PBML

The Prague Bulletin of Mathematical Linguistics
NUMBER 120 APRIL 2023 47–71

Transferring Word-Formation Networks Between Languages

Jonáš Vidra, Zdeněk Žabokrtský

Charles University, Faculty of Mathematics and Physics, Institute of Formal and Applied Linguistics

Abstract

We present a method for supervised cross-lingual construction of word-formation networks (WFNs). WFNs are resources capturing derivational, compositional and other relations between lexical units in a single language. Current state-of-the-art methods for automatically creating them typically rely on supervised or unsupervised pattern-matching of affixes in string representations of words, with few recent inroads into deep learning. All methods known to us work purely in a monolingual setting, limiting the use of higher-quality supervised models to high-resource languages. In this paper, we present two methods, one based on cross-lingual word alignments and translation and another based on cross-lingual word embeddings and neural networks. Both methods are capable of transfer of WFNs into languages for which no word-formational data are available. We evaluate our models on manually-annotated word-formation data from the Universal Derivations and UniMorph projects.

1. Introduction

A word-formation network is a dataset capturing information about how are lexemes created using derivation, compounding, conversion and other types of relations. Such networks can be created using various degrees of automatization. On one end of the spectrum, there are networks created by manually annotating the individual relations, resulting in a dataset that is highly precise, but either expensive to create or small in size.

In this article, we explore methods from the other, unsupervised, part of the scale: methods which do not require any human input or in-language annotations of word-formation relations. Instead, they transfer knowledge from existing word-formation networks in other languages. One method we present uses parallel texts and off-the-shelf tools for tokenization and lemmatization, another one uses cross-lingual word

Corresponding author: vidra@ufal.mff.cuni.cz

Cite as: Jonáš Vidra, Zdeněk Žabokrtský. Transferring Word-Formation Networks Between Languages. The Prague Bulletin of Mathematical Linguistics No. 120, 2023, pp. 47–71. doi: 10.14712/00326585.027.

embeddings. Parallel texts are significantly more abundant and easier to obtain than word-formation annotations and they are available for more languages – compare the OPUS collection (Tiedemann, 2012), where just the OpenSubtitles corpus is available for 65 languages, to a survey of available word-formation networks listing only 63 resources for 22 languages (Kyjánek, 2018). Similarly, cross-lingual word embeddings can be created for dozens of languages, e.g. XLM-R (Conneau et al., 2020) is pretrained on 100 languages from the CommonCrawl dataset.

As a result, our methods should allow for a cheap and rapid creation of word-formation networks for many languages, although at a cost of lower quality. We hope that it is possible to emulate the successes of transfer learning methods used for other similar tasks in natural language processing, such as syntactic parsing (McDonald et al., 2011), part-of-speech tagging (Zhang et al., 2016) or lemmatization (Rosa and Žabokrtský, 2019).

The main idea behind our methods is that translation of text between languages is supposed to preserve the pragmatic meaning of texts and it usually preserves also the meaning of individual sentences and words. Since word-formational relations connect words with similar semantics and orthography, multiple possible target-language translations of a single source-language word are word-formationally related with a higher probability than randomly selected words. Moreover, many types of word-formational relations have parallels across languages. For example, actor nouns are typically derived from verbs – and if we take two such nouns from two languages, which are translations of one another, chances are that their predecessor verbs will also be translation equivalents (e.g. the Czech and English relations *opravit* ("to repair") → *opravář* ("repairman") are parallel, even though one uses derivation and the other one compounding). Therefore, we believe that some information about word-formation relations can be shared across languages.

In practice, the transferred networks are too small to be usable, but they can serve as synthetic training data for a supervised machine translation model, which extracts word-formation patterns found therein and finds more examples of them across a large lexicon, thereby improving the recall of the resulting network. Synthetic training data are widely used in deep learning, e.g. in machine translation (Sennrich et al., 2016; Zhang and Zong, 2016).

The pilot experiments presented in this paper focus on one-to-one relations between lexemes. We omit compounding altogether and simplify the task of creating a word-formation network to a task of assigning each lexeme a single *parent* lexeme, or deciding that it is unmotivated and should function as a root of the morphological family.

2. Related work

Most existing word-formation data is in the form of manually- or semi-automatically-created word-formation networks. These are made individually for each language, using annotation schemas tailored towards that language's needs. Two larger projects aim to unify the annotation formats and provide data for more languages in a single format: Universal Derivations (Kyjánek et al., 2019) and, recently, UniMorph since version 4.0 (Batsuren et al., 2022).

Universal Derivations (UDer) extracts its data from word-formation networks created by linguists. The collection contains 31 resources covering 21 languages. Individual resources differ in annotation goals (some resources marking all word-formational relations, others e.g. only deverbal derivations), size (ranging from a thousand to a million lexemes), and quality. Some resources in the collection contain also other annotations, such as semantic labels of the relations or morphological segmentation.

UniMorph is a massively multilingual resource which aims at describing morphology in a general, language-universal way. The UniMorph data covers inflection of 168 languages, with 25 of them also containing word-formational information. The word-formational data, sourced from Wiktionary, describes derivational morphology only and contains no features other than derivational relations and annotation of the changed morpheme(s) in the successor lexeme. As with UDer, many datasets are small, covering only a few thousand relations.

In addition to the manually-created word-formation networks, multiple models for automatic construction have been proposed, typically working on the formal level (textual-string-wise) by detecting paradigmatic changes between the predecessor(s) and successor. Baranes and Sagot (2014) created a method that infers derivational relations from inflectional paradigms and reported a very high precision (80-98% depending on the language). The relations are detected by first extracting a list of possible prefixal and suffixal changes and then pattern-matching pairs of words against it. The inflectional paradigms are used for reducing problems with suppletion and allomorphy within stems, which would otherwise cause the prefix- and suffix pattern matching to fail – e.g. if we know that *spoken* is a past participle form of a lexeme with lemma *speak*, we can derive the lexeme *unspoken* from *speak* using the rule *X* → *un-X*.

A different solution to the problem of allomorphy is proposed by Lango et al. (2021), who use a pattern-mining method to detect rules of allomorphy jointly with affixation. The patterns are extracted automatically in an unsupervised fashion and the potential relations are ranked by a machine-learning model trained on a small manually annotated word-formation network.

Batsuren et al. (2019) deal with cognate detection (i.e. linking words of common origin, identical meaning and similar spelling in different languages) using a multilingual approach. The multilingual data they use is a specialized linguistic resource containing information about etymological ancestry, which means that their methods are not directly applicable in our semi-supervised setting.

Cognates can also be used as a clue for aligning parallel corpora and several methods for detecting cognate pairs were developed with the alignment task in mind, but these methods need not be very precise – e.g. Church (1993) uses identical character

4-grams and Simard et al. (1992) use pairs of words with identical first four characters; both methods are too imprecise to recognize exact word-formational relations.

