Transferring Word-Formation Networks Between Languages

Jonáš Vidra
Charles University,
Faculty of Mathematics and Physics,
vidra@ufal.mff.cuni.cz

Zdeněk Žabokrtský
Charles University,
Faculty of Mathematics and Physics,
zabokrtsky@ufal.mff.cuni.cz

Abstract

In this article, we present a proof-of-concept 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.

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.

In this article, we explore a method from the other, unsupervised, part of the scale: a method which does not require any human input or in-language annotations of word-formation relations. Instead, it transfers an existing word-formation network from another language using parallel texts and off-the-shelf tools for tokenization and lemmatization. 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).

As a result, our method 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 semantic 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.

By further filtering the transferred relations by orthographic distance, we obtain a moderate-precision and low-recall word-formation network. The recall can be improved by extracting the discovered string-wise word-formation patterns using a statistical machine-learning method and finding more examples of them across the lexicon.

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.

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 these are among the few languages for which a large, high-quality word-formation network already exists. The existing networks, Démonette (Hathout and Namer, 2014), DErivBase (Zeller et al., 2013) and DeriNet (Žabokrtský et al., 2016), serve a dual role as data for transfer on the source side, and evaluation datasets on the target side of each of the six possible independent translation pairs.

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 *X-e* → *X-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.

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.

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. Word embeddings are also used by Musil et al. (2019), who show that words created through similar word-formation processes have similar embedding differences; however, they do not use these results to actually construct a network out of word-embedding data.

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 target-side formal 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.

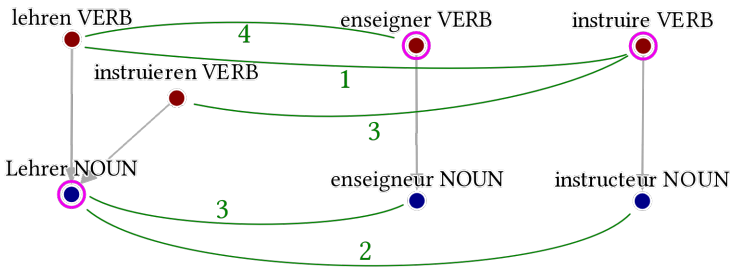

Figure 1: An example of finding a parent for the German lexeme *Lehrer* ("teacher") by transferring information from a French word-formation network, with word-formation relations in grey and alignments in green. *Lehrer* is aligned to *enseigneur* ⅗ times, which has *enseigner* available through 1 relation,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

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 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 $\mathrm{Xfer}(C, P)$ according to Equation 1 below, where $|\mathrm{align}(x, y)|$ denotes the number of alignments between lexemes x and y seen in the aligned data and $\mathrm{dist}(C', P')$ denotes the number of relations on the shortest path from C' to P' in the source network.

$$\mathrm{Xfer}(C, P) = \sum_{\forall C', P'} \frac{|\mathrm{align}(C, C')|}{\sum_{\forall x} |\mathrm{align}(C, x)|} \cdot 0.5^{\mathrm{dist}(C', P')} \cdot \frac{|\mathrm{align}(P', P)|}{\sum_{\forall x} |\mathrm{align}(P', x)|} \quad (1)$$

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. Therefore, we set the weight of the edit distance to 5, the weight of the translation to 1 and the threshold to 0.8 using grid search on the Czech → German transfer pair and use these hyperparameters on all pairs.

4 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.

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 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.

This method of obtaining relation candidates depends on the linguistic properties of the languages under consideration, namely Czech, French and German. All three derive words predominantly by affixation, with limited allomorphy in the stem and only rare examples of circumfixation, apophony or suppletive relations, which this method generally doesn't detect as possible relations. Therefore, 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 %. 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 produces inferior results on the three datasets we use, we therefore don't use it in our experiments.

