Multi-lect automatic detection of Swadesh list items from raw corpus data in East Slavic languages

Ilia Afanasev
University of Vienna
ilia.afanasev.1997@gmail.com

Abstract

The article introduces a novel task of multi-lect automatic detection of Swadesh list items from raw corpora. The task aids the early stage of historical linguistics study by helping the researcher compile word lists for further analysis.

In this paper, I test multi-lect automatic detection on the East Slavic lects' data. The training data consists of Ukrainian, Belarusian, and Russian material. I introduce a new dataset for the Ukrainian language. I implement data augmentation techniques to give automatic tools a better understanding of the searched value. The test data consists of the Old East Slavic texts.

I train HMM, CRF, and mBERT models, then test and evaluate them by harmonic F1 score. The baseline is a Random Forest classifier. I introduce two different subtasks: the search for new Swadesh list items, and the search for the known Swadesh list items in new lects of the well-established group. The first subtask, given the simultaneously diverse and vague nature of the Swadesh list, currently presents an almost unbeatable challenge for machine learning methods. The second subtask, on the other hand, is easier, and the mBERT model achieves a 0.57 F1 score. This is an impressive result, given how hard it is to formalise the token belonging to a very specific and thematically diverse set of concepts.

1 Introduction

The need for automatic tools that can aid human researchers has been pressing in computational linguistics for at least the last two decades (Mackay and Kondrak, 2005). There are turnkey solutions for the word list data (Jäger and Sofroniev, 2016; Jäger et al., 2017; Nath et al., 2022). However, when a researcher starts working with a new lect from scratch, they usually have nothing but raw data, from which they must extract this kind of a word list. This is where computational technologies may assist the researcher in the earlier stage of the study: they may execute preliminary detection of tokens that are of special interest – the Swadesh list items (Holman et al., 2008).

In this paper, I present a task of multi-lect automatic detection of Swadesh list items from raw corpus data. Swadesh list, named after its creator, Morris Swadesh, is a list of basic concepts that generally are universal among the human languages and may be used for historical linguistics purposes (Borin, 2012). I want to test, whether the computer can grasp the vague concept of *swadeshness* (Dellert et al., 2020), if even human researchers often struggle with its formalisation. I define swadeshness by the following set of criteria:

- **Historical stability**: lexical items that express Swadesh list concepts remain relatively unchanged during the history of language.
- **Frequency**: generally, Swadesh list concepts-expressing lexical units are among the more frequent ones of the language. However, it is a tendency, not a law. There is no distinct correlation, and by no means frequency should be considered the ultimate criterion (Burlak, 2021).
- **Stylistic neutrality**: concepts that represent Swadesh list items do not have a tendency to appear in a specific register or in statements with a specific sentiment.
- **Syntactic independence**: lexical items that express Swadesh list concepts should remain in the language not as a part of a bigger collocation, such as proverb (Kassian et al., 2010).
- **Semantic preciseness**: a member of the Swadesh list should have a distinct, easily identifiable meaning.

The multi-lect automatic detection of Swadesh list items from raw corpora is challenging. The tool

Proceedings of the 4th Workshop on Computational Approaches to Historical Language Change, pages 76–86
December 6, 2023 ©2023 Association for Computational Linguistics

(a rule-based, statistical, or neural network-based model) should be able to perform it zero-shot and from the first attempt: otherwise, human researchers are not going to need it at all. Ideally, the model should be able to grasp the concept of swadeshness and become proficient enough to perform the task on the languages, the relations of which to the others are completely unknown. Such a model ideally should be at the forefront, laying the groundwork for a human researcher. However, currently, automatic tools are not able to efficiently zero-shot detect Swadesh list items in the raw corpus of a randomly given language. It is only reasonable to start with an easier task, detecting Swadesh list items in the language for which there is a strong hypothesis of its genetic relationships. To carry out this detection, a researcher needs raw corpus material from this language and a model trained on the material of the language's hypothetical relatives. Thus, the task of multi-lect automatic detection of Swadesh list items from raw corpus data transforms into the task of multi-lect automatic detection of Swadesh list items from raw corpus data of a particular language group.

I propose to start with the East Slavic lects. In this paper, I use the term *lect* instead of dialect and/or language to denote any distinct variety without imposing any hierarchy, which generally distracts from the variation study. This is particularly relevant in the case of the Slavic group due to the political circumstances of the last three decades.

The East Slavic group seems especially well-fit for the task because a group is quite a small unit of language classification, for which the concept of swadeshness may be easier to grasp. The East Slavic group possesses some rather big corpora for both modern and historical data. I intend to train the models on the modern East Slavic data from different lects (Ukrainian, Russian and Belarusian) and to zero-shot test them on the historical data. I want to try different models, both simple probability-based tools and complex large linguistic models (LLMs).

