Semi-supervised machine learning methods for developing derivational networks

Ph.D. Thesis Proposal

Jonáš Vidra
Faculty of Mathematics and Physics, Charles University,
Malostranské náměstí 25
118 00 Prague, Czech Republic
vidra@ufal.mff.cuni.cz

Abstract

In this thesis proposal, we present a method for creating word-formation networks by transferring information from another language. The proposed algorithm utilizes an existing word-formation network and parallel texts and creates a low-precision and moderate-recall network in a language, for which no manual annotations need to be available. We then extend the coverage of the resulting network by using it to train a machine-learning method and applying the resulting model to a larger lexicon, obtaining a moderate-precision and high-recall result. The approach is evaluated on French, German and Czech against existing word-formation networks in those languages. We also report on experiments with modeling semantics of word-formation relations using word embeddings and present our plans for further development.

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 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.

The proposed thesis aims to explore methods from the opposite part of the scale: methods that require little or no human input or in-language annotations of word-formation relations. Instead, we seek to utilize a combination of unsupervised machine learning and transfer of existing word-formation networks from other languages. We hope that it is possible to emulate the successes of transfer learning methods used for lemmatization (Rosa and Žabokrtský, 2019b), part-of-speech tagging (Zhang et al., 2016) or syntactic parsing (McDonald et al., 2011); tasks, which are in many ways similar to ours. The resulting methods should allow for a cheap and rapid creation of word-formation networks for many languages, although potentially at a cost of lower quality.

1.1 Why is word formation difficult

There are many issues which make the task of correctly recognizing word-formational relations difficult. Chiefly, for derived lexemes, it is difficult to recognize the extent of allomorphy. Some derivatives accumulate so many morphological, phonological and orthographical changes at once that the relation becomes almost opaque (e.g. the Czech *sejmout* (“to take off”) → *sňatý* (“taken off”)). In other cases, misanalyzing allomorphy may lead to incorrect derivation (recognizing the incorrect derivation of *hůl* (“cane”) → *holka* (“girl”) by analogy with e.g. *stůl* (“table”) → *stolek* (“small table”) or *Eiffelův* (“Eiffel’s”) → *Eiffelovka* (“Eiffel Tower”)).

Very often, it is even difficult to recognize whether a lexeme is derived or non-derived. One useful feature is the length of its lemma, but it can be misleading e.g. in the case of loanwords, which are considered to be non-derived despite their length. Also, the degree of lexicalization may result in disagreement even among native speakers, with some e.g. deriving *kolej* (“rail”) from *kolo* (“wheel”) and others considering it non-derived. Such disagreements can also be caused by “folk etymology”, where different people interpret the derivations differently, e.g. *mísa* (“bowl”) can be derived from *mísit* (“to mix”) or considered a non-derived loanword from Latin *mēnsa* (“table”).

The direction of derivation is also a contested issue for many lexemes, especially for neoclassical formations and internationalisms. A useful feature for determining the direction can be the ratio of

corpus frequencies of the two words, but in many cases, semantic analysis is required (Panocová, 2017).

Another problem is delineating derivation from other processes, such as inflection. The research on word-formation is usually performed only on lemmas, but sometimes it is necessary to also consider inflected word forms. For example, according to long-standing Czech linguistic tradition, negation is considered to be an inflectional process, but negated words are used to form derivatives (e.g. nejistý-znejistět).

1.2 Scope of our current experiments

The experiments presented in this thesis proposal focus on one-to-one relations between lexemes. We omit compounding altogether, because apart from the fact that it requires correct identification of two or more parents instead of just one, it also often combines with derivation in a two-step process, such as in the Czech compound *pes* ("dog") + *vést* ("to lead") → *psovod* ("dog handler"), where the lexeme *vést* is first derived to form the unattested **vod*. This can also happen with pure derivation (some theories analyze e.g. the circumfixation *les* ("forest") → *polesí* ("forest district") as double derivation (Bauer, 2014, p. 127)), but it is rarer.

Therefore, we 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 word-formational family. This is done to simplify the task and to get a workable proof-of-concept, which we will expand on in the future.

Moreover, although we aim to produce algorithms and models which would be able to create word-formation networks for any language with mostly concatenative morphology and written in an alphabetic script, we currently focus on French, German and Czech, because we can these are among the few languages for which a large, high-quality word-formation network already exists (Kyjánek, 2018). These existing networks serve a dual role as data for transfer on the source side, and evaluation datasets on the target side.

1.3 Overview of this proposal

Other researchers have shown that a word-formation network can be created from scratch using pattern-mining approaches utilizing orthographic similarities and differences between words, or using word embeddings as a proxy for semantic similarity. We outline several possible methods in Section 2 below. Our work tries to extend and improve upon these results by transferring the necessary information from another language and we especially aim at supporting under-resourced and small languages. So far, the low-resource setting is merely simulated, because evaluating our approach on existing resources allows for more rapid development and verification of hypotheses.

The experiments we've conducted so far attempt to translate word-formation networks using parallel texts and off-the-shelf tools for tokenization and lemmatization. The main idea behind our methods is that 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.