More recently, algorithms working with word embeddings (as a proxy for a semantic representation) have also been proposed: Musil et al. (2019) show that word embedding differences between word-formationally related words reflect the word-formation paradigm of the relation, and perform clustering of word-formation relations to retrieve the paradigms, although they don't use the models to produce a word-formation network. Svoboda and Ševčíková (2022) use a fine-tuned Marian translation model (Junczys-Dowmunt et al., 2018) to directly produce parent lemma(s) for a given child lemma. The model requires a very large amount of data to train, and they solve this issue by creating synthetic training data with a simple manually-crafted morphology model, which creates nonsensical, but well-formed compounds. This works, because the focus of PareNT is Czech compounding, which has a simple formal structure, unlike typical derivational patterns in most languages.

The task of constructing word-formation networks is superficially similar to the task of dependency parsing – in both cases, one tries to attach words to typically a single parent (head or predecessor). However, there are also important differences: Dependency parsing is in many ways computationally simpler, because the space of potential heads for any single lexical unit is bounded by the length of a sentence (typically tens of units), while in WFN construction, any lexeme in the language can be the correct predecessor (typically hundreds of thousands of units). Also, when machine learning is used, data for syntactic parsing is more abundant, because the inventory of training sentences is potentially infinite and getting new ones from a corpus is relatively cheap, while with WFNs, the number of training examples is limited to the lexicon size.

3. Models

Our models process data in two steps: In the first step (projection), a cross-lingual method is used to create a small word-formation network in the target language using training data in other languages. In the second step (bootstrap extension), the small network from the first step is used as synthetic training data to train a supervised model of word formation, which produces a large word-formation network.

In this paper, we present two models for each step: The projection step can be performed either by the Transfer model (see 3.1.1), or by the Cross-lingual embedding model (see 3.1.2). The bootstrap extension step can be performed by the Statistical machine learning extension model (see 3.2.1) or by the Neural extension model (see 3.2.2).

3.1. Projection models

3.1.1. Transfer model

To transfer a word-formation network from a source to a target language, we view the network as a list of parent-child derivational relations and attempt to find the best parent for each target-side lexeme using a word-translation model together with target-side formal similarity metrics. Conceptually, the input lexeme C is first back-translated into the source language as C′, a suitable parent P′ of the translation is found in the source word-formation network and this parent is translated into the target language as P.

The translations and backtranslations are found using a probabilistic word translation lexicon induced from word-aligned data obtained by running FastAlign (Dyer et al., 2013) on a lemmatized parallel corpus. Since the present article does not consider compounding, univerbation or other word-formation relations connecting more than two lexemes, we count each pair of aligned lexemes separately, regardless of whether one of the lexemes has other alignments in that parallel sentence pair. As a result, a lexeme aligned to a multi-word phrase is considered to be equally translated from each member lexeme of that phrase.

Since there may be multiple possible translations of each lexeme, and because the most suitable parent needn't be the direct parent of C′, but rather another member of its word-formational family (e.g. the Czech lexemes *svoboda* ("freedom") → *svobodný* ("free") have the opposite derivational relation from English or German *frei* → *die Freiheit*), the process is conducted probabilistically, yielding many potential parents P for each C, each with a score. The target network is then found by finding the spanning tree of this graph of relations which maximizes the product of the scores (Chu and Liu, 1965).

The score s of each potential relation P → C is obtained as a weighted arithmetic mean (with weight w) of the translation score Xfer(C, P) and a relative edit distance computed from the Levenshtein distance l(C, P), according to Equation 1 below. The relative edit distance is the Levenshtein distance between the lemmas of C and P divided by the maximum of their lengths, yielding a number between 0 and 1.

$$s = \frac{\mathrm{Xfer}(C, P) + w \cdot (1 - \frac{l(C,P)}{\max(|C|,|P|)})}{w + 1} \tag{1}$$

We define the translation score of C and P as Xfer(C, P) according to Equation 2, where |align(x, y)| denotes the number of alignments between lexemes x and y seen in the aligned data and dist(C′, P′) denotes the number of relations on the shortest path from C′ to P′ in the source network.

$$\mathrm{Xfer}(C, P) = \sum_{\forall C', P'} \frac{|\mathrm{align}(C, C')|}{\sum_{\forall x} |\mathrm{align}(C, x)|} \cdot 0.5^{\mathrm{dist}(C', P')} \cdot \frac{|\mathrm{align}(P', P)|}{\sum_{\forall x} |\mathrm{align}(P', x)|} \tag{2}$$

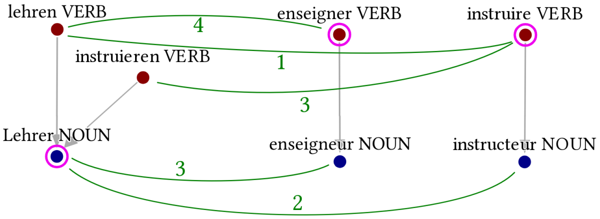

Figure 1: An example of finding a parent for the German lexeme *Lehrer* ("teacher") by transferring information from a French word-formation network, with word-formation relations in grey and alignments in green. *Lehrer* is aligned to *enseigneur* $^{3}/_{5}$ times, which has *enseigner* available through 1 relation, to which *lehren* is aligned $^{4}/_{4}$ times. *Lehrer* is also aligned to *instructeur* $^{2}/_{5}$ times, which has *instruire* available through 1 relation, to which *lehren* is aligned $^{1}/_{4}$ times and *instruieren* $^{3}/_{4}$ times. The translation score of *lehren* → *Lehrer*, calculated according to Equation 2 below, is therefore $\frac{3}{5} \cdot \frac{1}{2} \cdot \frac{4}{4} + \frac{2}{5} \cdot \frac{1}{2} \cdot \frac{1}{4} = 0.35$ while the score of *instruieren* → *Lehrer* is $\frac{2}{5} \cdot \frac{1}{2} \cdot \frac{3}{4} = 0.15$. The relative edit distance is $^{2}/_{6}$ for *lehren* → *Lehrer*, and $^{8}/_{11}$ for *instruieren* → *Lehrer*. Therefore, the final score of *lehren* → *Lehrer* is $\frac{0.35+5\cdot(1-{}^{2}/_{6})}{6} = 0.336$ and the score of *instruieren* → *Lehrer* is $\frac{0.15+5\cdot(1-{}^{8}/_{11})}{6} = 0.252$.

Therefore, the translation score is the product of the conditional probability of obtaining the backtranslated lexeme C' given the lexeme C and the conditional probability of obtaining the translated parent lexeme P given P', halved for each relation that has to be traversed between C' and P'. If there are multiple possible choices of C' and P' for the given C and P, their translation scores are summed.

To prevent relations with low scores from being selected in the case where there are no better candidates, a relation is only considered for inclusion if its score is higher than a threshold.

An illustration of the translation score calculation is given in Figure 1.

The transfer algorithm is parametrized by the weights used for calculating the weighted mean of the translation and edit distance scores, and by the threshold. Since we intend to use the transfer algorithm in an unsupervised setting, it is necessary to obtain the weights without training them using e.g. grid search or gradient descent on in-language annotations. We have, however, found that although the algorithm is moderately sensitive to the setting of the weights and the threshold, the optimal settings in all tested languages are nearly identical. This allows us to train the hyperparameters on one language pair in a supervised manner and use them on other pairs without further training. Using grid search on the Czech → German transfer pair, we set the weight of the edit distance to 5, the weight of the translation to 1 and the threshold to 0.8.

