Named Entity Recognition and Linking Augmented with Large-Scale Structured Data

Paweł Rychlikowski
University of Wrocław
prych@cs.uni.wroc.pl

Bartłomiej Najdecki
University of Wrocław
bnajdecki@gmail.com

Adrian Łańcucki
NVIDIA Corporation
alancucki@nvidia.com

Adam Kaczmarek
VoiceLab AI
adam.kaczmarek@voicelab.ai

Abstract

In this paper we describe our submissions to the 2nd and 3rd SlavNER Shared Tasks held at BSNLP 2019 and BSNLP 2021, respectively. The tasks focused on the analysis of Named Entities in multilingual Web documents in Slavic languages with rich inflection. Our solution takes advantage of large collections of both unstructured and structured documents. The former serve as data for unsupervised training of language models and embeddings of lexical units. The latter refers to Wikipedia and its structured counterpart - Wikidata, our source of lemmatization rules, and real-world entities. With the aid of those resources, our system could recognize, normalize and link entities, while being trained with only small amounts of labeled data.

1 Introduction

Intelligent analysis of texts written in natural languages, despite the advancements made with deep neural networks, is still regarded as challenging. The *lingua franca* of science is English, and new methods are typically evaluated firstly on English data, and often on other Germanic or Romance languages. This puts a certain bias on the development and design of modern NLP methods, which are not always transferable, and the metrics comparable, across languages and language families.

Due to the complexity and inherent vagueness of intelligent language processing, is has been naturally split into simple tasks, one of which is named entity recognition (NER), concerned in this paper. The output of a NER system is traditionally a set labelled phrases recognized in a given text. In order to process a document, one has to not only find and label the entities, but also link appropriately subsequent occurrences of the same entity. The task becomes harder, if the linking can be made across languages, when the entities are globally present.

We describe our submission to the *3rd Multilingual Named Entity Challenge in Slavic languages*, held at the 8th Workshop on Balto-Slavic Natural Language Processing (BSNLP) in conjunction with the EACL 2021 conference. The system was similar to the one submitted to the *2nd Multilingual Named Entity Challenges in Slavic languages* (Piskorski et al., 2019) held at 7th BSNLP Workshop in conjunction with ACL 2019 conference, and we discuss the differences between both systems.

The aim of those shared tasks was to recognize, normalize, and ultimately link - on a document, language and cross-language level - all named entities in collections of documents concerning the same topic, e.g., the 2020 US presidential election. Named entities have been split into five categories: PER (person), LOC (location), ORG (organization), PRO (product), and EVT (event). The 2019 edition featured four Slavic languages (Czech, Russian, Bulgarian, Polish), and the 2021 edition featured six languages (the previous four plus Ukrainian and Sloven).

In our solution we have combined models trained unsupervised on large datasets, and fine-tuned on small ones in a supervised way, with simple, white-box algorithms that perform later stages of processing in a stable and predictable manner. In addition, we have taken advantage of similarities between certain languages in order to augment the data and further improve the results.

2 Our Approach

Our system chains three modules for named entity recognition, lemmatization, and linking, which correspond to the objectives of the BSNLP Shared Task. We describe them in detail in the following sections. Our submissions for the 2019 and the 2021 shared tasks were similar, and differed

Proceedings of the 8th BSNLP Workshop on Balto-Slavic Natural Language Processing, pages 115–121
April 20, 2021.

Table 1: Class label mapping to the shared task label set in additional training datasets: KPWr (Marcińczuk et al., 2016), CNEC (Ševčíková et al., 2014), and FactRuEval (Starostin et al., 2016)

KPWr	PER	nam_adj_person, nam_liv_*
	LOC	nam_adj_city, nam_oth_address_street, nam_fac_*, nam_loc_*
	EVT	nam_eve_*
	PRO	nam_oth_tech, nam_pro_*, nam_oth_license, nam_oth_stock_index
	ORG	nam_org_*
CNEC	PER	p (personal names)
	LOC	g (geographical names)
	EVT	ia (conferences), tc (centuries), tf (feasts), tp (epochs)
	PRO	cs (article titless), mn (periodicals), oa (cultural artifacts), op (products), or (directives)
	ORG	ic (cultural/edu/science institutions), if (companies), io (govt. inst.), mt (tv stations)
FactRu	PER	name, surname, nickname, patronymic
	LOC	geo_adj, loc_descr, loc_name
	PRO	job, prj_name, prj_desc
	ORG	facility_descr, org_descr

only in the first element of the chain - the entity recognition method.

