incom.py – A Toolbox for Calculating Linguistic Distances and Asymmetries between Related Languages

Marius Mosbach†, **Irina Stenger, Tania Avgustinova, Dietrich Klakow**†
Collaborative Research Center (SFB) 1102: Information Density and Linguistic Encoding
† Spoken Language Systems, Saarland Informatics Campus
Saarland University, Germany
ira.stenger@mx.uni-saarland.de
avgustinova@coli.uni-saarland.de
{mmosbach, dietrich.klakow}@lsv.uni-saarland.de

Abstract

Languages may be differently distant from each other and their mutual intelligibility may be asymmetric. In this paper we introduce incom.py, a toolbox for calculating linguistic distances and asymmetries between related languages. incom.py allows linguist experts to quickly and easily perform statistical analyses and compare those with experimental results. We demonstrate the efficacy of incom.py in an incomprehension experiment on two Slavic languages: Bulgarian and Russian. Using incom.py we were able to validate three methods to measure linguistic distances and asymmetries: Levenshtein distance, word adaptation surprisal, and conditional entropy as predictors of success in a reading intercomprehension experiment.

1 Introduction

1.1 Related Work

Linguistic phenomena may be language specific or shared between two or more languages. With regard to cross-lingual intelligibility, various constellations are possible. For example, speakers of language A may understand language B better than language C, i.e. $[A(B) > A(C)]$ while speakers of language B may understand language C better than language A, i.e. $[B(C) > B(A)]$. For instance, Ringbom (2007) distinguishes between *objective* (established as symmetrical) and *perceived* (not necessarily symmetrical) cross-linguistic similarities. Asymmetric intelligibility can be of linguistic nature. This may happen if language A has more complicated rules and/or irregular developments than language B, which results in structural asymmetry (Berruto, 2004).

Recently, in the INCOMSLAV framework (Fischer et al., 2015; Jágrová et al., 2016; Stenger et al., 2017a,b), measuring methods were developed that are of direct relevance for modelling cross-lingual asymmetric intelligibility. While it has been common to use (modifications of) the Levenshtein distance (Levenshtein, 1966) to predict phonetic and orthographic similarity (Beijering et al., 2008; Gooskens, 2007; Vanhove, 2016), this string-edit distance is completely symmetric. To account for asymmetric cross-lingual intelligibility Stenger et al. (2017b) employ additional measures of conditional entropy and surprisal (Shannon, 1948). Conditional character adaptation entropy and word adaptation surprisal, as proposed by Stenger et al. (2017b), quantify the difficulties humans encounter when mapping one orthographic system to another and reveal asymmetries depending on stimulus-decoder configurations in language pairs.

Similarly, the research of Jágrová et al. (2016) shows that Czech and Polish, both West Slavic, using the Latin script, are orthographically more distant from each other than Bulgarian and Russian, South and East Slavic respectively using the Cyrillic script. Both language pairs have similar lexical distances, however, the asymmetric conditional entropy based measures suggest that Czech readers should have more difficulties reading Polish text than vice versa. The asymmetry between Bulgarian and Russian is very small with a predicted minimal advantage for Russian readers (Stenger et al., 2017b). Additionally Stenger et al. (2017a) found that word-length normalized adaptation surprisal appears to be a better predictor than aggregated Levenshtein distance when the same stimuli sets in different language pairs are compared.

Proceedings of Recent Advances in Natural Language Processing, pages 810–818,
Varna, Bulgaria, Sep 2–4, 2019.
https://doi.org/10.26615/978-954-452-056-4_094

1.2 This paper

To calculate linguistic distances and asymmetries, perform statistical analyses and visualize the obtained results we developed the linguistic toolbox `incom.py`. The toolbox is validated on the Russian-Bulgarian language pair. We focus on word-based methods in which segments are compared at the orthographic level, since orthography is a linguistic determinant of mutual intelligibility which may facilitate or impede reading intercomprehension. We make the following contributions.

1. We provide implementations of various metrics for computing linguistic distances and asymmetries between languages.

2. We demonstrate the use of `incom.py` in an intercomprehension experiment for the Russian-Bulgarian language pair.

3. We show how `incom.py` can be used to validate word adaptation surprisal and conditional entropy as predictors for intercomprehension and discuss benefits over Levenshtein distance.

The remainder of this paper is structured as follows. The considered distance metrics implemented in the `incom.py` toolbox are introduced in Section 2. Section 3 describes the linguistic data used in the experiments. Section 4 presents the evaluation results of the statistical measures and compares them with the intelligibility scores obtained in a web-based cognates guessing. Finally, in Section 5, conclusions are drawn and future developments outlined.

