more consistent performance for all sample sizes. This difference in the performance of this configuration can also be observed in Table 22 where the expansion configurations with lexical unit-roles-nouns have consistently outperformed the ones where only nouns was expanded. Whereas for verbs, overall both types of configurations have performed better. This also provides interesting insight into the behavior of verb and noun predicates to choose optimal expansion configurations for each. We can conclude that as opposed to targets and frames, semantic roles are a more diversified set of words and hence proved to be an ideal candidate to augment when data is insufficient.

5.4 Results with the BERT-based parser

Hyperparameters As this model is originally designed to work for verb type predicates, so we only report results for the verbs-based datasets here. For seed datasets, the model was trained for 50 epochs, while for augmented datasets, it was trained for 17 epochs to have the same number of training steps for both the seed and augmented datasets. We used BERT large cased with the batch size of 8 and the learning rate of 2e−5. All models were run 10 times to get mean and standard deviation values.

For the BERT-based parser, we present the learning curve and used both BERT and XLNet+embs lexical substitution models for augmented datasets for comparison. The learning curves for both models are shown in Fig. 6. From the top, the first row shows the performance with gold frames and the second row shows the performance without gold frames. It can be confirmed by the curves that lexical expansion is indeed helpful to obtain performance gain when the number of annotations is insufficient. However, the performance gain starts to diminish when moving to the right of the x-axis where the seed dataset size increases, this is also supported by p values that are consistently less than 0.01 for the sample sizes of 10 to 30. The gain in performance shows similar patterns in both situations with or without gold frame information. Whereas using gold frames information obtained significantly higher scores with F_1 going above 70.0 for all models and datasets in comparison to using predicted frames where it remains close to 65.0). These scores are significantly higher than the open-SESAME parser for the same datasets and show the advantage of using pre-trained Transformer models to learn the syntax and semantics of the sentence in comparison to using syntactic features such as part-of-speech tags. While there is no clear candidate when it comes to the comparison of lexical substitution models, both BERT and XLNet+embs performed similarly.

For nouns, the extensive set of experiments could not produce similar results as for verbs. There were no improvements in the performance for the augmented datasets, and where it showed improvements, results were not consistent and the variance between multiple runs of the model was excessive.

6 Conclusion

In this work, we performed a study of text augmentation methods for semantic frame processing based on (i) non-contextualized distributional models such as word2vec and syntax-based distributional thesauri, and (ii) contextualized lexical substitution methods based on neural language models, such as BERT and XLNet. We tested these methods in two extensive experimental setups.

In the first set of experiments, we perform generation of lexical representations of semantic frames. We demonstrated that a single frame annotated example can be used to bootstrap a fully-fledged lexical representation of the FrameNet-style linguistic structures. Non-contextualized models proved to be strong baselines, but failed to produce good substitutes for polysemous words (same word but different semantic frame), whereas contextualized models of BERT and XLNet produced competitive substitutes, especially when information about the target word is injected effectively.