2.1 Recognition

2.1.1 Additional Training Data

Because the training datasets were small, we looked for other labeled datasets. There is no common standard of labelling NER datasets, and those extra datasets had to be remapped into the label set of the shared task. However, their addition did improve the recognition scores, and we describe them in the following paragraphs.

PL We used 1343 documents from KPWr with Named Entity annotations pre-processed with `liner2-convert` (Marcińczuk et al., 2017) tool, flattening and mapping original categories as shown in Table 1.

RU, BG, UK For languages with Cyrillic script we used FactRuEval2016 (Starostin et al., 2016) corpus consisting of 255 documents with 11754 annotated spans. Interestingly, the addition of this dataset improved scores for BG and UK despite the language mismatch.

CS, SL For Czech and Slovene we used Czech Named Entity Corpus (Ševčíková et al., 2014) containing 8993 sentences with manually annotated 35220 named entities, classified according to a two-level hierarchy.

2.1.2 Flair-based Recognition System

Recognition in our 2019 submission was realized with Flair (Akbik et al., 2018), a model made of the embedding layer and a bi-directional LSTM with a Conditional Random Field output (BiLSTM-CRF). The embedding layer aggregated pre-trained word representations of varying granularity and origin (word embeddings, subword embeddings (Heinzerling and Strube, 2018), contextual *forward* and *backward* character embeddings inherent to Flair).

Because of the data scarcity, we adopted the philosophy of making our systems „neural gazetteers". To this end, we tried to collect as much various embeddings as possible. This line of reasoning applied especially to word-level embeddings. Ideally we wanted our systems to have, for every language, embeddings trained on Wikipedia, Common Crawl[1], and a collection of news articles.

We found it beneficial to mix word pieces and character embeddings between languages. For instance, our model for Russian used Bulgarian embeddings.This is especially useful when the model of specific granularity in the target language is unavailable. Lastly, we also found it beneficial to mix training data for seemingly related languages, and improved the scores by adding our additional FactRuEval data to the Bulgarian training dataset.

Our recurrent recognition model underperformed in comparison to the top 2019 contestants, notably those based on BERT (Arkhipov et al., 2019; Devlin et al., 2019). We present an excerpt from the 2019 recognition results in Section 3.1.

2.1.3 FLERT-based Recognition System

For our submission to the 2021 BSNLP Shared Task we have used FLERT (Schweter and Akbik, 2020), a state-of-the-art architecture for named entity recognition. It is a BERT-style transformer approach, in which a XLM-RoBERTa model (Conneau et al., 2019), initially trained on a 100-language Common Crawl corpus (Wenzek et al., 2020), is fine-tuned on a small, language-specific corpus. This model departs from training an output CRF. We found that FLERT models train fast, and outperform our previously used Flair models by a significant margin.

[1] http://commoncrawl.org

2.2 Lemmatization

In the process of lemmatization of compound phrases, some words are converted into their lemmas, and some words remain unchanged. Occasionally some words are changed into other forms, e.g., adjectives might be transformed to nominatives with an appropriate gender. In the low-data regime of the shared task, we have opted for a simple rule-based system and data augmentation.

We pose the lemmatization task as splitting a word w into two concatenated parts $w = w_1w_2$, and computing the lemma as w_1v_2, where $(w_2, v_2) \in R_{lem}$, and R_{lem} is a small set of single-word lemmatization rules.

We use two main additional sources of information:

Wikipedia We take advantage of numerous links between articles, from which we extract pairs `[[text anchor|document title]]` . The anchors often are the inflected forms, and document titles the lemmatized forms of the same entity. In order to filter out spurious we consider a pair (*anchor*, *title*) a correct lemmatization if both the anchor and the text have the same number of words, and every i-th word in a title is either equal to the i-th word in an anchor, or is its possible lemma.

Finally, we heuristically recognize a small set of of words for later use, which we call **stopper words**. We define them as words shared between the anchor and the title, such that all words that follow them are identical in the anchor and the title, e.g., in the (*anchor*, *title*) pair

(*Bazylikę św. Pawła za Murami*,
Bazylika św. Pawła za Murami),

a stoper word is *"św."*.

