Benchmarking Pre-trained Language Models for Multilingual NER: TraSpaS at the BSNLP2021 Shared Task

Marek Šuppa*
Comenius University in Bratislava
marek@suppa.sk

Ondrej Jariabka*
Comenius University in Bratislava
o.jariabka@gmail.com

Abstract

In this paper we describe TraSpaS, a submission to the third shared task on named entity recognition hosted as part of the Balto-Slavic Natural Language Processing (BSNLP) Workshop. In it we evaluate various pre-trained language models on the NER task using three open-source NLP toolkits: character level language model with Stanza, language-specific BERT-style models with SpaCy and Adapter-enabled XLM-R with Trankit. Our results show that the Trankit-based models outperformed those based on the other two toolkits, even when trained on smaller amounts of data. Our code is available at https://github.com/NaiveNeuron/slavner-2021.

1 Introduction

This paper describes the TraSpaS submission to the third shared task of the Balto-Slavic Natural Language Processing (BSNLP) Workshop at EACL 2021. The task focuses on recognizing named entities (NER), their normalization and linking across six Slavic languages. The participants are provided training data comprised of news articles crawled from the web, centered around four specific topics and labeled on the document level, while the testing data consists of news articles focused on two topics, which are completely different from those included in the training data. This setup poses a significant research challenge, as it features languages with Latin as well as Cyrillic scripts, substantial class imbalance in the training data, specific set of named entity tags (which makes it difficult to find comparable datasets) as well as document-level annotation, which does not map directly to the more generally used token-level annotation and requires custom preprocessing. Our solution utilizes three open-source NLP toolkits, namely **Tra**nkit (Nguyen et al., 2021), **spa**Cy and **S**tanza (Qi et al., 2020), yielding the name TraSpaS. We chose these particular toolkits for their popularity, ability to exploit large pre-trained language models for the NER task, as well as their potential to be used in production deployments.

The research question we are trying to pose in this work can be paraphrased as follows: can the universal open-source NLP toolkits (such as SpaCy, Stanza or Trankit) be used to achieve competitive performance on Multilingual Named Entity Recognition? This has been tested to some extent in the previous edition of the shared task with NLP-Cube[1], albeit with not very encouraging results: on average it reached F1 scores of around 0.2 whereas the top three best performing systems regularly reported F1 scores of over 0.85. We hypothesize that the situation has changed significantly since then, mostly thanks to large pre-trained Transformer-based language models, which are being made accessible thanks to Huggingface's Transformers (Wolf et al., 2020). Hence, we believe that the customized setup that was previously required in order for models like BERT to be used may no longer be necessary and that competitive performance can be obtained using an off-the-shelf toolkit. Furthermore, this comparison sheds more light on multilingual transfer learning, as Stanza uses LSTM-based pre-trained language models and both spaCy and Trankit generally utilize variants of BERT. Moreover, since spaCy and Stanza defaults to using single-source models (i.e. one model per each language) and Trankit works primarily in the multi-source setup, our experiments allow us to compare these two approaches as well. We further investigate the impact of various tokenizers and additional data on final performance and perform a detailed error analysis.

*These authors contributed equally to the work

[1] https://github.com/adobe/NLP-Cube

Proceedings of the 8th BSNLP Workshop on Balto-Slavic Natural Language Processing, pages 105–114
April 20, 2021.

Our results indicate that the multi-source approach implemented using Trankit yielded the strongest performance, both on the development as well as the test set. The impact of additional data is pronounced in the case of single-source approach but does not lead to significant increase in performance of multi-source systems. The error analysis shows that all of the presented system struggle with the EVT (events) tag, which is the least populated in the training set. The highest error rate is between the PRO (products) and ORG (organizations) tags. We speculate that this might be due to many ambiguities between these two domains, such as "Uber", a product of the Uber company or "Ryanair Choice", a specific product which contains the name of the Ryanair airline. Some of these errors might be caused by the disambiguation rules outlined in the Task Guidelines.

2 Related Work

The Balto-Slavic NLP Workshop has previously organized two shared tasks. In the first one, held in 2017, no training data was provided and as reported in (Piskorski et al., 2017), due to the relatively high complexity of the task, only two teams have submitted the results within the task's deadline. The second one was held in 2019 and featured document-level annotated data in four languages (Bulgarian, Czech, Polish and Russian) on four specific topics, two of which were released as training set, with the remained serving as the test set. As (Piskorski et al., 2019) reports, most teams participating in this task used embeddings provided by the BERT model (Devlin et al., 2018), often in its multilingual version, coupled with cross-lingual training (Tsygankova et al., 2019) and CRF (Arkhipov et al., 2019) or NCRF++ (Emelyanov and Artemova, 2019) as the top-most layer. The work presented in (Tsygankova et al., 2019) is of particular note, as the authors show that multi-source training outperforms the single-source approach.