1.1 Contributions

- I present a novel task of multi-lect automatic detection of Swadesh list items from raw corpora and its two subtasks: the search for new Swadesh list items and the search for the known Swadesh list items in new lects of a well-established group.
- I propose possible solutions for this task which achieve the highest score one may require from the computer, given that even the formalisation of swadeshness is quite hard for humans, as the definition I provide is far from being comprehensive.
- I prepare a new dataset for Ukrainian in the Universal Dependencies format, currently possessing silver morphological tagging, lemmatisation, and dependency parsing, performed with Stanza toolkit (Qi et al., 2018, 2020).

1.2 Paper structure

The second section is dedicated to the previous research, including works on automatic cognate detection, possible architectures, and evaluation in NLP. In the third section, I describe the dataset for the training models and the dataset to test them against. The fourth section includes a step-by-step description of the research method, including the architectures of the models I use and the metrics utilised to inspect the quality of their performance. In the fifth section, I report the results of the experiments. The conclusion provides an overall analysis of how well the models fulfilled the task and proposes possible ways to enhance their performance in the future.

2 Related Work

The desire to automatically extract Swadesh list items from new data manifested itself in historical linguistics almost as soon as the computing powers became sufficient for this type of task (Mackay and Kondrak, 2005). Generally, it falls within the greater historical linguistics trend of implementing computational methods as researcher's assistants (Dellert, 2019). HMM models are some of the most frequent solutions due to their simple yet effective architecture and overall dominance across the NLP horizon; with PairHMMs, adapted for working with parallel data (Wieling et al., 2007), being the most widely used. Further steps are connected with different techniques in multiple sequence alignment (List, 2012) and sequence comparison (List, 2014). After that, the automatic cognate detection and classification as a task emerges (Jäger and Sofroniev, 2016; Jäger et al., 2017; Nath et al., 2022). The methods to extract large Swadesh lists in the context of multi-lingual databases appear at this time (Dellert and Buch, 2016) and simultaneously the multi-lingual datasets for them to be

tested on arise (Dellert et al., 2020). The formalisation of *swadeshness* has become an important part of the discussion in recent years (Dellert and Buch, 2016).

The multi-lect automatic detection of Swadesh list items requires other approaches, as it utilises raw corpus data rather than lexical databases. One such approach is part-of-speech tagging. Part-of-speech tagging is mostly dominated by universal methods, based on recurrent neural networks (Qi et al., 2018) (Qi et al., 2020). Yet the tasks conducted on different language varieties demand agile models that can both be tuned for the needs of a specific tagset and work in the context of low-resourced and sparse data (Scherrer, 2021). Hidden Markov Model (HMM)-based taggers present this opportunity (Schmid, 1994, 1995; Özçelik et al., 2019; Lyashevskaya and Afanasev, 2021). The other probabilistic tool used for part-of-speech tagging is conditional random fields (CRF) (Behera, 2017). Both these methods are regularly applied in the context of historical linguistics and language variation (Mackay and Kondrak, 2005; Wieling et al., 2007; Gillin, 2022; Camposampiero et al., 2022). CRF is also used in named entity recognition, where it is rivalled by methods based on the use of transformer models (Yang et al., 2021). Historical linguistics study often requires efficient resource utilisation. This fits the current NLP trend that gave rise to the distilled and tiny versions of transformers (Sanh et al., 2019).

Historical data is usually quite low-resourced, which provides an additional challenge to the detection of sparsely distributed Swadesh items. This requires using special metrics for imbalanced data (Dudy and Bedrick, 2020). The harmonic F1 score, traditionally used for such cases (Chinchor, 1992), still finds its application in the analysis of NLP tasks (Scherrer, 2021).

3 Data

The data consists of two subsets of different sizes and coming from different languages, one used to train the models and to test them on the first subtask, the search for new Swadesh list items, and another – for the second subtask, to test their performance on completely new material. Both datasets are stored in Universal Dependencies (UD) format (Zeman; et al, 2022). I use UD format as it contains information on the lemma, which makes it significantly easier to prepare the datasets for the experiments.

The first subset is a large Modern East Slavic multi-lect dataset. It was vital to maintain the balance between these groups for the model to learn as many features of Swadesh list items across the East Slavic lects as possible. I call the main principle of balance a parent-node one, which means that the amount of data from the lects under the same node (i.e., sharing the last common ancestor) should be approximately equal. For instance, in the case of this research, it means that Ukrainian and Belarusian, the closest relatives out of the three present lects, should have the same amount of tokens on their part. Russian, the sole representative of their sister group, should be presented with a corpus of the same size.

