EAMT 2016

Graphonological Levenshtein Edit Distance: Application for automated cognate identification

Authors

Address

email@institution

Abstract: This paper presents a methodology for calculating a modified Levenshtein edit distance between character strings and applies it to the task of automated cognate identification from non-parallel (comparable) corpora. This task is an important stage in developing MT systems and bilingual dictionaries beyond the coverage of traditionally used aligned parallel corpora, which can be used for finding translation equivalents for the 'long tail' in Zipfian distribution: low-frequency and usually unambiguous lexical items in closely-related languages (many of those often under-resourced).

Graphonological Levenshtein edit distance relies on editing hierarchical representations of phonological features for graphemes (graphonological representations) and improves on phonological edit distance proposed for measuring dialectological variation. Graphonological edit distance works directly with character strings and does not require an intermediate stage of phonological transcription, exploiting the advantages of historical and morphological principles of orthography, which are obscured if only phonetic principle is applied. Difficulties associated with plain feature representations (unstructured feature sets or vectors) are addressed by using linguistically-motivated feature hierarchy that restricts matching of lower-level graphonological features when higher-level features are not matched. The paper presents an evaluation of the graphonological edit distance in comparison with the traditional Levenshtein edit distance from the perspective of its usefulness for the task of automated cognate identification and discusses the advantages of the proposed method.

Keywords: cognates; Levenshtein edit distance; phonological features; comparable corpora; closely-related languages; under-resourced languages; Ukrainian; Russian; Hybrid MT

1. Introduction

Levenshtein edit distance proposed in (Levenshtein, 1966) is an algorithm that calculates the cost (normally – the number of operations such as deletions, insertions and substitutions) needed to transfer a string of symbols (characters or words) into another string. This algorithm is used in many computational linguistic applications that require some form of the fuzzy string matching, examples include fast creation of morphological and syntactic taggers exploiting similarities between closely related languages (Hana et al., 2006), statistical learning of preferred edits for detecting regular orthographic correspondences in closely related languages (Ciobanu & Dinu, 2014). Applications of Levenshtein's metric for the translation technologies and specifically for Machine

Translation include automated identification of cognates for the tasks of creating bilingual resources such as electronic dictionaries (e.g., Koehn and Knight, 2002; Mulloni & Pekar, 2006; Bergsma & Kondrak, G. 2007), improving document alignment by using cognate translation equivalents as a seed lexicon (Enright, J & Kondrak, G., 2007), automated MT evaluation (e.g., Niessen et al., 2000; Leusch et al., 2003).

Levenshtein distance metrics has been modified and extended for applications in different areas; certain ideas have yet not been tested in MT context, but have a clear potential for benefiting MT-related tasks. This paper develops and evaluates one of such ideas for a linguistic extension of the metric proposed in the area of computational modelling of dialectological variation and measuring 'cognate' lexical distance between languages, dialects and different historical periods in development of languages, e.g., using cognates from the slow-changing part of the lexicon – the Swadesh list (Swadesh, 1952; Serva & Petroni, 2008; Schepens et al., 2012).

In this paper the suggestion is explored of calculating the so called Levenshtein's 'phonological edit distance' between phonemic transcriptions of cognates, rather than the traditional string edit distance (Nerbonne & Heeringa 1997; Sanders & Chin, 2009), based on the earlier linguistic paradigm introduced into the computational linguistic by Chomsky and Halle (1968). The idea is that each phoneme in a transcription of a cognate is represented as a structure of phonological differentiative features, such as:

[a] = [+vowel, +back; +open; –labialised] ;

Then the distance is calculated for rewriting of these feature representations rather than rewriting the whole character: so rewriting [o] into [a] (which, e.g., is a typical vowel alternation pattern in Russian and distinguishes some of its major dialects) would incur a smaller cost compared to the substitution of the whole character, since only two of its differentiative phonological features need to be rewritten:

[o] = [+vowel, +back; *+mid; +labialised*]

On the other hand, the cost of rewriting the vowel [a] into the consonant [t] (the change which normally does not happen as part of the historical language development or dialectological variation) would involve rewriting all the phonological features in the representation, so the edit cost will be the same as for the substitution of the entire character:

[t] = [*+consonant; –voiced; +plosive; +fronttongue; +alveolar*]

According to Nerbonne & Heeringa (1997:2) the feature-based Levenshtein distance makes it "…possible to take into account the affinity between sounds that are not equal, but are still related"; and to "…show that *'pater'* and *'vader'* are more kindred then *'pater'* and *'maler'*." This is modelled by the fact that phonological feature representations for pairs such as [t] and [d] (both front-tongue alveolar plosive consonants, which only differ by 'voiced' feature), as well as [p] and [v] (both labial consonants), share greater number of phonological features compared to the pairs [p] and [m] (differ in sonority, manner and passive organ of articulation) or [t] and [l] (which differ in sonority and the manner of articulation). However, the authors point out to a number of open questions and problems related to their modified metric, e.g., how to represent phonetic features of complex phonemes, such as diphthongs; what should be the structure of feature representations: Nerbonne & Heeringa use feature vectors, but are these vectors sufficient or more complex feature representations are needed; how to integrate edits of individual features into the calculation of a coherent distance measure (certain settings are not used, whether to use Euclidian or Manhattan distance, etc.

Linguistic ideas behind the suggestion to use Levenshtein phonological edit distance are intuitively appealing and potentially useful for applications beyond dialectological

modelling. However, to understand their value for other areas, such as MT, there is a need to develop a clear evaluation framework for testing the impact of different possible settings of the modified metric and different types of feature representations, to compare specific settings of the metric to alternatives and the classical Levenshtein's baseline. Without a systematic evaluation framework the usefulness of metrics remain unknown.

This paper proposes an evaluation framework for testing alternative settings of the modified Levenshtein's metric. This framework is task-based: it evaluates the metric's alternative settings and feature representations in relation to its success on the task of automated identification of cognates from non-parallel (comparable) corpora.

The paper is organised as follows: Section 2 presents the set-up of the experiment, the application of automated cognate identification; the design and feature representations for the metric and the evaluation framework. Section 3 presents evaluation results of different metric settings and comparison with the classical Levenshtein distance; Section 4 presents conclusion and future work.

2. Set up of the experiment

2.1. Application of automated cognate identification for MT

Automated cognate identification is important for a range of MT-related tasks, as mentioned in Section 1. Our project deals with rapid creation of hybrid MT systems for new translation directions for a range of under-resourced languages, many of which are closely related, or 'cognate', such as Spanish and Portuguese, German and Dutch, Ukrainian and Russian. The systems combine rich linguistic representations used by a backbone rule-based MT engine with statistically derived linguistic resources and statistical disambiguation and evaluation techniques, which work with complex linguistic data structures for morphological, syntactic and semantic annotation (Anonymized, 2099). In the project the translation lexicon for the hybrid MT systems is derived via two routes:

1. Translation equivalents for a smaller number of highly frequent words, which under empirical observations of Zipf's and Menzerath's laws (Koehler, R. 1993; 49) tend to be shorter (Zipf, 1935:38; Sigurd et al., 2004:37) and more ambiguous (Menzerath, 1954, Hubey, 1999; Anonymized, 2098:7), are generated as statistical dictionaries from sentence-aligned parallel corpora. However, as only small number of parallel resources is available for under-resourced languages, there remain many out-of-vocabulary lexical items.
2. The remaining 'long tail' in Zipfian distribution containing translation equivalents for a large number of low-frequent and usually unambiguous lexical items (as they typically have only one correct translation equivalent) is derived semi-automatically from much larger non-parallel comparable corpora, which are usually in the same domain for both languages. We use a number of different techniques depending on available resources and language pairs (e.g., Anonymised, 2097). For closely related languages (depending on the degree of their 'relatedness') the 'long tail' contains a large number of cognates. In our experiments for Ukrainian / Russian language pair this number reached 60% of the analysed sample of the lexicon selected from different frequency bands (see Section 3).