3.1.2. Cross-lingual embedding model

Our second proposed model is a pairwise classifier neural network. Its inputs are two lexemes (l_p the potential predecessor, l_s the potential successor) represented by their word and character embeddings, and the output is a score classifying l_s as derived from l_p or not[1].

The model uses non-pretrained character embeddings, which are merged to produce word-level states by passing them through a bidirectional recurrent neural network layer with GRU activation. The resulting GRU states are concatenated with pretrained word embeddings and passed through a single hidden layer with ReLU activation. The hidden states are then classified using SoftMax into two classes, derived or nonderived.

The architecture can be used both monolingually and cross-lingually, if cross-lingual word embeddings are available. In our case, cross-lingual training is used to obtain synthetic training data for each language, followed by either a second cross-lingual phase utilizing all synthetic datasets to train a single model, or a monolingual training phase training one model for each language. In the first cross-lingual phase, the model for each target language is trained separately, using only data from other languages. No model is therefore trained on the same language it predicts data for, simulating a semi-low-resource setting in which raw texts are available for training the word embeddings, but annotated word-formation data is missing.

Since the network classifies pairs of lexemes and classifying all pairs in the lexicon is computationally prohibitively expensive (the complexity increases quadratically with the lexicon size and the larger networks have on the order of 1 000 000 lexemes), the following heuristic is used to preselect pairs with a long common prefix and suffix. The lexicon is alphabetically sorted in a prograde and retrograde fashion, and for each lexeme, we test potential predecessors that lie within a 10 lexeme window around it in either sorting, for a total of 40 potential relations.

The selected lexemes needn't be the 40 ones with the longest common prefixes and suffixes, but at least 10 are guaranteed to share the longest prefix and 10 the longest suffix. We perform the lexicographic sorting on uppercased lemmas stripped of accent marks so that e.g. the German word *Wunsch* ("a wish") sorts close to *wünschen* ("to wish") despite the differences in case and the presence or absence of the umlaut.

This method of obtaining relation candidates depends on the linguistic properties of the languages under consideration. It works well with languages which derive words predominantly by affixation, with limited allomorphy in the stem and only rare circumfixation, apophony or suppletive relations, which this method generally doesn't detect as possible relations. Therefore, the preselection of classification exam-

[1] Two alternative formulations were considered and tested: A network directly producing l_p from a given l_s as a string of characters, and a network classifying a bag of potential successor words at once from a given l_p, but the model detailed above was found to outperform them from the start and research into the alternative architectures was not pursued further.

ples limits the potential performance of the models – if the true word-formational predecessor doesn't lie in the window of tested examples, it cannot be correctly classified by the model. However, testing has shown that the languages we selected for evaluation (see Section 4) all get reasonable results despite the simplicity of the method – in all gold-standard networks, the heuristic selects at least 90 % of true predecessors. Therefore, the relatively small window size is not the main limiting factor of the prediction performance.

For example, looking at a window of ±5 lexemes catches 85 % of all possible derivational relations in the German DErivBase word-formation network and ±10 catches 90 %. In the French Démonette network, 96 % of derivations are within ±5 and 98 % are within a ±10 window. In Czech DeriNet, a window of ±5 contains 85 % of all relations and ±10 contains 90 %. The method would perform poorly on languages with more frequent circumfixation or nonconcatenative morphology, such as transfixation or templatic morphology found in e.g. Hebrew or Arabic.

A possible systematic fix for detecting words derived by circumfixation would be to use a more complex measure of morphological similarity. A method we tried is the orthographic part of the model from Proxinette (Hathout, 2008), which approximates morphological relatedness by counting common n-grams of varying length, probabilistically weighting them by rarity in the corpus. Its construction allows enumerating lexemes most similar to an input lexeme in a computationally-tractable way, without considering all pairs. However, it produced inferior results on the Czech, German and French datasets we evaluated it on, and therefore we don't use it in our experiments.

A word-formation network is then constructed by calculating the maximum spanning tree of edges with a classification score ≥ 0.5, with the score used as the edge weight.

When training the network, the training data is sampled uniformly randomly from all positive examples in the training word-formation networks, supplemented by two sources of negative examples: Non-predecessor lexemes randomly sampled from the whole lexicon and non-predecessors sampled from the heuristic window around each lexeme. For each positive example pair, we sample 2 random negative pairs from the whole lexicon, and 3 random lexemes and for each of them one random non-predecessor from the window, for a 1:5 positive:negative sample ratio.

The word embeddings used are based on multilingually-aligned staticized XLM-R (Hämmerl et al., 2022). The XLM-R model (Conneau et al., 2020) provides high quality cross-lingual contextual embeddings, but since the task of identifying word-formational relations is lexical in nature, it is better suited for use with static embeddings. These are obtained using the X2Static method (Gupta and Jaggi, 2021), which distills static embeddings from contextual by a process similar to FastText's "continuous bag of words" (Bojanowski et al., 2017), but applied to contextual word embeddings instead of the words themselves. The staticization process transforms the embeddings one language at a time, so the cross-lingual relations of embeddings are partially lost, requiring realignment using VecMap.

3.2. Bootstrap extension

One issue with the aforementioned methods is that the word-formation networks they are able to produce are limited in size, because they both work only on lexemes with large-enough frequency in a corpus. Therefore, it is desirable to increase coverage of lower-frequency parts of the lexicon and lexemes not seen in the parallel data or embeddings lexicon. We propose two different methods to do this, both trained on data produced by one of the methods above. The first method is based on statistical machine learning with manually selected features, the second one reuses the neural network described above in a different setting.

3.2.1. Statistical machine learning extension model

One way of increasing recall of the produced word-formation networks is to take the networks created by the transfer model or cross-lingual embedding model described above, extract affixal patterns found therein and apply them to a larger lexicon.

The affixal pattern of a (proposed) word-formational relation is an unsupervised approximation of the morpheme difference between the related lexemes. We obtain it as the leftover substrings to the left and right of the longest common contiguous substring shared by lowercased lemmas of the lexemes. For example, the relation *Kampf* (“a fight”) → *kämpfen* (“to fight”) has the longest common contiguous substring *mpf* and affixal pattern *ka-* → *kä-* + *-en*.

We use the transferred network as a seed to train a machine learning method to predict derivational relations by classifying pairs of lexemes as either directly derived or non-derived from one another. The output network is obtained by finding the maximum spanning tree of the graph of predictions (Chu and Liu, 1965). The features used for classification are the one-hot-encoded part-of-speech categories of both lexemes, their edit distance, the difference of their lengths, whether each of them starts with a capital letter and the frequency of their affixal pattern as seen in the training dataset.

Since this method works by classifying pairs of lexemes, we again use the heuristic method for preselecting classification pairs described in Section 3.1.2 to decrease the computational complexity.

We evaluated multiple classification methods implemented in the scikit-learn package (Pedregosa et al., 2011), namely SVC, LogisticRegression, AdaBoostClassifier, KNeighborsClassifier, DecisionTreeClassifier, BernoulliNB and Perceptron and selected logistic regression for consistent evaluation performance.