2 Linguistic Distances and Asymmetries

2.1 Distance measures

We start with the introduction of basic notations and present the implemented distance measures.

2.1.1 Notation

Let L denote a language such as Russian or Bulgarian. Each language L has an associated alphabet – a set of characters – $\mathcal{A}(L)$ which includes the special symbol $\varnothing$[1]. We use $w \in L$ to denote a word in language L and $c_i \in w$ to denote the i-th character in word w. Note that while L is a set, w is not and may contain duplicates. Further, we assume the characters $c_i \in w$ are ordered with c_0 being the first and $c_{|w-1|}$ being the last character of word w, where the length $|w|$ of a word w is given by the number of characters it contains, including duplicates. Given two words w_i, w_j, the *alignment* of w_i and w_j results in two new words $\widetilde{w}_i, \widetilde{w}_j$ where $|\widetilde{w}_i| = |\widetilde{w}_j|$. We say character $s_k \in \widetilde{w}_i$ is aligned to character $t_l \in \widetilde{w}_j$ if $k = l$. That is, they occur at the same position.

[1] $\varnothing$ plays an important role when computing alignments. We will also refer to it as *nothing*

2.1.2 Levenshtein distance

Levenshtein distance (LD) (Levenshtein, 1966) is, it its basic implementation, a symmetric similarity measure between two strings – in our case words – $w_i \in L_1$ and $w_j \in L_2$. Levenshtein distance quantifies the number of operations one has to perform in order to transform w_i into w_j. Levenshtein distance allows to measure the orthographic distance between two words and has been successfully used in previous works for measuring the linguistic distance between dialects (Heeringa et al., 2006) as well as the phonetic distance between Scandinavian language varieties (Gooskens, 2007). When computing Levenshtein distance between two words $\mathrm{LD}(w_i, w_j)$, three different character transformations are considered: character deletion, character insertion, and character substitution. In the following we use $\mathcal{T} = \{\mathrm{insert}, \mathrm{delete}, \mathrm{substitute}\}$ to denote the set of possible transformations. A cost $c(t)$ is assigned to each transformation $t \in \mathcal{T}$ and setting $c(t) = 1\ \forall t \in \mathcal{T}$ results in the most simple implementation.

`incom.py` allows computing $\mathrm{LD}(w_i, w_j)$ based on a user-defined cost matrix $\mathbf{M}$, which contains the complete alphabets $\mathcal{A}(L_1), \mathcal{A}(L_2)$ of two languages L_1, L_2 as rows and columns, respectively, as well as the costs for every possible character substitution. That is, for two characters $s \in \mathcal{A}(L_1)$ and $t \in \mathcal{A}(L_2)$, $\mathbf{M}(s, t)$ is the cost of substituting s by t. This user defined cost matrix allows computing linguistically motivated alignments by incorporating a linguistic prior into the computation of the Levenshtein distance. For example, we assign a cost costs of 0 when mapping a character to itself. In case of $\mathbf{M}$ being symmetric, the Levenshtein distance remains symmetric. Along with the edit distance between the two words w_i and w_j our implementation of the Levenshtein distance returns the alignments $\widetilde{w}_i, \widetilde{w}_j$ of w_i and w_j, respectively. Given the length $K = |\widetilde{w}_i|$ of the alignment, we are fur-

ther able to compute the normalized Levenshtein distance $\mathrm{nLD}(w_i, w_j) = \frac{\mathrm{LD}(w_i,w_j)}{K}$. For computing both the alignment and the resulting edit distance `incom.py` uses the Needleman-Wunsch algorithm (Needleman and Wunsch, 1970) following a dynamic-programming approach.

2.1.3 Word adaptation surprisal

Given two aligned words $\widetilde{w}_i$ and $\widetilde{w}_j$, we can compute the *Word Adaptation Surprisal* (WAS) between $\widetilde{w}_i$ and $\widetilde{w}_j$. Intuitively, word adaptation surprisal measures how confused a reader is when trying to translate w_i to w_j character by character. In order to define WAS formally, we introduce the notation of *Character Adaptation Surprisal* (CAS). Given a character $s \in \mathcal{A}(L_1)$ and another character $t \in \mathcal{A}(L_2)$, the character adaptation surprisal between s and t is defined as follows:

$$\mathrm{CAS}(s, t) = -\log_2(P(t \mid s)) \tag{1}$$

Now, the word adaptation surprisal between $\widetilde{w}_i \in L_1$ and $\widetilde{w}_j \in L_2$ can be computed straightforwardly by summing over all characters of the aligned word pair, i.e.