In order to cover this part of the lexicon, the automated cognate identification from non-parallel corpora is used for generating draft ranked lists of candidate translation equivalents. The candidate lists are generated using the following procedure:

1. Large monolingual corpora (in our experiments -- about 250M for Ukrainian and 200M for Russian news corpora) are PoS tagged and lemmatised.
2. Frequency dictionaries are created for lemmas. A frequency threshold is applied (to keep down the 'noise' and the number of hapax legomena.
3. Edit distances for pairs of lemmas in a Cartesian product of the two dictionaries are automatically calculated using variants of the Levenshtein measure.
4. Pairs with edit distances below a certain threshold are retained as candidate cognates (in our experiments we used the threshold value of the Levenshtein edit distance normalised by the length of the longest word <=0.36, intuitively: 36% of edits per character)
5. Candidate cognates are further filtered by part-of-speech codes (cognates with non-matching parts of speech are not ranked)
6. Candidate cognates are filtered by their frequency bands: if the TL candidate is beyond the frequency band threshold of the SL candidate, the TL candidate is not ranked (in our experiment we used the threshold $FrqRange > 0.5$ for the difference in natural logarithms of absolute frequencies – see formula (1), intuitively: candidates should not have frequency difference several orders of magnitude apart.
7. Candidate cognate lists are ranked by the increasing values of the edit distance.

$$FrqRange = \frac{\min\left(\ln(FrqB), \ln(FrqA)\right)}{\max\left(\ln(FrqB), \ln(FrqA)\right)} \quad (1)$$

These ranked lists are presented to the developers, candidate cognates are checked and either included into system dictionaries, or rejected. Developers' productivity of this task crucially depends on the quality of automated edit distance metric that generates and ranks the draft candidate lists.

Different settings for modifications of Levenshtein edit distance can be systematically evaluated in this scenario by using human annotation of the candidate cognate lists (Table 1 shows the annotation labels used).

The task of creating parallel resources and dictionaries from comparable corpora is not exclusive to hybrid or rule-based MT. Similar ideas are used in SMT framework for enhancing SMT systems developed for under-resourced languages via identification of aligned sentences and translation equivalents in comparable corpora, which generally reduces the number of out-of-vocabulary words not covered by scarce parallel corpora (Pinnis et al., 2012). In these settings dictionaries of cognate lists can become an additional useful resource, so achieving a higher degree of automation for the process of cognate identification in comparable corpora is equally important for the SMT development. Under these settings an operational task-based evaluation for Levenshtein edit distance metrics will be performance parameters of the developed SMT systems.

Label	Interpretation
NC	No cognate: a word in source language (SL) does not have a cognate in the target language (TL)
0D	Zero difference: absolute cognates there is no difference in orthographic strings in the SL and TL
PN	Proper name: (usually) zero difference cognates which are proper names, e.g., names of people, places, organizations
FF	'False friends' cognates with different meaning in the SL and TL
CL	Correct cognate ranked higher by the baseline, string-based Levenshtein metric
CF	Correct cognates ranked higher by the feature-based Levenshtein metric
RL-	Cognate ranked lower by the baseline Levenshtein metric
RF-	Cognate ranked lower by the feature-based Levenshtein metric
ML	Cognate is missed by the baseline Levenshtein metric
MF	Cognate is missed by the feature-based Levenshtein metric

Table 1. Labels used for candidate cognate annotation

2.2. Development of Levenshtein graphonological feature-based metric

For the task of automated cognate identification a feature-based edit distance measure needs further adjustments, beyond the metric used in modelling dialectological variation.

Firstly, the metric has to work directly with word character strings, not via the intermediate stage of creating a phonological transcription for each word. While for modelling of dialects (many of which do not capture pronunciation differences in their own writing systems) the transcription may be a necessary step, MT systems normally deal with languages with their own established writing systems. There are practical reasons for extracting features from orthography rather than phonological transcriptions: automated phonological transcription of the orthographic strings may create an additional source of errors; resources for transcribing may be not readily available for many languages; for the majority of languages very little can be gained by replacing the orthography by transcription (apart from more adequate representation of digraphs and phonologically ambiguous characters, which can be addressed also on the level of orthography).