The first corpus I use, the Belarusian-HSE corpus (Shishkina and Lyashevskaya, 2022), consists of 305,000 tokens of different genres, such as fiction (including poetry), legal texts, non-fiction, news texts, Wikipedia, social networks texts.

Ukrainian UD (IU) corpus consists of only 122,000 tokens[1], so I need more data. For this purpose, I take the ua-gec corpus (Syvokon and Nahorna, 2021) and tag it with the existing Stanza model (Qi et al., 2018, 2020), acquiring silver data in UD format [2]. I get 183,961 samples of this corpus, and thus the Ukrainian-Belarusian branch of East Slavic remains in balance.

The Russian corpus Taiga (Shavrina and Shapovalova, 2017) consists of 197,000 tokens and is represented by a diverse set of genres, including poetry, fiction, non-fiction, Wikipedia, blogs, social media, and news. Taiga is designed to represent syntactic features of Russian lexical units (obviously, taking in Swadesh list items) in the best possible way.

To balance the Russian branch with the Ukrainian-Belarusian branch, I add data from SynTagRus (Droganova et al., 2018), a 1.5 million corpus of fiction, news, and non-fiction. I take 395,431 tokens, so the training corpus may achieve the balance.

One may point out that this makes the dataset imbalanced in favour of the Russian lect. However, it balances the Russian branch of the East Slavic tree with the Ruthenian branch, while the Ruthenian branch is still balanced within itself. This follows

[1] https://github.com/UniversalDependencies/UD_Ukrainian-IU/tree/master

[2] https://huggingface.co/datasets/djulian13/Swadesh-list-tagged-East-Slavic

Table 1: General characteristics of the training dataset.

Dataset	**Language**	**Token number**
IU	Ukrainian	122,000
UA-GEC	Ukrainian	183,961
Belarusian-HSE	Belarusian	305,000
Taiga	Russian	197,000
SynTagRus	Russian	395,431
Overall	**Various Slavic**	**1,203,392**

the historical-comparative principle of step-by-step reconstruction (see, for instance, Starostin (2019)). We illustrate this with Figure 1.

The corpora of the training dataset and their key features are presented in Table 1.

The test dataset is the two corpora of historical East Slavic lects, the Old East Slavic TOROT corpus (Eckhoff and Berdicevskis, 2015), containing nearly 246,000 tokens. The TOROT corpus is predominantly later Old East Slavic (Belarusian, Ukrainian and Russian ancestral lect continuum) and partly Middle Russian (when it split from Ukrainian and Belarusian) material. Its texts are mostly legal documents and non-fiction (chronicles). Old East Slavic being the ancestral form for all three modern East Slavic languages (thus containing within different texts proto-Belarusian, proto-Ukrainian, and proto-Russian features) is the main reason I use every one of them, and not only Russian.

Both datasets are additionally preprocessed to prepare them for the task. They are assigned a label *c* (non-Swadesh list item) or *i* (Swadesh list item). I use the 40-item Swadesh list (Holman et al., 2008), enriching it with some concepts from the 110-item list (Kassian et al., 2010), namely, *woman*, *kill*, *eat*, *all*, *man*, *me*, and *you (indirect)* (genitive stem). I chose these particular concepts as they are semantically close to the concepts of the 40-item list: *woman* to *breast*, *kill* to *die*, *eat* to *drink*, *all* to *full*, *man* to *person*, *me* to *I*, and *you (indirect)* to *you*. Hopefully, this aids the models to better grasp the semantic component of swadeshness. I tag each possible morphological form of Swadesh list items. Genitive stems of *you* and *I*, *you (indirect)* and *me* respectively, get the treatment of separate concepts. Yet this does not mean that I use only base forms for all the concepts in the dataset, as the East Slavic languages are highly inflective. I would risk losing a lot of forms, tagging only base forms as Swadesh list items. In this fashion, all the forms of *I* (я) and *me* (меня, мне, and мной) have an *i* (Swadesh list item) label. While picking the exact lexical item for a concept, I generally follow guidelines by Kassian et al. (2010).

The training dataset, while quite big, does not contain a lot of contexts for Swadesh list items for the model to learn on. The fully automatic generation of new examples, contrarily to grammatical error detection, currently seems impossible. However, I apply artificial augmentation, using token-level 3-grams that provide minimal left and right context. This is an approach that part-of-speech studies successfully implement (Lyashevskaya and Afanasev, 2021).

I wrote a script that generates 3-grams for each instance of the Swadesh list item in the text. These may be represented as *c i c*, where *i* is a Swadesh list item, and *c* is used for any other token, including *[CLS]* (this denotes fragment-starting token) and *[EOS]* (this denotes fragment-finishing token). An item of the dataset thus contains the original sentence and its labels and generated 3-grams with their labels. The script is also used for the test dataset. Artificial augmentations of the test dataset are not going to be used in the evaluation, as they may seemingly boost results for a poorer-performing model, and compromise the intention of evaluating the model on the raw data. Generated datasets are available on HuggingFace [3].