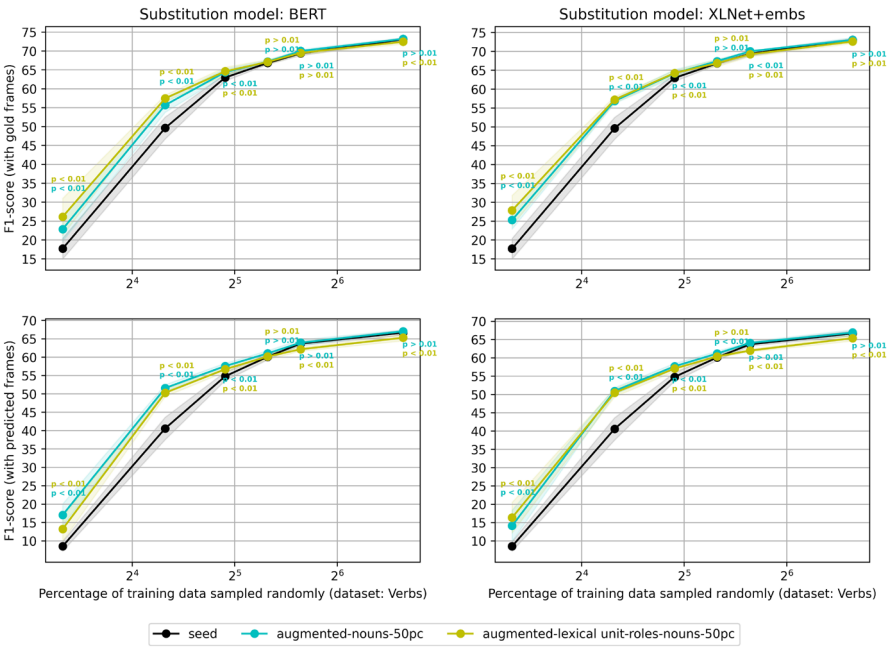

Fig. 6 Evaluation of lexical expansion for the BERT-based semantic role parser for the ArgId model over increasing size of the seed training dataset. The first row shows the performance using gold frames and the second row shows the combined performance where the first step is to predict the frames and then do the argument identification. The shaded region represents the standard deviation based on 10 runs of the model. The x-axis is in log scale. Source dataset: Verbs

Additionally, our experiments show that sometimes, combining individual models to generate lexical substitutes significantly helps to improve their individual performance.

Since automatic evaluation of lexical substitution is sensitive to completeness of the lexical resource itself, to further analyse the effectiveness of our method, we also did manual evaluation of these substitutes on small datasets to show that on the one hand, suitable lexical substitutes are sometimes absent from the gold datasets, while on the other hand, the present substitutes are not always good candidates for the purpose of lexical substitution since they can alter the sentence semantics.

In our second set of experiments, we deal with two neural FrameNet parsers by Swayamdipta et al. (2017) and Shi and Lin (2019). Namely, we demonstrate that text augmentation can be used to build more training samples from a few seed sentences, and these new frame representations help to improve the performance of semantic parsers for the semantic role identification and labeling tasks. These experiments suggest that expansion of roles (usually represented with nouns and noun phrases) and otherwise occurring nouns in the text significantly improves the performance of semantic parsing, while an expansion on verbs—which is an arguably harder task, as verbs does not have as many close co-hyponyms and synonyms—does not improve parsing results.

Overall, our results suggest that: (i) augmentation of lexical units can be of great use for expansion of lexical representation of semantic frames, and for (ii) building semantic parsers, which perform role identification in text, especially in situations where the number of training texts is small.

7 Future work

Going forward, we can expect further improvements from large foundation models like T5 (Raffel et al., 2020), BART (Lewis et al., 2020), FlanT5 (Longpre et al., 2023) and other pre-trained seq2seq transformers, especially those pre-trained on multiple word masking tasks helping to restore multiword expressions accurately. Experimenting with further contextualized lexical substitution methods, such as nPIC/PIC (Roller & Erk, 2016), may yield improvements in combined methods.

While large pre-trained language models are increasingly getting better at performing tasks in an end-to-end fashion, this is seemingly removing the need for explicitly expanding lexical-semantic resources for natural language understanding and generation tasks. However, there are still fields where lexical resources—with examples and their sources—are key to answering research questions or productive work, e.g. for the study of the structure of semantics, for the creation of dictionaries, as well as e.g. for controlled experiments in psycholinguistics and other fields. With our automatic expansion approach, we provide a method to aid the quicker development of these lexical resources in such situations, especially for under-resourced languages and domains.

Acknowledgements We thank the anonymous reviewers for their valuable feedback and acknowledge the support of the Deutsche Forschungsgemeinschaft (DFG) under the "JOIN-T 2" project (BI 1544/4-2), the German Academic Exchange Service (DAAD) and the Higher Education Commission (HEC), Pakistan, and Ministry of Science and Higher Education grant No. 075-10-2021-068.

Funding Open Access funding enabled and organized by Projekt DEAL.