However, there are more important theoretical reasons for preferring original orthographic representations. For instance, orthography of languages is usually based on a combination of three principles: *phonetic* (how words are pronounced), *morphological* (keeping the same spellings for morphemes – minimal meaning units, such as affixes, stems, word routes, irrespective of any pronunciation variation caused by their position, phonological context, regular sound alternations, etc.) and historic (respecting traditional spelling which reflects an earlier stage of language development, even though the current pronunciation may have changed; often orthography reflects the stage when cognate languages have been closer together). Example 2 illustrates the point why orthography might work better for cognate identification:

	Russian	*Ukrainian*
Orthography	s**o**baka (собака) 'dog'	sobaka (собака) 'dog'
Phonological transcription	[s**a**baka] (с[а]бака)	[s**o**baka] (с[о]бака)
Change	[o] -> [a]	[o] -> (no change)

(2)

The pronunciation change [o] -> [a], which in some (at that time) marginal Russian dialects dates back to 13th century (one of the explanations for this change is the influence the Baltic substratum), was not reflected in Russian educated written tradition, even at the later time when those dialects received much more political prominence and influenced the pronunciation norm of the modern standard Russian. In many cases such historic orthography principle makes the edit distance between cognates in different languages much shorter, and the phonological transcription in these cases may obscure innate morphological and historical links between closely related languages reflected in spelling. Therefore, using orthography to directly generate phonological feature representations has a theoretical motivation.

In this paper we use the term *graphonological features* to refer to representations of phonological features that are directly derived from graphemes. The approach adopted in our experiment is that each orthographic character in each language is unambiguously associated with a set of phonological features, even though its pronunciation may be different in different positions.

Secondly, we need to address the issue how to structure features sets in graphonological representations. In our earlier experiments the problems with structuring them as flat feature vectors became apparent. Even though in some examples there has been improvement in the rate of cognate identification caused by richer feature structures as compared to the baseline Levenshtein metric, in many more cases (and often counter to our earlier intuition) there structured caused unnecessary noise and lower ranking for true cognates, while non-cognates received smaller feature-based edit distance score – an unwanted result, which has been traced back to the use of feature vectors as graphonological feature structures.

The example (3) illustrates this. If feature vector representations are used, our graphonological metric calculates that following edit distances are the same, which is a counter-intuitive result (especially given that the traditional Levenshtein's metric clearly shows that the character-based edit distance is shorter:

robitnyk (робітник) 'worker' (uk) & *rabotnik* (работник) 'worker' (ru)
GrPhFeatLev =1.2 Lev=2.0

robitnyk (робітник) 'worker'(uk) & *rovesnik* (ровесник) 'age-mate, of the same age' (ru) (3)
GrPhFeatLev =1.2 Lev=3.0

The reason for this can be explained by checking edit matrices and feature vectors of the compared words shown in Figure 1 and Figure 2.

It can be seen from the figures that there is a specific problem when intuitively unrelated consonants (at least among Ukrainian-Russian lexical cognates) [b] and [v], or [t] and [n], which still receive very small rewriting scores. Table 2 and Table 3 show overlapping graphonological features for these words. In both cases, while one of the more essential features was not matched – *manner of articulation*, but instead the smaller edit distance resulted from matching less important features: *voice, active* and *passive* articulation organs. The problem with using feature vector representation is that all of the features stay on the same level, there is no way of indicating that certain features are more important for cognate formation and perception.

		r(р)	o(о)	b(б)	i(і)	t(т)	n(н)	y(и)	k(к)
	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0
r(р)	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
o(о)	2.0	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0
v(в)	3.0	2.0	1.0	1.0	2.0	3.0	4.0	5.0	6.0
e(е)	4.0	3.0	2.0	2.0	2.0	3.0	4.0	5.0	6.0
s(с)	5.0	4.0	3.0	3.0	3.0	3.0	4.0	5.0	6.0
n(н)	6.0	5.0	4.0	4.0	4.0	4.0	3.0	4.0	5.0
i(и)	7.0	6.0	5.0	5.0	5.0	5.0	4.0	3.0	4.0
k(к)	8.0	7.0	6.0	6.0	6.0	6.0	5.0	4.0	3.0