3.2.2. Neural extension model

The neural extension model reuses the architecture of the cross-lingual embedding model, but with different data. It is trained on synthetic word-formation networks produced by the cross-lingual embedding model described above. The use of the training data is different too.

The cross-lingual embedding model doesn't train on data for the language it predicts relations for to ensure correct separation between training and evaluation data, and therefore n models are trained to produce data for n languages. The extension model is fully cross-lingual – a single universal model is trained jointly on all languages and can classify word-formational relations for any language.

In addition to getting the benefit of supervised training on the target language, the neural extension model also benefits from an extended lexicon compared to the cross-lingual embedding model – while the dataset for the cross-lingual embedding model contains the intersection of the manually created WFNs with the embedding lexicon, the extension model uses the embedding lexicons directly, providing potentially more training examples.

4. Training and evaluation data

For training and evaluating the word-formation models, we use word-formational data from the Universal Derivations (Kyjánek et al., 2019) and UniMorph (Batsuren et al., 2022) projects.

The word embedding data required by the neural models is taken from pretrained X2S-MA (Hämmerl et al., 2022), which is a static embedding resource created from XLM-R (Conneau et al., 2020) by first staticizing the embeddings using X2Static (Gupta and Jaggi, 2021) and then realigning the resulting static embeddings cross-lingually. Although it doesn't use subword segmentation, and is limited to its training lexicon as a result, we've found it to outperform other sources of embeddings.

All resources mentioned above are available for many languages: UDer for 21, UniMorph for 25 and X2S-MA for 40. However, their intersection is more limited – only 14 languages have both a word-formation network (from either UDer or UniMorph) and pretrained embeddings available. From those languages, we selected the 13 listed in Table 1 for use with the neural-networks-based models. One language, Dutch, was excluded, because its word-formation network as contained in UDer has quality too low to be usable for either training or evaluation due to errors introduced in the UDer conversion process.

When there are multiple networks for one language, we train on concatenation of lists of all relations. Compounding relations are treated as multiple derivational relations with the same successor. The data sizes of the individual word-formation resources for languages which are also present in the X2S-MA embeddings dataset are summarized in Table 1.

Lang	Resource	Lexemes	Relations
deu	DErivBase*	280775	43367
deu	UniMorph	40155	29381
eng	CatVar*	82675	24628
eng	UniMorph	264690	225131
eng	WordNet	13813	7855
est	EstWordNet*	988	507
fas	DeriNetFA*	43357	35745
fin	FinnWordNet*	20035	11890
fin	UniMorph	48499	36997
fra	Demonette*	22060	13808
fra	UniMorph	93382	73259
hun	UniMorph*	38441	32477
ita	DerIvaTario*	8267	1783
kaz	UniMorph*	3158	1965
por	EtymWordNetPT	2797	1610
por	NomLexPT*	7020	4201
por	UniMorph	19236	12687
rus	DeriNetRU	337632	164725
rus	DerivBaseRU*	270473	134024
rus	EtymWordNetRU	4005	3227
rus	GCompAna	4931	1639
rus	UniMorph	19823	14048
spa	DeriNetES*	151173	42825
spa	UniMorph	42760	31293
tur	EtymWordNetTR*	7775	5838
tur	UniMorph	2836	1776

Table 1: Data sizes of different resources. Resources labelled UniMorph are Wiktionary data extracted by the UniMorph project (Batsuren et al., 2022), all other resources are taken from the UDer project (Kyjánek et al., 2019). Resources marked by a star are used for evaluation in addition to training.

The gold standard data for each language is always taken from one resource, even when multiple resources for the language exist, to avoid having multiple conflicting golden predecessors for a single lexeme. The datasets designated as golden are marked in Table 1 by a star. Due to the setup of the experiments, the resource used for evaluation on a language is never directly used for training of that particular language's model. However, in the second multilingual step, the resource is trained on indirectly, because models for other languages do use it. For example, the Portuguese data are left out when training the Portuguese model, but are used for training the English model. The second level model then uses both English and Portuguese data from the previous step. We deem that this is not an issue, because the neural network cannot get high scores by reproducing its training data, as the data are transferred cross-lingually twice before evaluation.

The transfer model was trained and evaluated on three languages only, namely Czech, French and German. These languages were selected for the large size and quality of their word-formation networks as present in UDer – DeriNet 2.0 (Žabokrtský et al., 2016) with 809 282 relations, Démonette 1.2 (Hathout and Namer, 2014) with 13 808 relations and DErivBase 2.0 (Zeller et al., 2013) with 43 368 relations, respectively. The transfer model fails to extract useful information from source data with low accuracy, and since (unlike the neural model) it works purely on pairs of languages, it is not possible to combine smaller resources for multiple languages to get one larger usable dataset.

We transferred each network into both other languages and compared the result to the existing network for that language. The transfer was realized using word dictionaries obtained from word alignments of parallel data. We used the OpenSubtitles dataset from the OPUS collection (Tiedemann, 2012) for all language pairs, lemmatizing them with UDPipe 1.2 (Straka and Straková, 2017) and extracting only words tagged as adjectives, adverbs, nouns and verbs. The lemmatizer uses pretrained models trained on treebanks from Universal Dependencies (Nivre et al., 2016). The lemmatized corpora are then aligned using FastAlign (Dyer et al., 2013). The data sizes are listed in Table 2.

Lang pair	Sentences	Tokens on left	Tokens on right
de — cs	15 237 340	48 320 109	45 922 280
fr — cs	25 838 124	83 108 504	87 983 667
fr — de	14 779 572	44 135 610	48 440 995

Table 2: Sizes of parallel data for each language pair after part-of-speech category filtering.

5. Evaluation Method

We evaluate the performance of our systems by measuring precision, recall and accuracy in the task of assigning a parent to a lexeme. We define precision as the ratio of correctly predicted relations to all predicted relations, recall as the ratio of correctly predicted relations to all gold relations and accuracy as the ratio of correctly assigned parents or correctly recognized unmotivated lexemes to all gold lexemes.

```
 1 for gold_child in gold.lexemes:
 2   if not gold_child.parent:
 3     true_negative++
 4   else:
 5     for t_child in translations(gold_child):
 6       for t_parent in family(t_child):
 7         for parent in backtranslations(t_parent, gold_child):
 8           if parent = gold_child.parent:
 9             true_positive++
10             continue_line 1
11     false_negative++
12   accuracy := ((true_positive + true_negative) / (true_positive +
       ↪ true_negative + false_negative))
13   recall := true_positive / (true_positive + false_negative)
```

Listing 1: Pseudocode for calculating oracle accuracy and recall of the transfer algorithm. The backtranslation function returns all backtranslations of `t_parent`, except those that translate to `gold_child`.

Therefore, the precision and recall don't take into account unmotivated lexemes, while the accuracy does. The gold-standard data is taken from the existing word-formation network for the target language.

Because the set of lexemes captured in the cross-lingually projected network differs from the one used in the gold-standard data, we calculate the metrics in two ways, which differ in their treatment of missing lexemes. "External" measures consider all gold-standard relations of lexemes missing from the evaluated network to be false negatives, while the "internal" measures ignore them instead measures and only measure scores on the intersection of the two lexicons. Therefore, the external measures quantify how close the method gets to reproducing the gold-standard data, while the internal scores show how good is the output itself. Precision is the same for both methods, but recall and accuracy differ. The baseline measures and the networks obtained by machine learning are created from the set of lexemes found in the gold-standard network, which makes the internal and external measures identical.