As one of the standard tasks of Natural Language Processing (NLP), Named Entity Recognition (NER) has been extensively studied in the past. In recent years, performance improvements have been obtained by the introduction of the Conditional Random Field (CRF) (Manning et al., 2014), especially when combined with neural representation of the input text, yielding the BiLSTM-CRF model (Lample et al., 2016). Further improvements can be attributed to the widespread use of contextual embeddings (Howard and Ruder, 2018; Peters et al., 2018; Devlin et al., 2018; Akbik et al., 2018) provided by large-scale language models. An example of such a model in the multilingual setup would be multilingual BERT (Devlin et al., 2018), which was trained on the text of top 104 largest Wikipedias. Its performance on the NER task has recently been improved by XLM (Lample and Conneau, 2019) which leverages cross-lingual language model pre-traning, and XLM-R (Conneau et al., 2019) which applies this pre-training approach to CommonCrawl data, which is significantly larger than data obtained from Wikipedia.

The increasing proliferation of open-source NLP toolkits has significantly lowered the barrier for inclusion of sophisticated language processing tools in larger systems. These include CoreNLP (Manning et al., 2014), FLAIR (Akbik et al., 2019), UDPipe (Straka, 2018), NLP Cube (Boroș et al., 2018) and spaCy[2], which is focused on industry usage and hence optimized for efficiency. Historically, the NLP toolkits have primarily supported a few major languages but recently the trend has reversed, with toolkits aiming to support the processing of multilingual text. Stanza (Qi et al., 2020) is one such toolkit, which provides pretrained models for 66 languages. These models are generally based on BiLSTM architecture, which is then amended for the target downstream task. Trankit (Nguyen et al., 2021) is a recently introduced toolkit based on the XLM-R language model, which also makes use of Adapters architecture (Pfeiffer et al., 2020). This allows the XLM-R model to be reused across languages and only a small number of parameters to be finetuned for a specific language and/or component of the toolkit.

3 The BSNLP2021 Shared Task

The BSNLP2021 Shared Task is the third installment of the Multilingual Named Entity Challenge in Slavic languages, organized as part of the 8th Workshop on Balto-Slavic Natural Language Processing. Similarly to its previous editions, it focuses on recognizing named entities in text of Web news articles, their normalization and linking across languages.

3.1 Dataset

In contrast to its previous editions, the dataset for this edition has been comprised of six languages:

[2] https://spacy.io

	Asia Bibi						Brexit						Nord Stream						other	Ryanair					
	BG	CS	PL	RU	SL	UK	BG	CS	PL	RU	SL	UK	BG	CS	PL	RU	SL	UK	SL	BG	CS	PL	RU	SL	UK
Documents	101	86	86	118	4	6	600	284	499	153	52	50	130	160	151	150	74	40	97	86	161	144	147	52	63
PER	976	712	863	933	41	43	3 539	1 159	3 120	1 523	651	319	350	567	540	375	742	64	934	168	136	148	67	138	35
LOC	543	400	504	664	27	60	4 690	1 541	4 912	853	552	607	1 425	1 981	1 952	1 730	2 018	608	765	431	976	946	995	527	586
ORG	231	247	336	455	9	36	4 366	1 460	5 141	1 035	533	193	649	590	1 009	941	804	613	753	283	1 035	817	842	618	462
PRO	70	46	57	48	3	1	499	265	845	140	35	18	396	451	955	536	343	8	176	74	74	122	74	148	20
EVT	17	3	13	1	0	0	1 813	546	1 426	445	375	208	7	19	14	5	54	15	64	6	12	6	0	11	0
Total Tags	1 837	1 408	1 773	2 101	80	140	14 907	4 971	15 444	3 996	2 146	1 345	2 827	3 608	4 470	3 587	3 961	1 308	2 692	962	2 233	2 039	1 978	1 442	1 103
Unique Tags	305	254	416	311	43	80	903	897	2 115	516	513	198	375	637	674	547	818	289	1 341	272	390	437	354	587	166
Ratio	6.02	5.54	4.26	6.76	1.86	1.75	16.51	5.54	7.3	7.74	4.18	6.79	7.54	5.66	6.63	6.56	4.84	4.53	2.01	3.54	5.73	4.67	5.59	2.46	6.64
Tokens	30 632	20 708	26 042	27 909	1 678	1 818	187 009	59 325	203 066	36 757	31 129	14 354	31 313	33 768	42 905	39 124	44 624	15 020	57 611	18 407	35 634	30 258	25 703	28 898	13 790
Unique Tokens	4 323	3 481	4 826	4 283	637	712	13 340	9 649	23 102	5 519	6 683	2 856	4 475	6 170	7 131	6 327	7 363	3 435	13 150	3 608	5 190	5 503	3 930	6 757	2 375