We model this sharing by constructing a word-translation lexicon using word alignments gathered from parallel texts. This lexicon is used to translate an existing word-formation network to create a list of potential word-formation relations ranked by probability. By further filtering the potential relations by orthographic distance and selecting best-scoring parents using a maximum spanning tree algorithm, we obtain a moderate-precision and low-recall set of word-formation relations. This model is described in Section 3.

In Section 4, we describe two experiments that aim at further improving the transfer algorithm by including semantic information from word embeddings. The recall of the transfer algorithm can be improved by extracting the discovered word-formation paradigms using a statistical machine-learning method and finding more examples of them across the lexicon, as shown in Section 5.

In section 6, we establish the methods and metrics used for evaluating the quality of word-formation networks, and we analyze our results in Section 7. Appendix A contains samples of outputs of the transfer method.

2 Related work

Several unsupervised methods of creating word-formation networks have been proposed before. 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 *worse* is a comparative form of the lemma *bad*, we can link the lexeme *worsen* to *bad* using the rule *-e* → *-en*.

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.

A method utilizing cosine distance between neural-network word embeddings was used by Üstün and Can (2016) to construct an implicit word-formation network as an intermediate step in morphological segmentation. These results show that word embeddings contain some information about derivational relations, which is further supported by the fact that it is (to some extent) possible to use them to automatically differentiate between derivational and inflectional relations in an unsupervised setting (Rosa and Žabokrtský, 2019a). In our prior research, we analyzed word embeddings to show that words created through similar word-formation processes have similar embedding differences (Musil et al., 2019); however, we did not use these results to construct a network out of word-embedding data.

Transfer learning is a general method useful for improving results in under-resourced settings by utilizing knowledge gained in different but similar settings. For example, it is possible to transfer delexicalized syntactic parser models between similar languages and get better results than fully unsupervised parsers, and the advantage improves when combining multiple models transferred from different sources (McDonald et al., 2011). It is also possible to transfer information between different tasks in the same language, not just between identical tasks for different languages. For example, many neural network models for processing natural language benefit from using pretrained word-embeddings, even though the embeddings are trained on a different task.

Generally, the more similar the settings, the more successful the transfer is – transferring models between languages with similar grammar gives a better result than using a distant source and target. The similarity can be approximated using metrics such as KL_{cpos^3} based on trigrams of part-of-speech tags (Rosa, 2018). Multi-source transfer has the additional advantage that it can utilize information even from more distant sources: Chen et al. (2019) demonstrate that an adversarially-trained neural network can learn common features for individual language pairs and efficiently transfer information in a massively multilingual setup without any parallel data or target-side annotations.

A useful tool for transfer learning is multilingual word embeddings. They allow us to train a model on a high-quality resource available for one language, and easily apply it on under-resourced languages, as long as the word-embedding spaces used for representing language data are similar enough. As an example, a successful transfer of a part-of-speech tagger using multilingual embeddings was performed by Zhang et al. (2016). Creating the multilingual word embeddings generally requires some amount of parallel data, but the amount needed is not as large as for training alignments (Mikolov et al., 2013b; Gouws et al., 2015).

Multilingual embeddings can also be used to infer word translation dictionaries, as shown by Artetxe et al. (2017). Another possible approach to creating a word translation dictionary is shown by Lample et al. (2018), who use adversarial neural networks with character-level embeddings to align pre-trained word embeddings for multiple languages.

We have previously shown that it is possible

to create a word-formation network using a neural network trained in a supervised way (Vidra, 2018). The neural network used a character-level encoder-decoder architecture with attention and achieved 90% accuracy in predicting the correct parent when trained on 600 000 derivational relations from DeriNet (Vidra, 2018, p. 49). The scores could be further improved by constraining the network to only generate words from the lexicon, instead of sometimes producing novel or erroneous words due to the character-by-character generation. However, the models produced by this architecture are not easily transferable between languages, as they use character-level embeddings, which are more language-specific than word embeddings, and they require a large amount of training data (at least 200 000 relations are necessary for reliable results, making even the largest word-formation networks, such as French Démonette with 96 000 or German DErivBase with 51 000 relations, too small).

3 Transfer algorithm

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 several target-side similarity metrics. Conceptually, the source lexeme C is first backtranslated 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 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.

The score of each potential relation is obtained as a weighted arithmetic mean of one minus the relative edit distance between C and P and their translation score. 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.

We define the translation score of C and P as $\sum_{\forall C',P'} \frac{|\text{align}(C,C')|}{\sum_{\forall x}|\text{align}(C,x)|} \cdot 0.5^{\text{dist}(C',P')} \cdot \frac{|\text{align}(P',P)|}{\sum_{\forall x}|\text{align}(P',x)|}$, where $|\text{align}(x,y)|$ denotes the number of alignments between lexemes x and y seen in the aligned data and $\text{dist}(C',P')$ denotes the number of relations on the shortest path from C' to P' in the source network.

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 using e.g. grid search or gradient descent. 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. Therefore, we empirically set the weight of the edit distance to 5, the weight of the translation to 1 and the threshold to 0.8.

The algorithm described above has been introduced in a paper accepted to the DeriMo 2021 workshop (Vidra, 2021).