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.

```
for gold_child in gold.lexemes:
  if not gold_child.parent:
    true_negative++
  else:
    for t_child in translations(gold_child):
      for t_parent in family(t_child):
        for parent in backtranslations(t_parent, gold_child):
          if parent = gold_child.parent:
            true_positive++
            continue_line 1
    false_negative++
  accuracy := ((true_positive + true_negative) / (true_positive +
   ↪ true_negative + false_negative))
  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`.

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. 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 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 measures and only measure scores on the intersection of the two lexicons. 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.

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.

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).

5.3 Experimental setting

For the purposes of this paper, 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.

6 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.

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 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.

The machine learning method provides a way of generalizing the output of the transfer method, as it

Alg	Lang pair	Size		Internal scores [%]				Gold scores [%]		
		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	743 469	45.70	73.81	56.45	48.90	73.81	56.45	48.90
	fr → cs	1 026 036	742 784	39.60	70.00	50.58	43.99	70.00	50.58	43.99
	cs → de	280 454	68 154	35.02	67.73	46.17	80.15	67.73	46.17	80.15
	fr → de	280 454	34 809	44.25	39.35	41.66	84.62	39.35	41.66	84.62
	cs → fr	21 288	15 136	60.33	88.64	71.79	66.30	88.64	71.79	66.30
	de → fr	21 288	4 700	35.57	13.79	19.88	36.69	13.79	19.88	36.69
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.

Lang pair	Scores [%]		Error cause [%]			WFN rel count	
	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.

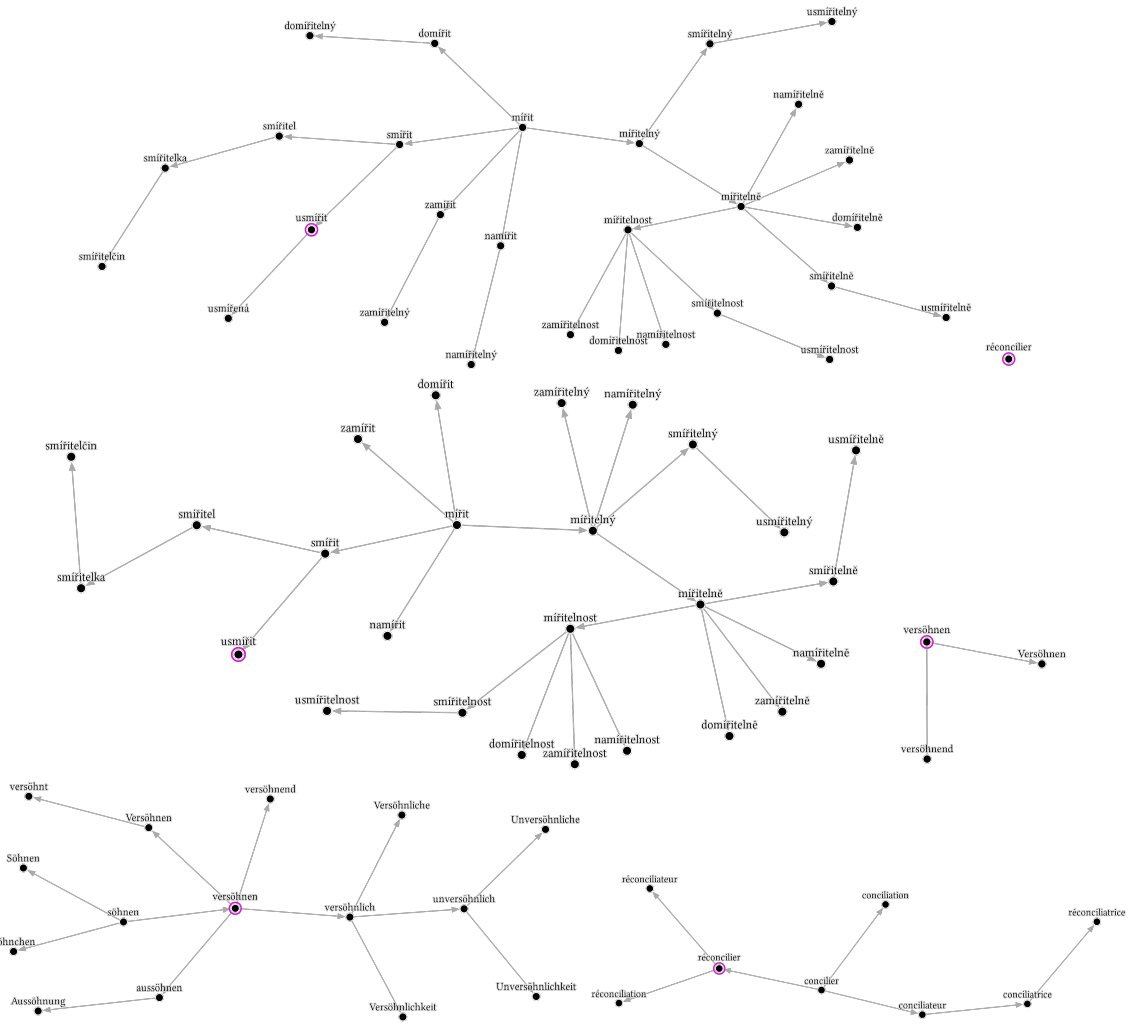

Figure 2: Word-formation networks generated by the machine learning expansion of the transferred networks, showing the family of lexeme *to reconcile* circled in violet for each of the six language pairs. Clockwise from top left: de-cs, de-fr (single lexeme), fr-de, cs-fr, cs-de, fr-cs.

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 2, 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 Figure 2.

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. 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 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.

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 fr-de pair has higher accuracy than cs-de, despite having lower recall – fewer relations are translated, resulting in more unmotivated 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 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.

7 Conclusion

In this paper, 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. The transferred small networks are then expanded by extracting paradigms using statistical machine learning and applying them to a larger set of lexemes. The resulting word-formation networks show moderately high precision and good recall on six language pairs.