Appendix 1: Effect of model size and masking on transformer-based models

As reported in Sect. 3.2.2, we used only *large-cased* variants of BERT and XLNet models without masking to reduce complexity of model choices in our experimental setup. Here, we compare their performance for all variants with and without masking to get an insight if there can be a single best configuration for all three datasets. Results are reported in Table 23. In case of BERT, the best configuration differs for all three tasks. For verbs-LU expansion task, *large-cased* variant without masking consistently outperformed all other variants in terms of P@k and MAP score. Larger variants seems to be better overall and masking has throughout negative affect in all variants. For nouns-LU expansion task, similar to verbs, the masking has negative affect overall, but in case of model size the effect is not consistent between cased and uncased variants. Overall the *base-cased* model without masking has performed best. Although the difference between the large and base model is not significant. For roles, it is shown that masking the target word has actually positive affect on all variants of BERT. But in comparison to LU expansion task for verbs and nouns, this difference in performance is more apparent in MAP scores than p@k scores. In terms of p@k score, *base-cased* with masking has performed best and in terms of MAP score the *large-uncased* model with masking has outperformed all other variants. For XLNet variants, results remained consistent and the *large* model without masking has performed best for all three datasets both in terms of p@k and MAP scores. For our final experiments, we just choose only one variant *large-cased* without masking to keep it simple.

Table 23 Comparisons of BERT and XLNet models for their sizes and masking affects

Model	Verb Lexical Unit Expansion			Noun Lexical Unit Expansion			Semantic Role Expansion		
	p@1	p@5	MAP@50	p@1	p@5	MAP@50	p@1	p@5	MAP@50
BERT large cased	**0.386**	**0.264**	**0.137**	0.403	0.288	0.114	0.471	0.388	0.118
BERT large cased with mask	0.270	0.213	0.108	0.307	0.242	0.095	0.482	0.387	0.128
BERT base cased	0.373	0.252	0.133	**0.407**	**0.297**	**0.119**	0.471	0.396	0.121
BERT base cased with mask	0.243	0.191	0.094	0.289	0.229	0.088	**0.491**	**0.401**	0.133
BERT large uncased	0.374	0.249	0.129	0.406	0.283	0.111	0.481	0.383	0.114
BERT large uncased with mask	0.262	0.207	0.106	0.302	0.239	0.095	0.490	0.398	**0.134**
BERT base uncased	0.329	0.225	0.115	0.384	0.276	0.107	0.452	0.365	0.105
BERT base uncased with mask	0.246	0.194	0.098	0.288	0.229	0.091	0.485	0.391	0.130
XLNet large	**0.352**	**0.26**	**0.137**	**0.386**	**0.293**	**0.122**	**0.513**	**0.421**	**0.144**
XLNet large with mask	0.295	0.228	0.119	0.323	0.255	0.104	0.479	0.392	0.134
XLNet base	0.323	0.236	0.122	0.370	0.282	0.115	0.490	0.405	0.139
XLNet base with mask	0.264	0.204	0.103	0.300	0.236	0.093	0.460	0.376	0.128

Appendix 2: Effect of pre-processing pipeline

Table 24 Evaluation of lexical substitutes for noun lexical units, with **lemminflect** library in post processing

Model	Noun Lexical Unit Expansion p@1	p@5	MAP@50
Non-contextualized models			
GloVe	0.390	0.294	0.123
fastText	0.410	0.309	0.130
word2vec	**0.415**	0.302	0.122
DT wiki	0.409	0.305	0.146
DT 59g	0.412	**0.324**	**0.156**
Contextualized models: Melamud			
Melamud add	0.384	0.282	0.123
Melamud balAdd	**0.421**	**0.313**	**0.141**
Melamud mult	0.340	0.254	0.109
Melamud balMult	0.416	0.306	0.139
Contextualized Transformer-based models			
BERT	0.417	0.301	0.123
XLNet	0.397	0.306	0.132
BERT [Tand-]	0.197	0.160	0.061
BERT [TandT]	0.416	0.301	0.122
BERT+embs	0.461	0.341	0.147
XLNet+embs	0.503	0.383	0.182
XLNet+embs (optimal)	**0.516**	**0.391**	**0.186**
Combined models			
nc-emb + DT	0.442	0.340	0.157
Melamud balAdd + DT	0.472	0.369	0.179
Melamud balAdd + nc-emb	0.472	0.357	0.164
XLNet + nc-emb	0.492	0.364	0.159
XLNet + DT	0.506	0.382	0.177
XLNet + Melamud balAdd	0.515	0.389	0.180
XLNet+embs + nc-emb	0.503	0.372	0.169
XLNet+embs + DT	0.521	0.396	0.191
XLNet+embs + Melamud balAdd	**0.534**	0.403	0.194
Melamud balAdd + nc-emb + DT	0.480	0.374	0.180
XLNet + nc-emb + DT	0.510	0.385	0.182
XLNet + Melamud balAdd + nc-emb	0.523	0.393	0.186
XLNet + Melamud balAdd + DT	0.530	0.405	0.201
XLNet+embs + nc-emb + DT	0.508	0.384	0.185
XLNet+embs + Melamud balAdd + DT	0.526	**0.407**	**0.205**