4 Using embeddings to model word-formation semantics

The transfer algorithm selects the best derivational edges based on two scores, corresponding to ortho-

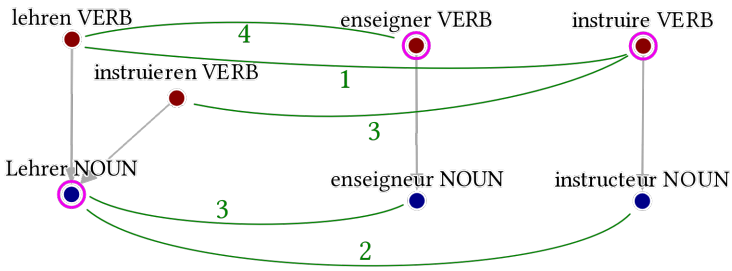

Figure 1: An example of finding a parent for the German lexeme *Lehrer* ("teacher") by transferring information from a French word-formation network, with word-formation relations in grey and alignments in green. *Lehrer* is aligned to *enseigneur* ⅗ times, which has *enseigner* available through 1 relation, to which *lehren* is aligned ¼ times. *Lehrer* is also aligned to *instructeur* ⅖ times, which has *instruire* available through 1 relation, to which *lehren* is aligned ¼ times and *instruieren* ¾ times. The translation score of *lehren* → *Lehrer* 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 ⅔ 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 $\frac{0.15+5\cdot(1-8/11)}{6} = 0.252$.

graphic similarity and translation context similarity. Upon inspection of the outputs of the transfer algorithm, we can observe two common types of errors produced when one score overpowers the other one, with authentic examples taken from the results:

1. Sometimes, the model connects semantically unrelated words with similar spelling, such as *Beruf* ("profession") and *liberal* ("liberal"), because the orthographic distance is relatively small (they both have *ber* in common).

2. In other cases, the model connects words which are not word-formationally related in the target language, such as *anbauen* ("to build on, to plant") and *kultivieren* ("to cultivate"), because they share common translations in the source language.

The first problem is caused chiefly by the fact that the optimal setting of the weights used to combine the translation and orthographic similarity scores diverges – increasing the weight of the orthographic similarity produces a marginally better model when evaluated against gold-standard data. This way, the model reduces the second problem while making the first one worse. We attempted to solve the issue by adding two other scores, both representing different aspects of semantic similarity. In theory, the additional information should help the model detect large semantic shifts between unrelated words and discard such wrong relations. These two scores are described in the following two subsections.

4.1 Modeling semantic distance

One score models semantic similarity of the two lexemes simply as their cosine similarity, based on the hypothesis that word-formationally related lexemes share a common root and therefore should be semantically similar. This hypothesis is true in practice – derivationally related lexemes have a higher cosine similarity between their embedding vectors than indirectly related or unrelated ones, as shown in Figure 2. Note that the relation between word-formational closeness and semantic similarity is a one-way implication. Many lexemes, e.g. synonyms, have similar semantics even without any common word-formational ancestry.

We've tested several types of embeddings: Word2Vec (Mikolov et al., 2013a) using publicly available vectors from the Gensim-data project (Řehůřek and Sojka, 2010) pretrained on unlemmatized data and vectors manually trained on lemmatized data from the CoNLL 2017 parsing shared task (Ginter et al., 2017); FastText (Bojanowski et al., 2017), again using both publicly available pretrained ones (Grave et al., 2018) and vectors we trained ourselves on lemmatized texts, and pretrained Multilingual BERT (Devlin et al., 2019). Since the BERT embedding vectors for a given word depend on its context, we've obtained concrete word embeddings by averaging the context-dependent embeddings of that word's word pieces across a corpus.

Each of the three types of embeddings has advantages and disadvantages; in the next few paragraphs, we will list those that are of particular interest to us.

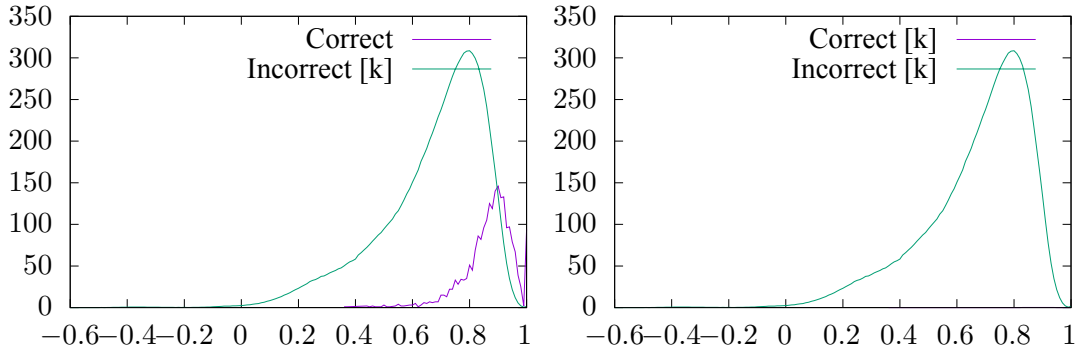

Figure 2: A histogram of FastText embedding cosine similarity between parents and children of relations proposed by the transfer algorithm on the Czech → German transfer pair; with curves for correct relations (found in the gold-standard data) and incorrect ones plotted separately. In the left plot, the negative examples are scaled down by a factor of 1 000. On the right, both curves have an identical scale; the curve for positive elements is too low to be visible.