Acknowledgments

This work was supported by the Grant No. GA19-14534S of the Czech Science Foundation, by the Charles University Grant Agency (project No. 1176219) and by the SVV project number 260 575. It has been using language resources developed, stored, and distributed by the LINDAT/CLARIAH CZ project (LM2015071, LM2018101).

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

References

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)*. European Language Resources Association (ELRA), Reykjavik, Iceland, pages 2793–2799. http://www.lrec-conf.org/proceedings/lrec2014/pdf/379_Paper.pdf.

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*. Association for Computational Linguistics, Florence, Italy, pages 3136–3145. https://doi.org/10.18653/v1/P19-1302.

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

Kenneth Ward Church. 1993. Char_align: A program for aligning parallel texts at the character level. In *31st Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Columbus, Ohio, USA, pages 1–8. https://doi.org/10.3115/981574.981575.

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*. Association for Computational Linguistics, Atlanta, Georgia, pages 644–648. https://www.aclweb.org/anthology/N13-1073.

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

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. http://ufal.mff.cuni.cz/techrep/tr61.pdf.

Lukáš Kyjánek, Zdeněk Žabokrtský, Magda Ševčíková, and Jonáš Vidra. 2019. 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)*. ÚFAL MFF UK, Praha, Czechia, pages 101–110.

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

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*. Association for Computational Linguistics, Edinburgh, Scotland, UK., pages 62–72. https://aclanthology.org/D11-1006.

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*. Association for Computational Linguistics, Stroudsburg, PA, USA, pages 173–180.

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

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.

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

Michel Simard, George F. Foster, and Pierre Isabelle. 1992. 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. https://aclanthology.org/1992.tmi-1.7.

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*. Association for Computational Linguistics, Vancouver, Canada, pages 88–99. http://www.aclweb.org/anthology/K/K17/K17-3009.pdf.

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

Ahmet Üstün and Burcu Can. 2016. Unsupervised morphological segmentation using neural word embeddings. In Pavel Král and Carlos Martín-Vide, editors, *Statistical Language and Speech Processing: 4th International Conference, SLSP 2016*. Springer International Publishing, Cham, Switzerland, pages 43–53. https://doi.org/10.1007/978-3-319-45925-7_4.

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*. Sofia, Bulgaria, pages 1201–1211. http://www.aclweb.org/anthology/P13-1118.

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*. Association for Computational Linguistics, San Diego, California, pages 1307–1317. https://doi.org/10.18653/v1/N16-1156.

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)*. European Language Resources Association, Paris, France, pages 1307–1314.