Figure 1 Baseline Levenshtein: Edit distance matrix for
robitnyk (робітник) ‘worker’(uk) & *rovesnik* (ровесник) ‘age-mate, of the same age’ (ru)

		r(р)	o(о)	b(б)	i(і)	t(т)	n(н)	y(и)	k(к)
	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0
r(р)	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
o(о)	2.0	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0
v(в)	3.0	2.0	1.0	0.4	1.4	2.4	3.4	4.4	5.4
e(е)	4.0	3.0	2.0	1.4	0.8	1.8	2.8	3.8	4.8
s(с)	5.0	4.0	3.0	2.4	1.8	1.0	2.0	3.0	4.0
n(н)	6.0	5.0	4.0	3.4	2.8	2.0	1.0	2.0	3.0
i(и)	7.0	6.0	5.0	4.4	3.4	3.0	2.0	1.2	2.2
k(к)	8.0	7.0	6.0	5.4	4.4	3.8	3.0	2.2	1.2

Figure 2. GPhFeatLev Levenshtein: Edit distance matrix with *feature vectors* for
robitnyk (робітник) ‘worker’(uk) & *rovesnik* (ровесник) ‘age-mate, of the same age’ (ru)

		r(р)	o(о)	b(б)	i(і)	t(т)	n(н)	y(и)	k(к)
	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0
r(р)	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
o(о)	2.0	1.0	0.0	1.0	2.0	3.0	4.0	5.0	6.0
v(в)	3.0	2.0	1.0	0.8	1.8	2.8	3.8	4.8	5.8
e(е)	4.0	3.0	2.0	1.8	1.2	2.2	3.2	4.2	5.2
s(с)	5.0	4.0	3.0	2.8	2.2	2.0	3.0	4.0	5.0
n(н)	6.0	5.0	4.0	3.8	3.2	3.0	2.0	3.0	4.0
i(и)	7.0	6.0	5.0	4.8	3.8	4.0	3.0	2.2	3.2
k(к)	8.0	7.0	6.0	5.8	4.8	4.6	4.0	3.2	2.2

Figure 3. GPhFeatLev Levenshtein: Edit distance matrix with *hierarchical features* for
robitnyk (робітник) ‘worker’(uk) & *rovesnik* (ровесник) ‘age-mate, of the same age’ (ru)

To address this problem instead of feature vectors we use *hierarchical representations of features*, where a set of central features at the top of the hierarchy needs to be matched first, to allow lower level features to be matched as well (Figure 4).

Figure 4 shows that for the feature hierarchy of the grapheme [b] to match the hierarchy of the grapheme [v] there is a need to match first the grapheme type: consonant (which is successfully matched), and then – a combination of *manner* of articulation and *active* articulation organ (which is not matched, since [b] is plosive and [v] is fricative), and only after that – low level features such as voice may be tried (not matched again, because the higher level feature structure of *manner* + *active* did not match). Note that the proposed hierarchy applies to Ukrainian–Russian language pair, and generalizing it to other translation directions may not work, as relations may need rearrangements of the hierarchy to reflect specific graphonological relations between other languages.

r (р)	['type:consonant', 'voice:sonorant', 'maner:thrill', 'active:fronttongue', 'passive:palatal']
o (о)	['type:vowel', 'backness:back', 'height:mid', 'roundedness:rounded', 'palate:nonpalatalizing']
b (б)	**['type:consonant'**, **'voice:voiced'**, 'maner:plosive', **'active:labial'**, 'passive:bilabial']
i (і)	['type:vowel', 'backness:front', 'height:close', 'roundedness:unrounded', 'palate:nonpalatalizing']
t (т)	**['type:consonant', 'voice:unvoiced'**, 'maner:plosive', **'active:fronttongue'**, **'passive:alveolar'**]
n (н)	['type:consonant', 'voice:sonorant', 'maner:nasal', 'active:fronttongue', 'passive:alveolar']
y (и)	['type:vowel', 'backness:front', 'height:closemid', 'roundedness:unrounded', 'palate:nonpalatalizing']
k (к)	['type:consonant', 'voice:unvoiced', 'maner:plosive', 'active:backtongue', 'passive:velar']