5.1. Baselines

To establish a lower bound of reasonably achievable scores, we created two baselines: one trivial, called "empty", and one inspired by the purely left- or right-branching parse, the standard baseline in syntactic parsing, called "closest-shorter".

The empty baseline for a given lexicon is calculated as the scores of an empty word-formation network created over that lexicon, i.e. a network without any relations. The lexemes from gold-standard data which have no assigned parent are therefore evaluated as correct, while all lexemes with parents are incorrect, resulting in unmeasurable (zero) precision, zero recall and moderate-to-high accuracy.

The closest-shorter baseline gives each lexeme four options for its parent and selects the one which has a shorter lemma and the closest orthographic distance, as measured by the ratio of the length of the longest common contiguous substring to the sum of lengths of the two lemmas. The options to choose from are the previous and next lexemes in prograde sorting of the lexicon, and the previous and next lexemes in retrograde sorting. The lemma length criterion means that lexemes surrounded by longer neighbors in both prograde and retrograde sorting of the lexicon remain unmotivated. We have already observed that both ends of most derivational relations lie within a small window on a sorted lexicon, making this baseline rather strong in terms of both precision and recall.

5.2. Oracle Score

As an additional measure of the potential quality of the transfer approach, we measured the oracle score of obtaining the gold-standard parent through any combination of back- and forward-translations of gold-standard child lexemes. Under this measure, unmotivated lexemes are always considered to be correct, and a derived lexeme is considered to be correctly connected to its parent if it can be backtranslated to a member of a word-formational family, which contains a member that can be translated to the correct parent. The pseudocode of this algorithm is present in Listing 1. The recall and accuracy obtained using this algorithm represent the maximum scores achievable with the transfer method, if it selected the gold parent for each lexeme every time it is available.

Any error in the recall can be broken down into three categories: first, where we cannot translate the child to the language of the transferring network; (no `t_child` on line 5 of Listing 1); second, where there are no translations of any members of the translated lexeme's family (no `parent` on line 7) and third, where no possible parent matches the gold one (predicate on line 8 is always false).

6. Evaluation Results

As can be seen in Table 3, the networks created by the transfer algorithm are rather small in size. Within the constructed network, precision and recall are moderate for most language pairs, but when compared to the gold standard data, recall is nearly zero for all of them.

To a large degree, difference in scores between languages can be attributed to the testing data – each language has its own independently developed dataset with different design decisions, size and quality. Even datasets with identical names (DerivBase, DeriNet) were typically created by different teams working with different constraints.

		Size [k]		Internal scores [%]				Gold scores [%]		
Alg	Lang pair	Lex	Rel	Prec.	Recall	F1	Acc.	Recall	F1	Acc.
Xfer	de → cs	18	6.0	40	33	36	54	0.29	0.58	1.2
	fr → cs	20	7.0	42	36	39	54	0.37	0.73	1.3
	cs → de	14	3.8	27	35	31	66	2.5	4.5	18
	fr → de	3	0.6	14	14	14	65	0.20	0.39	4.2
	cs → fr	3	1.2	24	31	27	43	2.1	3.9	7.7
	de → fr	0.4	0.1	3.5	11	5.3	59	0.04	0.07	1.8
ML	de → cs	1 026	743	46	74	56	49	74	56	49
	fr → cs	1 026	743	40	70	51	44	70	51	44
	cs → de	280	68	35	68	46	80	68	46	80
	fr → de	280	35	44	39	42	85	39	42	85
	cs → fr	21	15	60	89	72	66	89	72	66
	de → fr	21	5	36	14	20	37	14	20	37
closest-shorter baseline	cs	1 026	809	21	54	30	23	54	30	23
	de	280	225	5.2	57	10	21	57	10	21
	fr	21	17	32	83	46	39	83	43	39
empty baseline	cs	1 026	0	N/A	0.00	0.00	21	0.00	0.00	21
	de	280	0	N/A	0.00	0.00	85	0.00	0.00	85
	fr	21	0	N/A	0.00	0.00	35	0.00	0.00	35

Table 3: Evaluation scores of the results and baselines for each language pair. The lexeme and relation counts are in thousands. Internal scores are measured on the set of lexemes in the generated network, gold scores on the set of lexemes from gold data. Precision is identical for both. For the machine learning and baseline algorithms, the distinction between internal and gold scores does not matter, since the lexicon used for prediction is taken from the gold-standard data as is.

	Scores [%]		Error cause [%]			WFN rel count	
Lang pair	Recall	Acc.	No child trans	No parent trans	No match	Xferred	Gold
de → cs	5.1	29	91	0.08	3.8	43 368	809 282
fr → cs	6.8	32	90	0.05	3.6	13 808	809 282
cs → de	34	90	52	0.23	13	809 282	43 368
fr → de	26	93	51	0.02	22	13 808	43 368
cs → fr	35	80	57	0.20	8.3	809 282	13 808
de → fr	22	64	62	0.07	16	43 368	13 808

Table 4: Transfer oracle scores for each language pair. Precision is 100% in all cases. The error causes list percentage of cases where the lexeme cannot be translated to the language of the transferring network, where no possible parents can be translated back, and when none of the translated parents match the gold one, respectively. The error percentage points are relative to the total relation count, i.e. they sum up to 100 together with recall. The last two columns list sizes of the transferred and gold-standard word-formation networks, measured in relations.

For example, whether unconnected lexemes remain in the database or are elided has a dominating effect on accuracy – accuracy on the German, English and Spanish gold-standard WFNs is higher than on the other ones, because they contain $>$ 70 % lexemes without parents, which are comparatively easy to correctly predict, but don't contribute to either precision or recall.

The classification example preselection heuristic may be a bottleneck on performance, as it limits recall to approximately 90 % and several networks come rather close to that number. But it is still possible to improve performance by a large margin before being strictly limited by the heuristic.

The performance of the transfer method depends a lot on the size of the transferred network. Since the Czech DeriNet is an order of magnitude larger than the other networks, the gold scores for networks created by using it as a base are the highest ones, but even these don't match more than 2.5% of relations from the gold-standard data.

The precision of the constructed networks is also influenced by the translation quality. The alignment data trained on the deu-fra pair (in both directions) has many incorrect alignments. This doesn't affect the oracle score, since the correct translations will generally be found, but the wide distribution of the probability mass hurts the actual algorithm, which is unable to distinguish plausible and implausible translations.

		Internal scores [%]				Gold scores [%]		
Lang	Size	Prec.	Rec.	F1	Acc.	Rec.	F1	Acc.
deu	15765	14	34	20	42	5.7	8.1	20
eng	12819	30	47	37	47	18	22	30
fas	13877	22	82	35	31	11	15	13
fin	1616	16	10	13	38	2.4	4.2	15
fra	3767	27	66	39	35	7.9	12	11
hun	6158	41	35	37	23	8.7	14	7.7
ita	3465	9.3	57	16	27	24	13	23
kaz	522	58	37	46	30	17	27	16
por	2428	37	43	40	36	25	30	27
rus	8708	12	16	14	28	0.8	1.5	3.6
spa	13702	24	76	37	42	8.3	12	15
tur	2829	28	75	40	38	17	21	19

Table 5: Evaluation scores of the synthetic training data. Accuracy, precision and recall are in percent, size indicates the number of predicted relations.