The Word2Vec embeddings are trained on whole words using only their sentence-level context, while FastText and BERT break words down into subwords. Although subwords don't correspond to morphemes, syllables, phonemes or any other linguistically meaningful unit, this still gives the latter two methods partial access to the orthographic composition of the word in question. This is both a benefit and a curse. The advantage of subwords is that the model can produce an embedding of any word, including previously unseen ones (as long as they are composed using only characters from the training alphabet). The disadvantage is that these embeddings are created by averaging embeddings of individual subwords. If those subwords happen to correspond (even loosely) to morphemes, the difference between two words sharing the same stem will be wholly determined by the embeddings of the affixes. For example, assuming subword splits are found at "·", the differences in FastText vectors of *skříň* ("wardrobe") – *skříň·ka* ("cabinet") (diminution), *spoj* ("connection") – *spoj·ka* ("connector / conjunction / messenger / clutch") (instrument), *baron* ("baron") – *baron·ka* ("baroness") (grammatical gender change) and *čelen* ("bowsprit") – *čelen·ka* ("headband") (distant relation through *čelo* ("forehead")) will be identical. This would prevent us from distinguishing correct and incorrect derivations based on their semantics.

This model of semantics of word-formation relations suffers from one large issue – it considers semantically close pairs of words to be more probably related than distant words. But this reflects reality only in cases where the only difference between the derivational parent and child is mostly grammatical, such as conversion, where speakers create the child word to better fit a particular sentence context. Usually, the reason for using word formation in the first place is to create a semantically different word that is related to its parent only in some aspects. For example, the adjective *vodovatý* ("watery") describes a thing that shares some properties with water, but it is not water. Therefore, we expect that the two lexemes connected by a word-formation relation will exhibit systematic semantic shifts. We describe how we try to exploit these shifts in the following subsection.

4.2 Modeling semantics of affixal patterns

Note: The contents of this subsection are a result of collaboration with colleagues Tomáš Musil (Musil et al., 2019) and Jan Bodnár (paper in preparation).

One nice property shared by all types of embeddings we tested is, that they (to some degree) preserve semantic differences between words. For example, the vector difference between the embeddings for the words *Berlin* and *Germany* should be roughly similar to the difference between *Paris* and *France* (Mikolov et al., 2013a). We have previously verified that this property extends to derivational relations as well and that it is even possible to use unsupervised clustering methods to distinguish between different semantic types of derivational relations (Musil et al., 2019).

This means that the embedding difference between *Lehrer* ("teacher$_{\text{MASC}}$") → *Lehrerin*

("teacher$_{FEM}$") and *Maler* ("painter$_{MASC}$") → *Malerin* ("painter$_{FEM}$") should be similar, but at the same time distinct from the difference between e.g. *Fahrer* ("driver") → *Beifahrer* ("front passenger") or *schwer* ("heavy") → *Schwerin* ("Schwerin, city in Mecklenburg-Vorpommern"). A caveat is that the expected difference is sometimes identical even for words which are not word-formationally related, because the difference between e.g. *Lehrer* → *Lehrerin* is the same as for *Mann* ("man") → *Frau* ("woman"). We assume that such spurious relations would be rejected by their large orthographic distance in a different component of the combined model.

We exploit the systematicity by scoring how well the embedding difference of each potential word-formation relation corresponds to the expected difference. To do this, we first extract the affixal change pattern (see below) of each potential word-formation relation identified by the transfer algorithm. Extracting the exact morpheme difference is a difficult task (Vidra, 2018), but we approximate it by four steps: First, we lowercase the lemmas of both lexemes. Second, we find the longest common contiguous substring of the lemmas of the two lexemes and consider the leftover substrings to be the affixes. Third, we gather these affixal patterns for all relations. Fourth, we classify each relation as belonging to all affixal patterns that match the ends of the lemmas, regardless of what's left over. For example, the relation *Kampf* ("a fight") → *kämpfen* ("to fight") has the longest common contiguous substring *mpf* and affixal pattern *ka-* → *kä-* + *-en*, but it is also assigned to the pattern λ → *-en* found e.g. in *Bad* ("a bath") → *baden* ("to bathe") despite the umlaut change.

After we have a list of matching relations for each affixal pattern, we compute the mean of all the differences and consider it to be the prototypical semantic difference embodied by the affixal pattern. A semantic similarity score of any relation is found as the cosine similarity between the mean of its affixal pattern and the embedding difference of the relation.

The method of assignment relations to patterns results in any given relation potentially being assigned to multiple patterns. We do this to avoid most issues with allomorphy, resulting in more precise estimates of the means – without this, mean estimates for many patterns would consist of only a single example from the corpus.

The obvious downside is that many patterns will get mixed up with semantically unrelated relations. For example, *pošta* ("post (office)") → *poštovní* ("postal") has the affixal pattern *-a* → *-ovní*, but it is also assigned to *-a* → *po-* + *-í* found in e.g. *voda* ("water") → *povodí* ("drainage basin") and λ → *po-* found in e.g. *chodit* ("to walk") → *pochodit* ("to go well").