$$\mathrm{WAS}(\widetilde{w}_i, \widetilde{w}_j) = \sum_{k=0}^{K-1} \mathrm{CAS}(s_k, t_k) \tag{2}$$

where $K = |\widetilde{w}_i| = |\widetilde{w}_j|$. Similarly, the normalized word adaptation surprisal is computed as

$$\mathrm{nWAS}(\widetilde{w}_i, \widetilde{w}_j) = \frac{1}{K} \sum_{k=0}^{K-1} \mathrm{CAS}(s_k, t_k) \tag{3}$$

$$= \frac{1}{K} \mathrm{WAS}(\widetilde{w}_i, \widetilde{w}_j) \tag{4}$$

Note that in contrast to Levenshtein distance, word adaptation surprisal is not symmetric.

Computing CAS (and hence also WAS) depends on the conditional probability $P(t|s)$, which is usually unknown. `incom.py` estimates $P(t|s)$ by $\hat{P}(t|s)$ which is based on corpus statistics. Given the alignments of a corpus $\mathcal{C}$ of word pairs produced by the Levenshtein algorithm, we compute $P(t|s)$ by counting the number of times t is aligned with s and divide over the total number of occurrences of character s, i.e.

$$\hat{P}(L_2 = t|L_1 = s) = \frac{count(L_1 = s \wedge L_2 = t)}{count(L_1 = s)} \tag{5}$$

$$\approx P(L_2 = t|L_1 = s) \tag{6}$$

Certainly the quality of the estimate $\hat{P}(L_2 = t|L_1 = s)$ depends on the size of the corpus $\mathcal{C}$. In addition to the corpus based estimated character surprisals, `incom.py` provides functionality to modify the computed CAS values in a manual post-processing step. Based on this, the modified word adaptation surprisal can be computed as:

$$\mathrm{mWAS}(\widetilde{w}_i, \widetilde{w}_j) = \sum_{k=0}^{K-1} \mathrm{mCAS}(s_k, t_k) \tag{7}$$

where mCAS denotes the modified character adaptation surprisal. Similar to using a user defined cost matrix $\mathbf{M}$ when computing the Levenshtein distance, using modified character surprisal allows to incorporate linguistic priors into the computation of word adaptation surprisal.

2.1.4 Conditional entropy

Another asymmetric measure that is supported by our `incom.py` toolbox is *Conditional Entropy* (CE) (Shannon, 1948). Formally, the entropy of a discrete random variable X is defined as the weighted average of the surprisal values of this distribution. As discussed above, we can obtain the character surprisals based on the alignments obtained when computing the Levenshtein distance. Using these surprisal values we can compute the entropy of a language L as

$$H(L) = -\sum_{c \in L} P(L = c) \log_2 P(L = c) \tag{8}$$

In this work we are interested the entropy of a language L_1, e.g. Russian, that we compare to the entropy of another language L_2, e.g. Bulgarian. Thus we compute the conditional entropy between two languages L_1 and L_2.

$$\mathrm{CE}(L_1|L_2) = -\sum_{c_2 \in L_2} P(c_2) H(L_1|L_2 = c_2) \tag{9}$$

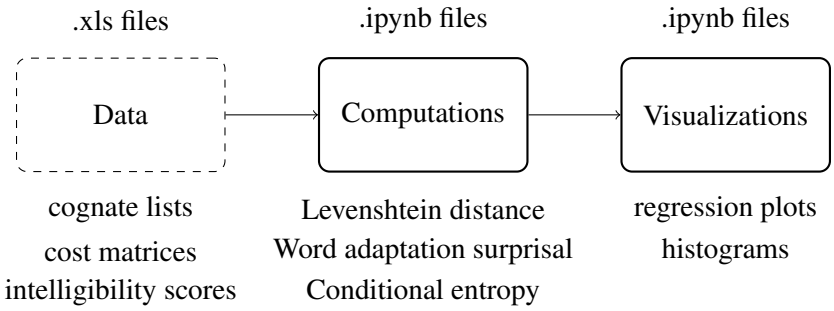


Figure 1: High-level overview of the `incom.py` toolbox.

Intuitively, $\mathrm{CE}(L_1|L_2)$ measures the difficulty for a L_1 reader reading language L_2. Note that similar to the word adaptation surprisal, both entropy and conditional entropy highly depend on the number of available word pairs and only serve as an approximation to the true (unknown) entropy and conditional entropy, respectively.