This library provides more robust way of POS tagging for words that can have multiple tags

Table 25 Evaluation of lexical substitutes for semantic roles, with stopword filtering

Model	Semantic Role Expansion		
	p@1	p@5	MAP@50
Non-contextualized models			
GloVe	0.317	0.238	0.071
fastText	0.192	0.133	0.029
word2vec	0.322	0.222	0.050
DT wiki	**0.365**	**0.283**	**0.090**
DT 59g	0.344	0.274	0.086
Contextualized models: Melamud			
Melamud add	0.367	0.273	0.069
Melamud balAdd	**0.376**	**0.279**	**0.071**
Melamud mult	0.353	0.260	0.065
Melamud balMult	0.372	0.272	0.070
Contextualized Transformer-based models			
BERT	0.510	0.413	0.126
XLNet	0.540	0.438	0.152
BERT [Tand-]	0.437	0.365	0.105
BERT [TandT]	0.467	0.385	0.120
BERT+embs	0.480	0.413	0.136
XLNet+embs	0.539	0.458	0.165
XLNet+embs (optimal)	**0.581**	**0.486**	**0.176**
Combined models			
nc-emb + DT	0.392	0.301	0.095
Melamud balAdd + DT	0.400	0.341	0.098
Melamud balAdd + nc-emb	0.453	0.358	0.098
XLNet + nc-emb	0.514	0.391	0.124
XLNet + DT	0.498	0.411	0.133
XLNet + Melamud balAdd	**0.556**	**0.465**	0.136
XLNet+embs + nc-emb	0.492	0.377	0.123
XLNet+embs + DT	0.480	0.410	0.137
XLNet+embs + Melamud balAdd	0.546	0.463	0.142
Melamud balAdd + nc-emb + DT	0.473	0.372	0.116
XLNet + nc-emb + DT	0.492	0.402	0.134
XLNet + Melamud balAdd + nc-emb	0.539	0.440	0.139
XLNet + Melamud balAdd + DT	0.535	0.454	0.145
XLNet+embs + nc-emb + DT	0.467	0.389	0.133
XLNet+embs + Melamud balAdd + DT	0.522	0.447	**0.147**

References

Amrami, A., & Goldberg, Y. (2018). Word sense induction with neural biLM and symmetric patterns. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 4860–4867). Association for Computational Linguistics. https://www.aclweb.org/anthology/D18-1523

Amrami, A., & Goldberg, Y. (2019). Towards better substitution-based word sense induction. CoRR arXiv:1905.12598

Anwar, S., Shelmanov, A., Panchenko, A., & Biemann, C. (2020). Generating lexical representations of frames using lexical substitution. In *Proceedings of the probability and meaning conference (PaM 2020)* (pp. 95–103). Association for Computational Linguistics. https://www.aclweb.org/anthology/2020.pam-1.13

Arefyev, N., Sheludko, B., Davletov, A., Kharchev, D., Nevidomsky, A., & Panchenko, A. (2019a). 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* (pp. 31–38). Association for Computational Linguistics. https://www.aclweb.org/anthology/S19-2004

Arefyev, N., Sheludko, B., & Panchenko, A. (2019b). Combining lexical substitutes in neural word sense induction. In *Proceedings of the international conference on recent advances in natural language processing (RANLP'19)* (pp. 62–70). http://lml.bas.bg/ranlp2019/proceedings-ranlp-2019.pdf

Arefyev, N., Sheludko, B., Podolskiy, A., & Panchenko, A. (2020). Always keep your target in mind: studying semantics and improving performance of neural lexical substitution. In *Proceedings of the 28th international conference on computational linguistics, international committee on computational linguistics* (pp. 1242–1255). https://www.aclweb.org/anthology/2020.coling-main.107

Baker, C. F., Fillmore, C. J., & Lowe, J. B. (1998). The Berkeley FrameNet Project. In *Proceedings of the 36th annual meeting of the association for computational linguistics and 17th international conference on computational linguistics* (Vol. 1, pp. 86–90). Association for Computational Linguistics.