Table 1: Summary statistics of the BSNLP2021 Shared Task dataset.

Bulgarian, Czech, Polish, Russian, Slovene and Ukrainian. Three of these languages use Latin script (Czech, Polish and Slovene), with the remaining three being written in Cyrillic script. The dataset contains both the training and the test data from the previous edition of the Shared Task. These relate to news stories about a varied set of topics: Asia Bibi (a Pakistani woman involved in a blasphemy case), Brexit, Ryanair and Nord Stream. These were selected such that their coverage could be expected across multiple languages. In the interest of completeness we also mention that the dataset also contains a topic denoted Other, which features news labelled text articles only in Slovene.

The training data is annotated with five Named Entity classes: person names (PER), locations (LOC), organizations (ORG), products (PRO) and events (EVT). The annotations are provided on the document level, in the form of a surface form along with the corresponding tag, the normalized form an the link to the cross-lingual concept. Note that this form of annotation does not provide token-level information and may lead to ambiguities during conversion to more common BIO tagging formats.

The summary statistics of the dataset can be found in Table 1. The distribution of documents shows that Brexit is by far the largest topic, having as much as 4.5x (in case of Bulgarian) documents as the second largest one. With regards to the document distribution across languages, we can conclude that it is on a similar level, except for Slovene (SL) and Ukrainian (UK), which tend to feature less documents across all topics as compared to the other four languages. This is especially pronounced in the case of Asia Bibi with only four and six documents written in Slovene and Ukrainian, respectively. This is particularly interesting, as the Asia Bibi topic has been used as the development set in previous work (Tsygankova et al., 2019; Emelyanov and Artemova, 2019). Despite losing the ability to compare directly with prior work, the presented statistics suggest that using a different topic as the development set may better approximate the test data (two sets of documents, each related to a specific topic, different from the topics in the existing training data sets).

The distribution of tags also highlights an imbalance in the provided training data. While the PER, LOC and ORG tags are distributed relatively evenly, the PRO and EVT tags have less examples in the training data. For instance, the Asia Bibi topic does not contain EVT-tagged tokens in Slovene and Ukrainian, whereas the Ryanair topic is missing them for Russian and Ukrainian. We therefore expect the recognition accuracy (as measured by the F-measure) to be lower for these two tags. This is in line with the results of the second edition of the BSNLP2021 Shared Task (Piskorski et al., 2019), where even the best model's performance on a specific language was markedly worse for these two tags.

3.2 Evaluation Metrics

When it comes to evaluation of NER systems, the most commonly used metric is the token-level F1 score, which further expects token-level annotations. Since data provided in this shared task are annotated on the document level a different type of metric needs to be used. The task definition[3] specifies two evaluation types that are to be used to evaluate this task: relaxed and strict evaluation.

Relaxed evaluation An entity is thought to be extracted correctly if the system response includes at least one annotation of a named mention of this entity. Whether the extracted entity is base form or not, does not play a role. Depending on whether partial matches count or the exact string must match, this type of evaluation can be executed with **partial match** or **exact match** criteria.

Strict evaluation The system response needs to capture and list all variant of an entity in order for it to be considered extracted correctly. Unless

[3] http://bsnlp.cs.helsinki.fi/shared-task.html

otherwise stated, all NER metrics in this document were computed via strict evaluation.

3.3 Subtasks

The shared task features three separate subtasks: Recognition, Lemmatization and Linking. We primarily focus on the Recognition part of the task by using the provided data to train custom models, as described in Section 4. For Lemmatization we use the pre-trained models that are provided by the toolkits in question. Since spaCy provides pre-trained Lemmatization models only for Russian and Polish, we use the surface form as the default lemma in all the other languages. As the Linking subtask is out of our focus, we return the same entity ID (`ORG-RAND`) for each detected entity.