2.2 `incom.py` toolbox

A high-level overview of the `imcom.py` toolbox is shown in Figure 1. The toolbox is a collection of jupyter notebooks based on the pandas and NumPy libraries[2]. To foster reproducibility and provide a resource for other researchers to easily compute linguistic distances and asymmetries we make `imcom.py` available online https://github.com/uds-lsv/incompy. In addition to computing distances and asymmetries based on a corpus of word pairs, `incom.py` readily supports visualizing the obtained results.

3 Data Sources

3.1 Language material

The Bulgarian (BG) and Russian (RU) data used in this work comes from a collection of parallel word lists consisting of internationalisms, Pan-Slavic vocabulary, and cognates from Swadesh lists. The words belong to different parts of speech, mainly nouns, adjectives, and verbs. We chose to use vocabulary lists instead of parallel sentences or texts in order to exclude the influence of other linguistic factors. The lists, each containing 120 words, were manually adjusted by removing non-cognates by possibly substituting them with etymologically related items, if such could be found, and adding further cognates[3]. Thus, for example, BG–RU ние–мы (nie–my) 'we' was removed and the BG звяр (zvjar) 'beast' instead of животно (životno) 'animal' was added to its RU cognate зверь (zver') 'animal, beast'. In a second step, a cross-linguistic rule set was designed taking into account diachronically motivated orthographic correspondences, e.g. BG–RU: б:бл, ж:жд, я:е, ла:оло etc. Following (Fischer et al., 2015) we apply the rule set to the parallel word lists in a computational transformation experiment and categorized all cognates in the respective pairs as either (i) identical, or (ii) successfully transformed, or (iii) non-transformable by this rule set. The stimuli selection for the online experiments (Section 3.2) is based on the successfully transformed ones: 128 items of a total of 935 (from all lists, excluding doublets). In this way we could exclude possible different derivational morphemes between related languages (e.g. BG–RU хладен–холодный (chladen–cholodnyj) 'cold') in order to focus on the impact of mismatched orthographic correspondences for cognate intelligibility. Even though it may seem artificial to test isolated words, the underlying assumption here is that correct cognate recognition is a precondition of success in reading intercomprehension. If the reader correctly recognizes a minimal proportion of words, he or she will be able to piece the written message together.

3.2 Web-based experiments

The orthographic intelligibility between BG and RU was tested in web-based experiments (http://intercomprehension.coli.uni-saarland.de/en) in which 71 native speakers of BG and 94 native speakers of RU took

[2] https://jupyter.org, https://pandas.pydata.org, https://www.numpy.org

[3] Shared inherited words from Proto-Slavic, shared loans, for example, internationalisms. Cognates are included in the definition; partial cognates are pairs of words which have the same meaning in both languages only in some contexts, for example, BG мъж (măž) 'man, husband' and RU муж (muž) 'husband'.

		Stimuli	
		Bulgarian	Russian
Native	Bulgarian	–	74.67%
	Russian	71.33%	–

Table 1: Intercomprehension scores from free translation tasks performed by humans.

part. The participants started with registration and then completed a questionnaire in their native language. The challenges were presented: 2 with each 60 different BG stimuli in each group for RU speakers and 2 with 60 different RU stimuli in each group for BG speakers. The order of the stimuli were randomized. The participants saw the stimuli on their screen, one by one, and were given 10 seconds[4] to translate each word into RU or into BG. It was also possible to finish before the 10 seconds were over by either clicking on the 'next' button or pressing 'enter' on the keyboard. After 10 seconds the participants saw the next stimulus on their screen. During the experiment the participants received feedback in form of emoticons for their answers. The results were automatically categorized as 'correct' or 'wrong' via pattern matching with pre-defined answers: some stimuli had more than one possible translation and we also provided a list of so-called alternative correct answers. For example, the BG word път (păt) 'way' can be translated in RU as путь (put') or дорога (doroga), so both translations were counted as correct.

The analysis of the collected material[5] is based on the answers of 37 native speakers of Bulgarian (31 women and 6 men between 18 and 41 years of age, average 27 years) and 40 native speakers of Russian (32 women and 8 men between 18 and 71 years of age, average 33 years). The mean percentage of correctly translated items constitutes the intelligibility score of a given language (Table 1).

The results show that there is virtually no asymmetry in written intelligibility between BG and RU: the BG participants understand a slightly larger number of the RU words (74.67%) than the RU participants understand the BG words they are presented with (71.33%). This can be explained by the fact that there are only slight differences between the two languages on the graphic-orthographical level (for more details see (Stenger et al., 2017b)).