		Internal scores [%]				Gold scores [%]		
Lang	Size	Prec.	Rec.	F1	Acc.	Rec.	F1	Acc.
deu	15	20	0.03	0.06	67	0.01	0.01	33
eng	6843	40	28	33	55	12	18	36
fas	186	15	0.08	0.51	35	0.08	0.16	14
fin	3864	16	40	23	28	6.1	8.8	11
fra	4241	24	80	37	32	8.1	12	9.7
hun	4765	54	30	38	24	8.5	15	8.0
ita	3816	9.3	72	16	23	28	14	20
kaz	830	57	70	63	46	30	39	24
por	1908	49	42	45	47	25	33	35
rus	11508	14	29	19	25	1.3	2.3	3.3
spa	13961	23	77	36	41	8.1	12	15
tur	2961	30	80	44	38	19	23	20

Table 6: Evaluation scores of the neural extension model applied on word-formation networks obtained by the cross-lingual embedding model. Accuracy, precision and recall are in percent, size indicates the number of predicted relations.

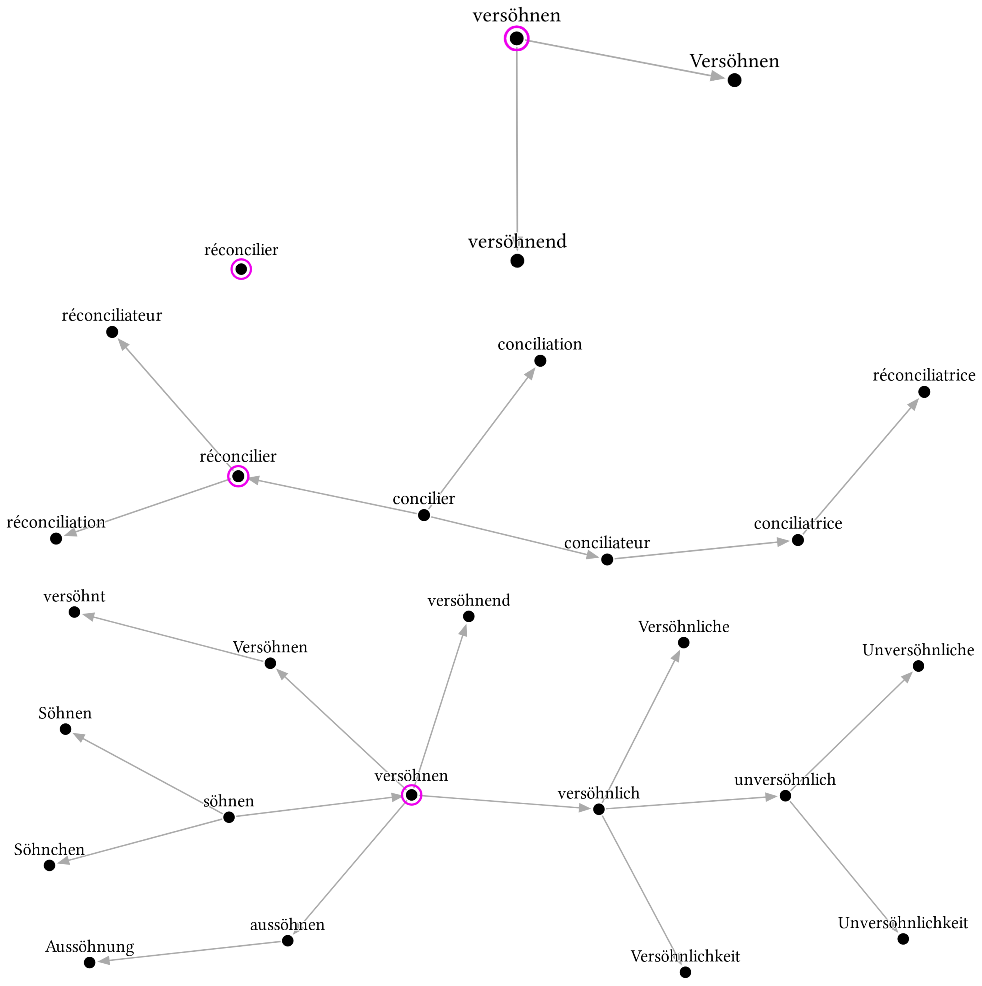

Figure 2: Word-formation networks generated by the machine learning expansion of the transferred networks, showing the family of lexeme *to reconcile* (encircled) for four of the six language pairs. Top: deu-fra (single lexeme) and fra-deu, middle: ces-fra, bottom: ces-deu.

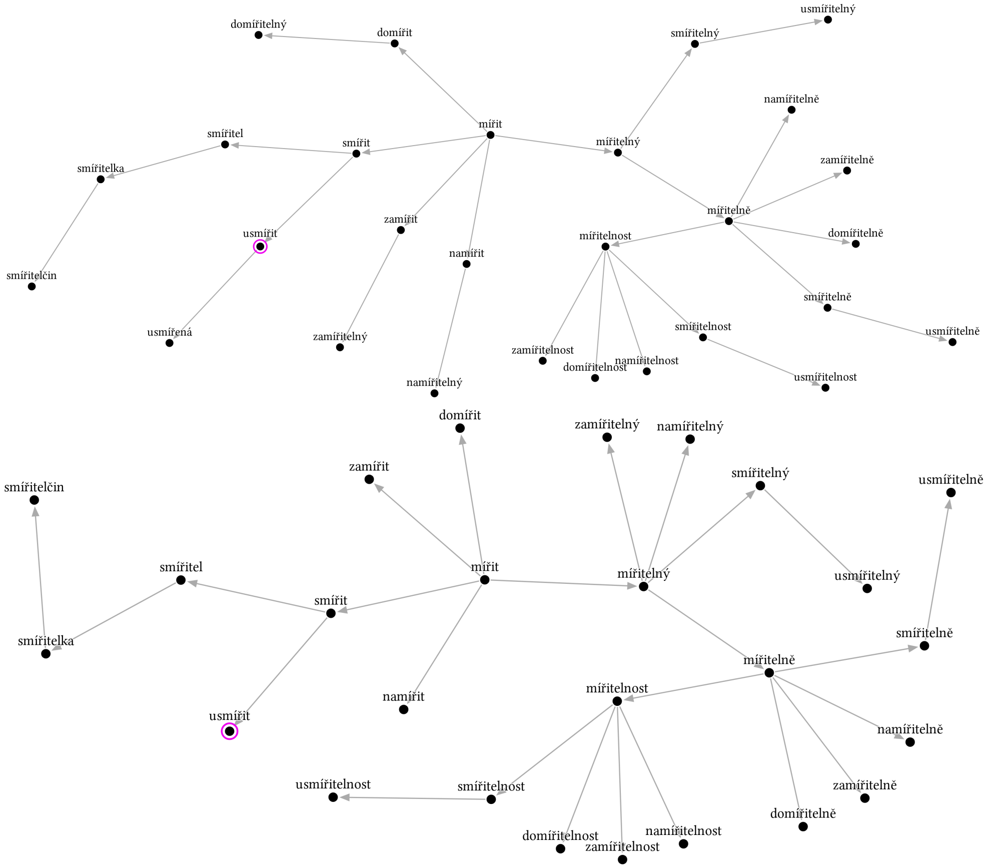

Figure 3: Word-formation networks generated by the machine learning expansion of the transferred networks, showing the family of lexeme *to reconcile* (encircled) for the other two of the six language pairs. Top: deu-ces, bottom: fra-ces.