4 Method

4.1 Task

I treat multi-lect automatic Swadesh list items detection as a sequence labelling and information extraction task, placing it among the part-of-speech tagging (Behera, 2017) and named entity recognition (Tjong Kim Sang and De Meulder, 2003) tasks, as it shares common features (a clear split of all the items into categories with part-of-speech-tagging and a heavy imbalance of two classes with named entity recognition) with both. One may see it as a reduced information extraction task with the extracted entity restricted to a single token, or as an unbalanced sequence labelling task with two labels, one of which is significantly less frequent than another. These different ways imply using

[3] https://huggingface.co/datasets/djulian13/Swadesh-list-tagged-East-Slavic

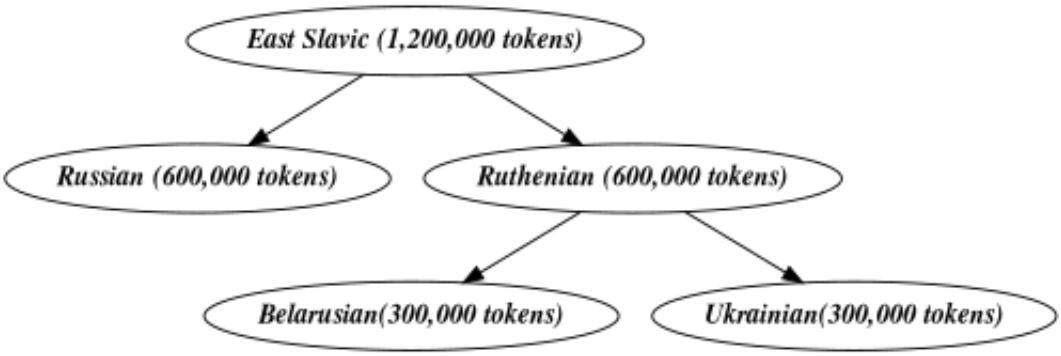

Figure 1: Application of step-by-step reconstruction principle to the training corpora size. On each historical division, the token number is equal between lects or groups of lects.

particular methods for both creating the tool and its evaluation.

Whether one frames the task as a reduced information extraction task or an unbalanced sequence labelling task, one should use metrics that fit the case of unbalanced classes the most. I propose to use the traditional harmonic F1 score between precision (the number of correctly predicted items of a particular class, divided by the number of all items) and recall (the number of correctly predicted items of a particular class, divided by the number of items that belong to this class) (Chinchor, 1992). The formula for harmonic F1 score is given in (1).

$$F = 2\frac{PR}{P+R} \tag{1}$$

I am going to provide information on precision and recall to present a clearer picture. As an evaluation method, I use only the F1 score for the Swadesh list items, as the average F1 score and F1 score for non-Swadesh list items, the dominating class, are going to be very high, and, at the same time, not informative.

4.2 Baseline

If I treat multi-lect automatic Swadesh list items detection as a sequence labelling task, the optimal methods are the ones used for part-of-speech tagging. Otherwise, if one sees the task as an information extraction one, the models, generally used for named entity recognition, are suitable.

Our intention to build the model able to generalise its knowledge on the previously unknown lects poses additional restrictions, making the use of rule-based methods, adjusted for a specific lect or set of lects, hard and probably not worthy of implementation. The possible tool is going to be based on machine learning methods.

As a baseline method, I use a random forest (Ho, 1995) classifier that utilises frequency (absolute and relative as different parameters), one of the most easily Swadesh list item quantifiable properties. The only tweaked parameter of classifier is random state, set to 1590.

4.3 Statistical methods

The first method I propose is a simple Hidden Markov Model (HMM), originally designed for part-of-speech tagging (Özçelik et al., 2019). It is a state machine that predicts the next state on the basis of the previous ones (Warjri et al., 2019). The particular implementation is enhanced with the Viterbi algorithm. Viterbi algorithm enhances HMM's ability to find the most likely tag sequence (Prajapati and Yajnik, 2019). The Hidden Markov Model nowadays almost never achieves state-of-the-art result quality and is not exactly well-adjusted for the unbalanced classification. However, it often demonstrates the ability to generalise on low-resourced heterogeneous datasets, sometimes exceeding modern state-of-the-art multi-lingual transformer neural networks (Lyashevskaya and Afanasev, 2021). This paper does not utilise any specific training setup, other than the one used in Lyashevskaya and Afanasev (2021)