Berant, J., & Liang, P. (2014). Semantic parsing via paraphrasing. In *Proceedings of the 52nd annual meeting of the association for computational linguistics* (Vol. 1: Long Papers, pp. 1415–1425). Association for Computational Linguistics. https://www.aclweb.org/anthology/P14-1133

Biemann, C., & Riedl, M. (2013). Text: Now in 2D! A framework for lexical expansion with contextual similarity. *Journal of Language Modelling, 1*(1), 55–95.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research, 3*, 993–1022.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics, 5*, 135–146.

Buljan, M., Padó, S., & Šnajder, J. (2018). Lexical substitution for evaluating compositional distributional models. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human language technologies* (Vol. 2 (Short Papers), pp. 206–211). Association for Computational Linguistics. https://www.aclweb.org/anthology/N18-2033

Chen, D., & Manning, C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 740–750). Association for Computational Linguistics. https://www.aclweb.org/anthology/D14-1082

Das, D., Chen, D., Martins, A. F. T., Schneider, N., & Smith, N. A. (2014). Frame-semantic parsing. *Computational Linguistics, 40*, 9–56.

Das, D., Schneider, N., Chen, D., & Smith, N. A. (2010). Probabilistic frame-semantic parsing. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics* (pp. 948–956). Association for Computational Linguistics. https://www.aclweb.org/anthology/N10-1138

Dementieva, D., Markov, I., & Panchenko, A. (2020). SkoltechNLP at SemEval-2020 task 11: exploring unsupervised text augmentation for propaganda detection. In *Proceedings of the fourteenth workshop on semantic evaluation* (pp. 1786–1792). International Committee for Computational Linguistics. https://www.aclweb.org/anthology/2020.semeval-1.234

Devlin, J., Chang, M. W., Lee, K., & 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* (Vol. 1 (Long and Short Papers), pp. 4171–4186). Association for Computational Linguistics. https://www.aclweb.org/anthology/N19-1423

Do, Q. N. T ., Bethard, S., & Moens, M. F. (2017). Improving implicit semantic role labeling by predicting semantic frame arguments. In *Proceedings of the eighth international joint conference on natural language processing* (Vol. 1: Long Papers), (pp. 90–99). Asian Federation of Natural Language Processing.

Fenogenova, A. (2021). Russian paraphrasers: paraphrase with transformers. In *Proceedings of the 8th workshop on balto-slavic natural language processing* (pp. 11–19). Association for Computational Linguistics. https://www.aclweb.org/anthology/2021.bsnlp-1.2

Ferraresi, A., Zanchetta, E., Baroni, M., & Bernardini, S. (2008). Introducing and evaluating ukwac, a very large web-derived corpus of English. In *Proceedings of the 4th web as corpus workshop (WAC-4) can we beat Google* (pp. 47–54).

Fillmore, C. J. (1982). Frame semantics. In The Linguistic Society of Korea, (Ed.) *Linguistics in the morning calm* (pp. 111–137).

Fossati, M., Giuliano, C., & Tonelli, S. (2013). Outsourcing FrameNet to the crowd. In *Proceedings of the 51st annual meeting of the association for computational linguistics* (Vol. 2: Short Papers, pp. 742–747). Association for Computational Linguistics. https://www.aclweb.org/anthology/P13-2130

Gao, Q., & Vogel, S. (2011). Utilizing target-side semantic role labels to assist hierarchical phrase-based machine translation. In *Proceedings of fifth workshop on syntax, semantics and structure in statistical translation* (pp. 107–115). Association for Computational Linguistics. https://www.aclweb.org/anthology/W11-1012

Goldhahn, D., Eckart, T., & Quasthoff, U. (2012). Building large monolingual dictionaries at the leipzig corpora collection: From 100 to 200 languages. In *Proceedings of the eight international conference on language resources and evaluation (LREC'12)*.