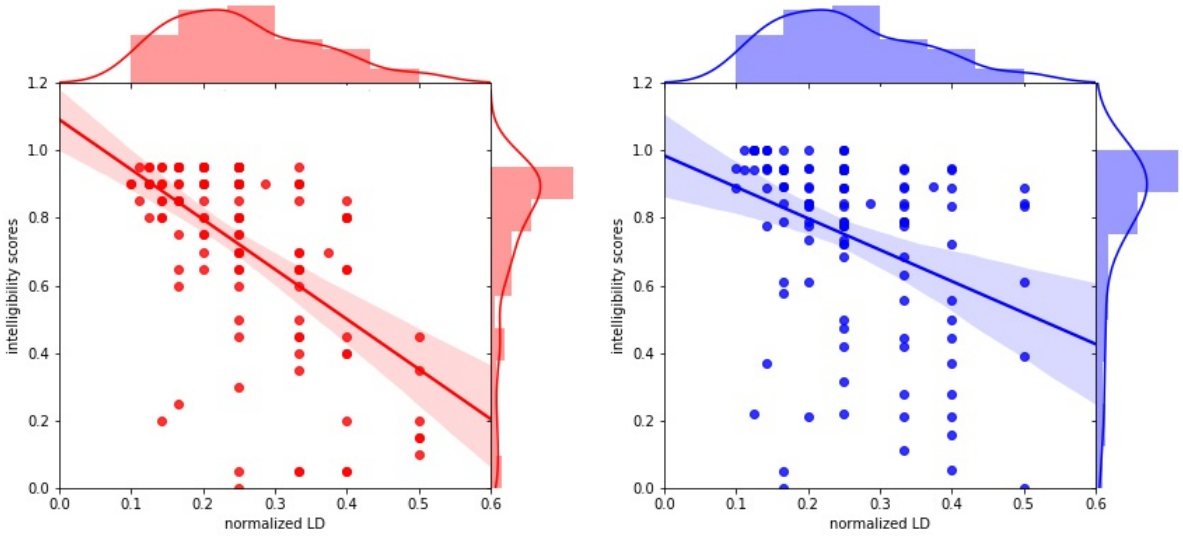
4 Results

4.1 Levenshtein distance and intelligibility score

Using `incom.py` we compute the orthographic LD in both directions and further consider the normalized Levenshtein distance nLD between the 120 BG and RU cognates motivated by the assumption that a segmental difference in a word of two segments has a stronger impact on intelligibility than a segmental difference in a word of ten segments (Beijering et al., 2008; Stenger et al., 2017a). There is a general assumption that the higher the normalized LD, the more difficult it is to translate a given word (Gooskens, 2007; Vanhove and Berthele, 2015; Vanhove, 2016). Thus, we correlate the normalized LD and the intelligibility scores from our experiments for both language pairs. The correlation results are presented in Figure 2. We find a correlation between orthographic distance (normalized LD) and the intelligibility of BG words for RU readers of $r = -0.57$ ($p = 1.4e-11$) and $r = -0.36$ ($p = 6.3e-05$) for BG readers. Both correlations are significant and confirm the above hypothesis. However, the LD accounts for only 32% ($R^2 = 0.32$) of the variance in the intelligibility scores for RU readers and for only 13% ($R^2 = 0.13$) of the variance in the intelligibility scores for BG readers, leaving the majority of variance unexplained. Recall from Section 2 that LD is a symmetric measure, and therefore it does not capture any asymmetries between correspondences. If, for instance, the RU vowel character а always corresponds to а for a BG reader, but in the other direction, BG а can correspond to а, о or я for a RU reader, then a measure of linguistic distance is required to reflect both this difference in adaptation possibilities and the uncertainty involved in transforming а. Such asymmetries are effectively captured by the next two intelligibility measurements of word adaptation surprisal and conditional entropy, both of which are implemented in the `incom.py` tool-

[4]The time limit is chosen based on the experience from other reading intercomprehension experiments. The allocated time is supposed to be sufficient for typing even the longest words, but not long enough for using a dictionary or an online translation tool.

[5]For the present study we exclude those participants who have indicated knowledge of the stimuli language(s) in the questionnaire and analyze the results only of the initial challenge for each participant in order to avoid any learning effects.