Conditional Random Fields (CRF) is a model that also often performs part-of-speech tagging (Behera, 2017) and named entity recognition (Jie and Lu, 2019). This model is based on computing the probabilities, which makes it similar to HMM, though some detailed implementations are different (Behera, 2017). CRF is a simple statistical tool, yet these currently demonstrate high results after slight augmentations, often competing with recurrent and transformer neural networks: it is especially relevant in non-standard conditions (Gillin,

2022) (Camposampiero et al., 2022). There is no specific parameter tuning for CRF: preprocessing includes adding special tokens for marking start and end of sentences, and training parameters are mostly default. The final set is the following:

- L-BGFS as gradient descent method,
- L1 regularisation coefficient = 0.25,
- L2 regularisation coefficient = 0.3,
- maximum number of iterations is 100,
- generation of transition features for all possible combinations of attributes and labels. This is especially important, as there are only two classes, and one heavily outweighs another. It is extremely necessary for the model to get the grasp of what Swadesh list items are not, not only what they are.

4.4 BERT

I also fine-tune multilingual cased BERT-base (Devlin et al., 2018) on the data, as one may fine-tune it for the task of named entity recognition (NER). Transformers are nowadays often used for this kind of task, showing state-of-the-art results (Yang et al., 2021). I do not implement the hierarchical architecture of (Yang et al., 2021) designed for nested named entity recognition. As Swadesh list items are not nested ones, the advantages it gives are not going to be useful.

NER is a much simpler task than swadeshness detection, and there is a high probability that the model used for NER may fail, yet this is probably the best shot there is. Models trained for other tasks, such as machine translation (MT), may become confused even more. They aim at direct transformation, while NER models grasp a concept, and thus, hopefully, will not only learn to find the known Swadesh list items but the ones the model does not know beforehand as well.The model trains for 1 epoch with batch size being equal to 1, due to the hardware restrictions.

The code for each of these models is present on GitHub [4].

4.5 Swadesh list split

I split the prepared Swadesh list into two halves presented in Table 2. The parts are designed for the model to be able to at least partially rely on vectorised semantics and syntactic behaviour, with pairs such as *come - path*, *one - two*, *ear - hear*. This is the motivation behind the addition of items to list (Holman et al., 2008). Not all the concepts find a pair (*name*), and some pairs, such as *horn - nose* may prove not as informative as one hopes. I also try to assign an equal amount of part-of-speech items to each part of the dataset.

5 Experiments and Results

The experiments start with splitting the modern dataset into three parts, α, β, and ω. ω is a full dataset, α and β contain sentences that include only tokens from the A part of the Swadesh list split, or the B one, respectively. I then augment each of the datasets with 3-gram addition. I train each architecture - HMM, CRF, BERT (but the baseline, random forest classifier) - separately on α, α-augmented, β, β-augmented, ω, and ω-augmented. The historical test dataset is not split, and later I refer to it as γ.

I cross-validate α- and β-trained models. This is the first subtask, the search for new Swadesh list items, and here the models are not going to show a high F1 score, as it is a hard task even for a human.

For the second subtask, I test the ω-trained model with the γ dataset. Here the results should be better, as there are obvious graphical similarities between modern and historical Swadesh list concepts, and their semantic and syntactic stability possibly may allow for an easier capture of historical Swadesh list concepts.

The models' results comparison should lead to the discussion of possible reasons why the model with the best performance was the most successful and why others failed.

5.1 Unknown Swadesh list items identification

The results of the experiments are in tables 3 and 4.

I provide only aggregated results, as with error rates this high there is no sense in the analysis of each concept precision/recall/F1-score. The numbers are going to be too low for us to get any valuable insights. I also do not attempt to simultaneously identify a token as a particular concept in addition to marking it as bearing swadeshness.

It is clearly easier for the models to predict β tokens than α. Mostly, this is due to the semantic closeness of *woman* and *person* concepts to *man*, and words that are very close to *one* (α-list) in β. It

[4] https://github.com/The-One-Who-Speaks-and-Depicts/SlavNLP-23

Table 2: Swadesh list split.

Half	Concepts
α	*come*, *ear*, *see*, *fire*, *hand*, *horn*, *I*, *leaf*, *mountain*, *skin*, *one*, *star*, *tongue*, *louse*, *breast*, *die*, *drink*, *full*, *man*, *you (indirect)*, *blood*, *fish*, *name*, *new*, *night*, *we*
β	*path*, *hear*, *eye*, *water*, *knee*, *nose*, *you*, *tree*, *stone*, *liver*, *two*, *sun*, *tooth*, *dog*, *woman*, *kill*, *eat*, *all*, *person*, *me*, *bone*

Table 3: Results on β-dataset of all the models trained on α-dataset rounded to the third decimal place. Best results here and afterwards are in **bold**.