Hartmann, S., Kuznetsov, I., Martin, T., & Gurevych, I. (2017). Out-of-domain FrameNet semantic role labeling. In *Proceedings of the 15th conference of the European chapter of the association for computational linguistics* (Vol. 1, Long Papers, pp. 471–482). Association for Computational Linguistics. https://www.aclweb.org/anthology/E17-1045

Hermann, K.M., Das, D., Weston, J., & Ganchev, K. (2014). Semantic frame identification with distributed word representations. In *Proceedings of the 52nd annual meeting of the association for computational linguistics* (Vol. 1: Long Papers, pp. 1448–1458). Association for Computational Linguistics. https://www.aclweb.org/anthology/P14-1136

Kawahara, D., Peterson, D. W., & Palmer, M. (2014). A step-wise usage-based method for inducing polysemy-aware verb classes. In *Proceedings of the 52nd Annual meeting of the association for computational linguistics* (Vol. 1: Long Papers, pp. 1030–1040). Association for Computational Linguistics. http://aclweb.org/anthology/P14-1097

Khashabi, D., Khot, T., Sabharwal, A., & Roth, D. (2018). Question answering as global reasoning over semantic abstractions. In *Proceedings of the 32nd AAAI conference on artificial intelligence, (AAAI-18)* (pp. 1905–1914). Association for the Advancement of Artificial Intelligence. https://aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/17406

Kilgarriff, A., Rychlý, P., Smrz, P., & Tugwell, D. (2004). The sketch engine. In *Proceedings of the 11th EURALEX international congress* (pp. 105–115). Université de Bretagne-Sud, Faculté des lettres et des sciences humaines.

Kriz, R., Miltsakaki, E., Apidianaki, M., & Callison-Burch, C. (2018). Simplification using paraphrases and context-based lexical substitution. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human language technologies* (Vol. 1 (Long Papers), pp. 207–217). Association for Computational Linguistics.

Kshirsagar, M., Thomson, S., Schneider, N., Carbonell, J., Smith, N. A., & Dyer, C. (2015). Frame-semantic role labeling with heterogeneous annotations. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing* (Vol. 2: Short Papers, pp. 218–224). Association for Computational Linguistics. https://www.aclweb.org/anthology/P15-2036

Lang, J., & Lapata, M. (2010). Unsupervised induction of semantic roles. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics* (pp. 939–947). Association for Computational Linguistics. https://aclweb.org/anthology/N10-1137

Lee, J., & Yeung, C. Y. (2019). Personalized substitution ranking for lexical simplification. In *Proceedings of the 12th international conference on natural language generation* (pp. 258–267) Association for Computational Linguistics. https://www.aclweb.org/anthology/W19-8634

Levy, O., & Goldberg, Y. (2014). Dependency-based word embeddings. In *Proceedings of the 52nd annual meeting of the association for computational linguistics* (Vol. 2: Short Papers, pp. 302–308). Association for Computational Linguistics. https://www.aclweb.org/anthology/P14-2050

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., & Zettlemoyer, L. (2020). BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In D. Jurafsky, J. Chai, N. Schluter, J. R. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics, ACL 2020*, July 5–10, 2020.

Association for Computational Linguistics (pp. 7871–7880). https://doi.org/10.18653/v1/2020.acl-main.703

Lin, D. (1998). Automatic retrieval and clustering of similar words. In *Proceedings of the 36th annual meeting of the association for computational linguistics and 17th international conference on computational linguistic* (Vol. 2, pp. 768–774). Association for Computational Linguistics.

Longpre, S., Hou, L., Vu, T., Webson, A., Chung, H. W., Tay, Y., Zhou, D., Quoc, V. Le, tBZ, Wei, J., & Roberts, A. (2023). The flan collection: Designing data and methods for effective instruction tuning. arXiv:2301.13688

Materna, J. (2012). LDA-Frames: an unsupervised approach to generating semantic frames. In A. Gelbukh (Ed.), *Computational linguistics and intelligent text processing* (pp. 376–387). Springer.