(a) BG-RU (b) RU-BG

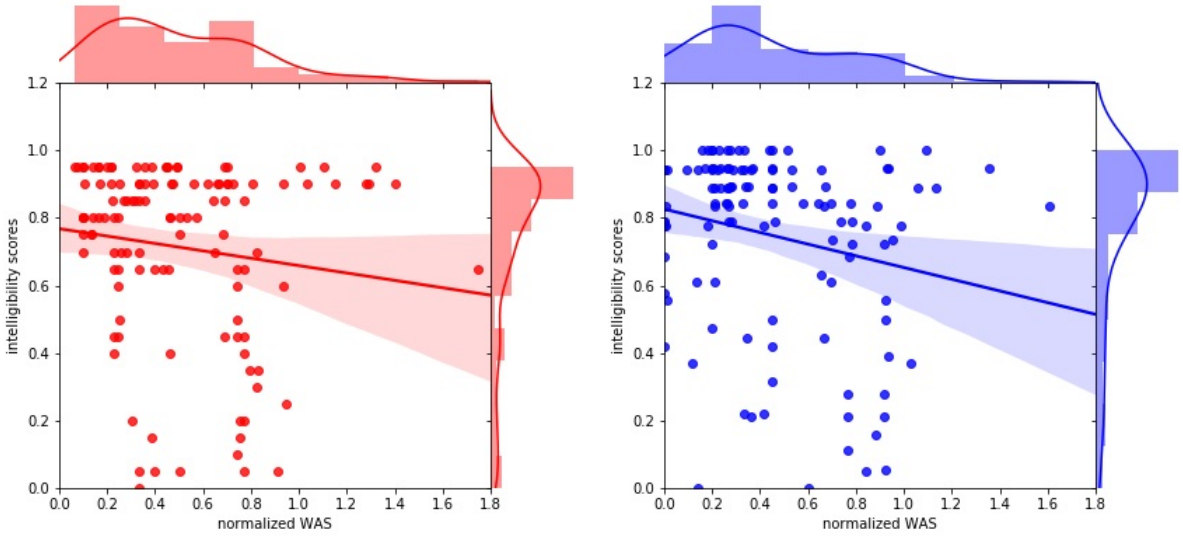
Figure 2: Normalized Levenshtein distance as a predictor for intelligibility. 2a Shows Russian native speakers reading Bulgarian. 2b Shows Bulgarian native speakers reading Russian.

box.

4.2 Word adaptation surprisal and intelligibility score

Word adaptation surprisal (WAS), in particular the normalized word adaptation surprisal (nWAS), helps us to predict and explain the effect of mismatched orthographic correspondences in cognate recognition. We assume that the smaller the normalized WAS, the easier it is to guess the cognate in an unknown, but (closely) related language. The correlation between the normalized WAS and the intelligibility scores is displayed in Figure 3. We find a low but significant negative correlation ($r = –0.22, p < 0.05$) between nWAS and written word intelligibility for BG readers. However, the negative correlation ($r = –0.13$) between nWAS and written word intelligibility for RU readers is not significant ($p = 0.14$). This can be explained by the fact that WAS values are given in bits and depend heavily on the probability distribution used.

Recall that with `incom.py`, we get the character adaptation surprisal (CAS) from our character adaptation probabilities (see Section 2 above). CAS and WAS values allow quantifying the unexpectedness both of individual character correspondences and of the whole cognate pair. This gives a quantification of the overall unexpectedness of the correct cognate. However, identical orthographic correspondences may still have a small surprisal value, for example, from a RU perspective the correspondence a:a has a surprisal value of 0.5986 bits resulting in an increase of the WAS value. Thus, we decided to manually modify our WAS calculation in such a way that all identical orthographic correspondences are measured with 0 bits. The calculated CAS values for mismatched orthographic correspondences remain unchanged in the modified calculation. Using the modified word adaption surprisal, we find a negative significant correlation between the modified nWAS and written word intelligibility also for RU readers ($r = –0.21, p < 0.05$). However, the modified nWAS accounts only for 12% ($R^2 = 0.123$) of the variance in the intelligibility scores for BG readers and the modified nWAS accounts for less than 5% ($R^2 = 0.044$) of the variance in the intelligibility scores for RU readers. This leaves the question why the correlation at the cognate level is so low. A possible explanation is that a cognate in an unknown closely related language will be easier to understand as it is more similar to the cognate in one's own language, because each cognate pair may have its own constellation of factors, affecting intelligibility, where one factor may overrule another factor, e.g., the number of orthographic neighbors in one's own language that



(a) BG-RU

(b) RU-BG

Figure 3: Normalized word adaptation surprisal as a predictor for intelligibility. 3a Shows Russian native speakers reading Bulgarian. 3b Shows Bulgarian native speakers reading Russian.

are very similar to the stimulus, the number of mismatched orthographic correspondences in the stimulus and their position, the word frequency in one's own language, the word length of stimulus etc. The estimated character values seem not to exactly reflect this constellation.