To reduce the impact of these spurious assignments, we can split a pattern into multiple means by clustering the assigned differences. The clustering is done using K-Means (Lloyd, 1982) with K ranging from 1 to 30, and selecting the K which minimizes the Akaike information criterion (Akaike, 1974). This generally results in 1-3 clusters per pattern, with several patterns with a high degree of spuriousness (such as the *-a* → *-í* mentioned above) being split into more. The semantic similarity score of a relation is then calculated against the closest cluster in the pattern.

5 Expansion through machine learning

The word-formation network obtained via cross-lingual transfer covers only lexemes with alignments, i.e. high-frequency ones. Therefore, it is desirable to increase coverage of lower-frequency parts of the lexicon and lexemes not seen in the parallel data. We perform this by extracting affixal patterns from the transferred network and applying them across the data.

To do this, 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. 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 classifying all pairs of lexemes found in the dataset is too computationally expensive, we only sample pairs of lexemes that are near one another when the dataset is lexicographically sorted by lemma, in both prograde and retrograde fashions. The prograde-sorted list puts lemmas with common beginnings near each other, meaning that pairs of words differing only in short suffixes will be selected for classification. The retrograde-

sorted one does the same with lemmas differing only in a short prefix.

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.

Empirically, looking at a window of ±5 lexemes catches 85% of all possible derivational relations in DErivBase, and ±10 catches 90%. On Démonette, 96% of derivations are within ±5 and 98% are within a ±10 window. In DeriNet, a window of ±5 contains 85% of all relations and ±10 contains 90%.

We've 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 its consistent evaluation performance.

6 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 non-derived lexemes to all gold lexemes. Therefore, the precision and recall don't take into account non-derived 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 transferred 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 and only measure scores on the intersection of the two lexicons. 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.

6.1 Baselines

To better evaluate the quality of the results, it is helpful to establish upper and lower bounds of reasonably achievable scores. For the lower bound, we came up with 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 non-derived. 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.

The upper bounds can be estimated by inter-annotator agreement on the task of derivational parent assignment, which we performed during the development of DeriNet 1.0 (Vidra, 2015). To measure it, we had two annotators with linguistic background assign parents to 1 000 lexemes sampled uniformly randomly from DeriNet 1.0, with the possibility of listing multiple parents if the lexeme in question was ambiguous (e.g. *unlockable* can be derived from either *lockable* or *unlock*; or *Karlův* ("Charles's") from either *Karel* or *Karl*). The annotations matched in at least one parent in 87.8% of lexemes.