Materna, J. (2013). Parameter estimation for LDA-frames. In *Proceedings of the 2013 conference of the North American chapter of the association for computational linguistics: Human language technologies* (pp. 482–486). Association for Computational Linguistics. http://www.aclweb.org/anthology/N13-1051

McCarthy, D., & Navigli, R. (2009). The English lexical substitution task. *Language Resources and Evaluation, 43*(2), 139–159.

Melamud, O., Levy, O., & Dagan, I. (2015). A simple word embedding model for lexical substitution. In *Proceedings of the 1st workshop on vector space modeling for natural language processing* (pp. 1–7). Association for Computational Linguistics. https://www.aclweb.org/anthology/W15-1501

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (Vol. 26, pp. 3111–3119). Curran Associates, Inc. https://papers.nips.cc/paper/2013/file/9aa42b31882ec039965f3c4923ce901b-Paper.pdf

Miller, G. A. (1995). Wordnet: a lexical database for English. *Communications of the ACM, 38*(11), 39–41.

Modi, A., Titov, I., & Klementiev, A. (2012). Unsupervised induction of frame-semantic representations. In *Proceedings of the NAACL-HLT workshop on the induction of linguistic structure* (pp. 1–7). Association for Computational Linguistics. http://www.aclweb.org/anthology/W12-1901

Oepen, S., Kuhlmann, M., Miyao, Y., Zeman, D., Cinková, S., Flickinger, D., Hajič, J., Ivanova, A., & Urešová, Z. (2016). Towards comparability of linguistic graph Banks for semantic parsing. In *Proceedings of the tenth international conference on language resources and evaluation (LREC'16)* (pp. 3991–3995). European Language Resources Association (ELRA). https://aclanthology.org/L16-1630

Padró, L., & Stanilovsky, E. (2012). FreeLing 3.0: Towards wider multilinguality. In *Proceedings of the eighth international conference on language resources and evaluation (LREC'12)* (pp. 2473–2479). European Language Resources Association (ELRA). http://www.lrec-conf.org/proceedings/lrec2012/pdf/430_Paper.pdf

Palmer, M., Gildea, D., & Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics, 31*(1), 71–106.

Parker, R., Graff, T. D., Kong, J., Chen, K., & Maeda, K. (2009). *English gigaword fourth edition. Linguistic Data Consortium LDC2009T13*. Web Download.

Peng, H., Thomson, S., Swayamdipta, S., & Smith, N. A. (2018). Learning joint semantic parsers from disjoint data. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: Human language technologies* (Vol. 1 (Long Papers), pp. 1492–1502). Association for Computational Linguistics. https://www.aclweb.org/anthology/N18-1135

Pennacchiotti, M., De Cao, D., Basili, R., Croce, D., & Roth, M. (2008). Automatic induction of FrameNet lexical units. In *Proceedings of the 2008 conference on empirical methods in natural language processing* (pp. 457–465). Association for Computational Linguistics. https://www.aclweb.org/anthology/D08-1048

Pennington, J., Socher, R., & Manning, C. (2014). GloVe: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532–1543). Association for Computational Linguistics. https://www.aclweb.org/anthology/D14-1162

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & 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* (Vol. 1 (Long Papers), pp. 2227–2237). Association for Computational Linguistics. https://aclweb.org/anthology/N18-1202

QasemiZadeh, B., Petruck, M. R. L., Stodden, R., Kallmeyer, L., & Candito, M. (2019). SemEval-2019 task 2: Unsupervised lexical frame induction. In *Proceedings of the 13th international workshop on semantic evaluation* (pp. 16–30). Association for Computational Linguistics. https://www.aclweb.org/anthology/S19-2003

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research, 21*, 140:1-140:67.

Roller, S., & Erk, K. (2016). PIC a different word: A simple model for lexical substitution in context. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies* (pp. 1121–1126). Association for Computational Linguistics. https://www.aclweb.org/anthology/N16-1131

Roller, S., Kiela, D., & Nickel, M. (2018). Hearst patterns revisited: Automatic hypernym detection from large text corpora. In *Proceedings of the 56th annual meeting of the association for computational linguistics* (Volume 2: Short Papers pp. 358–363). Association for Computational Linguistics. https://www.aclweb.org/anthology/P18-2057

Roth, M., & Lapata, M. (2015). Context-aware frame-semantic role labeling. *Transactions of the Association for Computational Linguistics, 3*, 449–460.