4.3 Conditional entropy and intelligibility score

For the BG–RU language pair the difference in the full conditional entropies (CE) is very small: 0.4853 bits for the BG to RU transformation and 0.4689 bits for the RU to BG transformation, with a very small amount of asymmetry of 0.0164 bits. These results predict that speakers of RU reading BG words are more uncertain than speakers of BG reading RU words. This is in accordance with the experimental results where the language combination with the slightly higher CE (RU speakers reading BG) had a slightly lower intelligibility score (see Table 1). Thus, CE can be a reliable measure when explaining even the small asymmetry in the mutual intelligibility.

Using `incom.py` we calculated entropy values of BG and RU characters in order to analyse asymmetries on the written (orthographic) level in more details. Figure 4 shows the entropy values of 6 BG characters е, ъ, а, щ, и, я, and the special symbol ∅ for RU readers and the entropy values of 5 RU characters о, е, я, у, л for BG readers (on the right). Note that the alignment of ∅ to any other character c corresponds to the case where Russian readers have to fill in a character. The entropy calculations reveal that, for example, BG readers should have more uncertainty with the RU vowel character о, while RU readers should have more difficulties with the adaptation of the BG vowel character е. This means that the mapping of the RU о to possible BG characters is more complex than the opposite direction. More precisely, the RU о can map into 4 BG vowel characters (о, а, ъ, е) or to *nothing* (∅), the BG е can map into 3 RU vowel characters (е, ё, or я). Certainly, in an intercomprehension scenario a BG or a RU reader does not know these mappings and the respective probabilities. However, the assumption is that the measure of complexity of the mapping can be used as an indicator for the degree of intelligibility (Moberg et al., 2007), because it reflects the difficulties with which a reader is confronted in 'guessing' the correct correspondence. Our experimental results indeed show that BG readers have greater problems with the RU о than RU readers with the BG character а or *nothing* (∅) in cognate pairs like RU–BG холод – хлад (cholod – chlad) 'cold', борода – брада (boroda – brada) 'beard', ворона – врана (vorona – vrana) 'crow'.

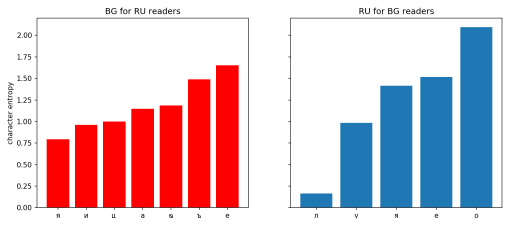


Figure 4: Character entropy values when translating from Russian to Bulgarian and vice versa.

5 Discussion and Outlook

Previous research in reading intercomprehension has shown that (closely) related languages may be differently distant from each other and their mutual intelligibility may be asymmetric. In this paper we present `incom.py` – a toolbox for computing linguistic distances and asymmetries. With `incom.py` we perform experiments on measuring and predicting the mutual intelligibility of Slavic languages, as exemplified by the language pair Bulgarian-Russian by means of the Levenshtein distance, word adaptation surprisal, and conditional entropy. Using a small corpus of parallel cognate lists we validated linguistic distances and asymmetries as predictors of mutual intelligibility based on stimuli obtained from written intelligibility tests. The results of our statistical analyses clearly support normalized Levenshtein distance as a reliable predictor of orthographic intelligibility at the word level for both language pairs tested. However, we find that only 32% (for RU readers) and 13% (for BG readers) of the variance in the intelligibility data is explained by the orthographic similarity quantified by means of the normalized Levenshtein distance. We find that the predictive power of the Levenshtein distance is different within the two language pairs. It must be mentioned here that the RU stimuli are in general longer (5.09 characters) than the BG stimuli (4.61 characters). Thus, the BG readers should intuitively delete more characters while the RU readers should add more characters in order to guess the correct cognate.

Previous research has shown that deletions and additions, the basic operations performed when computing Levenshtein distance, are not of equal value in the mutual intelligibility: it appears that deletions are more transparent for the participants in terms of subjective similarity than additions (Kaivapalu and Martin, 2017). This means that there is room for improvement in our orthographic distance algorithm. Word adaptation surprisal measures the complexity of a mapping, in particular, how predictable the particular correspondence in a language pair is. The surprisal values of correspondences are indeed different. However, they depend on their frequency and distribution in the particular cognate set. Most important and in contrast to Levenshtein distance, surprisal can be asymmetric. The character adaptation surprisal values between language A and language B are not necessarily the same as between language B and language A. This indicates an advantage of the surprisal-based method compared to Levenshtein distance. Our results show that the predictable potential of word adaptation surprisal was rather weak despite its modification. We assume that word adaptation surprisal should to a larger extent take into account relevant factors in reading intercomprehension, for example, orthographic neighbors (words that are very similar to the stimulus word and differ only in one character). Something we keep as future work.