6.2 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

```
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)
13              / (true_positive + true_negative + false_negative))
14 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`.

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).

6.3 Experimental setting

For this proposal, we conducted experiments on Czech, French and German, which are all languages with existing word-formation networks suitable for transfer – 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. For ease of use, we used their versions available in the UDer 1.0 collection (Kyjánek et al., 2019), which have been converted to a common format at a slight loss of information. 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 1.

7 Evaluation Results

As can be seen in Table 2, 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. Samples of outputs are in Appendix A.

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

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

		Size		Internal scores [%]				Gold scores [%]		
Alg	Lang pair	Lex	Rel	Prec.	Recall	F1	Acc.	Recall	F1	Acc.
Xfer	de → cs	18 118	5 971	39.66	33.11	36.09	53.71	0.29	0.58	1.19
	fr → cs	20 225	7 045	42.46	36.11	39.03	53.79	0.37	0.73	1.33
	cs → de	13 803	3 847	27.06	35.36	30.66	65.88	2.45	4.50	17.07
	fr → de	2 938	600	14.33	14.14	14.24	64.74	0.20	0.39	4.19
	cs → fr	2 769	1 219	23.54	30.50	26.57	42.72	2.10	3.86	7.65
	de → fr	439	144	3.47	11.36	5.32	59.45	0.04	0.07	1.84
ML	de → cs	1 026 036	914 097	37.83	84.94	52.35	38.64	84.94	52.35	38.64
	fr → cs	1 026 036	917 863	32.41	81.15	46.32	32.80	81.15	46.32	32.80
	cs → de	280 454	263 477	9.21	84.01	16.60	13.06	84.01	16.60	13.06
	fr → de	280 454	270 180	5.81	84.27	10.87	8.21	84.27	10.87	8.21
	cs → fr	21 288	21 287	43.88	100.00	61.00	43.88	100.00	61.00	43.88
	de → fr	21 288	21 287	17.28	100.00	29.47	17.28	100.00	29.47	17.28
closest-shorter baseline	cs	1 026 036	808 933	21.03	53.54	30.20	23.35	53.54	30.20	23.35
	de	280 454	225 092	5.22	56.51	9.55	20.70	56.51	9.55	20.70
	fr	21 288	17 451	31.65	82.71	45.79	38.55	82.71	45.79	38.55
empty baseline	cs	1 026 036	0	N/A	0.00	0.00	21.14	0.00	0.00	21.14
	de	280 454	0	N/A	0.00	0.00	84.62	0.00	0.00	84.62
	fr	21 288	0	N/A	0.00	0.00	35.15	0.00	0.00	35.15

Table 2: Evaluation scores of the results and baselines for each language pair. 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	Accuracy	No child trans	No parent trans	No match	Xferred	Gold
de → cs	5.10	29.14	91.05	0.08	3.77	43 368	809 282
fr → cs	6.75	31.74	89.62	0.05	3.59	13 808	809 282
cs → de	34.47	89.82	52.08	0.23	13.22	809 282	43 368
fr → de	26.24	92.69	50.60	0.02	22.14	13 808	43 368
cs → fr	34.67	80.11	56.81	0.20	8.33	809 282	13 808
de → fr	22.26	64.01	61.89	0.07	15.78	43 368	13 808

Table 3: 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.

influenced by the translation quality. The alignment data trained on the de—fr 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.

Not shown are results including the word-embedding-based improvements from Section 4. We've tried many combinations of their settings: different sources of embeddings, using cosine or Euclidean distance, using K-Means clustering or just a single mean for each affixal pattern, and including a word under all affixal subpatterns or only under its "native" pattern. However, a grid-search of the weights of translation score, orthographic similarity score and the two embedding similarity scores always tends to set the embedding weights to 0 and maximize the orthographic similarity weight. Ultimately, no improvement could be reached using these two models.

For the semantic similarity score, the reason can be seen in Figure 2. For any given score, there are at least 1 000 times more negative examples than positive ones, meaning that any useful signal invariably gets lost in the noise. The score could perhaps serve as a useful feature in a supervised machine learning method, but the experiments show that it is not helpful in our simple unsupervised setting.

The reasons for the failure of the affixal pattern score are more complex, and different variations of the score fail in different ways. When we classify each relation under just its single "native" pattern, many relations (especially wrong ones) will be the sole example in their cluster, giving them a similarity score of 1. To avoid this failure mode, we decided to require at least 30 samples in each cluster. But this means that most relations will not be able to get a score at all, because their affixal pattern is rarer.

Therefore, it is necessary to classify each relation under all patterns that can be applied to it, allowing many allomorphs to contribute to related patterns. However, this causes shorter patterns to accrue a large amount of spurious or wrong relations, splitting them into many clusters with high variance. Then, when trying to find the similarity between a given relation and its pattern, there are many possible clusters to compare against, increasing the probability that the embedding difference of the relation will be randomly close to one of them, which in turn increases the variance of the calculated similarity scores.