Table 2: Phonological feature vectors for Ukrainian word ‘robitnyk’ (робітник) – ‘worker’ overlapping features in intuitively unrelated characters are highlighted

r (р)	['type:consonant', 'voice:sonorant', 'maner:thrill', 'active:fronttongue', 'passive:palatal']
o (о)	['type:vowel', 'backness:back', 'height:mid', 'roundedness:rounded', 'palate:nonpalatalizing']
v (в)	**['type:consonant', 'voice:voiced'**, 'maner:fricative', **'active:labial'**, 'passive:labiodental']
e (е)	['type:vowel', 'backness:front', 'height:mid', 'roundedness:unrounded', 'palate:palatalizing']
s (с)	**['type:consonant', 'voice:unvoiced'**, 'maner:fricative', **'active:fronttongue'**, **'passive:alveolar'**]
n (н)	['type:consonant', 'voice:sonorant', 'maner:nasal', 'active:fronttongue', 'passive:alveolar']
i (и)	['type:vowel', 'backness:front', 'height:close', 'roundedness:unrounded', 'palate:nonpalatalizing']
k (к)	['type:consonant', 'voice:unvoiced', 'maner:plosive', 'active:backtongue', 'passive:velar']

Table 3: Phonological feature vectors for Russian word ‘rovesnik (ровесник) – ‘age-mate’, ‘of the same age’

r (р)	['type:consonant', 'voice:sonorant', 'maner:thrill', 'active:fronttongue', 'passive:palatal']
a (а)	['type:vowel', 'backness:back', 'height:open', 'roundedness:unrounded', 'palate:nonpalatalizing']
b (б)	['type:consonant', 'voice:voiced', 'maner:plosive', 'active:labial', 'passive:bilabial']
o (о)	['type:vowel', 'backness:back', 'height:mid', 'roundedness:rounded', 'palate:nonpalatalizing']
t (т)	['type:consonant', 'voice:unvoiced', 'maner:plosive', 'active:fronttongue', 'passive:alveolar']
n (н)	['type:consonant', 'voice:sonorant', 'maner:nasal', 'active:fronttongue', 'passive:alveolar']
i (и)	['type:vowel', 'backness:front', 'height:close', 'roundedness:unrounded', 'palate:nonpalatalizing']
k (к)	['type:consonant', 'voice:unvoiced', 'maner:plosive', 'active:backtongue', 'passive:velar']

Table 4: Phonological feature vectors for Russian word ‘rabotnik (работник) – ‘worker’

Consonant feature hierarchy	**Example (pl- prefix on lower level features enforces feature hierarchy)**
Type {Manner+Active} Voice Passive	[b]: ['type:consonant', {'maner:**pl**-plosive', 'active:**pl**-labial',} 'voice:pl-voiced', 'passive:pl-bilabial'

Figure 4. Hierarchical feature representations for consonants: non-matching higher levels prevent from matching at the lower levels

Finally, as the number of features for different graphemes may vary, we compute edit distance between partially matched feature sets as an F-measure between Precision and Recall of their potentially overlapping feature sets, and subtracting it from 1. As a result the measure is symmetric, (4):

$$Prec = len(FeatOverlap) \,/\, len(NofFeatA)$$

$$Rec = len(FeatOverlap) \,/\, len(NofFeatB)$$

$$OneMinusFMeasure = 1 - (2 * Prec * Rec) \,/\, (Prec + Rec)$$

$$\begin{aligned} matrix[zz+1][sz+1] \\ = \min(&\text{matrix}[\text{zz}+1][\text{sz}] + 1, \text{matrix}[\text{zz}][\text{sz}+1] \\ &+ 1, \text{matrix}[\text{zz}][\text{sz}] + OneMinusFMeasure) \end{aligned} \quad (4)$$

In these settings lower cost is given to substitutions; while insertion and deletions incur a relatively higher cost. As a result cognates that have different length are much harder to find using the graphonological Levenshtein edit distance, and in these cases the baseline character-based Levenshtein metric performs better. A general observation is that the feature-based metric can often find cognates inaccessible to character-based metrics, but sometimes misses cognates that involve several insertions, deletions and changing order of graphemes, as shown in Table 5.