The machine learning extension method provides a way of generalizing the output of the transfer method, as it learns frequent affixal patterns from the transferred data and applies them to a larger lexicon, omitting infrequent (often spurious) patterns. As seen in the second part of Table 3, this results in increased precision on the networks transferred to French and German, where the gold standard data consists of relatively few selected paradigms and therefore skews towards fewer, more productive patterns. The results on the Czech data, which is more varied, still reach precision comparable to the transferred networks we train on. Recall increases in all cases, even when compared to the "internal" scores, which are more favorable to the transferred networks. Due to this large increase, F1-score also increases. Sample outputs of the machine learning method can be seen in Figures 2 and 3.

The oracle scores for the transfer are in Table 4. The scores are influenced by the ratio of sizes of the word-formation networks used for transfer and evaluation; transferring a large network and evaluating on a smaller one gives an advantage in recall in comparison to the opposite scenario, simply because a larger source network offers more options to select from after transfer. The error causes listed in the table correspond to the sources of error in recall as categorized in Section 5.2.

For all language pairs, most of the errors (50-90%) are attributable to the first cause, where the gold data contains untranslatable lexemes. For the pairs that translate to Czech, this is again explainable by the size and composition of its DeriNet network, which contains many unattested lexemes – finding rare lexemes such as *přeskočitelnost* ("skippability") in the parallel data is unlikely. This is also the reason why the networks obtained through the machine learning expansion have better scores than the oracle of the transfer algorithm. The transfer lexicon is limited to the lexemes found in the parallel data, whose source-side alignments are found in the source word-formation network, and for evaluation purposes, we further limit the lexicon to lexemes from the gold-standard data. The machine-learning pipeline uses the gold-standard lexicon directly, eliminating the "No child trans" class of errors entirely.

Additionally, transfers of networks to German have higher accuracy than transfers to French, even though the recall is comparable. This is because the German network, DErivBase, contains many compounds, which don't have their parents annotated and are listed as unmotivated. These are counted in the accuracy scores (the definition of oracle score above considers missing relations to be always correctly recognized) but do not contribute to recall of relations. The unmotivated words are also the reason behind the fact that the fra-deu pair has higher accuracy than ces-deu, despite having lower recall – fewer relations are translated, resulting in more unmotivated words being correct.

Scores of the cross-lingual embedding model are in Table 5. The model produces results with roughly comparable internal scores (the F1 score on German is 31% for transfer vs. 20% for NNs, while on French it is 27% vs. 39%), but significantly higher gold scores, due to the networks themselves being several times larger. It does not, however, attain scores on par with the machine learning extension method.

The neural extension model exposes a large flaw in the training regime of the neural network. The network is optimized towards minimizing cross entropy between the predicted and gold binary classifications of individual word-formation relations, i.e. it maximizes gold accuracy. As seen in Table 6, the model generally succeeds at that, even though the scores don't necessarily increase on every language (there is a small but significant decrease on Finnish, French and Italian). However, this apparent improvement entirely destroys the usefulness of the model on German and Farsi, because the increase in accuracy is driven by correctly classifying unrelated lexemes at the expense of related ones, causing the recall to go to zero. Even then, the accuracy is still worse than the machine learning extension method. One solution could be a training objective focused on maximizing gold F1 score, or an improved model of word formation which doesn't predict individual relations, but focuses on larger units, e.g. whole instances of a word-formation paradigm or whole word-formation families.

7. Conclusion

In this paper, we presented a two cross-lingual methods for creating word-formation networks – one transfers an existing network using a word-translation lexicon induced from word alignments, the other one uses a neural network with pretrained cross-lingual word embeddings. The transferred small networks are then expanded by either extracting paradigms using statistical machine learning and applying them to a larger set of lexemes, or by bootstrapping the neural network on the small word-formation networks in a cross-lingual fashion. The resulting word-formation networks generally show moderately high precision and good recall.

Acknowledgments

This work was supported by the Grant No. START/HUM/010 of Grant schemes at Charles University (reg. No. CZ.02.2.69/0.0/0.0/19 073/0016935). It has been using language resources developed, stored, and distributed by the LINDAT/CLARIAH CZ project (LM2018101).

The OpenSubtitles corpus was kindly provided by `http://www.opensubtitles.org/`.

Bibliography

Baranes, Marion and Benoît Sagot. A Language-independent Approach to Extracting Derivational Relations from an Inflectional Lexicon. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2793–2799, Reykjavik, Iceland, May 2014. European Language Resources Association (ELRA). URL `http://www.lrec-conf.org/proceedings/lrec2014/pdf/379_Paper.pdf`.

Batsuren, Khuyagbaatar, Gabor Bella, and Fausto Giunchiglia. CogNet: A Large-Scale Cognate Database. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3136–3145, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1302. URL `https://www.aclweb.org/anthology/P19-1302`.

Batsuren, Khuyagbaatar, Omer Goldman, Salam Khalifa, Nizar Habash, Witold Kieraś, Gábor Bella, Brian Leonard, Garrett Nicolai, Kyle Gorman, Yustinus Ghanggo Ate, Maria Ryskina, Sabrina Mielke, Elena Budianskaya, Charbel El-Khaissi, Tiago Pimentel, Michael Gasser, William Abbott Lane, Mohit Raj, Matt Coler, Jaime Rafael Montoya Samame, Delio Siticonatzi Camaiteri, Esaú Zumaeta Rojas, Didier López Francis, Arturo Oncevay, Juan López Bautista, Gema Celeste Silva Villegas, Lucas Torroba Hennigen, Adam Ek, David Guriel, Peter Dirix, Jean-Philippe Bernardy, Andrey Scherbakov, Aziyana Bayyrool, Antonios Anastasopoulos, Roberto Zariquiey, Karina Sheifer, Sofya Ganieva, Hilaria Cruz, Ritván Karahóǧa, Stella Markantonatou, George Pavlidis, Matvey Plugaryov, Elena Klyachko, Ali Salehi, Candy Angulo, Jatayu Baxi, Andrew Krizhanovsky, Natalia Krizhanovskaya, Elizabeth Salesky, Clara Vania, Sardana Ivanova, Jennifer White, Rowan Hall Maudslay, Josef Valvoda, Ran Zmigrod, Paula Czarnowska, Irene Nikkarinen, Aelita Salchak, Brijesh Bhatt, Christopher Straughn, Zoey Liu, Jonathan North Washington, Yuval Pinter, Duygu Ataman, Marcin Wolinski, Totok Suhardijanto, Anna Yablonskaya, Niklas Stoehr, Hossep Dolatian, Zahroh Nuriah, Shyam Ratan, Francis M. Tyers, Edoardo M. Ponti, Grant Aiton, Aryaman Arora, Richard J. Hatcher, Ritesh Kumar, Jeremiah Young, Daria Rodionova, Anastasia Yemelina, Taras Andrushko, Igor Marchenko, Polina Mashkovtseva, Alexandra Serova, Emily Prud'hommeaux, Maria Nepomniashchaya, Fausto Giunchiglia, Eleanor Chodroff, Mans Hulden, Miikka Silfverberg, Arya D. McCarthy, David Yarowsky, Ryan Cotterell, Reut Tsarfaty, and Ekaterina Vylomova. UniMorph 4.0: Universal Morphology. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 840–855, Marseille, France, June 2022. European Language Resources Association. URL `https://aclanthology.org/2022.lrec-1.89`.