Unlike the embedding scores, the machine learning method is proven to be successful. It provides a way of regularizing 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 patterns. As seen in the third part of Table 2, this results in increased precision on the French Démonette data, which only contains a few selected paradigms and therefore skews towards fewer, more productive patterns. On the other target languages, some precision is traded for recall, which increases from 0-2.5% to 84%. Due to this large increase, F1-score also increases. In general, the resulting networks are overgenerated, with very high recall and smaller precision, but all our attempts at balancing the two resulted in a decrease in the F1 score.

The oracle scores are in Table 3. 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.

For all language pairs, most of the error in the recall is 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 lexemes such as *přeskočitelnost* ("skippability") in the parallel data is unlikely.

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 nonderived. 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 non-derived words are also the reason behind the fact that fr-de has higher accuracy than cs-de, despite having lower recall – fewer relations are translated, resulting in more non-derived words being correct.

The oracle scores show that the main bottleneck is the word translation dictionary – the "No child trans" category accounts for 50-90% of all errors. This is 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.

8 Future work

An immediately necessary task is to manually annotate test data, because the German and French word-formation networks we use for evaluation have a too low recall. The recall of DErivBase is approx. 44 %, recall of DeriNet is approx. 80 % and recall of Démonette is approx. 70 %. The low recall of DErivBase is mostly caused by compounding, which is not captured in the resource, but its lexicon nonetheless contains many compounds.

To remedy this, we've annotated parents of 200 German and 100 French lexemes, in addition to the already available 2 000 Czech lexemes (Vidra, 2018). However, these test sets are too small to reliably measure the quality of the transfer algorithm – cross-validation performed by extracting varying subsets of the input parallel data shows high variance, as opposed to the very stable scores measured on the word-formation networks themselves. This is caused by the fact that the annotated lexemes were sampled from the outputs of the transfer algorithm and mostly represent hapaxes, which vary a lot when subsampling the input corpus. In the future, we will annotate a sample of the data weighted by corpus frequency to get a more representative gold-standard dataset.

In the coming months, we plan to extend the current transfer experiments to cover compounding in addition to derivation. This should be doable by fully utilizing the information contained in the word alignments. At present, if a word on the source side is aligned to a phrase on the target side (or vice versa), we ignore the grouping and treat the alignment as several independent 1:1 relations. However, the fact that a word is aligned to several content words at once indicates a degree of relatedness to all of them, potentially signifying compounding. In addition to this, the Czech word-formation network contains annotations of compounding, which can be transferred even across 1:1 relations.

Another way of increasing the quality of generated networks would be to use multi-source transfer, combining the translation scores from multiple source languages before constructing the target-side network.

In a more long-term outlook, we would like to use deep learning techniques to improve the results. Although word embeddings are known to contain derivational information, we've been unable to successfully exploit them in the simple setting where we reduced them to just a cosine or Euclidean distance. We believe that using them as inputs to a neural network, possibly with joint training in an end-to-end architecture, would be more successful. This would also enable us to transfer information through multilingual embeddings, which could eliminate a large source of error stemming from the word translation dictionary that we use for transfer now.

9 Conclusion

In this thesis proposal, we presented a cross-lingual method for creating word-formation networks by transferring an existing network using a word-translation lexicon induced from word alignments. One attempt at improving the transferred data using word embeddings as a proxy for modeling semantics of word formation has failed, but a second attempt using a simpler machine learning method to expand the transferred small networks by extracting paradigms using statistical machine learning and applying them to a larger lexicon was successful. The resulting word-formation networks are somewhat overgenerated, but still show moderately high precision for several language pairs.

Acknowledgments

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

References

Hirotugu Akaike. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada. Association for Computational Linguistics.

Marion Baranes and Benoît Sagot. 2014. 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. European Language Resources Association (ELRA).

Khuyagbaatar Batsuren, Gabor Bella, and Fausto Giunchiglia. 2019. CogNet: A large-scale cognate database. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3136–3145, Florence, Italy. Association for Computational Linguistics.

Laurie Bauer. 2014. *The Oxford Handbook of Derivational Morphology*, Oxford Handbooks in Linguistics, chapter Concatenative Derivation. Oxford University Press, Oxford, United Kingdom.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information.