2.3. Evaluation sample

We performed evaluation of the baseline Levenshtein metric and our proposed feature-based metric with two settings: one using feature vectors for graphonological representations, and the other – using hierarchically organised features. Evaluation was done on a sample of 300 Ukrainian words selected from 6 frequency bands in the frequency dictionary of lemmas (ranks 1-50, 3001-3050, 6001-6050, 9001-9050, 12001-12050, 15001-15050), Russian cognates were searched in the full-length frequency dictionary of 16,000 entries automatically derived from the Russian corpus (as described in Section 2.1)

uk	***ru***	***GPhFeatLev***	***Baseline Lev***
рішення rishennia 'decision'	решение resheniye 'decision'	**Found**	Missed
сьогодні s'ogodni 'today'	сегодня segodnia 'today'	**Found**	Missed
колгосп kolgosp 'collective farm'	колхоз kolhoz 'collective farm'	**Found**	Missed
коментар komentar 'commentary'	комментарий kommentariy 'commentary'	Missed	**Found**
перерва pererva 'break'	перерыв pereryv 'break'	Missed	**Found**

Table 5. Examples of missed and found cognates for each metric

For 274 out of the 300 Ukrainian words either the baseline Levenshtein metric, or the experimental feature metric returned Russian candidate cognates (with the threshold of

$$\frac{LevDist}{\max\left(len(W1), len(W2)\right)} \leq 0.36$$

applied across all the metrics, as mentioned in Section 2.1. The 274 lists of cognate candidates provided by each metric were then labelled according to the annotation scheme described in Table 1, Section 2.1.

3. Evaluation results

Counts of annotation labels for each of the categories are shown in Table 6 and Table 7. It can be seen from the tables that while the baseline Levenshtein metric outperforms the feature-based metric that uses feature vector graphonological representations, but the feature-based metric outperforms the baseline when hierarchical graphonological feature representations are used. The improvement is about 4% (or nearly 5%, if trivial examples of absolute cognates are discounted). There is no improvement in ranking of found equivalents, which may be due to the noise related to a relatively higher cost of insertions, deletions and reordering of characters.

	per cent	count
0 Difference cognates	16.42%	45
Of which proper nouns	5.84%	16
Have no cognates	34.31%	94
False Friends	1.82%	5
All cognate candidates in sample	**100%**	**274**

Table 6. Parameters of evaluation sample

	Lev		***GPFeat Vectors***		**GPFeat Hierarch**		**Difference: GPFeatHierarchy - Lev**
	per cent	**#**	***per cent***	**#**	**per cent**	**#**	**per cent**
correct, higher better (+exclude 0 differences)	47.08% (36.68%)	129 (84)	*46.72%*	*128*	**51.09% (41.48%)**	140 (95)	**+4.01% (+4.80%)**
missing (lower better)	13.87%	38	*10.58%*	*29*	**9.85%**	27	**+4.02%**
lower rank (lower better)	**2.19%**	6	*10.58%*	*29*	2.55%	7	-0.36%

Table 7. Comparative performance of distance measures for the task of ranking cognates

4. Conclusion and future work

Even though the traditional character-based Levenshtein metric gives a very strong baseline for the task of automated cognate identification from non-parallel corpora, the proposed graphonological Levenshtein edit distance measure outperforms it. Our hierarchically structured feature representations capture linguistically plausible correspondences between cognates much more accurately compared to traditionally used feature vectors. These representations are essential components of the proposed graphonological metric. Feature-based metric often identifies cognates which are missed by the baseline Levenshtein character-based metric.

Different settings of the metrics were compared under our task-based evaluation framework, which requires a relatively small amount of human annotation and can calibrate further developments of the metric and refinements of the feature representation structures. This framework tests the metric directly for its usefulness for the task of creating cognate dictionaries for closely related languages.

For practical tasks both the traditional and feature-based Levenshtein metrics can be used in combination, supporting each other strengths, especially if boosting recall in the cognate identification task is needed.