Bojanowski, Piotr, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. ISSN 2307-387X. doi: 10.1162/tacl_a_00051.

Chu, Yoeng-Jin and Tseng-Hong Liu. On the shortest arborescence of a directed graph. *Science Sinica*, 14:1396–1400, 1965.

Church, Kenneth Ward. Char_align: A Program for Aligning Parallel Texts at the Character Level. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 1–8, Columbus, Ohio, USA, June 1993. Association for Computational Linguistics. doi: 10.3115/981574.981575. URL `https://aclanthology.org/P93-1001`.

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.747. URL `https://aclanthology.org/2020.acl-main.747`.

Dyer, Chris, Victor Chahuneau, and Noah A. Smith. A Simple, Fast, and Effective Reparameterization of IBM Model 2. In *Proceedings of the 2013 Conference of the North American*

Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 644–648, Atlanta, Georgia, June 2013. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/N13-1073`.

Gupta, Prakhar and Martin Jaggi. Obtaining Better Static Word Embeddings Using Contextual Embedding Models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5241–5253, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.408. URL `https://aclanthology.org/2021.acl-long.408`.

Hämmerl, Katharina, Jindřich Libovický, and Alexander Fraser. Combining Static and Contextualised Multilingual Embeddings. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2316–2329, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.182. URL `https://aclanthology.org/2022.findings-acl.182`.

Hathout, Nabil. Acquisition of morphological families and derivational series from a machine readable dictionary. In *Proceedings of the 6th Décembrettes.*, Cascadilla Proceedings Project, pages 166–180, Bordeaux, France, 2008. Cascadilla. URL `https://hal.archives-ouvertes.fr/hal-00382808`.

Hathout, Nabil and Fiammetta Namer. Démonette, A French Derivational Morpho-Semantic Network. *Linguistic Issues in Language Technology*, 11:125–162, 2014. doi: 10.33011/lilt.v11i.1369.

Junczys-Dowmunt, Marcin, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. Marian: Fast Neural Machine Translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-4020. URL `https://aclanthology.org/P18-4020`.

Kyjánek, Lukáš. Morphological Resources of Derivational Word-Formation Relations. Technical Report ÚFAL TR-2018-61, ÚFAL MFF UK, Praha, Czechia, 2018. URL `http://ufal.mff.cuni.cz/techrep/tr61.pdf`.

Kyjánek, Lukáš, Zdeněk Žabokrtský, Magda Ševčíková, and Jonáš Vidra. Universal Derivations Kickoff: A Collection of Harmonized Derivational Resources for Eleven Languages. In *Proceedings of the Second International Workshop on Resources and Tools for Derivational Morphology (DeriMo 2019)*, pages 101–110, Praha, Czechia, 2019. ÚFAL MFF UK. ISBN 978-80-88132-08-0.

Lango, Mateusz, Zdeněk Žabokrtský, and Magda Ševčíková. Semi-Automatic Construction of Word-Formation Networks. *Language Resources and Evaluation*, 55(1):3–32, 2021. ISSN 1574-020X. doi: 10.1007/s10579-019-09484-2.

McDonald, Ryan, Slav Petrov, and Keith Hall. Multi-Source Transfer of Delexicalized Dependency Parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 62–72, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics. URL `https://aclanthology.org/D11-1006`.

Musil, Tomáš, Jonáš Vidra, and David Mareček. Derivational Morphological Relations in Word Embeddings. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 173–180, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-4818. URL https://aclanthology.org/W19-4818.

Nivre, Joakim et al. Universal Dependencies v1: A Multilingual Treebank Collection. In *Proceedings of the 10th International Conference on Language Resources and Evaluation*, pages 1659–1666. ELRA, 2016.

Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Andreas Müller, Joel Nothman, Gilles Louppe, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Rosa, Rudolf and Zdeněk Žabokrtský. Unsupervised Lemmatization as Embeddings-Based Word Clustering, 2019.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1009. URL https://aclanthology.org/P16-1009.

Simard, Michel, George F. Foster, and Pierre Isabelle. Using cognates to align sentences in bilingual corpora. In *Proceedings of the Fourth Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages*, Montréal, Canada, jun 1992. URL https://aclanthology.org/1992.tmi-1.7.

Straka, Milan and Jana Straková. Tokenizing, POS Tagging, Lemmatizing and Parsing UD 2.0 with UDPipe. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 88–99, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/K17-3009. URL http://www.aclweb.org/anthology/K/K17/K17-3009.pdf.

Svoboda, Emil and Magda Ševčíková. Word Formation Analyzer for Czech: Automatic Parent Retrieval and Classification of Word Formation Processes. *The Prague Bulletin of Mathematical Linguistics*, 118:55–73, 2022. ISSN 0032-6585. doi: 10.14712/00326585.019.

Tiedemann, Jörg. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf.

Zeller, Britta, Jan Šnajder, and Sebastian Padó. DErivBase: Inducing and Evaluating a Derivational Morphology Resource for German. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1201–1211, Sofia, Bulgaria, 2013. URL http://www.aclweb.org/anthology/P13-1118.

Zhang, Jiajun and Chengqing Zong. Exploiting Source-side Monolingual Data in Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Lan-*

guage Processing, pages 1535–1545, Austin, Texas, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1160. URL `https://aclanthology.org/D16-1160`.

Zhang, Yuan, David Gaddy, Regina Barzilay, and Tommi Jaakkola. Ten Pairs to Tag – Multilingual POS Tagging via Coarse Mapping between Embeddings. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1307–1317, San Diego, California, June 2016. Association for Computational Linguistics. doi: 10.18653/v1/N16-1156. URL `https://aclanthology.org/N16-1156`.

Žabokrtský, Zdeněk, Magda Ševčíková, Milan Straka, Jonáš Vidra, and Adéla Limburská. Merging Data Resources for Inflectional and Derivational Morphology in Czech. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 1307–1314, Paris, France, 2016. European Language Resources Association. ISBN 978-2-9517408-9-1.

Address for correspondence:
Jonáš Vidra
`vidra@ufal.mff.cuni.cz`
Institute of Formal and Applied Linguistics
Faculty of Mathematics and Physics,
Charles University
Malostranské náměstí 25
118 00 Praha 1, Czech Republic