Conditional entropy can reflect the difficulties humans encounter when mapping one orthographic system on another. The underlying hypothesis is that high predictability improves intelligibility, and therefore a low entropy value should correspond to a high intelligibility score. This result is as we expected. We have calculated conditional entropy for Bulgarian and Russian using a cognate word list from intelligibility tests. In our experiments, conditional entropy – like the intelligibility task – reveals asymmetry between Bulgarian and Russian on the orthographic level: the conditional entropy in Bulgarian for Russian readers is slightly higher than the conditional entropy in Russian for Bulgarian readers. This means that the slightly higher entropy is found in the language pair where there is slightly lower intelligibility. Thus, we were able to show that conditional entropy can be a reliable measure when explaining small asymmetries in intelligibility. In future work we plan to extend `incom.py` with additional functionality to compute distances and asymmetries on the phonological level. Additionally, it might be interesting to consider the morphological level which has been shown to be helpful when processing words for humans with limited reading abilities (Burani et al., 2008).

Acknowledgments

We would like to thank Klára Jágrová and Volha Petukhova for their helpful feedback on this paper. Furthermore, we thank the reviewers for their valuable comments. This work has been funded by Deutsche Forschungsgemeinschaft (DFG) under grant SFB 1102: Information Density and Linguistic Encoding.

References

Karin Beijering, Charlotte Gooskens, and Wilbert Heeringa. 2008. Predicting intelligibility and perceived linguistic distance by means of the Levenshtein algorithm. *Linguistics in the Netherlands* 25(1).

Gaetano Berruto. 2004. Sprachvarietät-sprache (Gesamtsprache, historische Sprache) .

Cristina Burani, Stefania Marcolini, Maria De Luca, and Pierluigi Zoccolotti. 2008. Morpheme-based reading aloud: Evidence from dyslexic and skilled italian readers. *Cognition* 108(1).

Andrea Fischer, Klára Jágrová, Irina Stenger, Tania Avgustinova, Dietrich Klakow, and Roland Marti. 2015. An orthography transformation experiment with Czech-Polish and Bulgarian-Russian parallel word sets. *Natural Language Processing and Cognitive Science 2015 Proceedings, Libreria Editrice Cafoscarina, Venezia* .

Charlotte Gooskens. 2007. The contribution of linguistic factors to the intelligibility of closely related languages. *Journal of multilingual and multicultural development* 28(6).

Wilbert Heeringa, Peter Kleiweg, Charlotte Gooskens, and John Nerbonne. 2006. Evaluation of string distance algorithms for dialectology. In *Proceedings of the workshop on linguistic distances*. Association for Computational Linguistics.

Klára Jágrová, Irina Stenger, Roland Marti, and Tania Avgustinova. 2016. Lexical and orthographic distances between Bulgarian, Czech, Polish, and Russian: A comparative analysis of the most frequent nouns. In *Language Use and Linguistic Structure: Proceedings of the Olomouc Linguistics Colloquium*.

Annekatrin Kaivapalu and Maisa Martin. 2017. Perceived similarity between written Estonian and Finnish: Strings of letters or morphological units? *Nordic Journal of Linguistics* 40(2).

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. In *Soviet physics doklady*. volume 10.

Jens Moberg, Charlotte Gooskens, John Nerbonne, and Nathan Vaillette. 2007. Conditional entropy measures intelligibility among related languages. *Proceedings of Computational Linguistics in the Netherlands* .

Saul B Needleman and Christian D Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology* 48(3).

Håkan Ringbom. 2007. *Cross-linguistic similarity in foreign language learning*, volume 21. Multilingual Matters.

Claude Elwood Shannon. 1948. A mathematical theory of communication. *Bell system technical journal* 27(3).

Irina Stenger, Tania Avgustinova, and Roland Marti. 2017a. Levenshtein distance and word adaptation surprisal as methods of measuring mutual intelligibility in reading comprehension of slavic languages. In *Computational Linguistics and Intellectual Technologies: International Conference 'Dialogue 2017'Proceedings*. volume 16.

Irina Stenger, Klára Jágrová, Andrea Fischer, Tania Avgustinova, Dietrich Klakow, and Roland Marti. 2017b. Modeling the impact of orthographic coding on Czech–Polish and Bulgarian–Russian reading intercomprehension. *Nordic Journal of Linguistics* 40(2).

Jan Vanhove. 2016. The early learning of interlingual correspondence rules in receptive multilingualism. *International Journal of Bilingualism* 20(5).

Jan Vanhove and Raphael Berthele. 2015. Item-related determinants of cognate guessing in multilinguals. *Crosslinguistic influence and crosslinguistic interaction in multilingual language learning* .