Graphonological Levenshtein edit distance metric may find applications beyond the task of cognate identification. Future work will include integrating feature-based representations into algorithms for learning phonological and morphosyntactic correspondences between closely-related languages and for automatically deriving morphological variation models for automated grammar induction tasks, with a goal of building large-scale morphosyntactic resources for MT.

Bibliography

Bergsma, S., & Kondrak, G. (2007, September). Multilingual cognate identification using integer linear programming. In RANLP Workshop on Acquisition and Management of Multilingual Lexicons.

Chomsky, N., & Halle, M. (1968). The sound pattern of English. Harper & Row Publishers: New York, London.

Ciobanu, A. M., & Dinu, L. P. (2014). Automatic Detection of Cognates Using Orthographic Alignment. In ACL (2) (pp. 99-105).

Enright, J & Kondrak, G. (2007) A fast method for parallel document identification. Proceedings of Human Language Technologies: The Conference of the North American Chapter of the Association for Computational Linguistics companion volume, pp 29-32, Rochester, NY, April 2007.

Hana, J., Feldman, A., Brew, C., & Amaral, L. (2006, April). Tagging Portuguese with a Spanish tagger using cognates. In Proceedings of the International Workshop on Cross-Language Knowledge Induction (pp. 33-40). Association for Computational Linguistics.

Hubey, M. (1999). Mathematical Foundations of Linguistics. Lincom Europa, Muenchen.

Koehler. R. (1993). Synergetic Linguistics. In: Contributions to Quantitative Linguistics, R. Koehler and B.B. Rieger (eds.), pp. 41-51.

Koehn, P. & Knight, K. (2002). Learning a Translation Lexicon from Monolingual Corpora, , ACL 2002, Workshop on Unsupervised Lexical Acquisition

Leusch, G., Ueffing, N., & Ney, H. (2003, September). A novel string-to-string distance measure with applications to machine translation evaluation. In Proceedings of MT Summit IX (pp. 240-247).

Levenshtein, V. I. (1966). Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady 10 (8): 707–710.

Menzerath, P. (1954). Die Architektonik des deutchen Wortschatzes. Dummler, Bonn.

Mulloni, A., & Pekar, V. (2006). Automatic detection of orthographic cues for cognate recognition. Proceedings of LREC'06, 2387, 2390.

Nerbonne, J., & Heeringa, W. (1997). Measuring dialect distance phonetically. In Proceedings of the Third Meeting of the ACL Special Interest Group in Computational Phonology (SIGPHON-97).

Nießen, S.; F. J. Och; G. Leusch, and H. Ney. (2000) An evaluation tool for machine translation: Fast evaluation for MT research. In Proc. Second Int. Conf. on Language Resources and Evaluation, pp. 39–45, Athens, Greece, May

Pinnis, M., Ion, R., Ştefănescu, D., Su, F., Skadiņa, I., Vasiļjevs, A., Babych, B. (2012) Toolkit for Multi-Level Alignment and Information Extraction from Comparable Corpora // Proceedings of ACL 2012, System Demonstrations Track, Jeju Island, Republic of Korea, 8-14 July 2012.

Sanders, N. C., & Chin, S. B. (2009). Phonological Distance Measures. Journal of Quantitative Linguistics, 16(1), 96-114.

Schepens, J., Dijkstra, T., & Grootjen, F. (2012). Distributions of cognates in Europe as based on Levenshtein distance. Bilingualism: Language and Cognition, 15(01), 157-166.

Serva, M., & Petroni, F. (2008). Indo-European languages tree by Levenshtein distance. EPL (Europhysics Letters), 81(6), 68005.

Sigurd, B., Eeg-Olofsson, M., & Van Weijer, J. (2004). Word length, sentence length and frequency–Zipf revisited. Studia Linguistica, 58(1), 37-52.

Swadesh, M. (1952). Lexico-statistic dating of prehistoric ethnic contacts: with special reference to North American Indians and Eskimos. Proceedings of the American philosophical society, 96(4), 452-463.

Zipf, G. K. (1935). The psycho-biology of language.