Universal Dependencies (UD) (Universal Dependencies Consortium, 2021) is a large collection of treebanks in multiple languages. We extract morphosyntactic information (word, lemma, POS-tags and additional parameters[2]) from the words present in UD subsets for our target languages. Using that information, we construct single-word lemmatization rules. We say that the word w is a possible lemma of v if there is a one word lemmatization rule transforming v into w.

[2]We take the 'international version' of these parameters

PoliMorfologik For the Polish language, we additionally use PoliMorfologik (Woliński et al., 2012), a comprehensive morphosyntactic dictionary, which allows us to extract a large collection of lemmatization rules.

2.2.1 Lemmatization Schemas

Lemmatization of every phrase gives rise to a lemmatization schema. It works as follows: for every word we take its suffix (the longest suffix which occurs in the list of 2000 most popular suffices), in that way we obtain the left-hand side of the rule. The right-hand side describes, how this suffices should be transformed. For instance for the pair

(*Václavem Havlem, Václav Havel*)

we obtain a rule

(*-vem, -vlem*) $\longrightarrow$ (*-v, -vel*).

Our lemmatization algorithm takes a phrase (named entity found in the first stage) and returns its lemma. It follows that we do not consider every information from the words surrounding the phrase/context. Afterwards, we try to apply the following heuristics in a given order:

1. Try to find the (rightmost) stopper word. If there is one, then leave unchanged suffix of the phrase after the stopper (including the stopper itself), find the lemma for the prefix.

2. Try to apply rule based agreement phrase lemmatization (only for Polish)

3. Try to find the lemmatization schema suitable for the phrase. If there are more than one such rule, use the one which gives ‚more natural lemmatization' (which prefers common words and words occurring in lemmas)

4. Replace every word with its most popular lemma (in the training data, and in Wikipedia), if the word doesn't occur leave it unchanged

2.3 Entity Linking

A recognized entity, associated with a category and a normalized lemma, has to be linked with other occurrences of this entity (in this document, in other documents, and ultimately across the documents in all languages). The task is difficult due to the subtle differences between seemingly identical

Table 2: Relations between Wikidata categories and named entity categories

Label	Top-level Wikidata Entities
PER	human (Q5), nationality (Q231002), ethnic group (Q41710)
LOC	locality (Q3257686), location (Q2221906), spatial entity (Q58416391), geologic province (Q214045)
EVT	event (Q1656682), social phenomenon (Q602884), occurrence (Q1190554)
PRO	type of manufactured good (Q22811462), tangible good (Q1485500), broadcasting program (Q11578774), intellectual work (Q15621286), television station (Q1616075)
ORG	organization (Q43229), trade agreement (Q252550), company (Q783794)

entities. Consider *Donald Trump* entity: its one occurrence could be linked with *the 45th president of the United States*, or *Donald Trump Jr*, depending on the role in the text, but not with both at the same time.

We divide the task into two phases: 1) initial assignment of identifiers, and 2) refinement of identifiers. Our linking algorithm relies on three kinds of matches: exact matches of entity names, partial matches, and fuzzy matches with word embeddings. In order to ground the recognized entities regardless of the language, as well as extend our inventory of entities and their possible names, we use Wikidata[3] as a catalogue of entities.

2.3.1 Wikidata

Wikidata is a structured database of entities extracted from Wikipedia. Every entity has a unique identifier, e.g. `Q123456`, a list of labels and languages for each label, a description and subclasses/instances of properties, and relationships to other Wikidata entities (*instance of*, *part of*, etc.), which form a graph.

Thanks to the hierarchy of the relations, we have selected a handful of top-level Wikidata entities (Table 2), and collected all their descendants into sets of *wikidata_entities*. These are further weighted by their Term Frequency in Wikidata, so we could resolve collisions in favor of the most popular entities.

2.3.2 Initial Assignment of Identifiers

In a typical, coherent paragraph, the narrative develops with every new sentence. Upon introduction, the entities are named carefully (e.g., with a full name, expanded acronym), to be shortened later, when it is clear from the context what they refer to. For this reason we designed a stateful algorithm, that processes and refines a local list of *doc_entities* caught in the document.