Model	Precision	Recall	F1
Baseline	0	0	0
HMM	0.123	0.036	0.056
HMM (3-gram-augmented data)	0.02	0.036	0.026
CRF	0.011	0.003	0.005
CRF (3-gram-augmented data)	0.009	0.003	0.005
BERT	**0.795**	**0.082**	**0.149**
BERT (3-gram-augmented data)	0.5	0	0

Table 4: Results on α-dataset of all the models trained on β-dataset rounded to the third decimal place.

Model	Precision	Recall	F1
Baseline	0.01	0.004	0.005
HMM	0.034	0.012	0.018
HMM (3-gram-augmented data)	0.034	0.012	0.018
CRF	0	0	0
CRF (3-gram-augmented data)	0	0	0
BERT	**0.379**	**0.02**	**0.36**
BERT (3-gram-augmented data)	0.231	0	0

also seems that models may deduce that concept *eye* belongs to the Swadesh list.

Augmentation directly leads to overfitting, as the models trained on augmented datasets experience a significant drop in quality. HMM is probably the least influenced one, it seems to be heavily resistant to this kind of noise. Despite that, its precision gets down on β-dataset prediction.

The baseline model, a random forest classifier that is aware only of frequencies, is unable to predict new Swadesh tokens appearing in the dataset, which supports the theory that frequency is not a determining factor in choosing candidates for addition to lexicostatistical lists. There are, however, some words that may be interesting: месяц 'month', вы 'you (plural)', both from basic vocabulary lists. The baseline model clearly fails in the subtask - on the familiar data it achieves a much more optimistic 0.91 F1-score. In the same fashion CRF fails: it is good at memorising the exact tokens, not in generalisation over them.

The HMM model performs significantly better. HMM yet again proves that its simplistic design is exceptionally well-suited for classification tasks. In β-dataset, it detects наш 'our' that shares root with *we (indirect)*, a genitive stem of *we*, and хадзіць 'go', an aspectual pair for *come*. HMM also makes mistakes, tagging frequent words (such as м 'm') as Swadesh list items.

BERT is by far the best-performing model - probably, due to it being context-oriented, and thus able to grasp such properties of Swadesh list items as syntactic independence, stylistic neutrality, and semantic preciseness. It still has a low F1 score and its recall is not exactly high, but this is probably one of the best shots that a computer may have for a prediction of such a vague category. It also detects concepts, which are similar to the ones from the 110-item Swadesh list (Kassian et al., 2010), for instance, *somebody* (хтось is similar in form to хто 'who', a concept from the list).

Table 5: Results on γ-dataset of all the models trained on ω-dataset.

Model	Precision	Recall	F1
Baseline	0	0	0
HMM	0.384	0.36	0.371
HMM (3-gram-augmented data)	0.384	0.36	0.371
CRF	0.045	0.014	0.022
CRF (3-gram-augmented data)	0.045	0.014	0.021
BERT	0.734	**0.459**	**0.565**
BERT (3-gram-augmented data)	**0.737**	0.01	0.02

5.2 Swadesh list items identification for unknown lects

The results of the search for Swadesh list items in Old East Slavic texts are presented in Table 5.

The baseline score remains the same. It is probably due to the differences in size between ω- and γ-datasets and the distribution patterns of modern and historical East Slavic lects tokens. CRF architecture also lags behind the other models, barely beating the baseline.

Augmentation technique harms the results of Swadesh list items identification for unknown lects in a similar manner that it harms the results of the unknown Swadesh list items identification in the known lects. HMM yet again resists its negative effects, but the other models (even CRF, though slightly) do not.

Overall, the scores are significantly better than for the previous subtask. There are still choices that one may treat as mistakes. For instance, the model labels есми 'be-PRES.1.PL' as a Swadesh list item. At the same time, they find some tokens that may present interest as a potential Swadesh list material, for instance, ноць 'night'. Picking есми 'be-PRES.1.PL' here is more of an error, it is just very much alike to Ukrainian ми 'we'. However, ноць 'night' is a more interesting case: it is a historically stable, more or less frequent, stylistically neutral, syntactically independent and semantically precise unit. It is a Swadesh list concept (Kassian et al., 2010) in the East Slavic languages, and the model successfully discovered it. Cases like this prove that models generally may grasp the concept of swadeshness.

BERT performs the best out of all, mostly due to its ability to grasp the behaviour of the Swadesh list items and not their exact form. One additional explanation is that East Slavic languages are quite closely related, having started to split approximately 600 - 1,000 years ago (Starostin, 1989). BERT's F1 score steps over 0.5, which I see as a huge achievement, given the complexity and vagueness of the task presented even for humans (Burlak, 2021).