4 Experimental Setup

4.1 Preprocessing

Since the training data provided, as part of the shared task, is annotated on the document level, a conversion step into a more commonly used format is necessary. To do so, we first need to split the raw text of each document into sentences and tokens. We therefore use NLTK (Bird, 2006) as well as pre-trained tokenizers from Stanza and Trankit. As NLTK does not provide a tokenizer for Bulgarian, Russian and Ukrainian, we use the English tokenizer for these languages instead. Further discussion on the effect of various tokenizers can be found in Section 6.2.

After tokenization, we proceed to convert the document-level annotations to token-level annotations (BIO format), which can be readily consumed by all three considered NLP toolkits. We do so by iterating over the annotations for each document and assigning token-level tags belonging to the exact matches of the named entity found in the respective raw document. Any token that was not covered by this procedure was deemed to be "other" (`O`). This approach may lead to two types of errors

1. Tagging a token with a specific tag when in reality it was of the "other" (`O`) class.

2. Tagging a token with a wrong tag.

The first type of error is difficult to assess without manual inspection, as it can only take place when depending on context the same surface form is and is not considered to be a named entity. Given the rich morphology of the Slavic languages we are dealing with, we believe this kind of error will be quite rare and can safely be ignored.

The second type of error can happen in cases when the same surface form is assigned multiple distinct named entity tags. In our analysis we found that this happened in only 155 cases in the whole training dataset. As such, we concluded this type of error would have minimal effect on the final performance and in cases we were detected multiple named entity tags to be associated with the same surface form the first occurrence was used in the conversion procedure outlined above.

During preprocessing we also noticed that despite it not being mentioned explicitly, in at least one example[4] the annotated surface form could not be found in the document's raw text but was located in its title. To cope with this, we consider the title to be part of the document text as well and tag it with the procedure outlined above as well.

4.2 Training and Development sets

As discussed in Section 3.1, given the language and tag distribution of the provided dataset, we chose the RYANAIR topic as the development set[5] for our experiments. All the other topics were used for training.

4.3 Additional data

Encouraged by the results reported in (Tsygankova et al., 2019), we investigate the effect of providing additional data for model training. Specifically, we use two datasets: English CoNLL 2003 and the WikiANN dataset (Pan et al., 2017; Rahimi et al., 2019). The former dataset is, as its name suggests, comprised of NER-annotated English documents, whereas the latter consists of NER-annotated Wikipedia documents written in 282 languages.

Apart from the `PER`, `LOC` and `ORG` tags, the English CoNLL 2003 data also contains the `MISC` tag. To make the dataset compatible with our existing datasets, we simply replace all occurrences of the `MISC` tag with the "other" (`O`) tag. Since the WikiANN dataset contains only the `PER`, `LOC` and `ORG` tags, we simply take all the data it has for the six languages that are part of the shared task.

[4]The surface form in question is "Japonci" and can be found in the file named `brexit_cs.txt_file_10.out`.

[5]The terms "development set" and "validation set" are used interchangeably throughout the document.

Possible interpretation	Choose
ORG + PER	PER
ORG + PRO	ORG

Table 2: Named Entity type disambiguation rules applied in postprocessing

4.4 Models

SpaCy With spaCy we trained a single model per language, using its defaults for training custom, accuracy-optimized NER models. For all languages this meant using embeddings from multilingual BERT (`bert-base-multilingual-uncased`), except for Ukrainian, for which spaCy provides a pre-trained RoBERTa-style model. Since we were not able to fine-tune this model to a performance that would be different from random, we opted for using the efficiency-optimized version of the Ukrainian model, which utilizes word vectors trained on the training dataset.

Stanza A separate model was trained for each language. It uses a forward and backward character-level LSTM model, whose outputs at the end of the word are concatenated with word embeddings and passed to a Bi-LSTM sequence tagger combined with a CRF-based decoder. This yield an architecture similar to that of (Akbik et al., 2018).

We trained the NER tagger with pre-trained word2vec word vectors from CoNLL 2017 Shared Task.

Trankit While Trankit can work in single-source as well as multi-source setup, in preliminary experiments we found that its multi-source performance was consistently significantly better. Therefore, in our experiments, we used Trankit in the multi-source setup, in which XLM-R (`xlm-roberta-base`) serves as the basis of the model and a custom NER-specific Adapter (Pfeiffer et al., 2020) is trained.