Xilun Chen, Ahmed Hassan Awadallah, Hany Hassan, Wei Wang, and Claire Cardie. 2019. Multi-source cross-lingual model transfer: Learning what to share. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3098–3112, Florence, Italy. Association for Computational Linguistics.

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

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. 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. Association for Computational Linguistics.

Filip Ginter, Jan Hajič, Juhani Luotolahti, Milan Straka, and Daniel Zeman. 2017. CoNLL 2017 shared task - automatically annotated raw texts and word embeddings. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. BilBOWA: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 748–756, Lille, France. PMLR.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

Nabil Hathout and Fiammetta Namer. 2014. Démonette, a French derivational morpho-semantic network. *Linguistic Issues in Language Technology*, 11:125–162.

Lukáš Kyjánek. 2018. Morphological resources of derivational word-formation relations. Technical Report ÚFAL TR-2018-61, ÚFAL MFF UK, Praha, Czechia.

Lukáš Kyjánek, Zdeněk Žabokrtský, Magda Ševčíková, and Jonáš Vidra. 2019. Universal Derivations kick-off: 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. ÚFAL MFF UK.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *International Conference on Learning Representations*.

Mateusz Lango, Zdeněk Žabokrtský, and Magda Ševčíková. 2021. Semi-automatic construction of word-formation networks. *Language Resources and Evaluation*, 55(1):3–32.

Stuart P. Lloyd. 1982. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28:129–137.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. 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. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013b. Exploiting similarities among languages for machine translation.

Tomáš Musil, Jonáš Vidra, and David Mareček. 2019. Derivational morphological relations in word embeddings. In *The BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP at ACL 2019*, pages 173–180, Stroudsburg, PA, USA. Association for Computational Linguistics.

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

Renáta Panocová. 2017. Internationalisms with the suffix -ácia and their adaptation in slovak. In *Proceedings of the Workshop in Resources and Tools for Derivational Morphology (DeriMo)*, pages 61–72, Milano, Italy. Università Cattolica, EDUCatt.

Fabian Pedregosa, 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. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta. ELRA. `http://is.muni.cz/publication/884893/en`.

Rudolf Rosa. 2018. *Discovering the structure of natural language sentences by semi-supervised methods*. Ph.D. thesis, Charles University, Faculty of Mathematics and Physics, Praha, Czechia.

Rudolf Rosa and Zdeněk Žabokrtský. 2019a. Attempting to separate inflection and derivation using vector space representations. In *Proceedings of the Second International Workshop on Resources and Tools for Derivational Morphology (DeriMo 2019)*, pages 61–70, Praha, Czechia. ÚFAL MFF UK.

Rudolf Rosa and Zdeněk Žabokrtský. 2019b. Unsupervised lemmatization as embeddings-based word clustering.

Milan Straka and Jana Straková. 2017. 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. Association for Computational Linguistics.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey. European Language Resources Association (ELRA).

Ahmet Üstün and Burcu Can. 2016. Unsupervised morphological segmentation using neural word embeddings. In *Statistical Language and Speech Processing: 4th International Conference, SLSP 2016*, pages 43–53, Cham, Switzerland. Springer International Publishing.

Jonáš Vidra. 2015. Extending the lexical network DeriNet. Bachelor's thesis, Charles University, Faculty of Mathematics and Physics, Praha, Czechia.

Jonáš Vidra. 2018. Morphological segmentation of Czech words. Master's thesis, Charles University, Faculty of Mathematics and Physics, Praha, Czechia.

Jonáš Vidra. 2021. Transferring word-formation networks between languages. In *Proceedings of the Third International Workshop on Resources and Tools for Derivational Morphology (DeriMo 2021)*, Nancy, France. University of Lorraine. Accepted for publication.

Britta Zeller, Jan Šnajder, and Sebastian Padó. 2013. 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.

Yuan Zhang, David Gaddy, Regina Barzilay, and Tommi Jaakkola. 2016. 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. Association for Computational Linguistics.

Zdeněk Žabokrtský, Magda Ševčíková, Milan Straka, Jonáš Vidra, and Adéla Limburská. 2016. 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. European Language Resources Association.

A Sample outputs from the transfer algorithm

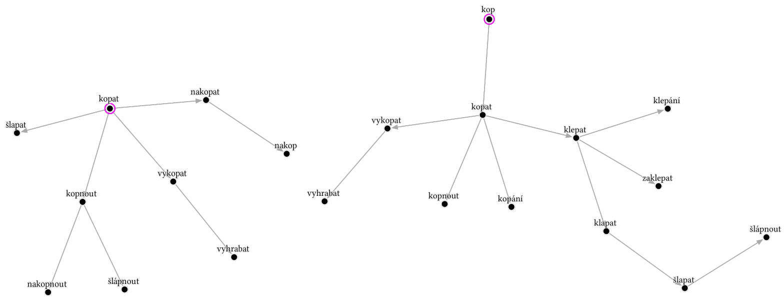

Figure 3: Transferred word-formation networks for Czech, showing the family of lexeme *kopat*. On the left, transfer from German, on the right, transfer from French.

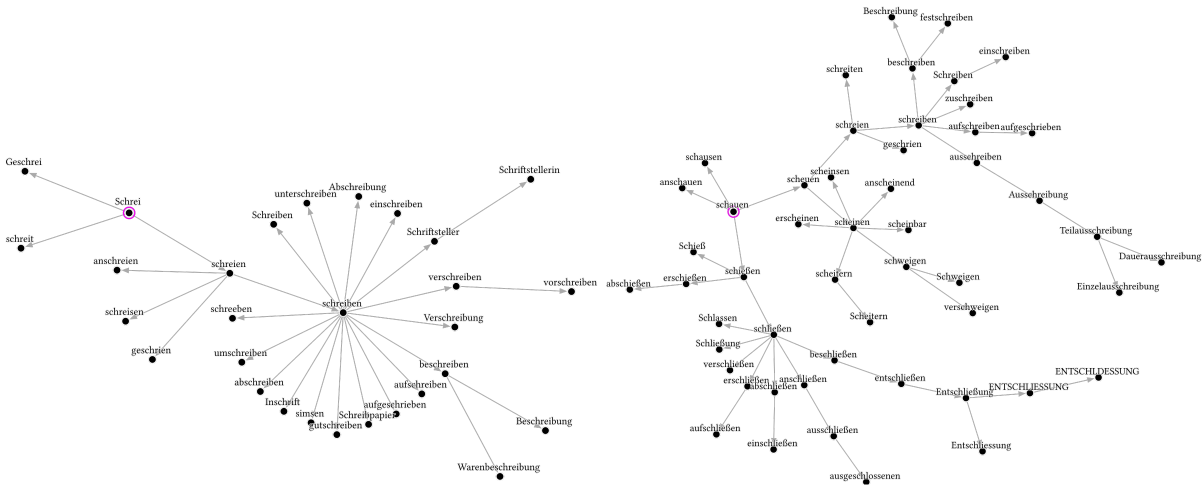

Figure 4: Transferred word-formation networks for German, showing the family of lexeme *schreiben*. On the left, transfer from Czech, on the right, transfer from French.

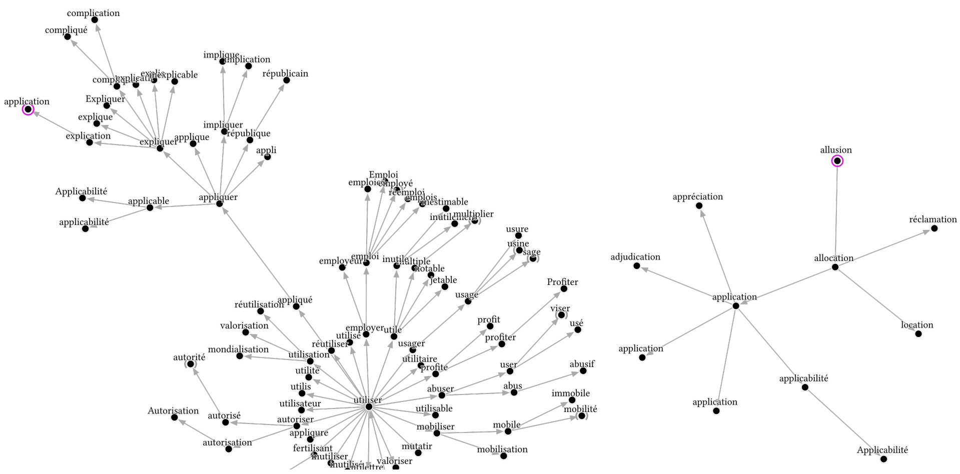

Figure 5: Transferred word-formation networks for French, showing the family of lexeme *application*. On the left, transfer from Czech, on the right, transfer from German. The left picture is cropped; root of the family is *Ie*, through *vie*, *vier*, *voir*, *avoir* and 13 other lexemes.