Algorithm 1 outlines the linking procedure. Assignment of identifiers is performed separately for every document with the ADD_AND_LINK function. It processes a lemmatized set of entities recognized earlier modules of our system. Two kinds of entity dictionaries: *doc_entities*, which is local to a function, and a global *wikidata_entities*, which we prepare earlier using Wikidata. Those dictionaries map the textual mentions to identifiers from Wikidata and the target language, e.g., *Donald Trump* maps to `[(Q22686, en), (Q22686, pl), (Q22686, cs), (Q3713655, cs)]` (the last identifier refers to *Donald Trump Jr*).

We process document entities starting from the longest ones, and for each select the best entity id with the BEST_ID function. It firstly prefers the matching entries from the *doc_entities* dictionary, and secondly the most popular Wikidata entries (by Term Frequency) from *wikidata_entities*. For instance, with the local *doc_entities* dictionary, after processing *Donald Trump*, a subsequent shorter mention *Trump* should be linked with it.

The function ALIASES handles only PRO and ORG labels, and returns a list of all short forms and acronyms specific to those labels, present in Wikidata, e.g., *Sony Ericsson* is aliased as *SE*.

2.3.3 Refinement of Identifiers

The refinement stage uses dense embeddings of phrases in order to uncover high similarities between them, that might have been otherwise missed. We use FastText (Bojanowski et al., 2017), which is suited for morphologically rich Slavic languages, since the representations are built from generic subword units.

The refinement is carried out in two phases. In the first one, all phrases with the same identifier are grouped together. In the second one, two groups are merged into one if there exist two mentions (one per each group) with sufficiently similar embeddings measured by their dot product. Phrases are embedded as sums of embeddings of their words. When we merge two groups, we assign to them the identifier with a higher Wikidata term frequency. We refine identifiers only on the single language level.

[3] http://www.wikidata.org

Algorithm 1 Basic routines of the linking algorithm

function ADD_AND_LINK($ners$)
  $doc\_entities, linked \leftarrow \{\}, \{\}$
  for $(phrase, lemma, type)$ **in** SORTED($ners$) **do** ▷ Descending by the # of words in a phrase
    $P_1 \leftarrow$ GET_IDENTIFIERS$(phrase, doc_entities)$
    $P_2 \leftarrow$ GET_IDENTIFIERS$(lemma, doc_entities)$
    $id \leftarrow$ BEST_ID$(lemma, P_1 + P_2 + [lemma + type])$
    $linked[(phrase, lemma, type)] \leftarrow id$
    $doc\_entities \leftarrow doc\_entities \cup$ ALIASES$(lemma, id)$
  return $linked$

function GET_IDENTIFIERS($phrase$, $doc\_entities$)
  $res \leftarrow []$
  for (doc_phrase, id) **in** $doc\_entities$ **do** ▷ IDs matching in the document
    if SAME_ENTITY$(doc_phrase, phrase)$ **then**
      APPEND(res, id)
  if $phrase \in wikidata\_entities$ **then** ▷ The most common ID for a phrase
    APPEND(res, $wikidata\_entities[lemma]$)
  return res

3 Evaluation

We present experiments carried out on different levels of the entity recognition pipeline. The data used in those experiments comes from the BSNLP 2019 Shared Task test set (*Nord Stream* and *Ryanair* subsets). Our algorithms are tested in the submitted form and have not been further adapted to those datasets.

3.1 The 2019 Shared Task

Recognition Table 3 summarizes *strict* recognition results on the test data.

Lemmatization We analyzed the influence of various part of lemmatization on the performance of our method. The results are shown in Table 4. Our baseline is the identity function, in which we assume a phrase being its own lemma.

One should be aware that due to the small amount of test data, the results should be treated as approximate. Some differences can be caused by bad lemmatization of one phrase (especially if the phrase occurs many times in test data). It seems that all implemented heuristic are reasonable and improve over the baseline. Moreover, it is easy to see that links from Wikipedia are useful source of information in this task.

Entity Linking Table 5 shows the result of linking. Even though our recognizer did not hold up to the competition, the linking algorithm was able to close the gap in F1 score. In order to test the algorithm in ablation, we include linking results on ground truth lemmatized data (Lemma Oracle).

Table 3: 2019 BSNLP Shared Task selected results (*strict* recognition evaluation, test set, F1 metric). For every submitter, the best solution is shown with respect to the average performance on all languages.