4.5 Postprocessing

When providing the the prediction output, we take the title as well as the text of the document and pass it to the pretrained model. The model's output is generally in the BIO format which is then converted to the output format specified by the shared task's guidelines.

Given the specifics of document-level annotation, the model can predict distinct tags for the surface form within the scope of a single document. If that happens, we choose the tag that was predicted most frequently for this surface form. In case of equal probability (i.e. two tags, each predicted only once) we apply the disambiguation rules[6] outlined in Table 2.

5 Results

The main results can be seen in Table 3, which lists the best performance of all benchmarked toolkits for each language. The `(nltk)`, `(st)` and `(tt)` tags highlight that a particular model's training data has been tokenized with the NLTK, Stanza or Trankit tokenizer, respectively. The `+WikiANN` and `+CoNLL2003` symbols indicate that the training data was extended with the WikiANN or CoNLL2003 datasets, as described in 4.3.

The table is split into four parts. In the first column are the names of the evaluated models, along with the aforementioned tags that alter their training set. The following six columns present the performance of respective models on the per-language subsets of the development set. The next five columns visualize the performance across the NER tags, while the last column is the F1 score computed on the whole development set. The presented metrics were computed using the official evaluation tool provided by the organizers of the shared task [7]. The test set results can be found in Table 5 in Appendix B.

6 Discussion

In this section we discuss several observations which follow from the results presented in Table 3. We can generally conclude that the multi-source approach implemented in Trankit tends to yield better results than the single-source (one model per each language) approach utilized by spaCy and Stanza, which is in line with the results reported in (Tsygankova et al., 2019). Across all modelsthe `PRO` and `EVT` tags report the worst performance, which is in line with our expectation from Section 3.1.

In the following subsections we discuss the impact of various pre-built models, the various evaluated tokenization methods, the inclusion of additional data and conduct an error analysis on the considered system.

[6] Rules adapted from `http://bsnlp.cs.helsinki.fi/System_response_guidelines-1.2.pdf`

[7] `http://bsnlp.cs.helsinki.fi/program/BSNLP-NER-Evaluator-19.0.4.zip`

	BG	CS	PL	RU	SL	UK	PER	LOC	ORG	PRO	EVT	All
spaCy												
+ (nltk) ③	75.59	88.69	90.29	84.73	83.01	79.11	78.32	92.29	81.75	57.38	40.68	85.14
Stanza												
+ (nltk)	79.54	82.82	78.21	85.81	71.12	86.78	58.04	90.87	78.71	48.08	18.87	80.83
+ (st) ①	81.14	82.71	78.46	85.60	73.34	90.14	64.33	91.84	77.25	46.98	10.35	81.61
+ (st) + WikiANN ②	86.25	90.58	88.42	89.09	83.90	92.29	76.43	95.17	87.61	56.42	28.07	88.55
Trankit												
+ (nltk) ④	85.82	92.72	**93.84**	92.62	87.41	**93.82**	87.48	**96.62**	90.42	**61.33**	**74.07**	91.60
+ (tt)	83.44	92.88	91.84	92.90	85.93	93.83	86.15	96.07	90.01	59.38	63.16	90.83
+ (st)	**86.43**	**93.31**	92.29	**93.26**	**88.52**	92.83	87.55	96.51	**91.41**	60.95	64.51	**91.81**
+ (st) + CoNLL2003	86.02	92.50	92.85	92.60	88.21	93.91	85.71	**96.62**	90.58	59.88	64.15	91.40
+ (st) + WikiANN ⑤	85.57	89.82	91.83	91.08	87.28	90.91	**88.54**	94.61	88.48	57.07	62.50	89.83

Table 3: F1 score results on the RYANAIR topic (dev set). The F1 scores are multiplied by 100 for convenience. Numbers in circles represent IDs of submitted systems. Their results on the test set can be found in Table 5.

6.1 Impact of Transformer-based pre-trained models

As the results across the whole dataset suggest (column All in Table 3), the Transformer-based pre-training using variants of BERT, which can be found in both spaCy and Trankit, generally yields better results than the pre-trained LSTM-based language model used in Stanza. While a Stanza-based model has eventually reported performance superior to that of spaCy, it only did so in a setup where additional data was used in its training.

A related effect can be observed in the case of Ukrainian (UK), where the spaCy model performed significantly worse than all the other presented models. We hypothesize this may be due to the fact that a BERT-based model was not used for this language/toolkit pair, as described in Section 4.4.