6 Conclusion

Automatic tools demonstrate modest yet inspiring results, achieving a maximum of 0.56 F1 score on the tokens they are familiar with in unfamiliar languages and a maximum of 0.15 F1 score on unfamiliar tokens in the familiar lects. This seems quite promising, as the Swadesh list items is a very sparsely distributed class of lexical units. The average probability of encountering them in raw text (across 1000 random samples, 100 lexical units each) is 0.02 for ω-dataset and 0.04 for γ-dataset. BERT outcompetes probabilistic tools, HMM and especially CRF, as it grasps the deep core properties of Swadesh list items, namely, syntactic independence, stylistic neutrality, and semantic preciseness. HMM, though, is the most stable one in terms of resisting the noise in the data. All the models perform better at memorising tokens than at generalising over the concept of swadeshness. This may still aid the search for concepts that are expressed by the forms most stable across the span of time, such as pronouns. They even sometimes find completely new candidates for Swadesh list items, such as *night*. Unfortunately, one still needs to deal with each case manually when a model labels something as a Swadesh list item. Effective evaluation systems are yet to appear. As for automatic evaluation, the last resort is still checking against an existing list.

Data augmentation, restricted to 3-gram generation from sentences, is harmful to both the probabilistic tools and the transformer models. It definitely leads to overfitting.

For the automatic tools to aid human researchers better, further enhancements must be provided in the future. The extension of the datasets and the im-

plementation of new, effective data augmentation techniques, such as providing quantified information on the described features of Swadesh list items, are required. It seems crucial to add verification of the method on other language groups, not only East Slavic.

The task may also be approached with other methods based on other NLP tasks. I believe that at least a random forest classifier will become a much better baseline with information on syntactic independence, semantic precision, and stylistic neutrality. The other models are also going to benefit from this kind of feature engineering.

7 Acknowledgements

I owe Olga Lyashevskaya and George Starostin for insightful discussion on the nature of swadeshness. I am grateful to anonymous reviewers from VarDial 2022, Slavic NLP 2023, and CNLPS 2023, whose comments helped to shape the research in its current form. I thank the anonymous reviewers from the LChange'23 conference for their useful feedback. The remaining errors are my own.

References

Pitambar Behera. 2017. An experiment with the CRF++ Parts of Speech (POS) tagger for Odia. *Language in India*, 17:18–40.

Lars Borin. 2012. *Core Vocabulary: A Useful But Mystical Concept in Some Kinds of Linguistics*, pages 53–65. Springer Berlin Heidelberg, Berlin, Heidelberg.

Svetlana Burlak. 2021. Stability and frequency: is there a correlation? *Journal of Language Relationship*, 19(3-4):293–307.

Giacomo Camposampiero, Quynh Anh Nguyen, and Francesco Di Stefano. 2022. The curious case of logistic regression for Italian languages and dialects identification. In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 86–98, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Nancy Chinchor. 1992. MUC-4 evaluation metrics. In *Fourth Message Uunderstanding Conference (MUC-4): Proceedings of a Conference Held in McLean, Virginia, June 16-18, 1992*.

Johannes Dellert. 2019. *Information-theoretic causal inference of lexical flow*. Language Science Press.

Johannes Dellert and Armin Buch. 2016. Using computational criteria to extract large Swadesh lists for lexicostatistics. In *Proceedings of the Leiden Workshop on Capturing Phylogenetic Algorithms for Linguistics*, Tübingen. Universitätsbibliothek Tübingen.

Johannes Dellert, Thora Daneyko, Alla Münch, Alina Ladygina, Armin Buch, Natalie Clarius, Ilja Grigorjew, Mohamed Balabel, Hizniye Isabella Boga, Zalina Baysarova, et al. 2020. NorthEuralex: A wide-coverage lexical database of Northern Eurasia. *Language resources and evaluation*, 54(1):273–301.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Kira Droganova, Olga Lyashevskaya, and Daniel Zeman. 2018. Data conversion and consistency of monolingual corpora: Russian UD treebanks. In *Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories (TLT 2018)*, pages 52–65, Oslo University, Norway. Linköping University Electronic Press.

Shiran Dudy and Steven Bedrick. 2020. Are some words worth more than others? In *Proceedings of the First Workshop on Evaluation and Comparison of NLP Systems*, pages 131–142, Online. Association for Computational Linguistics.

Hanne Eckhoff and Aleksandrs Berdicevskis. 2015. Linguistics vs. digital editions: The tromsø old russian and ocs treebank. *Scripta & e-Scripta*, 14-15:9–25.

Nat Gillin. 2022. Is encoder-decoder transformer the shiny hammer? In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 80–85, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Tin Kam Ho. 1995. Random decision forests. In *Proceedings of 3rd International Conference on Document Analysis and Recognition*, volume 1, pages 278–282 vol.1.