Model	Testset	BG	CS	PL	RU	All
RIS-slav_lemma	NordS	0.84	0.89	0.89	0.78	0.85
CogComp-7	NordS	0.84	0.89	0.86	0.72	0.83
IIUWR.PL-5	NordS	0.71	0.83	0.86	0.65	0.78
TLR	NordS	0.73	0.74	0.72	0.60	0.70
Cog_Tech_Cent-4	NordS	-	-	-	0.69	0.69
Sberiboba	NordS	0.63	0.71	0.68	0.60	0.66
JRC-TMA-CC-4	NordS	0.67	0.50	0.42	0.52	0.52
NLP_Cube	NordS	0.14	0.16	0.09	0.11	0.12
CogComp-6	Ryanair	0.88	0.94	0.91	0.94	0.92
RIS-slav_lemma	Ryanair	0.86	0.94	0.92	0.91	0.91
Cog_Tech_Cent-4	Ryanair	-	-	-	0.91	0.91
IIUWR.PL-4	Ryanair	0.76	0.87	0.84	0.79	0.82
TLR	Ryanair	0.76	0.83	0.82	0.83	0.82
Sberiboba	Ryanair	0.65	0.84	0.81	0.72	0.77
JRC-TMA-CC-1	Ryanair	0.64	0.55	0.52	0.79	0.64
NLP_Cube	Ryanair	0.15	0.13	0.19	0.18	0.16

3.2 The 2021 Shared Task

We present the results of our FLERT-based submission, which are partial results of the entire shared task available at the time of writing.

One of the sets of articles in the training data is devoted to COVID-19. This situation is unusual: the phrase very often used in test data, does not

Table 4: Accuracy of our rule-based lemmatization algorithm on the 2019 BSNLP Shared Task training data. Abbreviations: **p** – phrase lemmatization rules, **w** – separate lemmatization of words, **W** – additional Wikipedia data, **a** – handwritten agreement rules (Polish only), **s** – uses *stoper words*.

Method	BG	CS	PL	RU	Avg
Baseline	89.02	59.23	54.51	54.79	63.00
+a	89.02	59.23	58.53	54.79	64.12
+w	89.91	64.39	74.17	57.23	70.41
+p	89.18	67.47	79.12	57.53	72.34
+wW	92.73	71.29	81.27	86.71	82.62
+pW	88.53	81.78	80.77	89.16	84.97
+paswW	91.60	81.69	82.42	89.28	86.14
+pasW	92.33	81.86	83.57	89.99	**86.83**

appear at all in the training data (also in the data used to pre-train language model).

We have verified that our NER models struggle with assigning consistent labels to the phrase *COVID-19*, which is common in the test data. An additional difficulty is the ambiguity of this phrase, which may refer to a disease and possibly remain unclassified as a named entity, or a pandemic and be classified as EVT. We decided to do a simple post-processing which assigns EVT to all COVID-19 phrases recognized by the NER module.

We think that this situation is so unusual that in a real system, used in the industry, it would be handled using a special ad-hoc rule. Moreover, we wanted to know, what are the result of this fixed assignment, and submitted two versions of our solutions.

Table 5: 2019 BSNLP Shared Task results (cross-language linking, test data). For every team we present their highest scoring submission wrt. the F1 metric. (*) The oracle model (first row for every dataset) is our entity linking algorithm run on the ground truth lemmatized data after the competition.

Model	Testset	F1	Prec.	Rec.
Ours + Lemma Oracle	Ryanair	0.76*	0.83*	0.70*
Ours (IIUWR.PL-5)	Ryanair	0.49	0.80	0.35
JRC-TMA-CC-2	Ryanair	0.27	0.67	0.17
CogComp-3	Ryanair	0.13	0.07	0.73
RIS-merge	Ryanair	0.10	0.06	0.70
Sberiboba	Ryanair	0.10	0.06	0.30
NLP_Cube	Ryanair	0.00	0.67	0.00
Ours + Lemma Oracle	NordS	0.59*	0.74*	0.50*
Ours (IIUWR.PL-5)	NordS	0.42	0.73	0.29
JRC-TMA-CC-2	NordS	0.31	0.69	0.20
RIS-merge_lemma	NordS	0.11	0.06	0.72
CogComp-3	NordS	0.11	0.06	0.68
Sberiboba	NordS	0.06	0.03	0.36
NLP_Cube	NordS	0.00	0.46	0.00

Table 6: 2021 BSNLP Shared Task selected results (test set, F1 metric): *strict* recognition, normalization, language-level linking (coreference). NC refers to the submission without fixed labelling of COVID-19 occurrences as EVT.