6.2 Impact of tokenizers

We can observe the impact of various tokenizers introduced in Section 4.1 in the results reported for the Stanza and Trankit toolkits in Table 3. They indicate that the Stanza tokenizer (denoted by `+ (st)`) yields the best results but that its impact is relatively minor, with the largest difference on the order of 0.01 F1-score point (90.83 vs 91.81).

6.3 Impact of additional data

The effect of including additional data in training can be see in Stanza, as well as Trankit models. In particular, the results in Table 3 show that the inclusion of the WikiANN dataset (marked with `WikiANN`) helps the Stanza model better performance than that reported by spaCy (88.55 vs 85.14). The largest gains were reported on Polish (PL) and Slovene (SL) part of the development set, where the performance increased by more than 10% in absolute F1 scores, from 78.46 to 88.42 in case of Polish and from 73.34 to 83.90 for Slovene. The same phenomenon was not observed on Trankit-based models where the inclusion of the WikiANN dataset actually hampered the final performance, compared to the model that only used the training data provided with the shared task. Similarly, the inclusion of the CoNLL 2003 dataset (denoted as `CoNLL2003`) also did not help increase the final performance. We hypothesize this may be due to the very specific domain of the development set, which only tangentially overlaps with the data included in the additional datasets.

6.4 Error analysis

Stanza As the Table 3 shows, the worst performance of Stanza systems is recorded for the `EVT` tags and the best score is achieved for the `LOC` tags. Figure 1 indicates that the `EVT`'s error distribution does not prominently feature any other tag. We believe this may be caused by the limited population of `EVT` in the dataset shown in Table 1. Moreover, this is further supported by the fact that the best score is achieved on `LOC` tag, which is also the most populous one.

Furthermore, we see that by far the biggest error rate (69.79%) is between the `PRO` and `ORG` tags (last row of the confusion matrix). The system is incapable of disambiguation between said tags, as the dataset itself can contain such ambiguities as is also mentioned in the Task Guidelines. We speculate that these ambiguities may be resolved using a knowledge base, which would directly as-

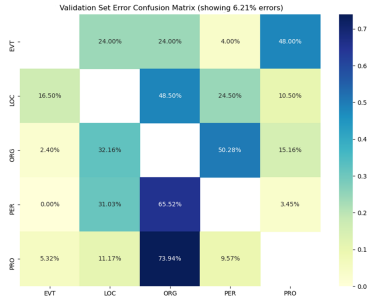

Figure 1: Confusion matrices showing *only* the errors made by the Stanza + (st) + WikiANN system. The error rate was 6.21%.

sociate specific terms with either products (PRO) or organizations (ORG).

Figure 2 shows the distribution of errors by language, where we can see that system achieves the worst performance on Russian language closely followed by Slovene. The models of both languages struggle with the ORG tag, with the highest error between true label PRO. In our experiments both languages benefited from longer training times. This fact might be used to improve the performance of the specific language models with longer training time and higher regularization as both languages are largely populated in the dataset.

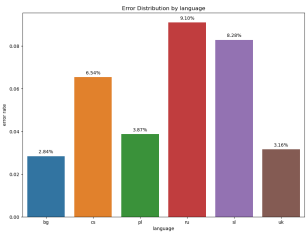

Figure 2: Error distribution across languages by Stanza + (st) + WikiANN system on the full dataset.

Trankit From Table 3, we can see that the best performing model is a Trankit with Stanza tokenizer. In Figure 3 we can see that overall errors made by all three subsystems on the validation set is quite lower compared to the one achieved by the Stanza system (Figure 1). This suggests that the Trankit system is more robust and can generalize better or that the Stanza system might suffer from overfitting, although we didn't see any evidence of such behavior during training.

Similarly to the other systems, we can see that Trankit also performs badly on the EVT tag and confuses it for other tags (LOC, ORG, PRO) with none of them having a prevalent error rate. We speculate that this is again symptom of its uniqueness and low frequency in the event tag domain.

Interestingly, the distribution of the LOC tag's errors changes with the addition of the Stanza tokenizer. This pushes most of the errors to the mismatch between LOC and ORG tags. This highlights the impact of various tokenizers on the overall performance of the model, and might indicate that better performance can be achieved by simply fine-tuning the tokenizers on a particular dataset.