Eric Holman, Søren Wichmann, Cecil Brown, Viveka Velupillai, André Müller, and Dik Bakker. 2008. Explorations in automated language classification. *Folia Linguistica*, 42:331–354.

Gerhard Jäger, Johann-Mattis List, and Pavel Sofroniev. 2017. Using support vector machines and state-of-the-art algorithms for phonetic alignment to identify cognates in multi-lingual wordlists. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1205–1216, Valencia, Spain. Association for Computational Linguistics.

Gerhard Jäger and Pavel Sofroniev. 2016. Automatic cognate classification with a support vector machine. In *Conference on Natural Language Processing*.

Zhanming Jie and Wei Lu. 2019. Dependency-guided LSTM-CRF for named entity recognition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language*

Processing (EMNLP-IJCNLP), pages 3862–3872, Hong Kong, China. Association for Computational Linguistics.

Alexei Kassian, George Starostin, Anna Dybo, and Vasiliy Chernov. 2010. The Swadesh wordlist. An attempt at semantic specification. *Journal of Language Relationship*, 16(59):46–89.

J.-M. List. 2014. *Sequence Comparison in Historical Linguistics*. Walter de Gruyter GmbH & Co KG.

Johann-Mattis List. 2012. LexStat: Automatic detection of cognates in multilingual wordlists. In *Proceedings of the EACL 2012 Joint Workshop of LINGVIS & UNCLH*, pages 117–125, Avignon, France. Association for Computational Linguistics.

Olga Lyashevskaya and Ilia Afanasev. 2021. An HMM-based PoS Tagger for Old Church Slavonic. *Journal of Linguistics/Jazykovedný casopis*, 72(2):556–567.

Wesley Mackay and Grzegorz Kondrak. 2005. Computing word similarity and identifying cognates with Pair Hidden Markov Models. In *Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005)*, pages 40–47.

Abhijnan Nath, Rahul Ghosh, and Nikhil Krishnaswamy. 2022. Phonetic, Semantic, and Articulatory Features in Assamese-Bengali Cognate Detection. In *Proceedings of the Ninth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 41–53, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Rıza Özçelik, Gökçe Uludoğan, Selen Parlar, Özge Bakay, Özlem Ergelen, and Olcay Taner Yıldız. 2019. User Interface for Turkish Word Network KeNet. In *2019 27th Signal Processing and Communications Applications Conference (SIU)*, pages 1–4.

Manisha Prajapati and Archit Yajnik. 2019. POS Tagging of Gujarati Text using VITERBI and SVM. *International Journal of Computer Applications*, 181(43):32–35.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. 2018. Universal Dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Yves Scherrer. 2021. *Adaptation of Morphosyntactic Taggers*, Studies in Natural Language Processing, page 138–166. Cambridge University Press.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of International Conference on New Methods in Language Processing*, Manchester, UL.

Helmut Schmid. 1995. Improvements in Part-of-Speech Tagging with an Application to German. In *Proceedings of the ACL SIGDAT-Workshop*, Dublin, Ireland.

Tatiana Shavrina and Olga Shapovalova. 2017. To the methodology of corpus construction for machine learning: «Taiga» syntax tree corpus and parser. In *Proceedings of the International Conference "CORPORA 2017"*, Saint-Petersbourg, Russia.

Yana Shishkina and Olga Lyashevskaya. 2022. Sculpting Enhanced Dependencies for Belarusian. In *Analysis of Images, Social Networks and Texts*, pages 137–147, Cham. Springer International Publishing.

George Starostin. 2019. *Reply to Pozdniakov' paper "On the threshold of relationship*, pages 215–220. Gorgias Press, Piscataway, NJ, USA.

Sergei A. Starostin. 1989. Sravnitel'no-istoricheskoe jazykoznanie i leksikostatistika. In *Lingvisticheskaja rekonstrukcija i drevnejshaja istorija Vostoka*, pages 3–39.

Oleksiy Syvokon and Olena Nahorna. 2021. Ua-gec: Grammatical error correction and fluency corpus for the ukrainian language.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Sunita Warjri, Dr. Partha Pakray, Saralin Lyngdoh, and Arnab Maji. 2019. Identification of POS Tag for Khasi language based on Hidden Markov Model POS Tagger. *Computación y Sistemas*, 23.

Martijn Wieling, Therese Leinonen, and John Nerbonne. 2007. Inducing sound segment differences using pair hidden markov models. In *Proceedings of ninth meeting of the acl special interest group in computational morphology and phonology*, pages 48–56.

Zhiwei Yang, Jing Ma, Hechang Chen, Yunke Zhang, and Yi Chang. 2021. HiTRANS: A hierarchical transformer network for nested named entity recognition. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 124–132, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Daniel Zeman; et al. 2022. Universal dependencies 2.11. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL),

Faculty of Mathematics and Physics, Charles University.