Task	Testset	CS	RU	BG	UK	SL	PL	All
Recon.	US Elect.	0.87	0.70	0.82	0.79	0.88	0.86	0.79
	COVID-19	0.80	0.57	0.72	0.75	0.77	0.81	0.73
(NC)	US Elect.	0.87	0.70	0.82	0.79	0.88	0.85	0.78
	COVID-19	0.80	0.55	0.68	0.72	0.75	0.78	0.71
Norm.	US Elect.	0.52	0.26	0.51	0.26	0.62	0.62	0.43
	COVID-19	0.45	0.27	0.33	0.51	0.53	0.57	0.45
Link.	US Elect.	0.66	0.39	0.69	0.52	0.66	0.70	0.56
	COVID-19	0.66	0.39	0.68	0.61	0.66	0.73	0.62
(NC)	US Elect.	0.66	0.39	0.68	0.52	0.66	0.70	0.55
	COVID-19	0.66	0.39	0.67	0.61	0.66	0.72	0.62

4 Conclusion

This paper describes our submissions to the 2019 and 2021 BSNLP Shared Tasks on named entity recognition on Slavic languages. Even though the training data was scarce, we have used large-scale datasets: corpora of unstructured text in the unsupervised training phase of training of the recognition model, and structured Wikipedia and Wikidata knowledge bases in order to extract rules and entities for lemmatization and linking phases. The linking algorithm is a strong point of our submission. In the 2019 task it allowed to close the performance gap between our solution and competitors, introduced by a weak initial recognition model. The results suggest that, perhaps, there is still a white spot in between supervised and unsupervised neural learning, where the structure of the data matters more than volume, and simple rule-based system excel.

Acknowledgment

The authors thank Polish National Science Center for funding under the OPUS-18 2019/35/B/ST6/04379 grant. We also would like to thank Adam Wawrzyński and Wojciech Janowski from VoiceLab AI for their support during conducting experiments and model training.

References

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, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Mikhail Arkhipov, Maria Trofimova, Yuri 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, Florence, Italy. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

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. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Benjamin Heinzerling and Michael Strube. 2018. BPEmb: Tokenization-free pre-trained subword embeddings in 275 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Michał Marcińczuk, Jan Kocoń, and Marcin Oleksy. 2017. Liner2 — a generic framework for named entity recognition. In *Proceedings of the 6th Workshop on Balto-Slavic Natural Language Processing*, pages 86–91. Association for Computational Linguistics.

Michał Marcińczuk, Marcin Oleksy, Marek Maziarz, Jan Wieczorek, Dominika Fikus, Agnieszka Turek, Michał Wolski, Tomasz Bernaś, Jan Kocoń, and Paweł Kędzia. 2016. Polish corpus of wrocław university of technology 1.2. CLARIN-PL digital repository.

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, Florence, Italy. Association for Computational Linguistics.

Stefan Schweter and Alan Akbik. 2020. Flert: Document-level features for named entity recognition.

Magda Ševčíková, Zdeněk Žabokrtský, Jana Straková, and Milan Straka. 2014. Czech named entity corpus 2.0. LINDAT/CLARIAH-CZ digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.

A. S. Starostin, V. V. Bocharov, S. V. Alexeeva, A. A. Bodrova, A. S. Chuchunkov, S. S. Dzhumaev, I. V. Efimenko, D. V. Granovsky, V. F. Khoroshevsky, I. V. Krylova, M. A. Nikolaeva, I. M. Smurov, and S. Y. Toldova. 2016. Factrueval 2016: Evaluation of named entity recognition and fact extraction systems for russian. In *FactRuEval 2016: Evaluation of Named Entity Recognition and Fact Extraction Systems for Russian*, pages 688–705.

Universal Dependencies Consortium. 2021. Universal dependencies.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Marcin Woliński, Marcin Miłkowski, Maciej Ogrodniczuk, Adam Przepiórkowski, and Łukasz Szałkiewicz. 2012. PoliMorf: a (not so) new open morphological dictionary for Polish. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation, LREC 2012*, pages 860–864, Istanbul, Turkey. European Language Resources Association (ELRA).