All Trankit systems show the highest error rates between PRO and predicted ORG tags (last row of the confusion matrix). As we mentioned in the Section 6.4, this inability to disambiguate can be a consequence of the training set containing many such ambiguities, which is also mentioned in the Task Guidelines. For example, *"Ryanair Choice"* from document with id cs-new-36 contains the name of the company within the name of a product. It seems that from the context of the sentence *"Mezi výhody* Ryanair Choice *patří možnost..."* (translated to English as "The advantages of *Ryanair Choice* include"), our models are not able to disambiguate between the two acceptable tags and the rules mentioned in Table 2 then push the prediction to ORG, resulting in an incorrect prediction. For instance, "Boeing 747" is another example of company name included in the product tag, which could be swayed to the ORG prediction by the nature of the disambiguation rules. More examples of such errors in the RYANAIR topic across various languages can be found in Figure 4 in Appendix A.

Another set of problematic examples consist of those the where products have company names and the only disambiguation clue comes from a small part of the context sentence. One such example is *Uber* from file pl-261 and the sentence *"... stolicy regionu tanich platform taksówkarskich, takich jak* Uber *i Cabify."* (translated to English as "... the region's capital of low-cost taxi platforms such as *Uber* or Cabify). The sentence mentions Uber (the platform) which ought to be categorized as product offered by a company named Uber. The only clue for disambiguation between PRO and ORG in the considered sentence is the word "platform", which might be easily missed by the models. Moreover, the sentence would still make sense even without the word "platform" and in that context "Uber" would probably be tagged as ORG.

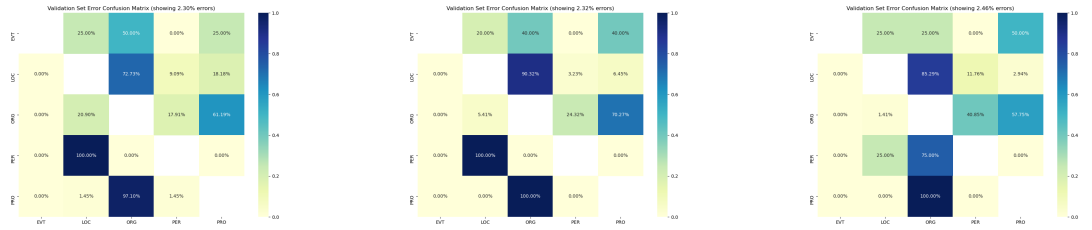

Figure 3: Confusion matrices showing *only* errors made by the Trankit + `(st)` + `WikiANN` (left, error rate 2.30%), Trankit + `st` (center, error rate 2.46%) and Trankit + `st` + `CoNLL2003` (right, error rate 2.46%) system on validation set (RYANAIR topic.). The 100% error rate in the fourth row, for the left and center figure, is caused by a single example and can be ignored.

SpaCy SpaCy placed somewhere in the middle between all tested models which is shown in Table 1 and it also suffers from issues mentioned in the previous sections. Tts errors are visualized in Figure 4, which presents the confusion matrix between the predicted and true tags only for the 3.02% of errors made by the spaCy system on the dev set.

SpaCy system is more similar to Trankit in the types of error it makes, which is visible on the `LOC` and `ORG` tags (rows 2 and 3). While Stanza errors for `LOC` are split between `ORG` and `PER` tags, most of spaCy's errors are concentrated on the `ORG` tag, similarly to Trankit. This is not surprising, as both frameworks use BERT-based models. Errors in location domain often comes from multiword names of locations, such as "Letitě Václava Havla" (Vaclav Havel Airport) in `cz-114`, "Letališče Franza Liszta" (Francz Liszt Airport) in `sl-75` or "Združeni arabski Emirati" (United Arab Emirates) in `sl-75`.

SpaCy does not improve on the `PRO` - `ORG` tag miss-matches (last row of the confusion matrix) and comparably to previously discussed systems, the language with the most errors on the validation set was Bulgarian, with the most miss-classified word "Райънеър", which stands for "Ryanair" and it's variants, such as, "Ryanaira" or "Ryanairze".

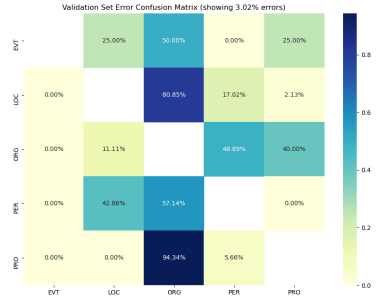

Figure 4: Confusion matrix showing only the errors of the SpaCy system on validation set (RYANAIR topic). The error rate was 3.02%.