Şahin, G. G. (2022). To augment or not to augment? A comparative study on text augmentation techniques for low-resource NLP. *Computational Linguistics, 48*(1), 5–42.

Shen, D., & 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)* (pp. 12–21). Association for Computational Linguistics. https://www.aclweb.org/anthology/D07-1002

Shi, P., & Lin, J. (2019). Simple BERT models for relation extraction and semantic role labeling. CoRR arXiv:1904.05255

Sikos, J., & Padó, S. (2019). Frame identification as categorization: Exemplars vs prototypes in embeddingland. In *Proceedings of the 13th international conference on computational semantics—long papers* (pp. 295–306). Association for Computational Linguistics. https://www.aclweb.org/anthology/W19-0425

Smedt, T. D., & Daelemans, W. (2012). Pattern for python. *Journal of Machine Learning Research, 13*, 2063–2067.

Swayamdipta, S., Thomson, S., Dyer, C., & Smith, N. A. (2017). Frame-semantic parsing with softmax-margin segmental RNNs and a syntactic scaffold. CoRR arXiv:1706.09528

Swayamdipta, S., Thomson, S., Lee, K., Zettlemoyer, L., Dyer, C., & Smith, N. A. (2018). Syntactic scaffolds for semantic structures. In *Proceedings of the 2018 conference on empirical methods in natural language processing* (pp. 3772–3782). Association for Computational Linguistics. https://www.aclweb.org/anthology/D18-1412

Titov, I., & Klementiev, A. (2012). A bayesian approach to unsupervised semantic role induction. In *Proceedings of the 13th conference of the European chapter of the association for computational linguistics* (pp. 12–22). Association for Computational Linguistics. http://www.aclweb.org/anthology/E12-1003

Ustalov, D., Panchenko, A., Kutuzov, A., Biemann, C., & Ponzetto, S. P. (2018). Unsupervised semantic frame induction using triclustering. In *Proceedings of the 56th annual meeting of the association for computational linguistics* (Vol. 2: Short Papers, (pp. 55–62). Association for Computational Linguistics. https://aclweb.org/anthology/P18-2010

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. U., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (Vol. 30, pp. 6000–6010). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Wang, R. C., & Cohen, W. W. (2007). Language-independent set expansion of named entities using the web. In *Proceedings of the 2007 seventh IEEE international conference on data mining* (pp. 342–350). IEEE Computer Society.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., & Brew, J. (2019). Huggingface's transformers: State-of-the-art natural language processing. arXiv:1910.03771

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., & Raffel, C. (2021). mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: Human language tech-*

nologies (pp. 483–498). Association for Computational Linguistics. https://aclanthology.org/2021.naacl-main.41

Yang, B., & Mitchell, T. (2017). A joint sequential and relational model for frame-semantic parsing. In *Proceedings of the 2017 conference on empirical methods in natural language processing* (pp. 1247–1256). Association for Computational Linguistics. https://www.aclweb.org/anthology/D17-1128

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., & Le, Q. V. (2019). Xlnet: generalized autoregressive pretraining for language understanding. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d' Alché-Buc, E. Fox, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 32, pp. 5753–5763). Curran Associates Inc.

Zhai, F., Zhang, J., Zhou, Y., & Zong, C. (2013). Handling ambiguities of bilingual predicate-argument structures for statistical machine translation. In *Proceedings of the 51st annual meeting of the association for computational linguistics* (Vol. 1: Long Papers, pp. 1127–1136). Association for Computational Linguistics. https://www.aclweb.org/anthology/P13-1111

Zhou, W., Ge, T., Xu, K., Wei, F., & Zhou, M. (2019). BERT-based lexical substitution. In *Proceedings of the 57th annual meeting of the association for computational linguistics* (pp. 3368–3373). Association for Computational Linguistics. https://www.aclweb.org/anthology/P19-1328