7 Conclusions

This paper describes the `TraSpaS` submission to the BSNLP 2021 shared task on multilingual named entity recognition in which three open-source NLP toolkits were benchmarked. The results show that the approach implemented using the Trankit toolkit, based on the XLM-R pre-trained language, produced a model with the best results on both the development as well as the test set. While additional data did help in the case of the other two frameworks, the effect on the Trankit-based models was mostly negative. Error analysis on the development set indicates that many errors of our models could be attributed due to ambiguities in the training data.

References

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. 2019. Flair: An easy-to-use framework for state-of-the-art nlp. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59.

Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. Contextual string embeddings for sequence labeling. In *Proceedings of the 27th international conference on computational linguistics*, pages 1638–1649.

Mikhail Arkhipov, Maria Trofimova, Yurii Kuratov, and Alexey Sorokin. 2019. Tuning multilingual transformers for language-specific named entity recognition. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 89–93.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Tiberiu Boroș, Stefan Daniel Dumitrescu, and Ruxandra Burtica. 2018. NLP-cube: End-to-end raw text processing with neural networks. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 171–179, Brussels, Belgium. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Anton A Emelyanov and Ekaterina Artemova. 2019. Multilingual named entity recognition using pretrained embeddings, attention mechanism and ncrf. *arXiv preprint arXiv:1906.09978*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Christopher Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60, Baltimore, Maryland. Association for Computational Linguistics.

Minh Nguyen, Viet Lai, Amir Pouran Ben Veyseh, and Thien Huu Nguyen. 2021. Trankit: A light-weight transformer-based toolkit for multilingual natural language processing. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*.

Xiaoman Pan, Boliang Zhang, Jonathan May, Joel Nothman, Kevin Knight, and Heng Ji. 2017. Cross-lingual name tagging and linking for 282 languages. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1958.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterhub: A framework for adapting transformers. *arXiv preprint arXiv:2007.07779*.

Jakub Piskorski, Laska Laskova, Michał Marcińczuk, Lidia Pivovarova, Pavel Přibáň, Josef Steinberger, and Roman Yangarber. 2019. The second cross-lingual challenge on recognition, normalization, classification, and linking of named entities across slavic languages. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 63–74.

Jakub Piskorski, Lidia Pivovarova, Jan Šnajder, Josef Steinberger, and Roman Yangarber. 2017. The first cross-lingual challenge on recognition, normalization, and matching of named entities in slavic languages. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*, pages 76–85.

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*.

Afshin Rahimi, Yuan Li, and Trevor Cohn. 2019. Massively multilingual transfer for NER. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 151–164, Florence, Italy. Association for Computational Linguistics.

Milan Straka. 2018. Udpipe 2.0 prototype at conll 2018 ud shared task. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 197–207.

Tatiana Tsygankova, Stephen Mayhew, and Dan Roth. 2019. Bsnlp2019 shared task submission: Multi-source neural ner transfer. In *Proceedings of the 7th Workshop on Balto-Slavic Natural Language Processing*, pages 75–82.

Thomas Wolf, Julien Chaumond, Lysandre Debut, Victor Sanh, Clement Delangue, Anthony Moi, Pierric Cistac, Morgan Funtowicz, Joe Davison, Sam Shleifer, et al. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45.

A Sample of miss-categorized terms (PRO vs ORG)

words	language
Pixabay	cs
Skyscanner	cs
Ryanair Choice	cs
Cabify	pl
Wizz Tours	pl
Uber	pl
CNN	sl
737	sl
БНТ	bg
БНР	bg
Марица	bg
Дарик	bg
DW	ru
Ryanair	ru
Corriere della Sera	ru
DW	ru
ЄП	uk
Boeing 737	uk

Table 4: Examples of miss-categorised words between PRO and ORG tags on validation set (RYANAIR topic).

B Test set results

ID	BG	CS	PL	RU	SL	UK	All
①	72.93	70.66	73.79	60.18	72.03	65.39	67.74
②	72.12	69.37	76.43	60.19	72.03	63.79	67.63
③	65.29	71.06	79.61	57.14	72.65	48.28	64.61
④	**79.00**	**78.33**	**82.08**	**64.12**	**81.48**	**75.36**	**77.41**
⑤	77.19	74.69	81.07	60.03	79.53	69.88	74.56

Table 5: Results of the submitted systems on the test set. The numbers represent the F1 scores, multiplied by 100 for convenience. The definition of the systems can be found in Table 3.